FOR REFERENCE ONLY

PKD

913/ ROS

SLC

Ref.

Large Vocabulary Semantic Analysis for Text Recognition

by

Tony Gerard Rose

This thesis has been submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

This work was carried out at Nottingham Trent University

Department of Computing

May 1993

Large Vocabulary Semantic Analysis for Text Recognition

Tony Gerard Rose

Doctor of Philosophy

Abstract

This thesis describes research work undertaken by the author from October 1988 to September 1992 concerning the automatic recognition of text (either handwritten or typescript) by computer. In particular, it details the use of semantic information (using lexical co-occurrence and collocational models rather than compositional theories) to improve the performance of a computerised handwriting recognition system. An important part of this work has been the systematic empirical testing and validation of the techniques so developed.

Such is the visual ambiguity of handwriting that a number of possible interpretations may be made for any written word. Indeed, this is true of any text, but especially handwritten text since the segmentation between the individual characters is particularly indistinct. Human readers cope with this by making selective use of visual cues and using an *understanding* of the text to compensate for any degradation or ambiguity within the visual stimulus. Word images occur within a meaningful context, and human readers are able to exploit the syntactic and semantic constraints of the textual material. Analogously, computerised text recognition systems would be enhanced by using higher level knowledge. Character recognition techniques alone are insufficient to unambiguously identify the input, particularly that of handwritten data.

Ideally, this higher-level knowledge would be acquired by the creation of a lexical database that contains all the relevant information. However, to create a semantic lexicon by hand for a large vocabulary is a considerable task - which is a major reason why so many semantic theories fail to "scale up" from the small, artificial domains in which they were developed. An alternative approach is to exploit existing sources of semantic information, such as machine-readable dictionaries and text corpora. This thesis describes the acquisition of semantic knowledge from such sources and its use in computerised text recognition systems.

Acknowledgements

I would like to thank the following people for their assistance and encouragement over the past four years:

Special thanks are due to my supervisors Lindsay Evett and Bob Whitrow. To Lindsay for support and ideas throughout the project and for providing constructive criticism on the many versions of this thesis. To Bob for funding and the opportunity to study for a research degree.

Thanks also to the people who have worked alongside me on the project: Cindy Wells & Frank Keenan.

And thanks to other colleagues at Nottingham Trent University (past and present), and to members of my family.

Cindy Wells developed the lexical analyser described in 1.4.2, and in so doing provided me with much needed test data.

Frank Keenan developed the simulator program described in 3.3.1, and the original indexing system described in 3.2.1, which I subsequently modified.

The sections of this thesis to have been published are:

- 3.2.4, 3.3.1, 3.3.3 and 4.3 (published as Rose & Evett, 1992);
- 3.7.2 and 4.5 (published as Rose & Evett, 1993a);
- 4.5, 6.2.1.2 and 6.2.2.2 (published as Rose & Evett, 1993b);
- 3.7.2, 4.5 and 6.2.2.2 (published as Rose & Evett, 1993c).

This research was funded by the European Commission under the ESPRIT initiative. Firstly as part of Project 295: The Paper Interface, continued in Project 5204: PAPYRUS, and utilised in Project 5203: INTREPID.

Copyright

Table of Contents

Chapter One

Introduction

This thesis describes work undertaken by the author over a period of four years in the Department of Computing at Nottingham Trent University. The work has been funded by the European Commission under the ESPRIT initiative. The area of research is the automatic recognition of handwriting and printed text by computer.

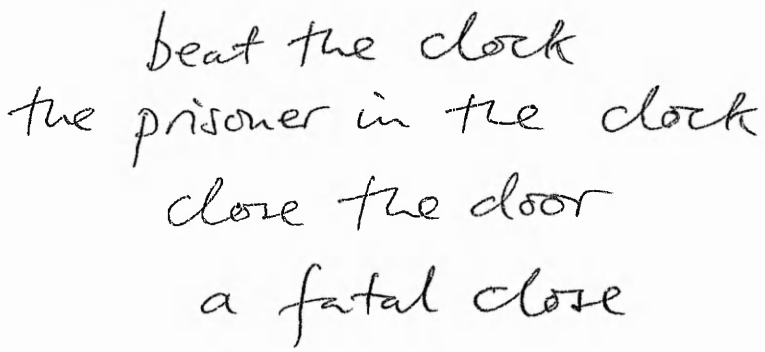
The development of reliable text recognition systems serves two important functions. Firstly, it allows a more user-friendly means of communicating with computers. For example, people who are unfamiliar with keyboards could choose instead to interact with the computer using their normal handwriting. Secondly, existing paper documents could be scanned into a computer and then converted to electronic form to allow further processing. For example, a library could convert its paper resources into electronic form, for reasons of space-saving, safe-keeping or filing.

1.1 Text Recognition

The visual ambiguity of handwriting is such that a number of possible interpretations may be made for any written word. Indeed, this is true of any text, but particularly handwritten text since the segmentation between the individual characters is often indistinct. For example, do the following words say "clock" or "dock", "close" or "dose"?

When seen individually, these words may be hard to disambiguate. However, when seen within a meaningful context, the correct interpretation seems almost obvious.

Indeed, it may seem almost so obvious that the alternative interpretation is not consciously perceived at all:



Effective text recognition, whether by human or computer, relies upon the successful disambiguation of confusions such as the above. Unfortunately, the English alphabet contains many similar-looking characters, for example: "n" and "h", "o" and "a", "c" and "e", "a" and "d". Upper case letters are also ambiguous, e.g. "U" and "V". Furthermore, letters can also be confused with digits, e.g. "O" and "0", "I" and "1", "Z" and "2", "S" and "5", etc. The problem is magnified when handwriting is cursive, since it is difficult to tell where one character finishes and another starts. Consider the word "minimum", written cursively:

Is it really clear where one letter finishes and another begins? Printed text is generally easier to recognise than cursive handwriting since the characters are (nearly always) physically separated from each other. However, once printed text has been photocopied a few times, or faxed, or degraded in some other way, the characters can become similarly indistinct. For the purposes of the current project, "text" is categorised according to how it is produced (either by machine or by human hand) and how it is recognised (e.g. dynamically or statically).

1.1.1 Methods of Text Production

Text can be produced either by machines or people. When typewriters and computers produce text, it is usually in the printed form. Often the text is in a variety of fonts (e.g. **Arial**, `Courier`, **Helvetica**, Modern, Roman, etc.) and font sizes. Handwritten text can also be printed (i.e. with each character written separately), but normally it is cursive. Both types of handwritten text are more difficult to process

than machine-printed text due to their greater variability. Figure 1.1 shows the three types of text, listed left-to-right in order of their recognition complexity. Since cursive script is the most difficult type to recognise, it represents the major focus of the work described in this thesis.

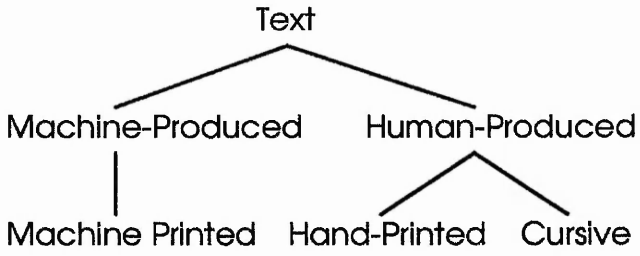


Figure 1.1: Types of Text

1.1.2 Methods of Text Recognition

The first stage of computerised text recognition is the input of data. Since the computer has no "eyes" with which to read, another form of input device must be used. There are two methods by which text may be input to a computer for the purpose of recognition. These are referred to as "static" and "dynamic" systems.

1.1.2.1 Static Systems

With static systems, recognition is performed some time *after* the handwriting has been produced. In other words, the text already exists on paper. Using this method, the input device is an optical scanner, which captures the image as a pixel representation. To a certain extent, this process may be seen as part of the more established technology known as OCR (Optical Character Recognition), but OCR has traditionally been associated exclusively with machine printed text rather than handwriting.

1.1.2.2 Dynamic Systems

Dynamic systems attempt to recognise handwriting in real-time, i.e., as the user is writing. The input device is either a digitising tablet or a device known as "electronic paper". The latter is of particular significance, since it allows two-way

communication: input and output through an LCD screen, using a special pen. Using this method the handwriting is represented as a sequence of 2-dimensional co-ordinates. In practice, this means the recognition process lags behind the production of the writing, usually by one or two characters, but keeps up with the speed of the writer [Tappert, 1989].

Dynamic systems have the facility for collecting further information such as the speed or direction of writing, number of strokes used, and the order in which they were written. This information can be used to improve the accuracy of the recognition process, since there is more information with which to identify characters and to separate overlapping points. However, dynamic systems are susceptible to "noise" in the data, which may take the form of spurious points created during the writing process. Likewise, the scanning procedure involved in static systems can introduce additional noise to the data. For both approaches, the accuracy of a particular system will be ultimately limited by the resolution of the hardware. In some ways, dynamic recognition is more restrictive since both the writer and a suitable input device must be present at the same time, and the technique is evidently only applicable to handwritten text.

1.2 Motivation for Text Recognition

The development of reliable text recognition systems serves two important functions. Firstly, it allows a more user-friendly means of communicating with computers. People who are unfamiliar with keyboards could choose instead to interact with the computer using their normal handwriting. Secondly, existing paper documents could be scanned into a computer and then converted to electronic form to allow further processing.

1.2.1 User-Friendly Interaction

Despite rapid advances in computer technology in the last three decades, there have been few changes in the methods by which communication with computers is achieved. The QWERTY keyboard was, and still is, the principal input device. However, to use the keyboard efficiently requires extensive learning and much practice. Its layout reflects the mechanical limitations of early typewriters and as such hardly constitutes an intuitive ordering that would facilitate rapid learning. Non-

typists therefore experience considerable difficulty in finding the desired keys, and errors are commonplace. Attempts have been made to change the layout of the keyboard, but since millions of typists throughout the world have learnt using the QWERTY arrangement, resistance to change is considerable. The development of the mouse constitutes a valuable extension to the keyboard, but this is a supplement, not a replacement. For many computers and computer-controlled machines, the keyboard remains the principal method of input. However, the proliferation of computer technology continues, and applications involving novice users are becoming increasingly prevalent. The need for an alternative input device therefore persists.

1.2.1.1 Speech or Handwriting: Practicalities

There are two methods of communication that are natural to human beings: speech and writing. The automatic recognition of human speech has long been the subject of science-fiction fantasy and (more recently) the subject of extensive scientific investigation. Speech is the most rapid form of human communication - faster than both handwriting and the output from a trained typist. From the human perspective, it is highly convenient since it is almost universal in its use and requires no special training on the part of the user. The automatic recognition of human speech is, however, an immensely difficult problem, and the actual progress so far achieved falls far short of original research expectations [Wheddon, 1990].

The other natural and widely used mode of human communication is that of handwriting. This medium has existed in nearly all societies for centuries, in a variety of forms. Most computer users are capable of reasonable handwriting, and can usually write quicker than they can type. Shorthand adds a further dimension of speed, since trained writers of shorthand can transcribe speech faster than keyboard entry [Leedham, 1989]. However, handwriting recognition has attracted far less research investment than speech, possibly due to the less "glamorous" image it possesses. Nevertheless, pen-based systems offer a number of distinct advantages over speech or keyboard based systems. This is because the pen as an input device can be used to facilitate many other sorts of interaction besides the input of freehand text. For example, it can also function as a pointing device. In this role, the pen can be used to select items, pull down menus, move objects around the screen; in effect, to handle any task that hitherto required the use of a mouse. Furthermore, the pen is more compact and requires no mouse mat. Secondly, pen-based interfaces allow the creation of sketches and drawings. Such drawings, along with any handwritten input,

can be subsequently edited or annotated, again using the pen (and a set of freehand gesture-based editing symbols). In short, the pen-based interface is highly versatile, and offers many further capabilities besides that of handwritten input.

Evidently, the choice between the use of speech, handwriting or keyboard for a particular application will vary according to a number of factors:

Ability of User: where the users of a system are likely to be casual or perhaps untrained in keyboard skills (a situation that is becoming increasingly more commonplace), the use of speech or handwriting may be more appropriate than typed input;
Noisy Environments: speech recognition is made difficult if interference is created by noisy machinery or extraneous conversations;
Quiet Environments: in a lecture theatre or library speech input would be unsuitable, and typed input could create a similar distraction;
Security: where confidentiality is required, speech input could be overheard and therefore unsuitable;
Social Constraints: In some environments (e.g. doctors' note-taking on hospital wards) speech or typed input would be deemed inappropriate for social reasons, whereas handwritten notes are already an established procedure;
Verification: systems developed for legal or commercial applications could include automatic signature verification as a useful part of their functionality;
Data Storage: with text (as opposed to speech) there is always a verbatim record (hard copy) of the dialogue or interaction.
Multi-Media Communication: in some environments it is necessary to listen and make notes - speech input would therefore not be suitable.

For a long time the hardware available for handwriting recognition was such that the input would take place on a graphics tablet, while the visual feedback appeared on a monitor screen. This division of attention inevitably presented a major distraction to users. However, recent hardware developments have created a device known as "*electronic paper*", in which input and output takes place through a combined unit. It uses an LCD screen and a special pen such that marks appear directly below the tip of the pen whenever contact is made with the surface. This has led to the emergence of a number of commercial pen-based systems, such as PenWindows, Paragraph and PenPoint. Although the performance of these systems is demonstrably adequate, they are constrained to hand-printed rather than cursive input.

Analogously, contemporary speech recognition systems impose a similar set of constraints, e.g. a limited vocabulary (possibly only a few hundred words), speaker dependence (new users require individual training) and the use of disconnected speech (rather than the more natural continuous speech).

1.2.1.2 Speech or Handwriting: Technicalities

A major problem associated with both speech and text recognition is variability of the input. This is particularly true of speech, whereby the pitch, volume and tempo of an utterance can vary according to meaning. For instance, the same sentence can be expressed as a statement or a question, simply by varying its pitch (e.g. "*It's OK*" versus "*It's OK*?"). Similarly, by altering the volume, a speaker can express anger (with a loud voice) or secrecy (with a whisper). By altering the tempo of an utterance, the speaker can express excitement (with rapid speech) or deliberation (slow speech). All these variations combine to the extent that even words spoken by the same person are never identical [Vaissiere, 1985]. Together, they represent a considerable problem for speech recognition systems: a given word may be uttered in a number of ways such that it never exactly matches the examples with which the system was trained. Handwriting is also variable, along the dimensions of size, slope and "connectiveness". However, written language tends to be more structured than speech, since it can use punctuation (e.g. question marks, exclamation marks, etc.) to indicate meaning within individual sentences and physical layout (e.g. headings, etc.) to identify the various components of a discourse.

There are certain operational difficulties associated with speech recognition. For example, different instances of sounds that human listeners perceive as the same may have very different waveforms. Similarly, there are certain words between which human listeners only hear one difference (e.g. "*cap*" and "*cab*"), yet there may be many differences between their waveforms. Another major problem for speech systems is interference from background noise, since it is necessary to remove sounds from the input data that are not part of the speech signal. In a noisy environment this presents a considerable problem. Moreover, with a variety of acoustic transducers in use there is yet no agreement on performance characteristics: different microphones can produce different acoustic signals, and these need to be standardised.

A particular problem for speech recognition is the reliable identification of word boundaries. Speech sounds are produced as a continuous sound signal rather

than discrete units, and knowledge of the language is required to determine where one word ends and another begins. This problem may be illustrated by listening to a foreign language (about which one has no knowledge) and trying to determine the location of the word boundaries. This is an extremely difficult task, since in normally articulated speech there are seldom pauses between the individual words. Moreover, when words are spoken in continuous speech they often sound different from when spoken in isolation. This is known as *co-articulation*. For example, some words may be concatenated, such that certain sounds are omitted, e.g. "*go away*" may be pronounced as "*go way*". Similarly, adjacent sounds may be modified to sound more like each other, e.g. "*gone back*" may be pronounced as "*gom back*". These problems add a further complication to the way in which speech recognition systems are designed, since training with isolated words may be inadequate for the recognition of connected speech.

With handwriting, there are similar problems concerning the identification of letter boundaries, as illustrated in the first example ("cl" and "d" can be easily confused). Furthermore, there may be problems regarding context dependent effects, whereby letters are written differently according to their surrounding context. However, the detection of word boundaries is relatively simple since they are usually indicated by physical spacing on the page. Consequently, the achievement of accurate segmentation is more problematic for speech than for handwriting. Furthermore, the number of different sounds (phonemes) used in English speech is greater than the number of letters used in handwriting, adding further complexity to the recognition of speech.

1.2.2 Document Processing

Although the use of electronic media in the business world is increasing, there remains a vast amount of communication that takes place on paper. Evidently, the arrival of the "paperless office" is still some years distant. In addition, there are numerous textual resources around the world that are only available in their original (paper) form. This is particularly true of libraries and archives in which valuable information is stored in a manner that takes up vast amounts of space, is prone to decay, and may not be easily accessible.

Consequently, there is a need for a means by which text can be translated from a paper form into an electronic form. When a document is in electronic form it can be

subjected to a range of further processes: stylistic analysis, statistical analysis, copying, editing, forwarding, filing, and so on. Until recently, the only means by which this could be achieved was by manual entry, involving many hours of repetitious labour. However, the development of the optical scanner has provided the hardware necessary for converting a paper document into an electronic form, albeit as a pixel representation. What is further required is the conversion of the pixel representation to a textual one, which is where optical character recognition (OCR) algorithms take over. A number of commercial OCR systems are currently available, such as ReadWrite and TextPert. Some systems are able to identify characters along with their size, location and other layout information, showing robust performance on high quality machine-printed documents. However, once documents have become degraded (i.e. faxed or photocopied) the performance rapidly deteriorates. There is yet no commercially available OCR system that can cope with handwritten (or even hand-printed) text.

1.3 The Need for Higher Level Knowledge

There is much evidence to suggest that there is more to the process of reading than just the recognition of individual characters. Studies from as long ago as the 19th century (e.g. Cattell, [1885]) have shown that characters are more easily recognised when they form part of a word than when they do not. This is known as the *word superiority effect*. Further work by Reicher [1969] has extended this to show that familiar words are perceived as units rather than strings of letters. Studies of eye movement during the reading process provide further evidence of the role of higher level knowledge. Javal [1879] showed that the eyes do not scan smoothly across the lines of print, but instead make a series of discrete fixations with rapid movements (known as *saccades*) in between. Analysis of these eye fixations during reading provides insight into the visual information being processed. For example, Just and Carpenter [1987] showed that typically only 68% of the words may be fixated during normal reading, suggesting that higher level knowledge must contribute to the processing of remaining 32%. Furthermore, they showed that over 80% of the content words were fixated, compared to only 40% of the function words. For this distinction to take place higher level knowledge must be affecting the reading process.

Eye movement studies have also been used to demonstrate the role of syntactic knowledge in the reading process. Carpenter and Just [1983] showed that

syntactically ambiguous words take longer to process than syntactically unambiguous words, indicating that the reader is trying to determine the syntactic role of the ambiguous word while fixating it. Similarly, eye movement studies have been used to demonstrate the role of semantic knowledge in the reading process. Pairs of sentences that have a coherent semantic relation have been shown to be more rapidly processed than pairs in which the relation is less distinct [Just & Carpenter, 1978].

Evidently, the most successful text recognition system to date is that of the human information processing system. Although it takes many years for a child to master the process of reading, once acquired, the skills are comprehensive and flexible enough to cope with a diversity of written material in a variety of fonts and formats (including previously unknown ones such as unfamiliar handwriting). The principal strengths of the human information processing system lie in its ability to make selective use of available visual cues and to utilise an *understanding* of the text to compensate for any degradation or ambiguity within the visual stimulus. Word images occur within a meaningful context, and human readers are able to exploit the syntactic and semantic constraints of the textual material [Rayner, 1983]. Analogously, computerised text recognition needs to use higher level knowledge to achieve comparable levels of performance. For both printed and handwritten input, the stimulus alone is insufficient to unambiguously identify the text.

An ideal OCR system would be 100% reliable and always output the correct character (and no others). However, in practice there is always some degree of error, which increases if the text is printed at an angle, or characters overlap, or an unknown font is used, and so on. Furthermore, in most fonts there will be confusions between letters (e.g. "e" and "c") and ambiguities regarding the correct segmentation ("d" can look like "cl"). Handwriting is subject to all the above problems plus further ones due to its increased variability. Reliable segmentation is particularly difficult, as demonstrated by the "minimum" example presented earlier. Most character recognisers handle this ambiguity by producing a list of characters for each ambiguous letter position, to represent the plausible matches between the data and a database of stored character templates. When placed within the context of a whole word, the combination of these candidate letters forms a "lattice", from which the correct word must be extracted. To reduce this ambiguity, it is necessary to eliminate the incorrect letter candidates in each position, or at least rank them according to some measure of their plausibility. This can only be achieved by using higher level information.

1.4 System Overview

The work described in this thesis was developed in the context of an integrated handwriting recognition system that uses a number of sources of higher level knowledge. These knowledge sources and the processes associated with them are shown in Figure 1.2. It should be noted however, that the serial layout is for illustrative purposes only: the syntactic and semantic analysers can and have been run effectively in parallel. In addition, the semantic analyser has since been successfully applied to other text recognition applications (notably OCR systems).

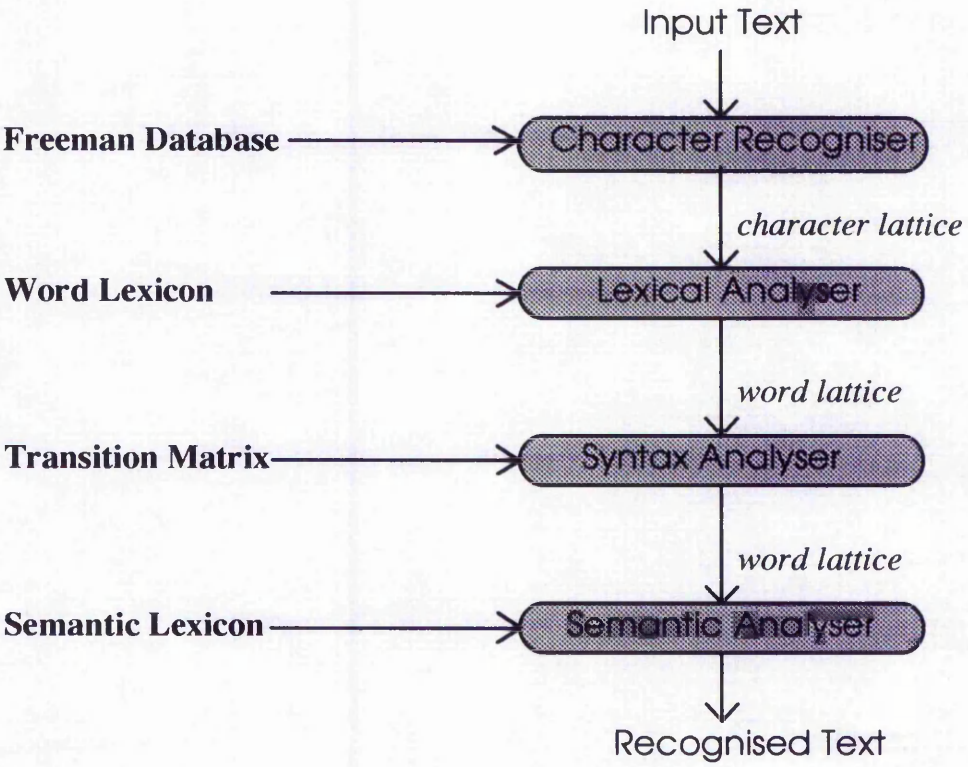


Figure 1.2: System Overview

1.4.1 Character Recognition

The first stage of the current system involves the process of character recognition. For dynamic input (e.g. using electronic paper or similar) the movements of the pen tip are captured as a series of x-y co-ordinates. There are a number of

algorithms by which characters may be extracted from this data, including spatial analysis methods, whereby strokes are coded by a numbering system on a grid, and topological feature methods, whereby attempts are made to identify the constituent shapes within letters. The present project uses a vector chain method known as Freeman Encoding, which codes letters into sequences of strokes that conform to a number (usually 8) of preset geometric directions. This is then further reduced to a combination of the five most significant vectors. A training database is created by encoding samples of handwriting using this technique, and at run-time the input is similarly encoded and then compared with the characters stored in the database. The stored characters that most closely match the input are identified as the most likely interpretation of that input.

For static recognition (e.g. using an optical scanner) the process is slightly different. The x-y co-ordinates of the input can still be calculated, but in this case, there is no information concerning the order in which they were created. Instead, a pre-processing stage is required whereby a probable sequence is determined [Wright, 1989]. However, this process is not completely reliable, and alternative methods are being investigated [Tappert et al, 1990].

1.4.2 Lexical Analysis

The principle underlying the use of the lexical analyser is that input to a text recognition system will normally consist of English words rather than arbitrary strings of characters. Therefore, letter strings that form words are considered to be more plausible than letter strings that do not form words. This may not always be the case, but when it does apply it constitutes a very tight linguistic constraint. For example, consider a sequence of four letters taken from a 26-letter alphabet. There are 26^4 possible permutations of these letters, i.e., 456,976 different combinations. However, the number of four letter words taken from a lexicon of approximately 14,000 English words is 1,323. This represents about 0.3% of the possible letter combinations.

The character recogniser produces strings of candidate letters for each word in the input. It is likely that many of these letter strings will not form English words. Therefore, if these non-word strings are eliminated from consideration, the output will consist of a much smaller list of genuine English words, rather then a large list of candidate letter strings. However, word-lookup methods such as the above are not the

only method of lexical analysis. Other systems have been developed that employ "sub-word" knowledge, i.e. information concerning strings of letters that are allowable in English, but do not necessarily form whole English words. Examples of this include n-gram techniques [Higgins & Whitrow, 1987], transitional probabilities, Markov models, etc. However, n-gram techniques have been shown to be less effective for the present project than word-lookup methods, as they fail to exploit letter-level constraints to the same extent [Wells, 1989].

1.4.3 Syntax Analysis

The principle underlying the syntax analyser is that input to a text recognition system will normally consist of grammatical English phrases and sentences rather than arbitrary sequences of words. Therefore, word sequences that are grammatically acceptable are considered to be more plausible than word sequences that are grammatically unacceptable. The output from the lexical analyser is a series of candidate words for each position in the input. Syntactic knowledge may be used to identify those word combinations that are grammatically acceptable.

There are two schools of thought concerning the application of syntactic knowledge. The first approach, advocated mainly by theoretical linguists, is based on the notion that there is a universal grammar underpinning human linguistic competence [Chomsky, 1957]. This approach argues that the grammar may be formalised from linguists' intuitions and encoded as a set of rules to which input must conform if it is to be considered acceptable. These rules may then be tested using selected "interesting" examples and counter-examples of grammatical constructs. In contrast, the second approach starts from an "unbiased" text corpus and attempts to describe everything that occurs in it. This approach is broadly statistical in nature, and concentrates on the most frequent constructs rather than the most intuitively "interesting". Although it has been criticised as a "descriptive" technique using "weak" methods, this approach has proved more suitable for the present project, due to its greater robustness, coverage and computational efficiency [Keenan, 1992]. Furthermore, this position is supported by the findings of other researchers (e.g. Atwell et al, [1993]).

1.4.4 Semantic Analysis

The principle underlying the semantic analyser is that input to a text recognition system will normally consist of meaningful English phrases and sentences rather than arbitrary sequences of words. Therefore, word sequences that are semantically acceptable are considered to be more plausible than word sequences that are not semantically acceptable. The output from the lexical analyser is a series of candidate words for each position in the input, and semantic knowledge may be used to identify those word combinations that are meaningful. The application of semantic knowledge to the problem of text recognition is the subject of this thesis.

1.4.5 Other Levels

There are further levels of knowledge that have yet to be incorporated into the present project. One such level is that of *discourse*. Knowledge of discourse aids the selection of coherent sentence progressions from incoherent ones, and involves aspects such as argument formation, dialogue continuity, story grammars, etc. Another knowledge level of is that of *pragmatics*, or "knowledge of the world". Pragmatic or contextual knowledge can be seen as a type of *meta-knowledge*, having effects throughout the other levels. Winograd [1983] demonstrates the importance of contextual knowledge using the following example:

"*She dropped the plate on the table and broke it.*"

Pragmatic knowledge of the context of this sort of sentence would normally indicate that it was the plate that broke, and not the table.

Human readers usually have little difficulty with most types of ambiguity, since they can effortlessly apply a variety of contextual information. Computers do not have the knowledge and experience of the average human reader, so for them to cope with the ambiguities shown above they need to have access to repositories of the different sorts of knowledge.

1.4.6 Other Applications

Evidently, higher level knowledge is required for effective text recognition. Indeed there are a number of other computational implementations that have required

the use of linguistic information, and the extent to which they have proved successful may identify useful avenues of research for the present project.

1.4.6.1 Speech Recognition

Considerable research effort has been invested over many years into both speech recognition and speech understanding, but very few robust implementations have been developed. Some of the more successful systems were products of the North American DARPA research program into speech understanding, which ran from 1971 to 1976. All the systems so produced were designed to handle the ambiguity in the signal and processing by using a number of diverse, co-operating knowledge sources (e.g. acoustic-phonetics, vocabulary, grammar, semantics, discourse, etc.). However, the systems differed in the types of knowledge they used, the interactions of the knowledge, the representation of the search space and the control of the search. Possibly the most well known is HEARSAY II [Lesser et al, 1977]. Also significant are HWIM [Wolf, 1980], the SRI System [Walker, 1980] and HARPY [Lowerre, 1980].

1.4.6.2 Machine Translation

Work in machine translation (MT) began in the 1950's with very high hopes but a somewhat naive attitude concerning the difficulties involved. Many researchers considered MT to be an extension of the code breaking techniques developed during World War 2, whereby foreign languages were little more than a complex coding of words and translation required merely the use of a bi-lingual dictionary. Consequently, early systems were unsuccessful, and the realisation that effective MT would not be possible without fundamental work on text understanding led to a cutback in funding. The more recently developed systems have addressed the need for higher level knowledge, by making greater use of linguistic information, particularly semantic (e.g. SYSTRAN [Toma, 1977] and EUROTRA [Raw et al, 1988]). Other contemporary systems make use of statistical information extracted from parallel corpora (such as English and French versions of parliamentary proceedings). Initial work at IBM [Brown et al, 1989] suggests that such an approach may be highly effective.

1.4.6.3 Information Retrieval

Since there is so much information available in natural language form - such as books, journals and reports - another application to receive attention is automatic information retrieval. Even though much of this information may already be in electronic form, retrieving the correct document from a database is not simple. Information retrieval (IR) systems attempt to allow the user to input a query and extract the relevant text from a database of documents. Difficulties arise due to the polysemous nature of English words and that documents may have been filed under keywords that are different to those used in the query. Progress in this area has led to developments in the field of knowledge representation, since the appropriate documents can only be retrieved if the content of the text can be matched with the content of the query [Hirschman & Sager, 1982].

1.4.6.4 Human-Computer Interfaces

Natural language is the most convenient mode for communication with interactive systems, particularly for users who are not computer literate. Some progress has been made in this area, since the input to such systems is typically simpler than the text to be processed in MT or IR systems, and the interactive nature of the application allows the system to resolve certain ambiguities by asking the user to rephrase the question. The technology has advanced to the point where systems are being used for real (albeit simple) applications rather than demonstrations [Grishman, 1986]. Typical human-computer interface applications include text-to-speech systems such as reading aids for the blind and partially sighted [Pugh, 1992].

1.5 Sources of Information

For human readers, the knowledge sources required for the recognition of text (or indeed language in any form) include those gained from experience and those which are inborn. Either way, the acquisition of this knowledge is essential, and in the case of computers it represents a considerable problem. Indeed, much natural language research has been addressed to precisely this problem. Some earlier researchers resorted to laborious hand-crafting of knowledge sources, which, for a substantial vocabulary, can prove an insurmountable task. Others have attempted to extract information from pre-compiled sources such as machine-readable dictionaries

and thesauri. In many ways, the issue of knowledge acquisition has been one that has separated the tractable representations of natural language semantics from those that can only remain as theories in textbooks. There are two major sources of semantic information:

(a) From a machine-readable dictionary (MRD). The definitions contain encyclopaedic information, syntactic information and semantic information in the form of sense relations that describe the relationship of any one word to a number of others. MRDs are the most accessible source of semantic information, since they provide it in a pre-compiled form;

(b) From large bodies of naturally occurring text (known as *corpora*) which can be processed to derive further information concerning language use and word patterns. Such resources must be compiled systematically, i.e., the corpus needs to be large enough to cover the requisite variety of linguistic structures, and to be representative of the type of language to be processed.

As the need for lexical resources has grown, greater numbers of machine-readable dictionaries have become available, and progress has been made regarding the issues of standardisation and format. However, the fact that a dictionary is in machine-readable form does not necessarily mean that the required information is instantly available - pre-processing may be necessary to organise the information. Indeed, such progress has not been common to all publishers - some still produce little more than typesetting tapes with cryptic codes labelling the various components of each entry, and little or no accompanying software or documentation for their search or extraction. By contrast, others have invested heavily in the natural language market, and the design of their dictionaries reflects those needs. Longman's Dictionary of Contemporary English (*LDOCE*) [Procter, 1978] in particular has been designed with computational applications in mind, and to this end it has formed the resource for many projects, e.g. machine-readable databases, syntactic parsing, semantic analysis [Boguraev & Briscoe, 1989]. Some dictionaries have used contemporary labelling standards such as *SGML* (Standard Generalised Markup Language) to facilitate the logical organisation and efficient extraction of required information. The Oxford Advanced Learner's Dictionary of Current English (*OALD*) [Hornby, 1974] uses such markup, and for this reason has become a major resource for the current project. Details of some widely available MRDs are shown in Table 1.1 (where CED = Collins English Dictionary [Hanks, 1986], W7 = Webster's

Seventh New Collegiate Dictionary [Merriam, 1963] and OED = Oxford English Dictionary [Murray, 1928]).

Dictionary	MBytes	Headwords	Bytes/Headword
LDOCE	14.0	55,000	254
OALD	6.6	21,000	290
CED	21.3	85,000	251
W7	15.6	69,000	226
OED	350.0	304,000	1,200

Table 1.1: Currently Available Machine-Readable Dictionaries

1.6 Summary

The development of reliable text recognition systems serves two important functions. Firstly, it allows a more user-friendly means of communicating with computers. For example, people who are unfamiliar with keyboards could choose instead to interact with the computer using their normal handwriting. Secondly, existing paper documents could be scanned into a computer and then converted to electronic form to allow further processing. For example, a library could convert its paper resources into electronic form, for reasons of space-saving, safe-keeping or filing.

The output from a character recogniser requires further processing to reduce the ambiguity and hence increase the accuracy of recognition. Three levels of knowledge have been investigated and incorporated into the current system: lexical, syntactic and semantic. Lexical analysis eliminates non-English words, syntax analysis ranks competing word sequences according to the grammatical plausibility, and semantic analysis ranks competing word sequences according to the plausibility of their combined meaning.

Text recognition systems developed to date have been mainly concerned with the pattern recognition level, and the use of higher level information is often restricted to some form of lexical analysis. Comparison between published systems is difficult, due to the lack of standard measures of assessment or "benchmarks". There are many parameters associated with the performance of a system, and their relevance will vary according to the use of the system: some systems may be designed for single

users with a particular style of writing; others may attempt to be more generalised and therefore need training. Furthermore, earlier systems were constrained by hardware restrictions such as a lack of available memory, which would then restrict the size of the vocabulary. Unless systems are tested in a comparable way, the measures of performance have no relative meaning.

As with speech, the most successful recognition system is the human information processing system. It takes many years for a human child to master the process of reading, even though they already possess established linguistic and cognitive subsystems. However, once mastered, these skills are comprehensive and flexible enough to cope with a diversity of written material, in a variety of fonts and formats (including previously unknown ones such as unfamiliar handwriting). Some handwriting may be difficult to read due to a lack of visual clarity, but most readers can perform some sort of recognition resulting in a meaningful interpretation. This skill relies on the human ability to consider information and constraints from a variety of knowledge sources to arrive at a "solution" to the "problem" of making a plausible interpretation of some arbitrary handwriting marks on a page. This solution represents the best compromise between information from each of the knowledge sources. As with many aspects of human performance, it is perhaps inappropriate to talk of "rights" and "wrongs" as in the *right choice* (i.e. correct) and the *wrong choice* (i.e. an error). A reader of any given text can never be 100% sure of the writer's original intentions; they can only select the *most likely* interpretation of the marks on the paper, based on their outward appearance and the various sources of linguistic and general knowledge.

Evidently, text recognition is a substantially different problem to text *understanding*. Whilst full understanding of a text is an immensely difficult and contentious problem (there are few agreed definitions of the word "understanding") the present project aims "only" to recognise text. Understanding implies recognition, but the converse is not necessarily true. Both objectives require the application of higher level knowledge to identify inconsistencies at individual levels, so it is possible that techniques developed for recognition applications could contribute to the development of text understanding systems. However, the difference, in practice, is that whereas human understanding involves the identification of the *most* consistent of a number of interpretations, computer recognition usually involves the identification (and hence elimination) of the *least* consistent ones. The emphasis in

human understanding can be seen as *top-down*, or *hypothesis-driven*, whereas computer recognition is usually *bottom-up* or *data-driven*.

This thesis is concerned with the application of semantic knowledge to the problem of text recognition. Semantics, in its strict linguistic sense, is concerned with the *meaning* of words, and not with non-linguistic facts about the world. However, it has become clear that to apply such knowledge in text recognition systems it is necessary to interpret the word "semantics" in a somewhat broader sense, i.e. including "general" or other sorts of knowledge. For example, the semantic information acquired from machine-readable dictionaries contains encyclopaedic knowledge (as well as the purely semantic sense definitions) and the information acquired from text corpora embodies a variety of types of linguistic knowledge, such as syntax, semantics and pragmatics. However, the label "semantic" will continue to be used, although it is intended that the reader should not constrain its interpretation to the strict linguistic sense.

The use of semantic knowledge, its theory, application and relevance to text recognition forms the basis of this thesis. The next section reviews published natural language applications that have in some way addressed the problem of semantic knowledge representation and processing.

Chapter Two

Literature Review

2.1 Introduction

This section reviews published research into natural language applications that involve some element of semantic processing; with particular emphasis on the applicability to text recognition systems. Much of this research is purely theoretical, in that it explores computational implementations of linguistic theories that are independent of any specific application area. However, techniques that are developed for theoretical purposes or for other applications may have relevance to the present project, and this is indicated where appropriate.

2.1.1 Computational Linguistics

Computational Linguistics could be defined as the study of computer systems for understanding, generating and processing natural language [Grishman, 1986]. The motivation for research in this field may be applied (e.g. the development of human-computer interfaces, machine translation systems, information retrieval, etc.) or theoretical (e.g. the investigation of grammars proposed by theoretical linguists).

Language understanding is generally regarded as being of greater importance than generation, since understanding requires the recognition of many paraphrases for the same command or information, whereas generation may be satisfied with the production of just one. Recognition is a prerequisite to understanding, since what has not been recognised can hardly be understood. However, both can involve similar stages of processing in applying orthographic, lexical, syntactic and semantic constraints to the perceived input. The difference is that with understanding, the semantic processing must eventually involve translation of the natural language input into an internal language with a semantics that is based on the knowledge representation structure of the system in question.

For example, if natural language is to be used to control a robot, then the English commands must ultimately be translated into the same knowledge representation formalism used by the robot to describe its own state, its goals, and the state of the environment around it. Once translated, this command can then be acted upon, and the goals and states of the robot and its environment can be updated accordingly. Recognition stops short of this stage, requiring "only" the use of semantic information to eliminate inconsistencies in the input data, and (hopefully) arrive at a unique interpretation. It is not necessarily essential to design knowledge representation formalisms into which the input text must be translated and subsequently acted upon. However, such formalisms may be used as a knowledge base against which input can be checked for semantic consistency (e.g. Grosz, [1986]).

2.1.2 Linguistic Semantics

The progression from theoretical representation to physical implementation is common to research in both natural language syntax and semantics. For example, linguists produce theories of syntax, which specify what a human (or machine-based) parser has to compute. Psycholinguists then produce theories of human parsing, and computational linguists produce theories of automated parsing. Similarly, semantic theories derived from a number of disciplines (e.g., linguistics, lexicography, philosophy, formal logic) have formulated theories of meaning, on which psycholinguists can base theories of how those meanings are computed in humans. Computational semantics can extend these semantic theories to determine how those meanings could be computed by machine.

2.1.2.1 The Meaning of Meaning

The ability to understand is not the same as being able to explicate the concept of meaning. Indeed, it is claimed that there are as many as sixteen senses of the word "*meaning*" [Ogden & Richards, 1923]. Although psycholinguists need not be concerned with all sixteen (and computational linguists perhaps less still), there remain some distinctions of which both disciplines should be aware.

One of the most often cited is the distinction between *reference* and *sense*. The reference (or *denotation*) of an expression is the thing that it stands for; e.g. the reference of the predicate "*is blue*" is the set of things that are blue. The sense of an

expression is more closely connected with its actual meaning, and is roughly equivalent to the content of the expression. The sense of an expression determines which things it can denote.

One must also distinguish between the way meaning is applied to words and to sentences. When applied to words in the form of a question, e.g., "*What does 'spider' mean?*" the required answer concerns spiders in general, not any specific spider. It therefore concerns the sense of the word "*spider*". However, questions about the meaning of sentences, such as "*Who did he mean by the woman he saw last night?*" can usually only be answered with reference to facts about the world, i.e., the specific denotation. What is in doubt, with this second question, is not the semantic information that it conveys (the *sense*), but the objects to which it refers (the *reference*).

A complete semantic theory, whether human or computationally oriented, must specify for each expression what semantic information that expression conveys, which in turn determines what that expression can refer to. It should therefore be able to compute all possible senses of an expression, so that the application of specific facts and world knowledge can then be used to determine the referents. The process of understanding needs to be based on a complete semantic theory and representation; recognition does not. To illustrate, consider the robot example given earlier. For the robot to understand an English command, it must compute the semantics of the input text, i.e. determine the senses of the expression and refer these senses to objects in the environment, before the intended action can be taken. Recognition, however, requires no such computation. Recognition can be facilitated by the use of semantic knowledge to eliminate semantic inconsistencies within the input. (Of course, this process could be guided by an algorithm that we can label as "semantic", but that hardly constitutes a semantic theory in any true sense.)

2.1.2.2 The Extent of Linguistic Semantics

Given that the origins of semantics are diffuse (logic, philosophy, linguistics, etc.) it may be also observed that the boundaries of the subject are equally amorphous. It has been said that the word "semantics" refers to the analysis of the meaning of single sentences. By contrast, the analysis of the meaning of collections of sentences is referred to as *discourse processing* [Grishman, 1986]. However, upon closer analysis this distinction proves somewhat superficial. A clearer distinction is

that between semantics and *pragmatics*. Morris [1938] proposed that semantics was the theory of the relation between signs and objects (i.e. words and their referents), and pragmatics that of signs and their users. So for example, the assignment of the referent of a personal pronoun such as "*I*" or "*you*" depends on who is speaking (i.e. the user), and as such remains within the domain of pragmatics.

Consequently, most semantic theories have focused their attention on single sentences, rather than larger units (such as paragraphs, etc.). These larger units of text are more than just sets of sentences, and convey more meaning than the sum of the individual sentences. Perhaps a better definition of discourse processing is to state that it attempts to describe the "extra meanings" that come about due to the combination of individual sentences within a larger passage of text. For example, consider the following fragment of discourse:

Harriet was hungry.

She walked over to the fridge.

People can understand the relationship between these two sentences, and hence the coherence of this discourse, through the activation of relevant knowledge sources and elaborative inferences. In this case, knowledge of human plans to relieve hunger and the location of typical food stores create the required coherence (i.e. the "extra meaning") between the two sentences. These knowledge sources are essential to the comprehension of all but the simplest discourse, and hence present a considerable acquisition problem for computers. The identification of the extra meanings mentioned above can only take place once the initial sentences have been in themselves understood. This is a creative, active process, and one that relies upon the translation of input sentences into an internal knowledge representation. This point is discussed further in Chapter Six.

Some researchers have attempted to model typical human knowledge sources in the form of "*scripts*" that could be activated, like human knowledge sources, when deemed relevant. However, there is no reliable algorithm for the identification of relevant scripts, or when the new scripts should be activated, or how detailed they should be (or indeed how they could be efficiently acquired).

2.1.2.3 Subdivisions within Theoretical Semantics

Semantic theory may be further subdivided into the fields of lexical semantics and structural semantics.

Lexical semantics refers to the meaning of individual words. In computational theories of semantics, these meanings may be stored as the sense definitions of a machine-readable dictionary, or by some other representation. In psychological theories of semantics, the meanings may be stored in one of a number of ways that are described in further detail below.

Structural semantics refers to the way in which lexical meanings combine to produce complex semantic expressions. It has its origins in formal logic, and owes much to the writings of Aristotle and Frege. Montague [1970] argued that translation from natural language into a logical notation (such as predicate calculus) provides the basis of a semantic theory for that language, and that a precise method of translation could be determined and executed mechanically. Tarski [1931] proposed the notion of semantic truth for a formalised language, arguing that the purpose of structural semantics is to show how sentences come to have the truth values they do, given the meanings of the individual words and the way the syntax combines them. The majority of work on structural semantics has remained philosophical or at best highly theoretical, and has inspired few computational implementations.

2.2 Semantics and Psycholinguistic Theory

2.2.1 The Meaning of Words

Psycholinguists are concerned with two particular questions about word meaning:

(i) *How is knowledge about word meaning stored in the mind?*
(ii) *How is it accessed during the process of understanding?*

The first question concerns the issue of representation; the second that of processing. Considering the first issue, most psycholinguists support the existence of a mental lexicon that contains knowledge about words. The entries in the lexicon do not actually contain semantic information, but have instead pointers to locations in a

separate store known as *semantic memory* in which the meanings are held. The meanings of words can also be represented as textual definitions within an ordinary dictionary, so a third question may therefore be:

(iii) *How might semantic memory be related to dictionaries?*

This is an important question since (a) dictionaries are attempts by people to represent externally what they know about language; and (b) they are an existing source of information that has evolved over hundreds of years. Dictionaries define one word in terms of others, and this characteristic may be shared to some extent by semantic memory. However, dictionaries and semantic memories have different purposes, and this fact is reflected in the way in which words are interconnected in each. For example, dictionaries can also be used to define unknown words, relying on the assumption that the majority of words in the definition will already be known by the user. In so doing, they assume that the users of dictionaries are linguistically knowledgeable. Semantic memory cannot work in this way, since in itself it is partly responsible for performing this function.

2.2.2 Theories of Natural Language Semantics

Psycholinguists and AI researchers have both contributed toward theories of word meaning, and in recent decades a number of distinct theories have evolved. Psychological theories of language semantics are especially relevant, for two reasons: (a) the most successful language processing system to date is that of the human information processing system; and (b) such theories attempt to produce information-processing models of cognitive processes that should, in principle, be computationally implementable. In so doing, they should provide insight into the way these processes work.

The theories discussed below may have evolved from different assumptions, but it is still difficult to discriminate completely between them. The representational aspects of the six theories differ widely, but all suffer from the same problems of knowledge acquisition and inefficiency when implementations are attempted.

2.2.2.1 Feature Theories

Chomsky [1965] argued that word meanings can be accurately described by sets of bivalent features, which he called semantic markers, e.g. male, animate, human,

etc. Where hierarchical relations exist they are represented by redundancy rules, e.g. if a word meaning has the feature HUMAN, then it also has the feature ANIMATE. In this way, semantic markers decompose the meanings of words into more primitive elements. Katz and Fodor [1963] proposed that there was a universal set of markers that could represent the meaning of words from every possible language.

Attempts to perform automated semantic analysis by means of the selectional restrictions provided by semantic features have almost universally exposed the theory as being crude and inefficient. At the simplest level, semantic features may help identify correct word senses in sentences such as:

John hit the post with a ball

(where both "*hit*" and "*ball*" have two possible senses and feature lists), since only one combination of features is possible. However, given a sentence such as:

James hit John at the ball

it can be seen that the feature list for the "*dance*" sense of "*ball*" needs to be modified to include the act of "*hitting*"; and with it, everything else that can occur at such a function. This list would thus grow to considerable length. Furthermore, it can be seen that feature lists would be duplicated for words such as "*dance*", "*ball*", "*party*", etc., which is grossly inefficient, and would result in very complex feature lists for each word in the lexicon.

2.2.2.2 Semantic Networks

Semantic networks became established in psycholinguistics through the work of Collins and Quillian [1969]. In a semantic network, concepts, which refer to word meanings, are represented by nodes. The nodes are joined by a variety of links that represent the different relations between concepts such as set membership, set inclusion, part-whole, property attribution, etc. The meaning of a word is determined by its place in the network as a whole, the most important characteristic being the hierarchical organisation of the set inclusion links (usually known as "ISA" links). These hierarchies are most easily demonstrated by concrete nouns (e.g. collie ISA dog ISA animal, etc.)

It can be shown that semantic network representations are formally equivalent to semantic feature representations, in as much as any information that can be

represented in one can be represented in the other. However, semantic theories must consider not only how knowledge is represented but also how it is used, so these two approaches are treated separately.

Semantic networks inspired a number of natural language understanding programs during the early 1970's, with notable contributions from Schank [1972], Rumelhart [1972] and Anderson [1973]. The limitations of this representation soon became apparent; in particular its inability to deal with quantifiers. Hendrix [1979] made some progress regarding this problem, introducing a technique known as partitioning to deal with quantifiers. However, other difficulties remained: sentences that could not readily be decomposed into "SUBJECT and PREDICATE" form also proved troublesome.

The basic problem with semantic networks, as indeed with many other techniques, is that they are a language in which the meaning of sentences can be expressed - a language in need of its own semantics. However, some progress has been made concerning the knowledge acquisition problem: Amsler [1982], Calzolari [1984], Chodorow [1985] and Alshawi [1988] have all demonstrated the use of a machine-readable dictionary in the automatic construction of semantic relations and networks.

2.2.2.3 Case Frames

An alternative approach is to define words according to the sentence contexts in which they occur. For example, Fillmore [1968] proposed the notion of a case grammar, in which each sentence was analysed into the cases attached to the verb:

Agent: animate being initiating action;
Instrument: inanimate entity involved in the action;
Recipient: animate being affected by the action;
Object: inanimate entity affected by the action;
Locative: location or direction of the action.
etc.

The assignment of these categories need not necessarily follow the grammatical assignments (e.g., the object case need not be the syntactic object). Using this technique, the lexicon defines each verb according to the cases it can take. The main difference between this method and that of semantic features is the level of detail they

specify. For a verb such as "*collide*", all that is specified by the case restrictions is that the object case can be any inanimate entity. Thus these case roles could be filled with nonsensical objects such as "*sincerity*" or "*steam*"; i.e., one could say "*steam collided with sincerity*". The case roles have been modified since Fillmore's original definition to include restrictions such as these, but the restriction lists grow to an interminable length and the technique is still unable to recognise that two expressions (e.g. "*the woman*" and "*she*") may refer to the same individual.

2.2.2.4 Prototypes

The theory of prototypes owes much of its origins to Wittgenstein's ideas about word meaning [1953], and its development to the work of Rosch [1975]. Wittgenstein argued that most words could *not* be defined in terms of lists of necessary and sufficient conditions for membership of the set for which they stood (his most noted example is that of the word "*game*"). Furthermore, network theories and feature theories fail to explain two important facts:

(a) the correct classification of an object may be in doubt when its features are not;
(b) some examples of a concept are more typical than others, and are more easily brought to mind.

Prototype theory sees entries in the mental dictionary as being centred on a representation of the prototypical member of the class to which the word belongs; e.g. a robin may be seen as the prototypical bird. A prototype is located in a multi-dimensional space with dimensions corresponding to the characteristics on which examples of the concept can vary. Boundary spaces can be drawn around the prototype that demonstrate the extent of the definition. To give meaning in prototype theory is to determine how far something can differ from the prototype and still be a member of the class.

Minsky's frame system [1975] is a computational implementation of prototype theory. Each concept is represented by a frame, which contains slots that may be filled differently for separate instances of the same concept. These slots may have default values that represent the characteristics of the prototype. The boundaries of the concept are implicit in the frame structure and in the constraints on the values of the slot fillers. As with the scripts mentioned above, attempts to represent semantic knowledge in frames have suffered from the problems of relevance and detail, along with the usual acquisition problem.

2.2.2.5 Semantic Primitives

Some researchers have attempted to capture the core meaning of words by decomposing them into a small set of "building blocks" known as semantic primitives [Wilks, 1973]. Schank [1972] drew up a list of 12-15 primitive actions that he claims underlie the meaning of all active verbs, of which a number are listed below:

ATRANS: transfer of possession
MTRANS: transfer of mental information
PTRANS: physical transfer from one location to another
MBUILD: build memory structures
ATTEND: sensory input

Schank argued that every verb in the lexicon could be expressed by a combination of these primitives, e.g. "*give*" can be *ATRANS* or in the case of "*giving advice*" it can be decomposed to *MTRANS*. So instead of specifying case frames (or indeed semantic features) for each individual word, Schank only needs to provide case frames for the primitives. Moreover, verbs that involve the same primitive automatically have the same case frame, eliminating the duplication of effort seen with the other approaches. However, this reduction of different verbs to the same primitive does have its disadvantages, as information may be lost (as in the case of "*joke*", "*say*" and "*preach*" all being reduced to *MTRANS*). Furthermore, systems based on semantic primitives also suffer from the usual acquisition problem.

Schank used the notion of semantic primitives as the basis for a number of semantic analysis programs, which took as their input natural language text and gave a semantic representation composed of primitives as their output. Typically, the program would look at the words from left to right, and test whether each word in the sentence was a likely candidate for the case slots of the main verb. If later words in the sentence suggested a different categorisation of the main verb, then re-interpretation of the sentence was possible with re-allocation of the case roles. In many ways, this type of semantic "parsing" had all the abilities of a syntactic parser, plus the ability to allocate case roles according to semantic information extracted from the lexicon (e.g., an object such as "*book*" cannot fill the Agent case role). The semantic analyser would then link each of the primitives into a structure known as a *conceptual dependency* network, which represented the causal relations between actions and states.

2.2.2.6 Meaning Postulates

A meaning postulate is a formula expressing some aspect of the sense of a predicate [Hurford, 1983], using a predicate-calculus-like notation that permits any number of arguments. Bar-Hillel [1967] argues for the superiority of meaning postulates over semantic markers due to their ability to represent arguments of lexical items, which are essential for expressing the relation between the meanings of words like "*buy*" and "*sell*":

for any x, y, z (x sells y to z if and only if z buys y from x)

Strictly speaking, theories involving meaning postulates have their origins in formal semantics rather than psycholinguistics. Although meaning postulates have been used by some psychologists to represent semantic relations, the theory has inspired little empirical research or computational implementation. This is mainly due their inability to adequately represent certain phenomena; for example:

- temporal relations, i.e. the time at which a predicate applies. This requires the development of a more elaborate logical framework;
- gradable predicates such as "*tall*", *short*", "*large*", "*small*", etc. These words do not have an *absolute* meaning, since it varies according to context. Meaning postulates are designed to account for truths that hold in *all* contexts, and are therefore less able to adequately represent such phenomena.

2.3 Semantic Processing

2.3.1 Requirements of the Present Project

The needs of the present project could be defined as "*to use the semantic constraints inherent in natural language to reduce the ambiguity in the output from a text recognition system*". It can be argued that the successful development of such techniques requires an adherence to semantic theory from both the computational and linguistic perspective, to provide a sound theoretical framework. However, during the present project the limitations of the established semantic theories have become apparent. Indeed, the techniques that have proved to be of greatest use are empirical or almost "trial and error" in their approach. To use the linguist's terminology, they would be referred to as *weak methods*, to reflect their lack of theoretical rigour.

However, there is another important reason why the established semantic theories are less relevant: they pursue the goal of *understanding* rather than *recognition*. Understanding requires the determination of the meaning of a sentence. This usually implies translation into a formal language with a simpler semantics, and, using Tarski's definition, the determination of its truth conditions. Recognition, however, does not necessarily require such processing (although under certain circumstances it may be desirable). Instead, it may be sufficient merely to apply semantic constraints to assess the plausibility of a particular combination of words in the input.

Criteria such as "*theoretical integrity*" and "*psychological plausibility*" dictate that any adopted techniques should be based on established semantic theory. However, it may nevertheless be possible to "simulate" the processes described by semantic theory using techniques based solely on empirical results. This would be described, using the linguist's terminology again, as using "weak" methods to achieve "strong" results. After all, it could be argued that the purpose of a semantic analyser used within a text recognition system is not to demonstrate the plausibility of a particular semantic theory, but to simulate the output of a human reader. Put another way, the process of human semantic processing as described by psycholinguistic theories may be simulated by programs that bear little resemblance to any established theory of linguistic semantics.

Published literature on the role of semantic processing within computerised text recognition is sparse. The majority of research has tended to focus on the pattern recognition level, with the higher level processes being progressively of lesser interest. However, a number of researchers have investigated semantic processing applied to related NLP problems, and some of the techniques and resources used have shown direct relevance to text recognition. In particular, dictionary definitions and co-occurrence statistics have been identified as valuable sources of semantic information. The acquisition and use of such information is discussed in the following sections.

2.3.2 The Acquisition of Semantic Information

2.3.2.1 From Machine-Readable Dictionaries

Many researchers are currently engaged in the processing of dictionaries in order to extract semantic information and reconstruct it in an alternative (more accessible) form; the objective often being the creation of lexicons for large-scale natural language processing. This trend towards the construction of lexicons using machine-readable dictionaries (MRDs) follows the realisation by many that a major restriction on the functionality of many NLP systems is the small size of their lexicons [Alshawi, 1988]. Zernik [1989] has identified a number of shortcomings with existing lexicons in terms of "lexical gaps"; such as:

Single words - where entries are missing from the lexicon;
Compound words - e.g., "*respective*" cannot be processed as "*respect*" + "*ive*";
Word senses - which vary according to topic or context;
Collocations - e.g. "*strong*" and "*powerful*" may be similar, but we cannot talk of "*a strong car*" or "*powerful tea*";
Idioms and phrases - of which the meaning is not a simple product of the constituents;
Metaphors - of which the interpretation is not literal;
Others including prepositions, noun group compounds, individual constraints, synonyms, etc.

Alshawi's [1988] analysis of the sense definitions in LDOCE attempts to provide sufficient semantic information to enable an NLP system to cope with unknown words. The process starts with a hand-coded classification of the core vocabulary, and then the propagation of these structures throughout the dictionary so that all sense definitions are included. In the case of processing nouns, this process involves the location of the semantic head (superordinate term), and the exploitation of other information present (e.g. modifiers and predications). The structures so produced have some properties of a linguistic analysis of the definitions and some properties of a semantic definition of word sense concepts, and they take the basic syntactic form of nested feature lists. Although the performance of the system seems respectable (correct semantic head located for 77% of the definitions, additional information recovered for 61% of definitions, of which 88% was correct), there are

still problems associated with the processing of idioms, phrasal verbs, circular definitions and cross references.

Vossen, Meijs and den Broeder [1988] have carried out similar studies of the meaning and structure in dictionary definitions, based on Dik's [1987] "stepwise lexical decomposition", which reduces the meaning of lexical items to a restricted set of basic lexical items. This procedure involves firstly the application of a grammar code to all the words in the core vocabulary (of the LDOCE) and their inflections. These codes are then inserted in each of the sense definitions to create a corpus of coded definitions. This is followed by the development of a syntactic typology for the meaning descriptions, and the subsequent creation of parser-grammars for each part of speech. These grammars can then be applied to the coded corpus, with the intention of identifying the premodifiers, kernel and postmodifiers of each definition. Finally, the development of a semantic typology allows the identification of horizontal and vertical links between words, the tracing of hyponyms and hypernyms of given words, and the identification of the properties of premodifiers and postmodifiers. Further investigation of the semantic typology has identified four distinct structural patterns within the meaning definitions of nouns:

(1) Links - in which the syntactic kernel is semantically a hypernym of the entry word, with pre- and post-modifiers expressing restrictions on the extension of the hypernym;
(2) Synonyms - all semantic information is expressed by just one word (no need for modifiers or other restrictions);
(3) Linkers - in which the kernel is somewhat meaningless and most of the semantic weight is carried by another part of the meaning description;
(4) Shunters - in which the interpretation is "shunted" from a nominal structure to a non-nominal structure (e.g. a verb phrase).

Further work on MRDs includes Guo's [1989] attempts to build a machine tractable dictionary (MTD) from the LDOCE, based on the fact that a set of 1,200 words (known as the Key Defining Vocabulary or KDV) is found to define the 2,219 words of the core vocabulary of the LDOCE. It was proposed that the entire LDOCE vocabulary could be defined by the KDV by a series of four "defining cycles" that progressively add more of the core vocabulary to the KDV, until after three cycles all the core vocabulary is accounted for and the fourth cycle defines the remaining 27,758 headwords. The knowledge structures used to represent the dictionary entries

are known as integrated semantic units, or ISUs. These structures can be regarded as the semantic primitives of the MTD, incorporating linguistic knowledge with general world knowledge in the representation of each word sense. The set of primitives that best suits a particular MRD can be found empirically, and an average of three basic senses of 1,200 KDV words requires the hand-crafting of 3,600 ISUs. An initial attempt at hand-crafting a small set has been successful, and alternative approaches to pure hand-crafting are currently being investigated.

Although the extraction of semantic information from dictionary definitions may be considered an objective in itself, many researchers pursue a specific application, or at least have a range of applications in mind. For Jensen and Binot [1988], that application was to create a knowledge source for syntactic disambiguation. In particular, they addressed the issue of the attachment of prepositional phrases and relative clauses, but also considered anaphoric reference and the interpretation of dangling modifiers. It was their intention that the natural language of the dictionary definitions should be used as the knowledge representation language, eliminating the need for hand-coding. For example, given the sentences:

I ate a fish with a fork

I ate a fish with bones

it can be seen that the prepositional phrase can have two sources of attachment. The objective, in each case, is to determine that source of attachment. In the first example, the system compares the link between "*ate*" and "*fork*" with that of "*fish*" and "*fork*". It finds that according to the dictionary definitions there is a link between "*ate*" and "*fork*" in that they both have the phrase "*taking up*" in common. The absence of any link between "*fish*" and "*fork*" confirms the choice that the prepositional phrase "*with a fork*" should be attached to "*ate*". In the second example, the system compares the connection between "*ate*" and "*bones*" with that of "*fish*" and "*bones*", finding a connection in the latter case through the word "*vertebrate*", and the prepositional attachment is chosen accordingly. A similar technique is used to resolve anaphoric references and similar syntactic ambiguities.

Chodorow, Byrd and Heidorn [1985] have investigated the use of MRDs in the construction of semantic hierarchies, based on the assumption that all noun definitions have "genus" and "differentiae" terms. The genus extraction is performed by a limited form of parsing, usually involving the identification of the head of the defining

phrase. From this data, two types of hierarchy can be created, the processes involved being known as "*sprouting*" and "*filtering*". The sprouting process involves the creation of a semantic tree from a specified root, organising the results of the head-finding into a "hyponym index". For example, "*vehicle*" may have as an entry:

vehicle: ambulance...bicycle...car...tanker...

Although all the words have at least one sense bearing the property for which the root was originally selected, it must be noted that to locate all words bearing a particular semantic feature must involve the careful selection of several roots (e.g. to find nouns with a [+female] feature, sprouts should begin from "female", "woman", "girl" and possibly "wife").

The filtering process also produces lists of words bearing a certain feature, but only those words of which all the senses have the feature. It is created using a "hypernym index", in which each word is listed with its hypernyms. The data so produced may be of use to parsing systems, whenever it becomes necessary to know whether a noun *must* have a certain feature, not merely that it *may* have it. Byrd et al [1987] have extended this work, complementing the above two processes with other lexicographic tools and methodologies, such as Head Finding (a method for automatically discovering hypernyms of words); Matrix Building (for clustering synonyms into senses and analysing senses in bilingual dictionaries); TUPLES (a system for finding frequent words and phrases) and others.

Markowitz [1986] investigated the creation of large lexicons for NLP using semantically significant patterns in Webster's Seventh New Collegiate Dictionary (W7). She saw the problem as that of making the information implicit in a dictionary explicit, by finding taxonomies, set membership, recognising human nouns, etc. For example, noun definitions that begin with the word "*Any*" usually indicate a taxonomic relationship between the noun being defined and the word following the word "*Any*". Similarly, definitions beginning with "*A member of ...*" indicate a member-set relation; usually human. Generic agents were signalled by the sequence "*One that...*" and human nouns were often identified by the suffices "*-er*", "*-ant*", etc.

2.3.2.2 From Co-occurrence Statistics

A technique for describing text types based on statistical data has been investigated by Huizhong [1986]. His use of frequency of occurrence and distribution

data to identify scientific and technical terms suggests that it is possible to characterise text types according to the collocational behaviour of these terms. Three criteria for text characterisation have been identified: firstly, that of subject matter; secondly, that of genre (for whom the texts are written); and thirdly, that of topic-type (information flow and concept structure). Preliminary results have shown good differentiation between science texts and general texts; although to what level of detail this can be applied (i.e., identification of topic?) remains unspecified.

Plate [1989] has investigated the use of co-occurrence statistics obtained from LDOCE. He restricted his investigation to the 2,200 or so words in the LDOCE core vocabulary, and took the sense definitions as the textual units over which to collect co-occurrence data. The data obtained forms the triangle of a 2,200 by 2,200 matrix (requiring 4.7Mb disk space), and as such required further processing to be rendered comprehensible (in this case, the application of programs called PATHFINDER and BROWSE). Tests comparing concept relatedness based on co-occurrence data with that of human judgement showed a strong correlation. Co-occurrence data has been used by the present project; the data having been collected from a number of corpora rather than LDOCE. Such free-text sources were identified as being more representative of the type of text that the system would eventually have to recognise. An alternative representational structure was chosen to reduce the need for extended processing and memory overheads. This information has been shown to make a significant contribution to the recognition process, and is described in detail in Chapter Four.

Fraenkel [1980] has investigated the semi-automatic construction of "semantic concordances", in which homographs are distributed into disjoint classes with one semantic value per class. The technique involves the partitioning of words in the text into classes based on certain similarities, and then further partitioning based on small word contexts. An editor can then check one representative word from each small class as to its correct meaning in context, and this meaning can then be assigned to all other words in this class.

Several corpus researchers (e.g. Hughes & Atwell [1993], Finch & Chater [1991], Atwell & Drakos [1987]) have investigated clustering techniques to *automatically* learn a word classification set from a training corpus, using word co-occurrence patterns. It transpires that the clusters so derived tend to reflect both

syntactic and semantic constraints: small, closed function classes (e.g. prepositions) are syntactic, but nouns and verbs generally cluster on semantic grounds.

Other researchers [McKinnon, 1975] have applied the statistical techniques of cluster analysis to samples of language that conform to regular patterns within a specific subject domain (known as "*sublanguages*"). Sager [1981] has identified a number of sublanguages that are sufficiently regular for this technique, and provides a methodology for the reliable identification of their constituent semantic classes. These semantic classes can then be used to define a *sublanguage semantic grammar*, which constitutes a further level of semantic constraint on the text. She does, however, point out that these techniques are only suitable for text that displays the specialised sublanguage characteristics.

Smadja [1989] used co-occurrence data as an aid to language generation. He observed that there are certain classes of English word combinations that neither syntax nor semantics can justify; for example, although the words "*strong*" and "*powerful*" may have a similar meaning, people prefer saying "*drink strong tea*" to "*powerful tea*" and similarly prefer "*drive a powerful car*" to "*strong car*". Smadja argues that knowledge of relations such as these is necessary to both understanding and generation, and he outlines an approach for automatically acquiring such restrictions from a corpus, and then using it to augment an existing lexicon. He also makes the distinction between *lexical* and *conceptual* collocations; the latter being word pairs that co-occur simply because they are associated to the same context or topic (e.g., "*bomb*" and "*soldier*", "*trouble*" and "*problem*", etc.). Although Smadja suggests no specific use for conceptual co-occurrences in his lexicon-building research, it is shown in Chapter Four that both types are of value to the current project.

Choueka [1988] has designed an algorithm for locating collocational expressions in corpora that does not include any morphological or syntactic component and does not require any dictionary lookup. His algorithm was applied to the New York Times News Wire Service and produced various lists of collocational expressions from length two to six. Lancashire [1987] also gives details of such an algorithm, and this has been adapted to meet the needs of the present project (see Chapter Four).

2.3.3 The Use of Semantic Information

Semantic information may be used for a variety of linguistic purposes. A good example is that of word sense disambiguation, since it is relevant to many NLP applications. Word sense disambiguation may be seen as a knowledge-intensive problem. Jacobs [1989] has identified some of the knowledge sources contributing to sense discrimination as:

Morphology - senses cannot always be derived by affix-stripping, e.g. "*conductivity*" derives from "*conduct*", but corresponds only to the electrical sense (one would not talk of the "*conductivity*" of an orchestra);
Word frequency - some word senses only occur in specific expressions;
Topic - word senses vary according to subject area or domain;
Word senses from a dictionary - generally, dictionaries have more senses than necessary for broad applications but too few for specific applications;
Collocations - some word senses appear only when used in particular idioms or collocations; Intersections - e.g. if "*conduct*" and "*violin*" appear together, this would suggest a different sense of "*conduct*" than if it appeared with "*wire*";
Semantic preferences - often more structured than collocations and intersections, e.g. "*the sun rises*";
Syntactic information - e.g. the word "*conduct*" in "*John's conduct at his violin lesson*" invokes the sense of behaviour due to its status as a noun, despite other connections between "*conduct*" and "*violin*".

Lesk [1986] described dictionary-based techniques for determining the senses of words used in text and choosing the most appropriate one according to the sentential context. The correct sense is chosen by accessing the word's sense definitions within the OED, then counting the words that each sense definition has in common with the definitions of the other words in the immediate context. The sense definition with the highest commonality is the one chosen. This technique has been applied extensively by the present project to the problem of discriminating between different candidate words in a single sentential position (rather than different senses of the same word). It has been shown to make a significant contribution to text recognition, and the results are described in Chapter Three.

There are a number of variations on this basic theme. Demetriou [1993] investigated word sense disambiguation using the LDOCE rather than the OED.

LDOCE has been compiled using a defining vocabulary of some 2,000 words, and it is argued that this core vocabulary constitutes a set of semantic primitives that may be used to embody semantic constraints. Guthrie et al [1991] also investigated word-sense disambiguation but based their work on subject-dependent co-occurrence information, exploiting the subject classification codes of LDOCE. They produced subject-specific "neighbourhoods" for each word within the LDOCE, and intersected these with words in the test data to determine the correct sense. Guthrie [1993] states that the best success rates for word-sense disambiguation are around 70% in all experiments except other than Kelly and Stone [1975], who used extensive hand-coding and tuning to achieve 90% success rates.

Kelly and Stone restricted their sense determination research to that of a KWIC concordance of 1,815 types taken from a 510,976-token corpus. Their objective was to use the concordance and dictionary definitions to produce an ordered set of disambiguation rules that would determine the correct sense of a word by testing for both part of speech and membership of sixteen specific semantic categories. The results of each test would then either assign a specific sense to the word or activate other rules within the set.

Black [1988] has described a number of methods for discriminating English word senses, again by examining the sentential context. The first method, after Debili [1982], uses a listing of word pairs observed to have entered into certain syntactic relations in a previously analysed corpus (e.g., subject/main verb or noun/adjectival modifier, etc.). Word pairs on this list are given a "validity score" of 1, and those not on it are given 0. When the program finds a word with multiple senses, it produces a list of all the synonyms of each of the senses of that word, which are then known as "word families". The program then chooses a neighbouring word (the sense of which is unambiguous) and determines the syntactic relation between this word and the word in question. The maximum is now calculated of the products of the validity scores of the word families and the unambiguous word, and the word family yielding this maximum determines the sense chosen. This technique may be seen as an extension of the basic co-occurrence methods, as it includes the concept of matching using a group of semantic relatives (rather than just the word itself) and also matches according to specific syntactic relations.

The second method, after Gross [1985], uses a lexicon-grammar, in the form of a two-dimensional matrix. Columns of this matrix are labelled with possible syntactic

properties of each entry word and semantic properties, e.g., "*concrete*", "*animate*", etc. Candidate word senses are determined by inspecting the context of a given word and eliminating those words whose conditions of usage (according to the matrix) are not met. This technique may be seen as an extension of the established corpus-based parsing techniques to include a limited coverage of semantic features.

Sinclair [1970] used the collocation patterns extracted from the 7.3 million-word Birmingham Collection of English Text [Renouf, 1987] to partition a set of tokens of one word-type into separate senses, which could then be used to guide the word sense disambiguation process. Moreover, this general technique was then applied on a large-scale project to build the Collins COBUILD dictionary on empirical grounds, with the sense distinctions based on explicit collocational patterns [Sinclair, 1987]. Indeed, this dictionary is of particular relevance, and for a number of reasons. Firstly, it shows how collocations can be used to compile dictionary definitions, which contrasts with the work of others (e.g. Guthrie et al [1991], Plate [1989]) whereby definitions are used to compile collocations. Secondly, as Chapter Three will illustrate, the use of examples in dictionary definitions is particularly important in text recognition, and it would be interesting to see how a machine-readable version of COBUILD would fare compared to other dictionaries such as LDOCE or OALD.

Amsler and Walker [1986] have used the subject categories (referred to as domain codes) present in the LDOCE as a source of data for sense disambiguation. Each word within a paragraph is assigned all its possible subject categories, and the category most frequently represented over the whole paragraph is deemed to be the subject area of the text. The senses of each word within the paragraph carrying this domain code are then selected as being correct, in favour of alternative senses not bearing this code. This technique is significant for two reasons - not only does it aid the process of word sense disambiguation, it also provides a method whereby the topic of the text can be identified. These codes, along with a further set derived using an original corpus-based method, are investigated in Chapter Six. Slator [1989] also used the domain codes within the LDOCE for the same purpose, but only after having restructured the hierarchy of the coding system. He suggests that this restructuring gives a better intuitive ordering of the important concepts in each text, and enables a knowledge-based and context dependent strategy for making word sense selections.

Dahlgren [1986] approaches the problem of sense discrimination using an ordered set of categorial rules that are applied in sequence to the text to be discriminated. The first type of rule uses data on concordances, referred to as "frequent collocates". The second type of rule is based on syntax, and checks for dependency relations such as (in the case of a noun) the presence of an associated definite article, personal pronoun, etc. The third sort of rule uses "common-sense knowledge" as defined by a number of psycholinguistic studies, and represented by a "tangled hierarchy" of ontological predicates. These rules test for similarity between the word in focus and its neighbours, the highest similarity indicating the correct sense.

Black [1988] compared experimentally three techniques for sense discrimination: a domain general method and two domain specific methods. Five experimental words were chosen; four of which had four senses and the remaining word three. About 2,000 concordance lines were obtained for each test word, taken from a 22 million-token corpus. Each experimental method consisted of a set of 81 "contextual categories"; such that the context of a word was represented by the pattern of presences or absences of each of the 81 categories within each concordance line. The domain general method (henceforth known as "DG") was based on Amsler and Walker's subject category approach, using the domain codes in the LDOCE. Each of the 500 most frequently appearing words in the 2,000 concordance lines was analysed with respect to their definitions, to produce a profile of the concordances from the point of view of the domain codes in LDOCE. The requisite 81 categories were then derived from this profile.

The first domain-specific method ("DS1") was based on the frequencies of different lexical items in a training corpus of 1,500+ concordances. Two classes of category were identified: the first consisted of the 41 types occurring most frequently in a window of + or - 2 word positions, to capture those words in close grammatical construction with the node. The second class consisted of the 40 most frequent words excluding function words and extending over the entire concordance line, to represent collocates of the node. The second domain-specific method (DS2) resembles the first in that 20 of its 81 categories were one or two-word sequences occurring most frequently in a window of + or - 1 and 2 word positions, respectively. The remaining 61 were derived from the concordances of 100 randomly chosen types occurring in the corpus.

In the experiment it was found that all three methods produced results that were better than random selection - but the domain specific methods were far superior to the domain-general method (DG was 27% better than chance, the DS methods both roughly twice as good as chance). It was suggested that the DG method failed to represent the thematic organisation of the concordances analysed, as the categories that might have been chosen on an "intuitive" basis do not seem to attract any of the 500 most frequently appearing words of a test item. Concerning the usefulness of the three techniques, it may be noted that all suffer from the necessity for hand-labelling of concordances, which remains a task of considerable size. Whilst the automation of this process remains only a distant possibility, it must be conceded that none of these techniques is directly usable as described.

Other researchers, concerned with word sense disambiguation for the purposes of thesaurus creation, have also used the overlap technique. For example, Byrd [1989] solves the problem of matching word senses in different thesauri by computing a set of "sense property vectors" for each entry from the two thesauri, and then computing an "optimal" mapping of the two sets. This "optimal" mapping is based on maximising the overlap between the sense property vectors (which in this case take the form of synonym lists). Byrd has also investigated the use of the overlap process to map between dictionaries, matching word sense definitions. He describes his results as showing "limited" success; with fewer than 50% of the mappings being correct. Two types of failure were observed; with incorrect mappings being proposed in some cases, and correct mappings being missed in others. Byrd concludes that lists of undifferentiated definition words are not selective enough for adequate mappings, but the overall plan shows promise. He suggests that assignment of specific sense properties in the definitions and then matching on a property-by-property basis may offer the greater degree of constraint necessary.

The identification of the "theme" or subject area of a passage of text can provide valuable information concerning the likely semantic content of that passage. In applications where the domain is restricted or known in advance, semantic knowledge structures can be used to constrain the range of words that are statistically likely to occur within that subject area. Grosz [1986] has demonstrated the use of semantic networks as a knowledge source to aid knowledge-based discourse processing. Alternatively, the work of Critz [1982] has demonstrated the possibility of automatic theme recognition used frame-based representations. His system determines the thematic continuity of English texts by first parsing to find the head

noun, which is chosen initially to represent the theme, and declared as the "theme noun". Then a list of frames indexed by that theme noun is searched to associate it with any frames currently active (i.e., those indexed by previous theme nouns). If no direct association is found, interpretive rules are applied to attempt an indirect association, through one of the frames normally associated with the object but not yet associated with the text.

The work of Walker and Amsler [1986] on word sense discrimination (described above) involves the use of subject codes in LDOCE to identify the topic of a piece of text. A similar process of topic identification can contribute to the present project, by reinforcing the choice of words whose senses contain subject codes that have been identified as being representative of the overall text. For static recognition, this could be achieved pre-processing the whole text to determine the topic. For dynamic recognition, processing may proceed from left to right through the text with subject codes of new words being compared to a "running profile" of subject codes taken from previous words. These techniques are discussed in greater detail in Chapter Six.

2.4 Current Resources

2.4.1 Machine-Readable Dictionaries

It was mentioned earlier that MRDs were designed for human rather than computational use (see Table 1.1). In many cases, they are provided as little more than typesetting tapes, in need of considerable "cleaning up" or normalisation. This involves the removal of typesetting codes, removal of errors, and the identification and labelling of all the information such that it can be easily accessed without detailed knowledge of the formatting conventions [Byrd et al, 1987]. Consequently, it is possible to identify types of information that are needed by language processing systems but are either absent from or wrongly presented within the dictionary. Krovetz [1987] has identified four such types:

Sense frequency information - which can give preference to one sense when other factors are equal; and can be biased according to sublanguage/domain;

Collocation information - the words that a word sense co-occurs with, along with an indication of frequency (N.B. an exception to this is the COBUILD dictionary, which provides many such examples);
Proper nouns - a greater coverage of these is needed;
Semantic class - the grouping of words according to similar semantic behaviour offers potential for predicting semantic roles based on their classification.

Braden-Harder and Zadrozny [1989] have identified what they refer to as a "wish-list" for MRD organisation; in terms of enhancements to existing information and additional information:

Short hierarchical definitions - i.e. collections of short sentences, listing the more important facts first (it is easier to parse short sentences, and controlling the "spread of activation" of background knowledge is easier);
Cross reference - which requires the disambiguation of word senses in the definitions to control the spread of activation caused by cross-referencing;
Synonyms and Antonyms - which should be easily accessible from a given entry;
Example sentences - which are a good source for typical subjects and objects; compressed versions are difficult to parse and therefore best avoided;
Information from bilingual dictionaries - which may enhance the lexicon. For example, typical prepositions may be included in verb definitions.
Preference/frequency - ordering the word senses according to frequency is desirable, and the inclusion of co-occurrence information shows promise;
Combining multiple dictionaries - to solve the problem of mapping sense information from one MRD to another.

It must be noted that the use of MRDs is not a panacea to the problem of lexicon construction. They were constructed for human interpretation and they are thus designed to provide verbal explanations and translations of words, rather than the morphological, syntactic and semantic data required by NLP lexicons. McNaught [1988] has commented:

> *"...publisher's MRDs have the wrong form, and the wrong content, and while some ad hoc programming may achieve rapid partial results...further study or exploitation of existing MRDs will lead to diminishing returns."*

Jacobs [1989] adds the deficiencies of circularity of definitions and obsolescence to the above; and argues that MRDs are only a piece of the solution to

the lexical acquisition problem. Furthermore, he observes that most NLP systems ignore a major source of knowledge, i.e. that of the input itself. Instead of maintaining the general domain of a text topic and preserving partial results to form hypotheses about new words and meanings, most natural language programs preserve little or nothing from one sentence to the next.

Amsler [1989] has drawn attention to the inadequacies of MRDs in the context of developing lexical knowledge bases for NLP. He describes a comparison between the word forms found in a sample of text taken from the New York Times Newswire Service and those listed as entries in the Merriam-Webster Seventh Collegiate Dictionary, which showed that 64% of the words in the text were not in the dictionary. Of these omissions, one quarter were inflected forms, one quarter were proper nouns, one sixth were hyphenated forms, one twelfth were misspellings and one quarter were not yet resolved, but were likely to be new words occurring since the dictionary was published. Amsler also draws attention to the assumptions made by early lexical knowledge bases, in particular the notion that a word is a contiguous sequence of alphabetic characters. Amsler shows that this notion ignores important classes of words such as open nominal compounds, phrasal verbs and idioms. He states that NLP systems "*...lacked a complete lexicon of the language they were attempting to manipulate intelligently and had no rules for understanding how to recognise these lexical concepts when they appeared in text.*" Furthermore, he argues that proper-nouns have a grammatical structure, and that the compositional rules for such compounds will have to be written into NLP systems.

By contrast, Sampson [1989] investigated a derivative of the OALD by running it over some 50,000 words from the LOB Corpus, and found the coverage to be surprisingly high: of the 45,622 tokens he believed should be handled by the dictionary, 43,490 were found as they stood. When minor morphological changes were made (e.g. hyphen removal, etc.) there remained only 1,477 word tokens not found in the dictionary: that is, 3.24% of the target domain. He identified the largest category of omitted types to be (predictably) proper names, but noted also a bias against technical vocabulary and derived forms with negative meanings. He concludes that "*...a modest standard dictionary ... is remarkably successful at covering the vocabulary of ordinary printed documents*".

Ahlswede and Evens [1988] have investigated techniques for extracting information from MRDs using the concept of a defining formula. Defining formulas

are certain words or phrases that are frequently used in definitions, such as "*the quality or state of being*" or "*of or relating to*", and these have been used to identify a variety of lexical-semantic relations in dictionary definitions. In particular they demonstrate how definitions can be used to generate word-relation-word triples that are then used as data to build the lexicon.

2.4.2 Text Corpora

A corpus is a body of text or speech that provides a representative sample of a language. Text corpora provide empirical data concerning language usage, and as such may be used in the design and testing of NLP systems. However, they are of limited use in their raw state - they must be statistically analysed to provide meaningful information, such as word frequency data or collocations. Yet for many years the idea of using probabilistic information within an NLP system was viewed with some disdain by the linguistic community. Many linguists felt that since corpora were finite and degenerate they were unable to deal with many of the phenomena present in language, and could offer no insight to the "real" question of how people process language. Stylistic analysis was one of the few tasks for which such statistical information was deemed appropriate [Ellegard, 1962]. Furthermore, the use of text corpora was hampered by severe practical difficulties: computers were rare and expensive, and the only method of acquisition was manual input. Consequently, text corpora were limited both in size and availability.

However, in recent years the processing power and storage capacity of computers have increased dramatically, and many more textual resources are now available in electronic form. These developments have fostered a proliferation of corpus compilation projects, with some of the more recent ones having target sizes set at 100 million words, e.g.:

- The Lancaster-Oslo-Bergen Corpus [Johansson, 1980] - 1 million words;
- The COBUILD Corpus [Sinclair, 1987] - 20 million words;
- The Longman/Lancaster Corpus [Crowdy, 1992] - 30 million words;
- The TEI Corpus [Walker, 1989] - 100 million words;
- The British National Corpus [Leech, 1993] - 100 million words.

Evidently, corpora have increased in size as resources have expanded and techniques become more refined. This is a reflection of the fact that there is a huge

imbalance in the frequency of words in the English language, and large corpora are needed to provide adequate coverage of low frequency words. However, size is not the only important factor: for a corpus to be truly representative of a language it must also be "balanced". There are a number of variables against which a language can vary, e.g. time-span, geographical origin, gender of author, discourse type, subject area, etc., and the sources from which the corpus is compiled must be selected carefully to maintain a representative balance. Additionally, some corpora are available in annotated form (e.g. the Tagged LOB Corpus [Johansson et al 1986]), but the annotations to date have been mainly syntactic rather than semantic. However, some "parsing schemes" used in the annotation process have a semantic orientation (e.g. the Systemic-Functional Grammar parse-trees in the Polytechnic of Wales Corpus [Souter, 1990]) and more recently attempts have been made at semantic tagging (e.g. Jost & Atwell [1993]).

In general, theoretical semantics offers no large-scale reusable resources that can be applied to text recognition. Consequently, MRDs and text corpora remain the prime source of semantic information. However, there is one possible exception: the WordNet semantic network [Miller, 1985]. This system has been built "from linguistic intuition" and aims to be a large-scale general-purpose semantic network. It is conceivable that it could be used to apply semantic constraints on the text recognition data, and to thereby identify the correct words from alternative candidates. This is suggested as an area for further research.

2.5 Semantics and Text Recognition Systems

The most successful text recognition system to date is that of the human information processing system. This is largely because human readers use an *understanding* of the text that can guide the reading process. Word images occur within a meaningful context, and human readers are able to exploit the syntactic and semantic constraints of the textual material [Rayner, 1983]. Indeed, it is argued that the conspicuous gap between the reading performance of people and that of algorithms may reflect the fact that few text recognition systems utilise the many knowledge sources or recognition strategy of the human reader [Hull, 1987].

Handwriting is inherently more ambiguous than printed text, and consequently the role that higher level knowledge plays in its recognition is particularly significant.

It would be reasonable to assume, therefore, that particular attention would be given to the incorporation of such knowledge in the development of handwriting recognition systems. However, almost exclusively, early systems have made little or no attempt to use any such knowledge beyond that of the word level (e.g. Earnest [1962], Eden [1964], Sayre [1973], Tappert [1984]). Indeed, a recent and very comprehensive survey of handwriting recognition techniques and systems considered its significance to merit just one sentence: "*Higher level linguistic rules such as syntax and semantics can also increase the recognition rate*" [Tappert et al, 1990].

Even if the argument concerning *whether* semantic knowledge should be incorporated is upheld, there still remains the question of *how* it should be incorporated. Ultimately, the issue of *integration* within a complete system architecture will have to be addressed. There are those who argue that psychologically plausible NLP systems cannot be constructed by conjoining various knowledge-specific modules in series or hierarchically; they must instead be massively parallel and strongly interactive. Some systems have addressed this problem using a *blackboard approach* (e.g. HEARSAY, [Erman, 1975]) in which a neutral working area is set up for components to store the results of their analysis. This approach has experienced some degree of success [Erman, 1980]. The work of Waltz and Pollack [1985] shows a different commitment to parallelism, involving the co-operation of many knowledge sources, such as word use, word order, phrase structure and "real-world" knowledge. Their model offers insights into a variety of linguistic phenomena, such as:

Ambiguity - the system can compute multiple readings, and shows increased processing load with ambiguous language;
Single interpretation - the system can consider only one interpretation of an ambiguous sentence at a time, but can easily "flip" between interpretations (as in visual disambiguation of the Necker Cube);
Comprehension errors - "garden path sentences" have more natural and complete explanations as side-effects of strongly interactive processes;
Non-grammatical text - humans are able to interpret non-grammatical language, relaxing constraints as necessary to handle ill-formed input.

The structure of their model is that of a network displaying the characteristics of both *spreading activation* and *lateral inhibition*. The network can be seen as divided into four layers: the first shows the syntactic parse tree for the sentence; the

second, the actual input words; the third, a cluster of meanings for individual words (with mutually inhibitory links); and the fourth, a contextual interpretation of the input (with activatory and inhibitory links between lexical categories and meanings). The model can be used to represent various components of human comprehension, including:

Semantic priming - to bias the competition in the network and influence the stable states;
Autonomy and integration - to represent competing alternative parses and account for garden path sentences with priming;
Errors in comprehension - effects that depend on the arrival time of words can be modelled, allowing "snapshots" to be taken during the processing of a sentence that induces a "cognitive double-take";
Case frames - selectional restrictions of various words can be modelled by the type of links attached to their representative nodes.

Dyer [1989] has suggested ways in which connectionist and symbolic systems may be combined, demonstrating desirable characteristics from each (i.e., variable bindings, logical rules, hierarchies and inheritance from symbolic systems, and reconstructive memory, graceful degradation, category formation, etc. from connectionist systems). Using a technique known as symbol recirculation, Dyer shows that distributed symbol representations can be learned through recurrent networks that will generate expectations concerning words that will occur next in tasks such as script-learning and comprehension. Using this approach, words gain their meanings through how they are used in context (by iteration through a training set), such that each word in the lexicon implicitly represents all the language-use experiences in which it has so far been involved. Although much further research is required, evidently connectionist models can provide a framework for modelling comprehension phenomena that cannot be tackled using ordinary serial or symbolic models.

Hull [1987] has investigated a computational theory of reading and made progress towards defining an algorithmic realisation of this theory. His theory is based on psychological studies of the reading process; acknowledging the extent to which the integration of understanding and recognition is responsible for the fluent reading capabilities displayed by people. Inspired by the human model, the theory states that there are two steps of visual processing, which are influenced by higher-

level cognitive processes. A gross visual description of a word is used as the starting point, to which a small number of feature tests are applied to discriminate between the possible interpretations. Further selective analysis is performed through the application of language characteristics such as syntax and semantics. This technique has demonstrated recognition rates of 96% when applied to images of 12,600 words. Ramsay [1987] has identified a number of linguistic levels as being necessary components within an NLP system architecture, and these are shown in Table 2.1.

Level	Subject Matter
Lexical analysis	Words and word endings
Morphological analysis	The significance of word endings
Syntax	Rules about word order
Semantics	Relationships and identities
Discourse rules	What you can say when
World knowledge	What you can assume

Table 2.1: Levels of Knowledge

He argues further that any system that assumes a simple uni-directional flow of information (either bottom-up for generation or top-down for comprehension) will be ineffective. This is because at any level there are competing interpretations of the data that cannot be disambiguated, for which the appropriate information is available at some other level. Ramsay identifies two solutions to this problem. The first is to use the blackboard approach described above [Erman, 1975], but this is criticised as having serious implementational problems relating to the format of entries on the blackboard and the control over the resources that should be available to each component. The second solution is to try to carry ambiguities around in the form of constraints [Sussman & Steele, 1980]. Using this approach it is not necessary to resolve alternative interpretations immediately - they may instead be maintained until their combination with other constraints produces in a single interpretation.

2.6 Summary

Semantics is the study of meaning, and as such derives much of it theoretical inspiration from disciplines such as philosophy, psychology and linguistics. A semantic theory attempts to formulate ways in which meanings can be represented and processed, either in the abstract, or by people, or by machines. Theoretical

semantics operates on two levels: (a) lexical semantics, which is concerned with the meaning of individual words; and (b) structural semantics, which deals with complex expressions produced by the combination of words within sentences. In principle, linguistic semantics is concerned with the analysis of single sentences; the analysis of collections of sentences being generally referred to as discourse processing. It does not concern itself with encyclopaedic or "world" knowledge - that is deemed the province of pragmatics.

Psychological theories of semantics have been shown to be important, since the human information processing system is the best example of a language processing system, and that such theories attempt to produce cognitive models that should be computationally implementable. A number of semantic theories have been discussed; these are shown to differ widely in terms of their representational aspects. However, all suffer from a variety of implementational difficulties, the most common of which being the problems of knowledge acquisition and inefficiency.

A number of sources of semantic information are identified, the most notable of which being machine-readable dictionaries and text corpora. Several techniques for the extraction of such information are discussed and evaluated. A variety of natural language applications that make use of such semantic information are described, and their relevance to the present project is indicated where appropriate. None of these applications is specifically concerned with text recognition. Moreover, in many cases the objective is language *understanding* rather than language *recognition*.

The issue of lexical acquisition has been discussed, and a number of suggested improvements to the design of MRDs has been identified. The ways in which semantic information may be used within a text recognition system have been described, and a number of possible system architectures discussed. These have included bottom-up and top-down systems, blackboard systems, constraint-based approaches and connectionist models.

Chapter Three

Dictionary Definitions

3.1 Introduction

Natural language semantics can be defined as the study of the meaning of utterances, and to this end a number of semantic theories have been proposed. These theories attempt to define the method by which meanings are computed (by people or machines). Traditionally, semantic theories have been more concerned with language *understanding* rather than *recognition*; and have in most cases attempted a full exposition of language in all its semantic complexity. This contrasts sharply with the needs of the present project. Recognition (not understanding) is the objective; and when applied to a limited domain, only a subset of the language may be necessary. To what then, should the present project turn, for its semantic theories, principles, and data sources?

The established semantic theories are severely limited in terms of their computational applicability. Although these theories may work for artificial domains that are both small and concrete, extending them for large, real world vocabularies is difficult. Firstly, there is the problem of acquisition. The hand-crafting of semantic information for a large vocabulary would be a complex and time-consuming job. Secondly, while some theories may work for concrete subjects, they may not be as applicable to abstract concepts, such as "justice", "insurance" or "business". Thirdly, such theories can easily become unwieldy and inefficient when applied to larger domains [Bookman, 1987]. As an alternative, it is suggested that simpler techniques involving the processing of machine-readable dictionaries and data within the text itself are more practical, and offer better prospects for a successful implementation.

Several applications have successfully used MRDs as their source of semantic information, e.g. taxonomy creation [Chodorow, Byrd & Heidorn, 1985] and knowledge-based parsing [Jensen & Binot, 1988]. Lesk [1987], Guthrie et al [1991]

and Demetriou [1993] have all used dictionary definitions to disambiguate polysemous words within a passage of text. For example, consider the sentence:

I swim across the river to the bank

This sentence (like most others) is composed of the two types of words: function words and content words. Function words are those which give structure to a sentence, such as articles, pronouns, prepositions, etc. In this example, the function words are "*I*", "*across*", "*the*", and "*to*". The content words are the remainder, i.e. "*swim*", "*river*", and "*bank*". It is this latter set of words that is of semantic interest, as it is they that "seem to contain more meaning" [Bolinger & Sears, 1981].

However, the meaning that these words contain is not always clear and unambiguous. For example, the word "*bank*", in isolation, can refer to a financial institution or to the sides of a river (among other things). The actual interpretation chosen by people is guided by context - in this case most would choose the second meaning. This is because their world knowledge tells them that the banks found near rivers are not usually of the financial type (although it must be stressed that this interpretation is not *wrong*, it is just less *likely*). By contrast, a computer has no such knowledge. It has no general knowledge source to aid the process of disambiguation. However, there is much information contained in dictionary definitions, and this can be used as a crude replacement for some aspects of human knowledge.

Lesk's method is to compare the various sense definitions of a word with the definitions of other words in the immediate sentential context. The degree of commonality or "overlap" between definitions is measured, and the word sense showing the highest overlap is selected as the appropriate sense for that context. Consider the above example. The word "*bank*" would have a number of sense definitions, including a financial one and a geographical one. The geographical one might contain words like "*land*", "*river*", "*side*", "*water*", etc. and the financial one words like "*establishment*", "*keep*", "*money*", "*safe*", etc. When these are compared with the word "*river*" and its definition (which might contain words like "*stream*", "*water*", "*flow*", etc.), we can immediately see a greater overlap with the geographical definition of "*bank*". A similar result is obtained if the senses of "*bank*" are compared with the word "*swim*". In this way, word senses can be disambiguated simply by comparing them with the definitions of neighbouring words. Experimentation with this technique has yielded accuracies of 50-70% on short samples of text [Lesk, 1987].

It should be noted, however, that the disambiguation required by the present project is not between *multiple word senses*, but instead between *multiple interpretations* of the input. This is best illustrated by considering the flow of information through the various stages of recognition, as shown in Figure 3.1. The first stage of recognition is pattern recognition, which outputs a list of candidate letters for each letter position. Post-processing then begins with the lexical analyser, which differentiates the acceptable character strings (i.e. English words) from all the other permutations identified by the recogniser. The syntactic analyser identifies the most syntactically acceptable word strings, whilst the semantic analyser identifies the most semantically plausible.

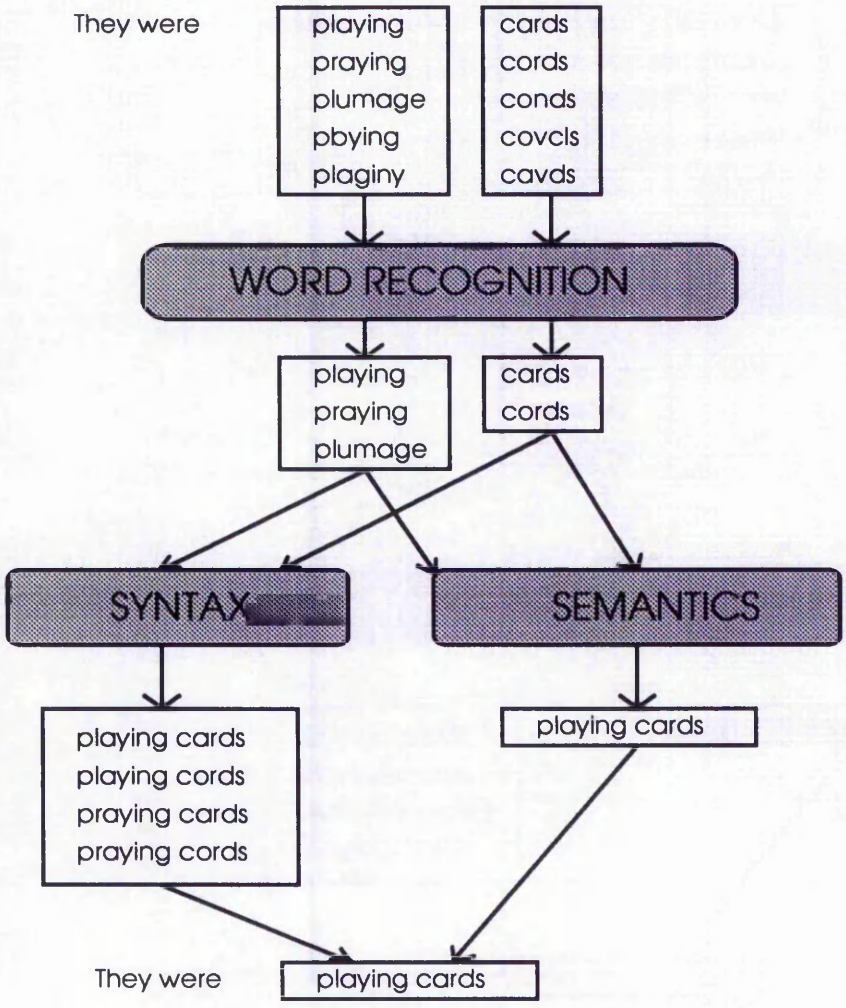


Figure 3.1: Data Flow through the System

For example, consider the following set of phrases:

superbly restored victorian terraced house ...
superbly restored victorian terraced rouse ...
superbly restored victorian terraced rouge ...

The object is to determine which of these is the most semantically acceptable. So in word positions where there are a number of alternative candidates (e.g. the last position) the semantic analyser must identify one as being the most plausible (e.g. "*house*", "*rouse*" or "*rouge*"). Lesk's technique has been adapted to meet these requirements: the definitions of these three candidates are compared with those of the rest of the sentence to find the one with the highest overlap. The commonality between two definitions can be measured by counting the number of words they have in common. There are two ways in which this can be measured:

(i) Strong overlap: This occurs when word1 is found in the definition of word2 or vice-versa (or both), e.g. "house" is found in the definition of "terraced";
(ii) Weak overlap: This occurs when a third word is common to both definitions, e.g. "brick" is found in both "terraced" and "house".

In the example above, if the other candidate words ("rouse" and "rouge") showed negligible overlap, then "house" would be selected as the semantically most plausible word in that position. Most human observers of these three sentences would also confirm this as the most likely choice. Understandably, the behaviour of various words and their overlaps will vary according to the dictionary used - the current project has used the Collins English Dictionary (CED), the Oxford Advanced Learner's Dictionary of Common English (OALD), and Longman's Dictionary of Contemporary English (LDOCE).

Definitional overlap uses dictionary definitions as a source of semantic knowledge, and follows a sequential comparison algorithm to select one from a number of alternative word candidates as being the most "semantically plausible" within that sentential context. It is perhaps misleading to state that definitional overlap selects the "correct" word in any particular case, because ultimately the correct word is a product of the writer's original intentions, and is therefore subjective. Concepts such as "semantically correct" and "semantically incorrect" remain somewhat contentious, and in practice are inessential to the text recognition problem: the measure of success is not found in adherence to some formal semantic proof but simply the ability to choose the same word as a human observer would.

This, as will be seen, can be achieved through methods that have very little to do with the linguistic notion of semantics but much to do with the empirical processing of text-based knowledge sources.

3.2 Definitional Overlap

3.2.1 Data Structures

A dictionary, in either machine-readable or paper form, constitutes a very large textual resource. If this resource is to be stored and processed efficiently, some thought needs to be given to the manner in which it is represented. Clearly, to store it as a character-based file involves a considerable storage requirement and much complex processing to relate words in the dictionary to words in the data. It is desirable therefore to devise an *indexing system*, by which words can be related to some other data structure that is more easily processed and stored. One such form is the *integer*. By representing words in the definitions as integers, storage requirements decrease dramatically and efficient sorting routines become easily applicable. This process, by which words are replaced by integers, is known as *indexing*.

Furthermore, it is necessary to relate inflected forms to root forms. For example, the words "*made*", "*making*" and "*makes*" are all inflections of the root form "*make*", so there seems little point in assigning them separate indices when their origins (and hence much of their semantic content) are shared. This process, by which inflected forms are related to their root forms, is known as *lemmatisation*.

All the dictionaries used during the present project have been processed in the above manner. To eliminate the possibility of incompatible representations, a "standard" list of words and their indices was produced. This list was designed to serve as an ultimate reference in the indexing process, and was compiled using the following algorithm:

1. A definitive list of words was derived from a number of machine-readable dictionaries. This list comprised as many root forms as possible plus their inflections.
2. Each root form in this list is given a unique index, starting from 1000.

3. All inflections inherit the index of their root form. For example, the words "*made*", "*making*" and "*makes*" are all assigned the same index as the word "*make*". This was achieved using the information in Text710 version of the OALD [Mitton, 1986] and a degree of manual post-editing and checking [Keenan, 1992].

4. All function words in this list are given a unique index, below 1000 (starting from 1), to indicate their lack of semantic content and hence non-participation in the overlap process.

Words that can behave as either function or content words were treated as the latter. Indeed, there is some debate concerning the precisely what constitutes a function word [Wilson, 1984]. For the present project, the function word list was created by extracting those that were tagged appropriately in the Text710 version of the OALD (i.e. labelled as preposition, article, etc.), and then reducing this list to the 250 most frequent. This was because it had been decided that the semantic analyser should give a score based on some constant to function words to reflect their high frequency of occurrence. However, some of the function words extracted from Text710 were very rare, so this list of function words was modified to represent only those of higher frequency. Not surprisingly, it is also open to judgement whether function words should be excluded from the semantic analysis at all - although Guthrie [1993] states "*it is typical in techniques based on this work to use a stop list of words*" that are excluded from processing, it is by no means essential to do so.

This indexed list so produced is referred to as the *lexicon*. It can be used to replace the words in the dictionaries by the appropriate indices. Evidently, when data is processed by the current system, candidate words must also be replaced by their indices to unify the representations. However, the lexicon should not be seen as a static repository of data. As new dictionaries become available, such information should be exploited, and for this reason the lexicon has been continually updated as the project has progressed. Consequently, some dictionaries have been indexed with lexicons of differing sizes. For example, in early investigations, the largest list available consisted of some 5,240 root forms plus their inflections. Later investigations had the benefit of a 18,800 lexicon, which provided a much greater coverage of the English language. The effects of such differences are discussed later in this chapter. An example from this lexicon is as follows:

```
a               1
abaci           1000
aback           1001
abacus          1000
abacuses        1000
abaft           1002
abandon         1003
abandoned       1003
abandoning      1003
abandonment     1003
abandons        1003
abase           1004
abased          1004
abasement       1004
abases          1004
abash           1005
abashed         1005
abashes         1005
etc.
```

Dictionary entries may thus be represented by an index corresponding to each headword, followed by a list of indices corresponding to each of the definition words. These lists are sorted in numerical order and delimited by square brackets. For example, the entries for "*abase*" and "*abash*" in the CED may be represented as follows:

```
1004 [13082 18188 21047 23151 27745 28017]
1005 [12386 15430 17632 17846 18645 21225 23334]
```

This structure is represented at run time using an array of structures and an array of integers, and is stored as a binary file.

3.2.2 The Overlap Algorithm

The definitional overlap technique is currently implemented as a C program running under UNIX. The input to the program is the output from the lexical analyser, which consists of a number of candidate words for each word position in the text. Each candidate has a score associated with it, which is initially set by the pattern recogniser and then updated by each of the analysers (this score is not shown in the example below for reasons of clarity). Consider the sentence "*this is a new savings account which you can open with one pound*" written as input to a handwriting recogniser. A typical output from the lexical analyser, showing the alternative candidates in each column, is as follows:

```
this is a hen savings gallant which you can open with one round
tail      new         account       boy car oxen pick ore pound
tall      see         accept        nos oar oven lick due found
trio                                  our           bra hound
```

The definitional overlap technique compares the definitions of the content words and ascribes a score to each, proportional to the number of words in common with the definitions of its neighbours. Once a complete sentence has been processed, the semantic scores are normalised according to an appropriate scale (see Chapter 5). At this point, any function words are assigned a score equal to the maximum overlap score multiplied by a constant (they were not overlapped by the program and therefore have a score of zero so far). Clearly, there are a number of choices to be made regarding this algorithm. For example, what should the value of this function word constant be? (At present, it is set at 0.66, which reflects the relatively high frequency of function words in normal text.) What constitutes a neighbouring word: one that is adjacent, or any word in the same sentence? Issues such as these are investigated and discussed in detail in later sections.

The definitional overlap technique has been successfully applied to the problem of sense disambiguation [Lesk, 1987]. Text recognition is, however, an entirely different problem. To test whether definitional overlap could contribute to such an application, it is necessary to show that where genuine semantic relationships are present between word pairs, the technique is sensitive to them. One way to achieve this is to identify semantically related pairs of words and compare the performance of the technique using these pairs to that of unrelated pairs. A positive result would provide evidence of the technique's ability to identify genuine semantic relationships between words, independent of any particular application area.

3.2.3 Semantic Priming

Theories of human word recognition allow for both bottom-up and top-down influences on processing. Bottom-up influences include those of the input stimulus and its environment, and top-down influences include expectations and hypotheses that come from higher-level cognitive functions. It has been demonstrated that there are various types of context that can influence the speed and ease with which words are identified (and hence recognised), including lexical, syntactic and semantic contexts.

Regarding the latter context, it has been shown that when a word is preceded or accompanied by a "semantically" related word, recognition of that word is facilitated with respect to unrelated controls. For example, Meyer and Schvanevelt [1971] presented subjects with pairs of letter strings, to which they would answer "yes" if both letter strings were words, otherwise responding "no". They found that recognition of semantically related words (such as "knife" and "fork", or "doctor" and "nurse") was faster than that of unrelated words. This evidence suggests that the semantic relation between two words can affect recognition performance.

This phenomenon precipitates an interesting question: is the effect repeatable by a computer? In other words, can semantically related word pairs be differentiated from semantically unrelated word pairs by a computer? If this proves to be the case, then evidently the process is sensitive to semantic relationships between words; specifically those that people are sensitive to. Therefore, it may be adapted to identify semantically related words in text recognition data.

3.2.4 The Semantic Priming Effect

Objective: To determine whether definitional overlap can distinguish between semantically related and unrelated word pairs.

Method: Forty semantically related word pairs were selected, drawn from Postman and Keppel [1970], with a control for each pair [Evett & Humphreys, 1981]. These pairs were chosen to be balanced for frequency, imageability, etc., in the same way as if selected for human subjects. Example related pairs included:

Sweet Bitter
Butter Bread
Smooth Rough

whilst the non-related (control) pairs included:

Sweet Notice
Butter Class
Smooth Court

The definitional overlap program was run on both sets of word pairs. In each case, the program output constituted a set of scores, which corresponded to the

number of strong overlaps and weak overlaps for each pair. Definitions were taken from the machine-readable version of Collins English Dictionary.

Results: The semantically related pairs were given higher scores than the control pairs in 34 out of 40 cases. The breakdown of scores between the two sets is as in Table 3.1. This data can be subjected to the Student's t-test for statistical significance (see Appendix A), giving the value: $z = 4.24$. This z-score can then be checked against statistical tables to determine the level of significance: (z [df 40] = 2.021, $p < 0.05$); (z [df 40] = 2.704, $p < 0.01$). This shows a significant difference between related and unrelated pairs. The result provides evidence that definitional overlap can identify semantically related word pairs.

	Related Pairs	Unrelated Pairs
No. Strong Overlaps	98	2
Total Scores	4726.0	430.0
Average Score	118.15	10.75

Table 3.1: The Semantic Priming Effect

3.3 The Choice of Dictionary

There are a number of commercially available dictionaries that can be obtained in machine-readable form. The CED is one of them; others are shown in Table 1.1. All these dictionaries are similar, inasmuch as they can all be used to create a list of words and definitions. However, there the similarity ends. MRDs are generally compiled by human lexicographers (with the aid of some computer-based tools), and hence are exposed to the subjective design guidelines and style of the particular publisher. Although there is much current research effort directed toward standardising the design of dictionaries, there still remains a large degree of variability in the format and content of each. Often this variation is due to differences in purpose or target readership. For example, the OALD is a learner's dictionary, and it employs a style that seems almost informal or colloquial when compared to the comprehensive, encyclopaedic style of the OED. Evidently, different dictionaries can provide quite different definitions for the same word. Consider the following definitions taken from four dictionaries for the noun sense of "*deposit*" (omitting details such as grammar, phonology, etc.):

(1) The Shorter Oxford English Dictionary (SOED):
Deposit *sb.* 1624 [- L. *depositum*, subst. use of neut. of pa. pple. of *deponere*; see DEPONE, DEPOSE.] **1.** Something laid up in a place, or committed to the charge of a person, for safe keeping. Also *fig.* 1660. **b.** *spec.* A sum of money deposited in a bank 1753. **c.** something committed to another person's charge as a pledge 1737. **2.** The state of being deposited; in phr. *on, upon, in d.* 1624. **3.** Something deposited, laid or thrown down; *esp.* matter precipitated from a fluid medium, or collected in one place by a natural process. In *Mining* an accumulation of ore, *esp.* of a somewhat casual character, as in pockets. 1781. **4.** The act of depositing; cf. prec. senses, and DEPOSIT *v.* 1773. **5.** A depository, a depot. (Chiefly *U.S.)* 1719.

(2) The Collins English Dictionary (CED):
deposit ... ~n **6. a.** an instance of entrusting money or valuables to a bank or similar institution. **b.** the money or valuables so entrusted. **7.** money given in part payment or as security, as when goods are bought on hire-purchase. See also **down payment**. **8.** a consideration, esp. money given temporarily as security against loss of or damage to something borrowed or hired. **9.** an accumulation of sediments, mineral ores, coal, etc. **10.** any deposited material, such as a sediment or a precipitate that has settled out of solution. **11.** a coating produced on a surface, esp. a layer of metal formed by electrolysis. **12.** a depository or storehouse. **13. on deposit**. payable as the first instalment, as when buying on hire-purchase. [C17: from Medieval Latin *depositare*, from Latin *depositus* put down]

(3) The Oxford Advanced Learner's Dictionary of Current English (OALD):
deposit *n* [C] **1** money that is deposited(2,3): *The shopkeeper promised to keep the goods for me if I left/paid/made a ~.* **money on ~**, money deposited in this way. **~ account**, money deposited in a bank, not to be withdrawn without notice, on which interest is payable. *current account* at current(3). **~ safe**, safe in the strongroom of a bank, rented for the custody of valuables. **2** layer of matter deposited(4): *A thick ~ of mud covered the fields after the floods went down.* **3** layer of solid matter left behind (often buried in the earth) after having been naturally accumulated: *Valuable new ~s of tin have been found in Bolivia.*

(4) Longman's Dictionary of Contemporary English (LDOCE):
deposit n. **1** something deposited : *There are rich deposits of gold in those hills.* | *There's some deposit at the bottom of this bottle of wine* **2** usu. sing. a part payment of money, which is made so that the seller will not sell the goods to anyone else : *You must pay a deposit to the hotel if you want them to keep a room free for you* compare earnest(1) **3** an act or action of depositing : *The rate of the river's deposit of mud is about one inch a year* **deposit account** n. a bank account which earns interest and usu. from which money can be taken out only if advance notice is given. compare *savings account, current account*

It can be seen that the SOED contains historical information, is formal in its style and content, and tends to use short punctuated phrases to provide coverage of

each sense definition. By contrast, the OALD is more straightforward, uses whole sentences wherever possible, and makes extensive use of example sentences. The CED is somewhere in between these two styles. LDOCE is of particular interest, since it is claimed that its entries are defined using a *controlled vocabulary* of around 2,000 words, and that the entries have a simple and regular syntax [Boguraev & Briscoe, 1989]. The effect that this has on LDOCE's efficacy for text recognition is discussed in later sections.

Lesk [1986] has used the OED as a source of information for his work on automatic sense disambiguation. He compared the OED with the OALD, CED and W7 and concluded that the OED was the most suitable due to its sheer size (the greater the number of headwords, the more chance that a particular word will be included) and because "*a quick count of several dictionaries indicated that the OED surpassed all others in the number of useful content words in its definitions and quotations*". It is not clear, however, what criteria were used in the measure of "usefulness", or just how "quick" the count was.

It may be true that some dictionaries are more suitable for language processing than others, but such distinctions should not be based on size alone. The OED may contain a greater overall quantity of "useful" information than other dictionaries, but just how *concentrated* is this information? A brief glance at their relative sizes shows that the OED is several times larger than any of the other dictionaries, resulting in a considerable computational overhead in terms of memory requirements and search times. Moreover, many of the headwords appear to be archaic or non-standard English: this may suggest that the other dictionaries are more suitable for text recognition (unless the test data itself is archaic or non-standard English!). A comparison of the definitions themselves suggests that the concentration of useful information may be proportionately less in the OED than in other dictionaries.

In general, a dictionary needs to be above a certain size to provide sufficient coverage over a wide range of domains. However, it may transpire that the most important factors are information *concentration* rather than sheer size, and the degree to which a dictionary is up-to-date and covers contemporary material (this after all is the type of material a text recognition system would typically have to handle). To resolve this issue, a version of the OALD was obtained and indexed in the same way as the CED. By running the definitional overlap program on some sample text using definitions extracted from both dictionaries, an objective comparison could be made.

3.3.1 The OALD

Objective: To determine the effectiveness of the definitional overlap technique using definitions extracted from the OALD.

Method: Due to constraints on the availability of data pads it was not always possible to obtain test data from the original source. As large samples of data were required, further programs were written to facilitate this process, by simulating recognised output for any given input sentence. This confusion simulator program gives an output similar to the real pattern recogniser, working on the same database of characters, probabilities and dependencies [Keenan, 1990].

The OALD was indexed using the same technique as that applied to the CED, and used as the source of definitions by the overlap program. A typical business letter of some 153 words was selected from a corpus of business documents. This was put through the confusion simulator and then used as input to the overlap program. It was then necessary to determine the "neighbourhood size" around each word, i.e. the limit beyond which words are deemed to be too far apart to be considered as neighbours. This quantity is also referred to as the *window size*, since semantic relations are said to take place within a limited *window* of so many words in the text. It was known from collocation studies (see Chapter 4) that the information from co-occurrence relations is optimised at a distance of four words, so the window size was provisionally set at this distance.

Results: The breakdown of scores is shown in Table 3.2:

correct choices	70%
ties (no decision)	15%
incorrect choices	15%
average no. of candidates per position	3.06
%correct expected from random	32.68

Table 3.2: Performance of the OALD

Discussion: Firstly, it can be seen that the process of definitional overlap selects the correct word in 70% of cases. At first sight, this may appear to be a poor performance but it is in fact over twice as frequent as a random selection (which for this data would be 32.7% of cases). Percentage measures of performance should therefore be

compared to that expected from random selection. Secondly, this performance measure is somewhat crude; it measures whether the correct word was chosen or not, irrespective of the score or margin by which it succeeded over the other candidates. Such quantities cannot be ignored if measures of the program's performance are to be valid.

One method of quantitative assessment would be to use a comparison between the score given to the correct word and the score given to the highest other content word in the same sentence position. These scores may then be recorded as "correct-score:other-score" pairs, and subjected to the relevant statistical test for significant mean difference. The methodology used in the treatment of results of subsequent investigations therefore includes the following procedure:

(1) Iterate through the sentence positions in the output;
(2) If the correct word is a function word, proceed to next sentence position;
(3) If the correct word is a content word, add its score to the "correct-word scorelist". Find the highest score of all the other content words in that position, and add that score to the "highest-other-word scorelist".
(4) When all candidate content words have been assigned a score, use the Student's t-test (see Appendix A) to determine whether the difference between the means of the correct-word scores and the highest-other-word scores is statistically significant.

When this test is applied, the following result is obtained: $z = 2.15$. This z-score can then be checked against statistical tables to determine the level of significance: (z [df 19] = 2.093, $p < 0.05$), and (z [df 19] = 2.861, $p < 0.01$). This shows a significant difference (to 95%) between correct-word scores and highest-other-word scores. The student's t-test is a recognised test for statistical significance of the difference between two means and allows for small sample sizes. Use of the OALD therefore surpasses the 95% confidence limit.

Conclusions: This investigation has demonstrated two important findings concerning the definitional overlap technique:

(1) An appropriate performance metric is required and has since been designed and applied to the original results, to reveal more objective patterns in the data;
(2) The OALD makes a contribution significant to the 95% confidence level in reducing the ambiguity in the output of a handwriting recognition system.

3.3.2 The CED

Objective: To determine the effectiveness of the definitional overlap technique using definitions extracted from the Collins English Dictionary.

Method: The CED was coded using the 5,240-word lexicon. The document used above was presented as input to the semantic analyser, and the CED used in place of the OALD as the source of definitions.

Results: The breakdown of scores is shown in Table 3.3. The z-score can be checked against statistical tables to determine the level of significance: (z [df 20] = 2.086, $p < 0.05$), and (z [df 20] = 2.845, $p < 0.01$), i.e. this difference is not significant.

correct choices	62%
ties (no decision)	0%
incorrect choices	38%
average no. of candidates per position	3.06
%correct expected from random	32.68
z-score	0.56

Table 3.3: Performance of the CED

Discussion: There are two effects to be explained. The first is that the CED worked with the semantic pairs, in the sense that it gave a statistically significant result, but it does not in this investigation. There are two reasons for this: (a) the correct words do not have a proven semantic relationship (unlike the semantic pairs), and (b) the control words are not artificially selected but instead are random orthographic derivations of the correct words and therefore not necessarily unrelated. The second effect is the inferior performance of the CED when compared to the OALD. Possible reasons for this difference are revealed through closer examination of the definitions within these two dictionaries. In both dictionaries, the purpose of the definition is to provide a precise, sense-based statement of the meaning of each word, preferably in terms of their hyponyms, synonyms, etc. However, the OALD is more encyclopaedic in its exposition, and more liberal in its use of examples. Consider the definition of the word "*payment*":

In the CED:
payment n. 1. the act of paying. 2. a sum of money paid. 3. something given in return; punishment or reward.

In the OALD:
payment n. 1. paying or being paid: *demand prompt ~; a cheque in ~ for services rendered.* 2. sum of money (to be) paid: *$50 cash down and ten monthly ~s of $5.* 3. reward; punishment.

The OALD evidently provides more encyclopaedic knowledge in the form of examples, and provides simpler, more concrete definitions, using everyday language. The design of the definitions within the OALD is therefore more likely to reflect the patterns of word usage found in everyday text than those in the CED. The use of examples in each definition encodes information concerning "typical collocations", so the OALD effectively provides both lexical overlap and some collocational information, whereas the CED tends to provide only the former. (NB - this also further justifies the investigations with collocational semantics in Chapter Four.) It thus appears that the OALD definitions contain more information of direct use to a text recognition system, and the results of this investigation reflect this difference.

However, the OALD differs in another important way: the indexing. In the discussion of data structures earlier in the chapter, it was stated that a lexicon, derived from a number of machine-readable dictionaries, was used as the ultimate reference in the indexing process. However, this lexicon has been updated as the project has progressed, and for this reason the OALD has been indexed using a different list. The first list comprised some 15,350 words (inflections of some 5,240 root forms), and this was used to index the CED. When the OALD became available, a more comprehensive lexicon had been compiled, comprising some 56,940 types (from 18,800 root forms), and this was used to index the OALD. So the two dictionaries differed on another dimension. To what extent did this difference in indexing affect the performance of each dictionary? To answer this question, a further investigation was carried out.

3.3.3 The Re-indexed CED

Objective: To assess the use in semantic analysis of definitions extracted from the CED and re-indexed using the 18,800-lexicon, to reduce the ambiguity of output from a text recognition system.

Method: The CED was re-indexed using a lexicon of some 18,800 lemmas. The data used above was presented as input to the semantic analyser, with the newly indexed CED definitions as the source of definitions.

Results: The breakdown of scores is shown in Table 3.4.

correct choices	70%
ties (no decision)	0%
incorrect choices	30%
average no. of candidates per position	3.06
%correct expected from random	32.68
z-score	0.98

Table 3.4: Performance of the Re-indexed CED

This z-score can then be checked against statistical tables to determine the level of significance: (z [df 19] = 2.093, $p < 0.05$), (z [df 19] = 2.861, $p < 0.01$), i.e. this difference is not significant.

Discussion: In both investigations involving definitions taken from the CED the results were well short of the 95% confidence level. However, the z-score for the newly indexed definitions (using the longer lexicon) is almost double that of the earlier version. This suggests quite strongly that the indexing process is important. There are two factors associated with this difference:

(1) Lexical Coverage: Consider the sizes of the two lexicons used in the indexing process:

Lexicon1 = 15,350 words, from 5,240 roots
Lexicon2 = 56,940 words, from 18,800 roots

When a dictionary is indexed, any word that has no entry in the lexicon is ignored (and subsequently discarded). The second lexicon, being much larger, will obviously cover a much larger subset of English, and hence cover much more of the contents of a definition. The larger the lexicon used in the indexing process, the more information in the definitions is retained, and the greater the contribution of these definitions to the recognition process. The process of indexing should therefore employ as large a lexicon as possible.

(2) Grain-Size: The first list relates 15,350 types to 5,240 roots. This is a ratio of 2.93:1. The second list relates 56,940 types to 18,800 roots. This is a ratio of 3.03:1. In this example, the ratios are comparable. However, it is possible to produce lexicons of comparable size that relate words to a much smaller number of roots.

Depending on the purpose for which it is intended, the lexicon can be compiled in many ways. For example, syntactic applications may require some discrimination between different grammatical categories, e.g. between the verb forms such as "*work*", "*works*" and "*working*" and the noun forms such as "*worker*" or "*workers*". This would suggest the assignment of one index to derivatives of the verb form and another to those of the noun form. However, the rules by which such indices are assigned to words are not totally reliable. This is because they are based on the spelling of each word, which is inconsistent due to the non-deterministic morphology of the English language. For example, in assigning an index to a word like "*making*" or "*baking*", it may be possible to work backwards to arrive at the root ("*make*" or "*bake*") by removing the "*ing*" ending and adding the letter "*e*". But what of words like "*king*" or "*fling*"? The same rules for finding the root are no longer relevant. Similarly, what is the root of "*leaves*"? Is it "*leaf*", "*leave*" or both?

There are many other such examples throughout English, and all are reflections of morphological inconsistency. The compilation of a lexicon therefore requires considerable manual intervention to ensure that indices have been assigned in a reliable manner that fits the application. The lexicon used by the present project reflects both the needs of the various analysers and the subjectivity of manual intervention to resolve the morphological inconsistencies. For example, consider the words "*pay*" and "*payment*". Neither is an inflection of the other, so strictly speaking their differing linguistic origin should dictate separate indices. However, by manual intervention they may be deemed sufficiently semantically related to justify a common index. Now consider the sentence fragment:

... payment on your account ...

In the OALD, the word "*account*" occurs in the definition of "*pay*", but not in the definition of "*payment*". So if "*pay*" and "*payment*" are indexed as having separate roots, the above fragment would show no overlap. However, if "*payment*" were assigned the same index as "*pay*", then a strong overlap would result. The need to represent such relationships may be accommodated using a smaller list of roots (and hence a coarser *grain-size*), and more meaningful overlaps may be the result. It is suggested that this issue of root-assignment (or lemmatisation) and grain-size form the basis of further investigation.

One final point concerns the calculation of the z-score. Although both the OALD and the re-indexed CED produce 70% correct choices, the z-score for the

OALD is 2.15 whereas for the CED it is 0.98. This is due to the number of incorrect choices, which being higher in the case of the CED lowers the mean difference and hence the z-score (see Appendix A). Moreover, to be able to perform a reliable statistical analysis the sample size must be considerably larger. The original text may have consisted of 153 words, but many of these are function words and a further number produces no alternative candidates when put through the simulator. Only positions of "semantic interest" are subjected to statistical analysis, i.e. those positions where the correct word is a content word accompanied by a number of alternatives. This sample text yielded 30 or 40 such positions, henceforth referred to as *data points*. With the standard deviations being relatively high (in the order of 8.0 to 12.0), clearly a much larger data sample is needed (to provide at least 100 data points, i.e. 25-30 sentences).

3.4 Definitional Overlap and Domains

It is expected that a fully functional text recognition system will be required to work with a range of material, taken from a range of domains (e.g. *banking, insurance, estate agents, medical, technical,* etc.). Although some degree of "tailoring" of individual systems may be possible (or even desirable in some cases), the basic techniques on which the system relies must work consistently across a range of domains.

So far, all the investigations of definitional overlap using the MRDs have been based on one sample of data, which was selected at random from a corpus of business letters. It may be typical of many business letters, but it is still only one sample of data, and as such represents only one domain - in this case that of *banking*. The significant results achieved using this text may have been specific to this domain and hence not necessarily repeatable in other domains (i.e. the OALD may provide unusually good coverage of commercial or financial terminology). It is necessary, therefore, to assess the reliability of the process across a range of domains, which requires the selection of a number of domains and the acquisition of appropriate data samples. The overlap program can then be run on these samples to establish the relationship between performance and domain.

3.4.1 Domain Specificity

Objective: To investigate the extent to which the definitional overlap technique contributes to the text recognition process across a range of domains, using the OALD coded with the larger lexicon.

Method: Three domains were chosen for investigation: *Banking, Estate Agents* and *Music*. This choice reflected both the potential application of the eventual system (i.e., the *Banking* and *Estate Agents*) and ease of availability (the *Music* documents could be easily collected from a research bulletin board). Test data was gathered by random selection from each domain until the target of approximately 17 sentences per domain had been met. These texts were then processed by the confusion simulator program, and the output used as input to the definitional overlap program.

Results: Before considering these results, it must be appreciated that the t-test is very sensitive to changes in sample size (see Appendix A). For this reason, these results show a much larger z-score for a relatively small percentage correct. Note also that the test is also sensitive to the standard deviation - a wide "spread" of scores (as demonstrated by the results for music) will produce a lower z-score than a domain with the same percentage correct but a narrow spread of scores (e.g. *estate agents*).

(a) *Banking*: The breakdown of scores is shown in Table 3.5.

correct choices	75%
ties (no decision)	3%
incorrect choices	22%
average no. of candidates per position	2.58
%correct expected from random	38.76
z-score	7.13

Table 3.5: Performance within the Banking Domain

This z-score can then be compared to the value of z required for 95% and 99% significance level (obtained from statistical tables): (z [df 90] = 1.99, $p < 0.05$), (z [df 90] = 2.63, $p < 0.01$). $7.13 > 2.63$ therefore reject null hypothesis at 99% significance level; i.e. it can be said with 99% confidence that the technique selects the correct word in favour of alternative candidates within the domain of *banking*.

(b) *Estate Agents*: The breakdown of scores is shown in Table 3.6.

correct choices	55%
ties (no decision)	5%
incorrect choices	40%
average no. of candidates per position	3.03
%correct expected from random	33.0
z-score	3.69

Table 3.6: Performance within the Estate Agents' Domain

This z-score can then be compared to the value of z required for 95% and 99% significance level (obtained from statistical tables): (z [df 133] = 1.98, $p < 0.05$), (z [df 133] = 2.61, $p < 0.01$). $3.69 > 2.61$ therefore reject null hypothesis at 99% significance level; i.e. it can be said with 99% confidence that the technique selects the correct word from alternative candidates within the domain of *estate agents*.

(c) ***Music***: The breakdown of scores is shown in Table 3.7.

correct choices	57%
ties (no decision)	2%
incorrect choices	41%
average no. of candidates per position	2.48
%correct expected from random	40.32
z-score	1.44

Table 3.7: Performance within the Music Domain

This z-score can then be compared to the value of z required for 90%, 95% and 99% significance level (obtained from statistical tables): (z [df 87] = 1.66, $p < 0.1$), (z [df 87] = 1.99, $p < 0.05$), (z [df 87] = 2.63, $p < 0.01$). $1.44 < 1.66$ therefore accept null hypothesis; i.e. no evidence to suggest that the technique makes a significant contribution to the recognition process within the domain of *music*.

Discussion: It is necessary to appreciate a number of factors concerning the t-test. Firstly, the use of a statistical test is only as a guideline to highlight trends in the data - not to define immutable standards of performance. Moreover, the t-test is quite strict, as are the levels of significance selected for this investigation (and others). Furthermore, the t-test takes into account background variation, unlike the percentage measures - which partially explains the less than total symmetry between the two

measures of performance. (Other explanations concern the variation in the "strength of the competition", as indicated by the "%correct expected from random" figure.)

Regarding the results themselves, it can be seen that the performance of definitional overlap across a range of domains is consistently positive but highly variable. It ranges from the highly significant (well above the 99% level in the case of *banking*) to the statistically insignificant (in the case of *music*). These results are based on considerably expanded sample sizes so may be accepted as being more representative of the expected performance within each domain. Possible reasons for this spread of results include issues related to both the origin of the dictionary and the origin of the test data:

(i) The origin of the dictionary: This explanation is concerned with the "accessibility" of the domain, i.e. the degree to which its constituent terminology is commonly understood. The type of terminology used in a business letter taken from the domain of *banking* is relatively well used in everyday life and consequently well understood. Most people to some extent have to manage and understand their own financial affairs and its concomitant terminology, and media coverage of financial affairs encourages this basic knowledge of terminology. So although lexicographers may purposely employ experts to contribute to the compilation of definitions in more esoteric domains, it is nevertheless the case that more everyday words will have more widely understood patterns of usage that are reflected in both the definitions they possess and the manner in which they are used in a typical business letter. In other words, the more "everyday" a domain is, the more "everyday" its constituent words will be, and the more they fit into stereotypical patterns of usage that are quoted as examples in the dictionary definitions.

More abstract or esoteric domains such as *music* are reliant upon expert knowledge to provide precise definitions and typical examples of usage, and hence are more subjectively compiled. The words themselves, being less "everyday", are less likely to have widely accepted definitions or stereotypical patterns of usage that may then be quoted as examples in their dictionary definition. In this sense, the dictionary and its definitions are less suited to the recognition of domains that are more abstract or esoteric. It is possible therefore that specialist dictionaries may be necessary to maintain performance in these domains.

(ii) The origin of the test data: Another explanation for this pattern of results is based on the purpose of the test documents. The *banking* document and *estate agent's*

document were both designed to be readable by lay people, and hence used language structures that could be immediately understood. This was achieved firstly by avoiding unusual (e.g. low-frequency) words, and secondly by adhering to commonly used contexts or patterns of usage (e.g. "*...saving for a rainy day...*" and "*...free to open, easy to run...*"). The *music* document, on the other hand, was not restricted by such limitations. It was designed to be a communication between experts, discussing technical issues within their own subject field and hence was not intended for "public consumption". The patterns of usage therefore reflected the writers' own individual styles, and their choice of words similarly reflects their own subjective knowledge. In this way, the dictionary, being a general-purpose source of knowledge designed for use by non-experts, is being used to contribute to the recognition of text that is both highly specific and intended for a specific audience only. It is possible that this disparity of purpose contributes to the discrepancy in performance indicated above.

The two explanations above show a high degree of commonality in their reasoning. This is because there is a high degree of commonality in the processes being described: usage of a word contributes to the way in which it is defined in a dictionary, and a dictionary definition contributes to the way in which a word is actually used.

Conclusion: There is a great degree of variability in the performance of the overlap technique across different domains. This is due to factors relating to both the design of the dictionary from which the definitions are taken, and the purpose of the text being recognised. It is suggested that the use of general dictionaries may be insufficient to recognise text taken from more esoteric domains or documents that are intended for a specific audience. Text such as this may require the acquisition of specialist dictionaries to maintain the high performance shown in other domains.

3.5 Definitions and Semantic Networks

3.5.1 Introduction

Most human observers would describe the words "*mortgage*" and "*money*" as being semantically related. However, to find such a connection in a dictionary it may be necessary to go beyond the "first level" of definition. Consider the definition of "*mortgage*" in the OALD:

mortgage ~ *(to) (for)*, give sb a claim on (property) as a security for payment of a debt or loan: ~ *a house (to sb for $40000); land that may be ~d. n* act of mortgaging; agreement about this: *raise a ~ (on one's house) from a bank. I can buy the house only if a ~ of $40000 is obtainable. We must pay off the ~ this year ...*

The word "*money*" is not present, although the use of the currency sign does suggest a monetary value (the investigation of the semantic role of such special characters will form part of future studies). However, the definitions of "*payment*", "*debt*" and "*loan*" all contain the word "*money*", so it can still be reached if the search goes far enough. The "semantic tree" thus created has already been exploited by researchers to provide information relevant to other natural language systems [Jensen and Binot, 1988]. It may thus transpire that definitional overlap could operate more effectively by considering both the definition of a word and the *expansion* of that definition. However, the average length of a definition (after reduction, indexing and sorting) is 21.81 words in the OALD and 14.55 in the CED. So to expand a typical definition from the OALD to just the second level would involve the processing of $(21.81)^2$ words, i.e. approximately 475. This produces a considerable combinatorial explosion, making processing beyond the first level somewhat impractical. What is required therefore is a method of *compressing* these definitions so that this combinatorial explosion is reduced.

One reason for the length of the definitions is that English is a highly polysemous language, and within a typical entry there will be separate definitions for every sense of the word. However, when language is used within a specific domain, it is often the case that only a subset of those senses is appropriate. For example, in financial documents, any references to the word "*bank*" are unlikely to involve the sense related to rivers and canals. It follows therefore that if the domain is known, it may be advantageous to eliminate the alternative word senses from a definition so that processing focuses only on those that are relevant to the domain. This reduces the combinatorial explosion and decreases the potential for spurious overlaps through the co-incidence of alternative word senses. For example, if definitions within a domain can be halved, i.e. reduced to an average of 11 words in length, the processing at the second level involves some 11^2 words, i.e. 121 or 25% of the previous total. (Of course, one problem with constraining semantics to a single domain is determining which domain is to be used for a given input - a problem discussed in greater detail in Chapter Six.)

3.5.2 The Filtering Method

This compression of definitions is achieved by a process involving the "*filtering*" of definitions through a *filter set* chosen to represent the core vocabulary of a domain (in this case, that of *banking*). Consequently, the overlaps can be seen as being "pre-processed" into the definitions of each word. This process consists of the following steps:

(1) Create the wordlist for which filtered definitions are required. This may be done using a frequency distribution from a domain corpus.

(2) Derive the filter set. This may be based on the core vocabulary, or a subset thereof. The core vocabulary is usually derived by taking a frequency distribution of a domain corpus, comparing this with a frequency distribution from a general corpus, and sorting according to distinctiveness. In the example below, the filter set consisted of the first 30% of the core vocabulary.

(3) Iterate through the wordlist, repeating the following steps for each word:

(i) overlap the word with each member of the filter set;
(ii) when a strong overlap occurs, place the index responsible in a "strong overlap" list;
(iii) when a weak overlap occurs, place the index responsible in a "weak overlap" list;
(iv) when the word has been overlapped with all members of the filter set, take those weak overlap indices with a frequency greater than a certain threshold (this was set to 1 for the example below) and append them to the strong overlap list.

(4) The list so obtained forms the new definition for that word, containing all strong overlaps (with the filter set), and the more frequent weak overlaps. In this way, words in a definition that represent senses inappropriate to a domain will be unlikely to overlap with the filter set, and so be excluded from the new definition.

The following example shows the effect of filtering the definition of the word "*charge*" (taken from the CED), using the domain of *banking*:

Length before filtering	= 129 words
Length after filtering	= 84 words
Ratio (new:old)	= 0.65

Example words removed from the old definition:

Military sense: *battle, command, control, horse*
Legal sense: *accusative, assault, commit, evidence, fault, injunction, judge, legal*
Electrical sense: *electricity, electron, explosive, formic, negative, phenomena, solution*

Example words remaining: *account, debit, demand, finance, liable, price, service, set*

The original definition contained 129 words, related not only to the financial sense (i.e. charge as in "*require payment*") but also to the "*electrical charge*" sense and the "*military charge*" sense, etc. When this definition is filtered through the *banking* domain filter set it is reduced to 84 words in length. Evidently, the definition has been reduced to 65% of its original length. The words relating to the non-financial senses have been largely eliminated, without losing words relevant to this domain. Sense-based definitions of words could contribute to the subsequent overlap process by providing more concise, pertinent definitions and reducing the chances of spurious overlaps due to inappropriate but co-incident word senses.

3.5.3 Discussion

Although initial studies using this technique have shown promise, further tests are required to determine the optimum settings for the various parameters. It may transpire that tightening (raising) the weak overlap threshold produces more concise definitions in some domains, whereas in others valuable information in the original definitions starts to be lost. Moreover, the choice of filter set is not at all fixed (e.g. what proportion of the core vocabulary is an adequate representation of the domain for these purposes?)

There are other ways in which definitions may be filtered. Another possibility, for example, uses the same concept of a filter set, but instead of taking the definition as a whole, takes the separate senses and progressively eliminates those that show the least overlap with the filter set. However, in its simplest form, this technique results in words having one single sense definition remaining. This is inappropriate, since in most domains, more than one sense of a word is relevant.

To conclude, the filtering process provides an automatic way of eliminating irrelevant material to produce sense-based definitions. This can help provide more domain-specific definitions, and reduce the combinatorial explosion produced by expanding definitions beyond the first level. Practical uses of this technique, and the

interrelation with other knowledge sources within the semantic analyser are discussed below. Whether they are filtered or not, expanding dictionary definitions may provide a way of accessing semantic relations that are not accessible at the first level. To resolve this issue, a further investigation was carried out.

3.5.4 Definition Expansion

Objective: To assess the extent to which the expansion of dictionary definitions provides information of use in semantic analysis.

Introduction: It has been suggested that dictionary definitions may be "expanded" to create "semantic trees", and that these trees can be traversed and searched to provide information for NLP systems [Chodorow, Byrd & Heidorn, 1985]. Evidently, the information so obtained may provide a further constraint of use in semantic analysis. However, is necessary to firstly define what is meant by "useful" information, and then to measure the quantity of useful information obtained from this process.

"Useful" semantic information is that part of a word's definition (or its expansion) that is semantically related to the sentential context of that word in typical usage. Useful semantic information therefore facilitates the incidence of meaningful strong overlaps in normal text. Since documents usually have an overall structure, objective and topic, they can be said to represent a particular domain (e.g. *banking*, *insurance,* etc.). Therefore, in measuring the amount of useful information contained in a definition, we cannot simply measure the total number of words. Instead, we need to measure the number of words that have senses related to the same domain, since these are more likely to co-occur in typical usage. This can be accomplished by collecting a corpus of texts for a chosen domain and producing a frequency distribution for the words therein. The definitions and expansions of various words can then be compared with this list to check for membership. The greater the degree of common membership, the greater the coverage of that definition or expansion, and hence the greater the amount of useful information.

This amount may be expressed as a ratio of the amount of useful information compared to the amount of redundant information. For example, consider the domain of *banking* and the definition of the word "*mortgage*" as derived from the CED:

mortgage n. 1. a conditional conveyance of property, esp. real property, as security for the repayment of a loan. 2. the deed effecting such a transaction. 3. the loan itself ~vb (tr.) 4. to convey (property) by mortgage. 5. Informal. to pledge.
mortgage rate n. the level of interest charged by building societies and banks on house-purchase loans.

We find that a proportion of these definition words are relevant to the domain of *banking* (as defined by membership of the domain wordlist), whilst the remainder are not (or have negligible relevance). In the definition above, if the words "*charged*", "*property*", "*interest*", "*repayment*" and "*purchase*" were all members of the domain wordlist, we could say that expansion of the word "*mortgage*" provides a total of 25 content words, of which 5 are directly relevant and hence useful. Expressed as a ratio, the coverage of this definition is therefore 5/25, i.e. 20%. A similar procedure may be applied to determine the coverage of the 2nd level expansion, i.e. expand each definition word (all 25 of them), calculate the number of relevant words produced and express this as a percentage of the total number of words produced. The relative contributions of the two levels can now be compared.

The use of frequency distributions facilitates the investigation of a second issue: the relationship between relative frequency (distinctiveness) and coverage. In other words, does the expansion of highly distinctive words result in a greater proportion of useful information? The objective of this investigation can now be re-stated as attempting to determine the following quantities:

(i) the coverage given by expansion to the 1st level (definition);
(ii) the coverage given by definition expansion to the second level;
(iii) the relationship between relative frequency and coverage.

Furthermore, the investigation of these three issues allows a fourth quantity to be determined: the relationship between syntactic category and coverage. In other words, do nouns have more useful definitions than verbs, adjectives or adverbs?

Method: A number of documents related to the domain of *banking* were collected, and a frequency distribution produced. Every tenth word from this list was selected for investigation. This process entailed two stages:

(i) Expansion of the word to its 1st level (definition), and subsequent calculation of the coverage;

(ii) Expansion of the definition to the next (second) level, and subsequent calculation of the coverage.

Coverage for each level was calculated as above. Spearman's Rank-Order Correlation Coefficient was calculated to determine the correlation between relative frequency and coverage.

Results: Table 3.8 shows the results for the first ten words selected from the relative frequency distribution list. "Rank" refers to their position on that list. "1st Level" refers to the coverage given by expansion to the first level, expressed as a percentage (as described above). "2nd Level" refers to the coverage given by expansion to the second level, also expressed as a percentage.

Word	Rank	1st Level (%)	2nd Level (%)
mortgage	1	28	16
hesitate	10	0	0
reassure	20	16	5
forward	30	17	11
account	40	24	5
confident	50	11	4
customers	60	14	9
exception	70	11	8
payment	80	21	5
reduce	90	16	6
complement	100	19	6
Mean Coverage		14.0	8.0

Table 3.8: Definition Expansion and Coverage

Calculation of Rank-Order Correlation Coefficient:

$$\rho = 1 - \frac{6\sum D^2}{N}(N^2 - 1)$$

Where N = the number of pairs

ρ = the correlation coefficient

$$\rho = 1 - \frac{6(2990)}{27}(27^2 - 1)$$

$$\rho = 0.09$$

Correlation needed for 95% confidence level = 0.3809, i.e. this correlation is insignificant.

The above table displays the words in rank order. These words can be re-grouped according to the syntactic categories of noun, verb, adjective and adverb. The average coverage of the first-level definitions in each group may then be calculated. The relationship between syntactic category and coverage may then be expressed by Table 3.9.

Category	1st Level	2nd Level
noun	20	10
verb	13	7
adjective	10	9
adverb	8	8

Table 3.9: Syntactic Category and Coverage

Discussion: It can be seen from the first of the above tables that the mean coverage given by the second level (8%) is just over half that of the first level (14%). From this we can infer that the proportion of useful information obtained by expansion to the second level is less than that available at the first level. At present, the text recognition system employs only first level information. To combine second level information with first level information would result in an overall decrease in the ratio of useful information to redundant information. In practical terms, this would imply an overall increase in the number of spurious overlaps and a proportional decrease in the number of meaningful overlaps.

Suggested explanations for this pattern of results focus predominantly on the highly polysemous nature of the English language. This is reflected in a multiplicity of senses for any given word in the dictionary. Whereas at the first level there may be a number of words representative of the typical sentential context (say 10%), beyond this point each of the definition words is in itself polysemous, involving further unrelated sense definitions. The percentage of related words at this level is therefore 10% of 10%, i.e. 1%. Hence the expansion of dictionary definitions descends into progressive generality, displaying a weaker and weaker semantic relationship with the original word. It may be possible to reduce or even eliminate this effect by "filtering" the definitions in the manner described earlier, but the pattern is extremely persistent (almost every word investigated exhibited a descent into generality) and it is unlikely

that the filtering process could substantially alter this trend. This issue is suggested as an area for further research. It can also be seen that there is a negligible relationship between relative frequency and coverage. There is thus no evidence to suggest that definition expansion may provide useful information when applied selectively to highly distinctive words.

There is a strong relationship between syntactic category and coverage. On average, nouns are two and-a half times more useful than adverbs (coverage is 20% and 8% respectively). Verbs and adjectives are in between these two extremes (13% and 10% respectively). Possible explanations for this pattern include the fact that nouns generally refer to concrete objects that can be easily described and related to other objects. There are more of them, so they can refer to more specific concepts within a definition. Verbs, on the other hand, and especially adverbs, are more likely to be abstract concepts that are harder to define and more unpredictable in their use. There are less of them, so they have to be more general and "domain-free" in their use. In this respect, a noun that is strongly related to a particular domain will have much useful information in its definition (e.g. "*mortgage*", "*account*" and "*payment*" are all strongly related to finance). A verb or adverb, however, is unlikely to be as strongly constrained to one domain and will therefore be more general in its use (e.g. "*withdraw*", "*open*" and "*save*" all have a variety of meanings besides the financial sense). However, the descent into generality applies to all words, and is not confined to any one syntactic category.

It should be noted that this investigation was based on definitions derived from the CED. Other studies have shown the Oxford Advanced Learner's Dictionary to be more suited to the needs of a text recognition system. However, there is no evidence at this stage to suggest that the expansion of OALD definitions would descend into generality at a significantly different rate.

Conclusions: At present the semantic analyser uses only first level information. The inclusion of second level information would not increase the proportion of useful information to redundant information. There is no evidence to suggest that expansion to the third level would reverse this process of increasing generality. Filtering the definitions may go some way towards reducing this effect, and is suggested as an area for further research. There is no evidence at this stage to suggest that the expansion of definitions derived from a different dictionary would descend into generality at a significantly different rate. The correlation between relative frequency and coverage

is negligible, therefore there is no evidence to suggest that descent into generality can be avoided through the selective expansion of distinctive words. The strong relationship between syntactic category and coverage exists only at the first level of information. Descent into generality cannot be avoided by using syntactic information.

3.6 The Overlap Algorithm

3.6.1 Introduction

There are a number of variables associated with the overlap process. It is necessary to identify the effect of these variables on the performance of the semantic analyser, and to determine the values by which optimum performance is obtained.

Variable 1 - The complexity of the algorithm: There are two ways in which the overlap program can iterate through the word positions in the input data:

(1) The simple approach

1. Identify the sentence position (at the start this is word position 1, candidate 1). This word is known as the "active word".

2. Scan forward, within the window size (4 positions), overlapping the active word with each of the candidates in each position.

3. Calculate the score for each overlap and attach this score to the active word and the candidate word in each case.

4. Move on to the next candidate, make this the active word, and repeat steps 1-3 for all candidates in the data.

(2) The complex approach

This algorithm assumes that for each sentence position only one of the candidates is correct (this condition generally holds, except in cases where the correct word is missing from the list of candidates).

1. Identify the sentence position (at the start this is word position 1, candidate 1). This word is known as the "active word".

2. Scan forward and backward within the window size (4 positions), overlapping the active word with each of the candidates in each position.

3. For each word position, calculate the best overlap between each of the candidates and the active word, and attach this score (only) to the active word.
4. Move on to the next candidate, make this the active word, and repeat steps 1-3 for all candidates in the data.

The second algorithm is necessarily more complex than the first. This is because it assumes that only one of the candidates in each position is correct, so the assignment of scores to words has to be delayed until the maximum for any given position is known. For this reason, the algorithm needs to scan backward as well as forward, and therefore takes twice as long as the simple algorithm. The question is, does this added complexity and the assumption on which it is based add anything to the overall performance?

Variable 2 - Window Size: It is known from studies of collocations that the information derived from co-occurrence information is optimised at a distance of four words [Jones & Sinclair, 1974]. However, when other parameters are varied it may transpire that the optimum window size varies as well. For this reason, in each of the trials discussed below the window size is varied from 1 to 10 words.

Variable 3 - Strong Overlap versus Weak Overlap: Throughout the earlier investigations it became apparent that the contribution of weak overlap was considerably smaller than that of strong overlap. Furthermore, the computational overhead associated with weak overlap is much greater. Considering these factors, is its inclusion in the overlap algorithm justified? Would strong overlap on its own suffice?

Variable 4 - Definition Length Compensation: The larger the definition, the greater the chance of a successful overlap occurring by chance. This factor can be compensated for, by dividing the semantic score between two words by the joint length of their definitions. This should reduce the biasing effect of large definitions, but to what extent does it improve performance?

3.6.2 Investigations with the Overlap Algorithm

Objective: To determine the optimum settings for a range of parameters associated with the overlap algorithm.

Method: It was decided to investigate each parameter in succession, i.e. to investigate one, find the optimum, set this as the default, and then turn to the next parameter. The exception to this is the window size, which was varied from 1 to 10 words in each trial, for the reasons outlined above. The test data was as before (i.e. documents taken from the domains of *banking*, *estate agents* and *music*, after having been processed by the confusion simulator).

Results and Discussion: Figure 3.2 shows the performance of each algorithm, measured as a ratio between correct and incorrect choices. The complex algorithm is consistently superior, particularly when the window size is greater than 6 words. Evidently, there can only be one correct word in a given sentence position, and this result may reflect the effective exploitation of this constraint. However, the window size is currently set at four words, and at this distance the difference in performance is very slight. (NB - although this setting is said to represent the distance at which collocational information is optimised [Jones & Sinclair, 1974], it is evident that window sizes of two or even one also capture much semantic information. Other researchers (e.g. Jelinek et al, [1983]) have also found this to be the case.) Consequently, the marginal improvement offered by the complex algorithm may be insufficient to justify the increased computational overhead associated with this algorithm. To resolve this, the execution time required by each algorithm to process all three documents was measured, and the result is shown in Figure 3.3.

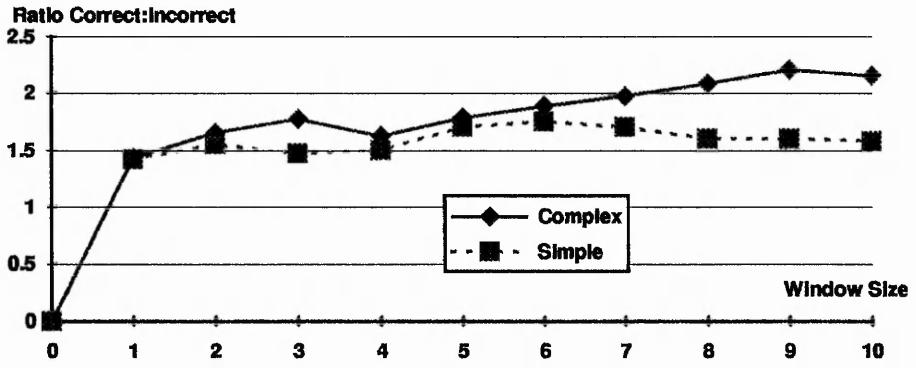


Figure 3.2: Effect of Algorithm Choice on Performance

Not surprisingly, execution time increases with larger window sizes in a fairly linear fashion, due to the greater number of word positions to consider. What is more important, the complex algorithm is shown to take roughly twice as long as the simple algorithm across all window sizes. This is understandable, since it must make

twice as many comparisons as the simple one. It would appear that the slightly superior performance of the complex algorithm does not justify the increase in execution time. The simple algorithm has therefore been be adopted as the default in subsequent trials. (N.B. - these timings are based on a prototype AWK implementation running under UNIX and are therefore unrepresentative of the current semantic analyser, which has been coded in C and optimised for efficiency.)

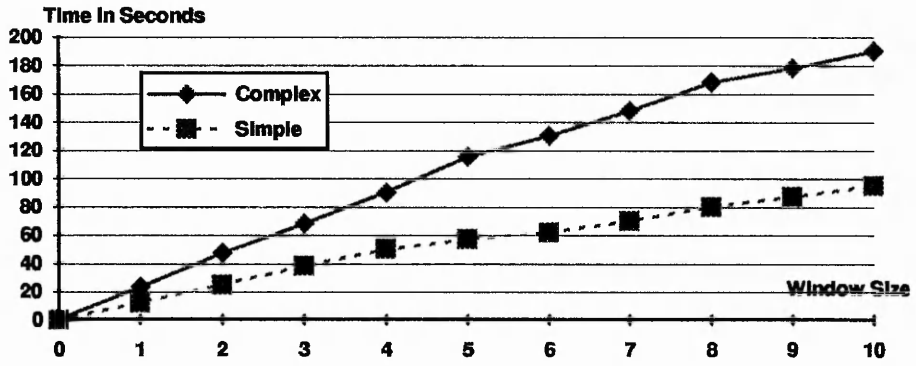


Figure 3.3: Effect of Algorithm Choice on Speed of Execution

Another aspect under scrutiny was the effect of domain on performance, which is shown in Figure 3.4. Evidently, once the window size exceeds two or three word positions in any domain, the performance tends to stabilise. When the window size is four words, the result for *banking* is 78% correct, *estate agents* is 58% and *music* is 50%. The average for all three domains is 61% correct. The percentage incorrect is 39%, which gives the ratio correct:incorrect of approximately 1.5:1.0. This can be cross-referenced with Figure 3.2.

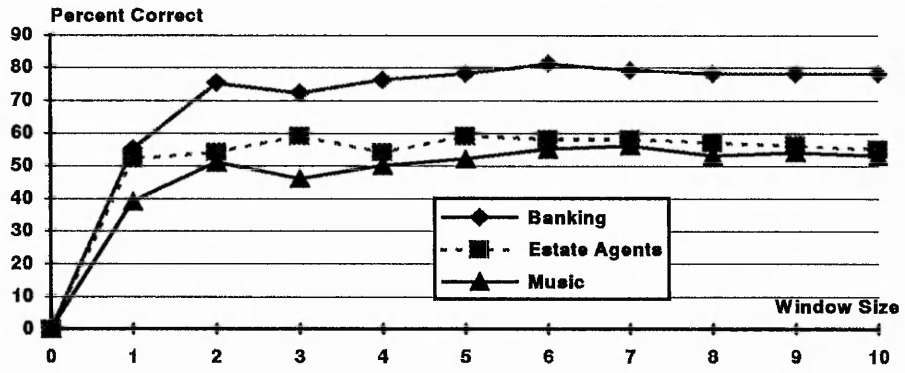


Figure 3.4: Effect of Domain on Performance

Another important parameter in the overlap process is definition length. To illustrate, let us consider the case of a word overlapping with two other candidates, the first of which has a long definition and the second has a short definition. All other things being equal, there will be a higher probability of a random overlap with the first candidate than with the second. As a result, some words will provide more overlaps than others purely as a result of the length of their definitions. This is particularly significant in the case of weak overlaps. However, it is possible to compensate for definition length by dividing the score assigned to any overlapping pair of words by the joint length of their definitions. Figure 3.5 clearly shows the improvement in performance obtained when scores are calculated in this manner. In future trials, therefore, definition length compensation is a default parameter setting.

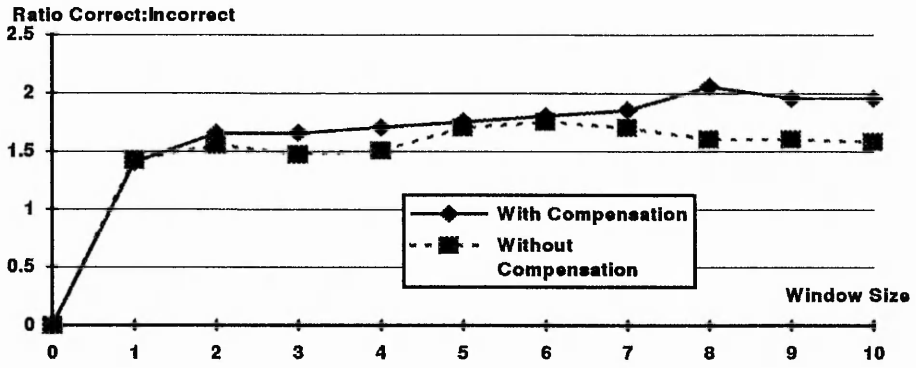


Figure 3.5: Effect of Definition Length Compensation on Performance

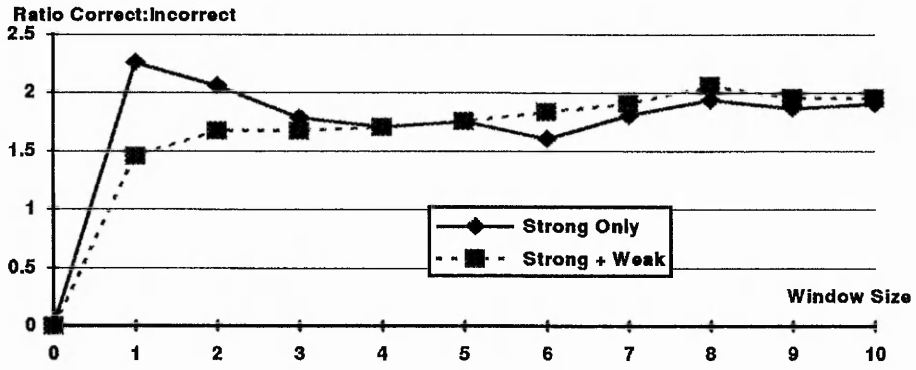


Figure 3.6: Effect of Weak Overlap on Performance

One further important aspect of the overlap process is the role of weak overlap in the allocation of scores. Would strong overlap on its own be sufficient? Figure 3.6 shows the comparison between two sets of scores; one based on strong and weak

overlap and the other based on strong overlap alone. With window sizes greater than four words, weak overlap has a beneficial effect, albeit minimal. Conversely, for window sizes of less than four words, strong overlap alone is the most effective. However, the peak on the left hand side of the graph is deceptive: with small window sizes and no weak overlap, the vast majority of positions remain unaffected; i.e. they remain as ties. To operate the program in this manner would therefore be pointless, since the vast majority of results would remain undecided. At a window size of four words the performance is identical, suggesting that the information obtained from weak overlap is redundant.

3.7 Another Choice of Dictionary

3.7.1 Introduction

As with all research, investigations are performed within the context of available time, resources and knowledge. For this reason, it is not always possible to carry out the preferred size or type of investigation until certain data or resources become available. For example, all the earlier investigations in this chapter have used either the CED or OALD. The LDOCE, widely used and recommended by many other language researchers, was simply not available. When it did become available, it was possible to repeat some of the earlier investigations with the CED and OALD using the larger sample size that was recommended, and to include the LDOCE as a further choice of dictionary. The subsequent acquisition of Longman's English Language Corpus (LELC) provided two further benefits: there was no longer a shortage of suitable test data available in machine readable form, and the raw corpus could be analysed to produce collocation dictionaries (see Chapter 4).

3.7.2 Investigations with LDOCE

Objective: To repeat the investigations of the OALD and CED with much increased sample size, and to compare their performance with that of the LDOCE.

Method: Fifteen documents were extracted from the Longman Corpus (LELC) and retained as test data, *not* to be used in any subsequent lexical processing (e.g.

dictionary creation, etc.). These test documents covered a wide range of domains (see Chapter 4 for a fuller discussion of the structure of LELC). Each of these test documents was at least 500 words in length, which compares favourably with the average of 200 words in previous investigations. Furthermore, where previously 3 domains had been investigated, there were now 15. These test documents were passed through the confusion simulator to produce alternative candidates as their output. This was then used as input to the overlap program that was run separately for each of the 15 documents and for each of the 3 dictionaries.

Results: The performance of each of the dictionaries across each of the domains is shown by Table 3.10.

	CED	LDOCE	OALD
Computing	79.6	71.9	69.9
Energy	66.7	70.1	74.1
Engineering	64.7	57.9	59.4
Business	69.9	74.3	68.4
Employment	62.9	70.8	61.3
Finance	66.7	73.1	68.7
Biology	69.2	72.4	72.3
Chemistry	76.0	76.9	71.4
Maths	67.4	62.9	56.9
Education	63.8	63.8	59.1
Medicine	65.8	67.9	63.2
Sociology	73.1	69.6	70.0
Economics	69.2	74.8	67.8
History	63.6	63.5	67.4
Politics	66.7	76.9	78.2
Mean	68.3	69.6	67.3
Std. Dev.	4.57	5.41	5.86

Table 3.10: Performance of each Dictionary for each Domain

Discussion: Evidently, the LDOCE outperforms the CED and the OALD. The main reason for this must surely be related to the manner in which LDOCE definitions are constructed. It is claimed that the entries within LDOCE are defined using a controlled vocabulary of about 2000 words, and that the entries have a simple and regular syntax [Boguraev & Briscoe, 1989]. This has the effect of reducing the *entropy* of the definitions, by cutting down on the randomness with which their constituent words are chosen. In so doing, the chance of strong (or weak) overlaps are

increased, since the probability of two semantically related words being defined using common terms is now proportionately increased. (Consider the nonsensical case where the core vocabulary is only half a dozen words - strong overlaps between words would be almost inevitable!). This reduction in the "noise" within definitions means that where semantic relations are present, the overlap technique is more likely to detect them.

Performance across domains, is however, highly variable, with no obvious pattern emerging. The CED is the most consistent, with 12 of the 15 scores being in the 60-70% range, and a standard deviation of 4.57. The LDOCE shows more variability, with 9 scores in the 70-80% range, and one particularly low score (57.9%, for engineering) which gives it a slightly higher standard deviation of 5.41. The OALD shows the most variability, with 5 scores in the 70-80% range and 3 in the 50-60% range. Consequently, this has the highest standard deviation; in this case 5.86.

As mentioned previously, the Longman Corpus consists of superfields that are in turn subdivided into subdomains. For example, the superfield *Applied Science* contains the subdomains *Computing, Energy and Engineering* (amongst others), The superfield *Commerce* contains the subdomains *Business, Employment* and *Finance*, and so on. Given that the eventual needs of a working system may be biased towards the domain of *Commerce*, the results for the *Business, Employment* and *Finance* documents take on a particular relevance. The LDOCE scores consistently in the 70-80% range for these documents, whilst the CED and OALD are both consistently in the 60-70% range. This result provides further justification for the recommendation of the LDOCE as the most suitable of the machine-readable dictionaries.

On the whole, these results are more reliable than those of previous investigations due to the vastly increased sample size. Whereas before the test data consisted of 3 domains, with a test document of 200 words each, there are now 15 domains, each document being at least 500 words in length. This means that the figure for the average performance (at the bottom of the table) is based on some 7,500 words of text.

Throughout this chapter it has been assumed that semantic relationships actually exist between words in ordinary sentences. However, it is possible that the definitional overlap effects observed were due to other factors. To this end, a further investigation was designed to test whether such semantic relationships exist in the texts studied. Pairs of words that had shown a strong overlap were selected from a

group of test sentences. A number of subjects (25) judged these to be semantically related compared to a control group of candidate pairs that had not shown a strong overlap (Mann-Whitney U test: $z = 5.977$, $p<0.0001$). This result supports the assumption that words within ordinary sentences exhibit genuine semantic relationships, and these can be identified by the definitional overlap process.

3.8 Summary

Dictionary definitions constitute a valuable source of semantic knowledge, and the definitional overlap technique has been shown to be a suitable method for applying such knowledge. The results of the semantic priming investigation provide independent evidence of its ability to identify semantically related words. The technique has been adapted to suit the format of text recognition data, and has been shown to be effective as a means of identifying correct words from alternative candidates.

Dictionaries differ widely in their style and content, and this has been shown to affect recognition performance. LDOCE is slightly superior to the CED and OALD in this respect, most probably due to its use of a core vocabulary. The way in which a dictionary is indexed is also important. A larger lexicon provides a greater coverage of English and therefore can be used to provide indexed definitions that are more representative of their verbal originals. However, such a lexicon may not always represent semantic relationships as effectively; the ratio of foot forms to inflections (i.e. the "grain-size") is also important. Further research of this issue is suggested.

The efficacy of definitional overlap varies greatly across domains, such that specialist dictionaries may be required for more esoteric or specialist domains. A technique for compiling such dictionaries is described, and the assessment of this technique together with the nature of domains in general are also suggested as areas for further research.

The expansion of dictionary definitions does not appear to provide any further useful information than that which is available at the first level. Instead, the information descends into generality in a manner that cannot be avoided by recourse to either word-frequency or syntactic information. It is suggested that semantic analysis using machine-readable dictionaries is restricted to their definitions and not their expansions.

Several aspects of the overlap algorithm have been investigated, and their optimal values (where appropriate) have been identified. These include:

- the use of a window size of four words;
- the use of the simple rather than complex algorithm;
- compensation for the length of definitions;
- the use of strong overlap only (rather than strong and weak).

The importance of a reliable and valid method by which results may be analysed has also become evident. A number of assumptions are related to any method of analysis, and these must be appropriate to the data to retain any validity. For example, if the process selects the correct word one in four times this does not necessarily imply 25% accuracy - it depends on the number of candidates from which each choice is made. The importance of adequate sample sizes has also been made evident. In addition, the presence of semantic relationships between words in ordinary sentences has been validated in an independent investigation.

Chapter Four

Collocations

4.1 Introduction

There are certain classes of English word combinations that cannot be explained using existing syntactic or semantic theories. For example, consider the use of "*strong*" and "*powerful*" in the following phrases:

to drive a powerful car
to drink strong tea

Both fulfil the same syntactic role (as an adjectival pre-modifier), and both make a similar semantic modification to the subject. However, to interchange them ("*powerful tea*" & "*strong car*") would undoubtedly be judged anomalous by most English speakers. These predisposed combinations are called co-occurrence relations or *collocations*, and account for a large proportion of English word combinations [Smadja, 1989]. Similarly, the notion of collocation may be explained with reference to the Oxford Advanced Learners Dictionaries of Current English (OALD):

collocate : ~ ***(with)***, (of words) combine in a way characteristic of language: *'Weak' ~s with 'tea' but 'feeble' does not.* **collocation**: coming together; collocation of words: *'Strong tea' and 'heavy drinker' are English collocations; So are 'by accident' and 'so as to'.*

These collocations could be regarded as extensions to the base meaning of a particular word; for example, in the case of "*weak*", we could regard the base meaning as "*lack of physical strength*" and then acknowledge modifications to this base meaning when used in the context of describing solutions. Indeed, this separation of meaning is reflected by the definition of "*weak*" in the OALD, with a distinct sense reserved for its use when pertaining to that of solutions. This is perhaps less surprising when one considers that these definitions are derived by examining and

grouping the actual collocations found for any particular word, and then working backwards to a definition from the separate contextual groupings [Mackin, 1978].

Collocations represent a further linguistic constraint upon text, and as such may be exploited by the semantic analyser. However, unlike dictionary definitions, there is no convenient repository from which to extract them. Evidently, it is necessary to compile some sort of "collocation dictionary" by automatic means. In the case of the human language processing system, collocations are learnt or compiled by experience, using feedback from language use, performance mistakes, etc. However, if collocations like "*weak tea*" and "*powerful car*" are so numerous as to evade any method of acquisition other than years of learning, how then should a machine-readable collocation dictionary be compiled? What type of collocations should be included?

One metric by which collocations may be measured and grouped is to rate them on a scale of probability. At the bottom would be little-used expressions and combinations, and at the other end would be uniform, predictable or fixed combinations such as clichés, sayings, metaphors, etc. A threshold could then be determined beyond which certain combinations could be deemed too infrequent to be worthy of inclusion within the dictionary.

Another metric by which collocations can be classified is according to the behaviour of the constituent words within the immediate context or *concordance*. Some collocations such as "*mortgage-property*" or "*insurance-client*" come about because both words are associated with the same context or subject domain. These may be referred to as *paradigmatic* or *conceptual collocates* [Smadja, 1989], and are characterised by an equal distribution of one term about the other within a given context (e.g. "*mortgage*" & "*property*" can occur anywhere within the same sentence in relation to each other).

Other collocates exist as lexical phenomena, and are referred to as *syntagmatic* or *lexical collocates*. In such cases, the order of the terms is important. For example, in business letters the words "*hesitate*" and "*contact*" may form a collocation, as in the phrase "*please do not hesitate to contact me*". However, because this is a stock phrase, it usually occurs as a fixed pattern. There is no longer an equal distribution of one term about the other: "*hesitate*" always precedes "*contact*" by two words.

4.1.1 Collocation Dictionaries

Previous methods of collocation dictionary compilation have included:

- the use of other dictionaries;
- using one's own "competence";
- corpus analysis.

The first method is uncreative and derivative, and the second is notoriously unreliable [Mackin, 1978]. The third, however, is both reliable and objective; and furthermore has benefited greatly from the advancement of computer technology and a proliferation of textual resources in electronic form. Increased processing power and memory capacity have greatly reduced the manual "word-crunching" that was previously involved in corpus analysis, rendering this a viable technique for the compilation of collocation dictionaries.

The precise format of a collocation dictionary depends largely upon the application. A "glossary" of sayings, proverbs, clichés, etc. for human use may require no more than a simple alphabetical listing of all recorded combinations; but for text recognition, the design of the dictionary and its subsequent use must be considered in conjunction, to reflect the run-time processing needs.

4.1.2 Collocation Analysis

There are many ways in which language can be analysed to produce collocational information. Phillips [1985] identifies three broad groupings:

(i) Classical Approaches: statistical approaches; viewing texts as random samples from a population described by a theoretical stochastic distribution;
(ii) Multi-Dimensional Scaling: conceiving the set of collocations of a word as its co-ordinates in multi-dimensional space;
(iii) Cluster Analysis: conceiving of each word as being characterised by its set of collocations, allowing subsequent fusion of individual data points.

Each has their relative advantages and disadvantages. For the purposes of the current project, the two major criteria were computational feasibility and rapid prototype development. The classical approach, as implemented by Berry-Rogghe [1970] proved the most suitable. This method is based upon an algorithm to determine the

likelihood that two adjacent words are truly a collocation, rather than an accidental association.

The process starts from a list of words for which collocation information is required. Very often this involves no more than manual selection, or simple extraction of the higher frequency words from a distribution taken on the corpus. The lower frequency words exhibit statistically unstable behaviour, because the sample of contexts in which they participate is too small to be an adequate representation. These lemmas are then analysed in turn, regarding each one as the "node" of a set of linguistic collocations. So in the case of "*money*", the corpus would be searched for all appearances of that word, and then the immediate contexts of each occurrence (the *concordances*) would be collected and truncated to extend no more than four words either side of the lemma. This four-word span or *window size* has been derived empirically and used effectively by other researchers (e.g. Sinclair et al [1970]). This collection of concordances now forms a subset of the corpus and can be treated to a separate frequency analysis, to discover the collocates of the original node (in this case the word "*money*").

A simple statistical procedure known as the z-score assigns a score to the strength of association between the lemma and each of its collocates. This distance relation score measures the degree of attraction between a lemma and its collocates, by comparing the frequency of a collocate in the concordances with the frequency that would be expected *were all words distributed in the text at random.* (For example, the frequency of the collocate "*pay*" in the context of "*money*" would be compared with its frequency outside of that context: this may return a high positive value, indicating a high degree of association, or a low or even negative value indicating non-association or even repulsion.) Lancashire [1987] has suggested the value of 1.49 as a threshold for association, although he provides no formal justification for this. A typical set of collocates for the word "*mortgage*" could be:

COLLOCATE	Z-SCORE
lend	9.45
property	8.13
advance	7.62
pay	5.67
increase	4.59
insurance	3.62
advice	2.56
term	1.81

Certain words, such as "*lend*", "*property*" and "*advance*" are highly associated with the node ("*mortgage*"); whilst "*advice*" and "*term*" show a weaker association. This information, given suitable formatting, can form the basis of a collocation dictionary and hence be used by the semantic analyser.

4.1.2.1 The Collocation Program

The AWK programming language has been used to implement a variation of the collocation technique described above. The program uses an algorithm based on that of Berry-Rogghe [1970], which is described in Appendix B. However, instead of producing a concordance represented as a list of z-scores, it reformats the collocates into a list headed by the lemma being analysed. The degree of repetition of any one collocate in the list corresponds to the degree of association between the lemma and that collocate. For example, the above concordance of the word "*mortgage*" could be represented as:

```
mortgage
[
lend lend lend own own own property property property advance advance
advance authority authority pay pay money money increase increase
insurance advice term
]
```

with the degree of repetition of each collocate representing the z-score divided by a constant (in this case 3.0) and rounded to the nearest integer. The value of this constant has been chosen so that the lists of collocations so produced are comparable in size with the definitions taken from machine-readable dictionaries. The format is also comparable, i.e. a headword followed by a list of related words in parentheses. The output of the program may be modified by adjusting a number of parameters:

(1) Window Size: the extent of the concordance around the node;
(2) Z-score Threshold: the minimum z-score for inclusion in the collocate list;
(3) Replication Constant: the constant by which z-scores are divided before inclusion in the list of collocates;
(4) Distribution Threshold: the level below which certain collocates are deemed to be of too low frequency to be statistically stable.

To date, the collocation program has been tested on a variety of corpora, including the 1 million-word LOB corpus, and a 5 million-word subset of Longman's English Language Corpus. In so doing, the program extracts and organises

information concerning likely sequences of words, based on analysis of genuine text (rather than subjective judgement, or ad hoc collection techniques). This information can then be used by the semantic analyser to discriminate between alternative sequences of candidate words, by comparing their collocational information. Since the format of the collocation dictionary is compatible with that of the MRDs used earlier, it can be substituted directly into the overlap program described in Chapter 3.

Initially, the program was run on a corpus of financial documents. It was appreciated at an early stage that the efficacy of collocational information could vary greatly according to the domain from which it was taken. For example, collocations extracted from the domain of *Banking* may be of little use when processing a medical report, since words like "*charge*" behave differently when used to describe a type of payment rather than a type of nurse. This financial corpus was relatively small, totalling 5,113 running words in its unedited state. After reduction and lemmatisation it comprised 2,344 lemmata. A frequency analysis of this corpus was produced, showing a distribution of some 651 types amongst these 2,344 lemmata. The first 156 items on a frequency ranked list were selected as input to the concordance program. Beyond this threshold the absolute frequency of each lemma was 2 or under; at which point the sample size becomes too small and the statistical behaviour unstable.

The output from the program was a domain-based collocation dictionary consisting of 156 entries, with headwords listed in alphabetical order and collocation information in the form of wordlists appended to each headword. Its efficacy could then be assessed by substituting it for the dictionary definitions in the overlap program.

4.2 The Pilot Study

Method: The collocation program was run on a corpus of financial text in the manner described above. A financial document that had NOT been used in the compilation of the corpus was selected as test data (this was the same document as used in the earlier investigations of Chapter Three). Simulated recogniser output was produced for this text, and the overlap program run on this data using the collocation dictionary as its source of information. Details of this algorithm are described fully in Chapter Three, so only a brief outline is given here:

(1) Iterate through the sentence positions and their candidate words;

(2) Compare the collocation list of each word with that of its neighbours (up to four sentence positions away);
(3) Record the number of "strong" and "weak" overlaps associated with each word - worth 50 points and 1 point respectively (NB - strictly speaking, it is inappropriate to talk of "strong overlaps" or "weak overlaps" in this context, but the terminology is used to indicate the consistency of the algorithm);
(4) Total up each word score and output finished sentence with scores normalised over the scale of 0-100 for each word.

Results: In many word positions the overlap process was biased since the correct word appeared in the collocation dictionary whilst the alternative candidates did not. Only the fair comparisons should be submitted to statistical analysis, and these fall into two categories:

(i) Cases where both the correct word and one other candidate had an entry as headwords in the collocation dictionary; or
(ii) Cases where none of the words in a position appeared as headwords in the dictionary, but some nevertheless received a score due to their inclusion in the collocation list of some other word.

The results were analysed in percentage terms and using the Student's t-test, as shown in Table 4.1. The t-score can be checked against statistical tables to determine the level of significance: (t [df 19] = 2.093, $p < 0.05$), and (t [df 19] = 2.861, $p < 0.01$). $3.04 > 2.861$ therefore: reject null hypothesis at 99% significance level; i.e. it can be said with 99% confidence that the technique selects the correct word in favour of other candidate words.

correct choices	60%
ties (no decision)	10%
incorrect choices	30%
average no. of candidates per position	3.06
%correct expected from random	32.68
z-score	3.04

Table 4.1: Performance of the Collocation Dictionary

Discussion: Evidently, collocational information can significantly improve the recognition process. However, this investigation used collocations extracted from a

domain-specific corpus, and test data taken from the same domain. Therefore, it is more accurate to state (so far) that collocation information "*extracted from and used within a specific domain*" can significantly improve the recognition process.

It remains to be seen whether this process can be extended to other domains. In theory, a collocation dictionary created from a large and general corpus should be sufficiently comprehensive to aid the recognition of text taken from almost any domain. To test this hypothesis, a further investigation was carried out.

4.3 The LOB Corpus

Method: The collocation program was run on the LOB corpus. A collocation dictionary of 7,130 entries was produced (after tidying and removal of empty entries), and this was used as the source of semantic knowledge in the overlap program. Test data was as in the previous investigation.

Results: An important difference with this investigation is the coverage of the collocation dictionary. In the pilot study, the collocational dictionary was very small, and therefore covered the just correct words (being from the same domain) and a few alternative candidates. However, the collocation dictionary extracted from the LOB consists of some 7,130 entries, and therefore covers a much larger proportion of the candidates in the data. So in every sentence position, many more alternative candidates are now able to compete with the "correct" word. In effect, the correct word now has more "competition". The results were analysed in percentage terms and using the Student's t-test, as shown in Table 4.2.

correct choices	74%
ties (no decision)	20%
incorrect choices	6%
average no. of candidates per position	3.06
%correct expected from random	32.68
z-score	2.84

Table 4.2: Performance of the LOB Collocation Dictionary

The t-score can be checked against statistical tables to determine the level of significance: (z [df 18] = 2.101, $p < 0.05$), and (z [df 18] = 2.878, $p < 0.01$).

2.84 > 2.101 therefore: reject null hypothesis at 95% significance level; i.e. it can be said with 95% confidence that the technique selects the correct word in favour of other candidate words.

Discussion: The LOB collocation dictionary makes a contribution to the recognition process that is significant to the 95% confidence level. We can conclude therefore that both domain-specific collocations and a general collocation dictionary extracted from the LOB corpus make a significant contribution to the recognition process. The domain-specific dictionary appears to perform slightly better than the general dictionary. This difference in performance may be explained by factors related to the dictionary compilation process.

The domain-specific dictionary has been compiled from documents associated purely with *Banking*. The senses of polysemous words (such as "*charge*" or "*rate*") are therefore most likely to have been used in the sense related to financial affairs. The collocations so formed are therefore representative of such financial material and hence will be more specific than those from the general dictionary. For example, consider the definition of the word "*access*" taken from the general collocation dictionary:

```
access
[
house house give give right terrace terrace terrace access access
access pupil pupil point west various usual trade route route road
require provide mean market important given gave freedom free found
experience ensure educate direct deny committee case cabinet cabinet
]
```

As can be seen, it refers most strongly to the physical sense of "*access*", with words like "*give*", "*right*", "*terrace*", "*trade*", "*route*", "*road*" & "*freedom*", etc. This is a reflection of the composition of the corpus from which it was taken. This entry may now be compared with the entry in the domain-specific dictionary:

```
access
[
addition build combine confident exception exception facility future
good knowledge manager month mortgage notice society sum time want
world give high instant instant interest money
]
```

It can clearly be seen that this entry indicates most strongly the financial sense of "*access*" (e.g. "*mortgage*", "*sum*", "*high*", "*instant*", "*interest*", "*money*", etc.). Hence, when used to disambiguate a document taken from the domain of *Banking*,

the second entry is more appropriate, because it is more representative of the likely word senses and collocations found in such a text.

This result implies that domain-specific collocations may be superior to general collocations in analysing documents from the same domain. However, it is not necessarily the case that these characteristics will be exhibited within other domains. Indeed, there is much evidence to suggest that many collocations found in natural text are *domain-independent*, and that only the analysis of a sufficiently large and general corpus will provide coverage of such structures. This issue is considered in further detail in the following sections.

It is possible that a domain such as *Estate Agents* will be adequately covered by the LOB corpus, due to its nature as a concrete, well-understood domain that is concerned with "everyday" words and concepts such as houses, towns, rooms, etc. In such a case, domain-specific dictionaries would be unnecessary, since the LOB provides adequate coverage of such words. To test this hypothesis, three different domains were selected and investigated.

4.4 Domain Dictionaries

Introduction: The above investigations have demonstrated the significant contribution of collocation dictionaries to the semantic analyser. However, these results were based on sample documents taken from a single domain. In the pilot study, a domain-specific dictionary was tested with a document from the same domain. In the second investigation, a general collocation dictionary was tested using the same document.

Although the pilot study demonstrates that a domain-specific dictionary can make a significant contribution to the recognition of a same-domain document, it is not necessarily the case that this effect will be repeated in other domains. Moreover, it is desirable to quantify the degree of reciprocity between domains - i.e., the extent to which collocations from domain X contribute to the recognition of text from domain Y, and vice-versa. For example, domains that are closely related may have a large number of collocations in common, such that the recognition of one could be facilitated by a dictionary taken from the other. Conversely, it may transpire that apparently similar domains make radically different use of those constituent words and hence demonstrate extremely different collocational patterns. Indeed, it is likely

that cross-domain recognition is constrained by the coverage of a particular dictionary. For example, it is unlikely that a *Banking* dictionary would have sufficient coverage to aid the recognition of medical texts, regardless of how suitable the collocations were.

Evidently, the general collocation dictionary derived from the LOB corpus can make a significant contribution to the recognition of domain-specific documents. However, it has only been tested on one domain. Therefore it is desirable to apply this dictionary to a range of domain-specific documents and to compare its performance with that of the appropriate domain-specific dictionary.

Method: Three domains were chosen for investigation: *Banking*, *Estate Agents* and *Music*. This choice reflected both the potential application of the eventual system and ease of availability. Corpora were built up in each case to over 10,000 words, then domain-specific collocation dictionaries compiled using the method described earlier. The size of each domain-specific dictionary is shown in Table 4.3 Test documents were selected for each of the domains, approximating to 17 sentences in each (the same documents as used in the early investigations in Chapter Three). Each of these documents was processed by the confusion program to produce simulated recognition output. The performance of each dictionary was tested using each of the three documents, and measured in terms of percentage correct and z-score (the t-score could now be replaced by the z-score since sample sizes were sufficiently large).

Dictionary	No. of Entries
General	7,130
Banking	1,022
Estate Agents	1,327
Music	1,713

Table 4.3: Sizes of the Domain-Specific Collocation Dictionaries

Results: This investigation involved four dictionaries and three documents, and hence gave a total of twelve combinations, as shown in Table 4.4. The scores in each column show a triple score representing the percentage correct/tied/incorrect, with the z-scores underneath. By way of comparison, the significance levels for a comparable sample size are as follows: (z [df 120] = 1.980, $p < 0.05$), (z [df 120] = 2.617, $p < 0.01$).

Test Data	DICTIONARY			
	General	Banking	Estate	Music
Banking	72/3/25 2.69	83/2/16 6.46	56/10/34 0.88	43/17/40 -1.09
Estate	68/7/25 7.18	53/20/27 2.57	68/12/20 5.51	45/22/33 -0.35
Music	58/6/36 3.19	41/21/38 -0.99	57/11/32 0.96	51/6/43 3.99

Table 4.4: Cross-Domain Performance of each Collocation Dictionary

Discussion: The general dictionary proved significant to the 99% confidence level across all three domains, which justifies the effort required to produce such a comprehensive general collocation dictionary. However, the spread of results was quite wide (z = 2.69 to z = 7.18). Indeed, the range of results may reflect the proportions of text-types represented in the LOB corpus. The high result for the *Estate Agents'* text may be a reflection the commonality between *Estate Agents'* literature and other genres contained in the LOB Corpus. For example, words that are high frequency in such text (e.g. "*buy*", "*house*", "*room*", "*door*", etc.) are concrete, everyday terms that are also high frequency words within other genres such as fiction, hobbies, DIY, etc. They may therefore be found within many other text types contained with the LOB, and used in a manner that tends to be consistent across each domain. This commonality is demonstrated by examination of entries taken from each dictionary. Consider the entry in the *Estate Agents'* dictionary for the word "*buy*":

```
buy
[
ability active add advantage afford aim alike auction average borrow
breaker cash certain charter consult consultant counsel couple
distribute fair feel find found general go grow guide happy leasehold
majority marry necessary opportunity package permit post potential
present probable prohibit raise reach reason reduce result safety see
specialise splendid step stop telephone tenth thatch waive whole wont
young arrange cent ideal likely lot new seek take want week finance
part people purchase sell house right home will first time
]
```

As would be expected, the majority of senses of the constituent words are related to property and its purchase (e.g. "*leasehold*", "*house*", "*home*", "*first*", "*time*", etc.). Such references to property purchase are also highly evident in the entry in the general dictionary (e.g. "*house*", "*money*", "*build*", etc.):

```
buy
[
house house money money build afford afford afford small save save
vote vote people make able store store sell rent part need cheap
cheap car book want ton ton stamp society run provide price paper
large income improve home farm expense cost buy bin bin white told
start rich public process politic pack operate library instance firm
encourage conservative colour activity wise win tool style site
secret sand risk remember purchase proportion property proper potato
market luxury likely invest hotel holding heavy finish feed favour
export equip enter enlarge dress distribute distinct department dear
client client clean charge champagne champagne cent cement cement
business box bird big appeal aerial advantage accuse
]
```

Although the entry in the general dictionary understandably includes a variety of collocations that are representative of other domains (e.g. "*vote*", "*politic*", "*conservative*", "*society*", and "*public*", all of which suggest an origin in parliamentary proceedings), there are still a large number of words common to both entries. These include "*advantage*", "*afford*", "*cent*", "*distribute*", "*home*", "*house*", "*likely*", "*part*", "*people*", "*purchase*", "*sell*", and "*want*". It is this high degree of commonality, of which the entry for "*buy*" is typical, that enables the general dictionary to provide a highly relevant source of information for recognising text from the estate agent's domain.

The performance of the domain-specific dictionaries varies greatly across domains (the *Banking* dictionary varies from z = -0.99 to z = 6.46). Each domain-dictionary achieved its highest z-score when used in the recognition of text taken from the same domain. This is to be expected, since the purpose of a domain-specific dictionary is to capture precisely those collocations that are specific to and therefore representative of that domain, in preference to any others with which the individual words may otherwise be associated.

Another objective of this investigation was to determine the extent to which collocations taken from one domain could aid the recognition of text taken from another. It was suggested that cross-domain recognition may be constrained by a lack of mutual dictionary coverage and the specificity of the collocations they represent. However, this was not exclusively the case. The *Banking* dictionary made a significant contribution (to 95% level) to the recognition of *Estate Agents'* text. This suggests that many of the language structures used in the *Estate Agents'* corpus are also present in the *Banking* corpus. Moreover, this result may simply be a reflection of the generality and "everydayness" of *Estate Agents'* text: just as the LOB gave

good coverage of this domain, the *Banking* corpus also contains language structures that are representative of estate agent's text.

However, this cross-domain recognition does not appear to be mutual. Although the *Estate Agents'* dictionary contributes to the recognition of the *Banking* text, it is nowhere near significant (z=0.88). This implies that the *Estate Agents'* dictionary does not cover a large enough subset of the language used in the *Banking* text to provide significantly representative collocations.

The *Music* dictionary contributes only to the recognition of *Music* text. For both *Estate Agents'* and *Banking* it makes a negative contribution (-1.09 and -0.35 respectively). This reflects the specificity of musical language and terminology, and its consequent inability to adequately represent the language structures found within other domains. What is more important, this result underlines the need for accurate & reliable domain identification. Evidently, this particular specific dictionary is only of use in recognising text from its own domain. Were it to be used on any other domain, it is likely that a negative (and therefore potentially damaging) effect would result.

From this set of results it is not clear whether separate domain dictionaries are a necessity. Although the specific dictionary out-performs the general dictionary for *Banking* text, for the *Estate Agents'* text the general dictionary out-performs the domain dictionary. For *Music* text the performances are comparable. However, it may transpire that the result for the *Estate Agents'* text is somewhat unrepresentative. The general dictionary works well in this case possibly because the "concreteness" and generality of *Estate Agents'* literature is well represented in the LOB corpus. Other domains, however, do not share these characteristics and may not be so well represented within the LOB corpus. For such domains separate dictionaries may remain a necessity. Alternatively, it may be preferable to adopt a hybrid approach, whereby the domain dictionary is merged into the general dictionary. Indeed, it may be possible to do this at run time, such that the domain dictionary is constantly dynamically updated over a large moving window by "remembering" patterns from the last N words written (where "N" is several thousand). To clarify these issues, a further investigation was set up, involving the processing of a much larger corpus and the testing of a greater number of domains.

4.5 The Longman Corpus

Introduction: The Longman English Language Corpus (LELC) is a collection of texts divided into mainly 40,000-word chunks, taken from over 2,000 sources (books, magazines, journals, leaflets, advertising material, etc.). The corpus can be subdivided in many ways; one of which is by subject area. The main subject areas are referred to as *superfields*, of which there are ten. These superfields are in turn subdivided into smaller subject areas, which are referred to as *subdomains*. The number of subdomains within each superfield is variable, as shown in Table 4.5.

Superfield	Subdomains
Natural & Pure Science	Maths, Physics, Chemistry, Biology, Astronomy
Applied Science	Engineering, Communications, Technology, Computing, Energy, Transport
Social Science	Sociology, Geography, Anthropology, Medicine, Psychiatry, Psychology, Law, Education, Linguistics
World Affairs	History, Government, Politics, Military, Archaeology, Economics, Development
Commerce & Finance	Business, Finance, Industry, Employment, Occupations
Arts	Visual Arts, Architecture, Performing, Media, Literary, Design
Belief & Thought	Religion, Philosophy, Occult, Mythology, Folklore
Leisure	Food, Travel, Fashion, Sport, Household, Antiques, Hobbies, Gardening
Fiction	General fiction, Historical fiction, Science fiction, Romantic fiction, Mystery, Adventure
Not Fiction	Poetry, Drama, Humour

Table 4.5: Structure of the Longman Corpus

The LELC is available on a custom basis, and the present project has acquired some 13 million words of text from this corpus. Unfortunately, these 13 million words are not evenly distributed across the 10 superfields. Fiction is heavily over-represented whilst many other domains are heavily under-represented. For this reason, the largest "balanced" corpus that can be derived from these 13 million words consists of 5 million words, and represents approximately 500,000 words from each of the 10 superfields. However, the problem does not end there, since within each

superfield the subdomains are not evenly represented. For example, the superfield of Applied Science should adequately represent all the subdomains listed above, but within the text so far obtained this superfield contains 6 computing texts, 3 engineering texts, 1 energy text, 1 transport text, and none on technology or communications. Due to this distorted coverage, it has been necessary to choose test data with some care, ensuring that they are adequately covered by the corpus. If this were not the case, then the collocation dictionaries would not adequately represent the test data.

Method: Fifteen subdomains were chosen for investigation, such that five of the ten LELC superfields were represented by texts from each of three constituent subdomains. The criterion for selection was mainly that of sufficient coverage in the corpus, as outlined above. A general collocation dictionary of some 12,475 entries was produced from the 5 million-word general corpus, and separate domain dictionaries created for each of the five superfields. The size of each dictionary is shown in Table 4.6. This table also shows the fifteen subdomains selected for investigation, and the superfields to which they belong.

Test documents were selected for each of the fifteen domains, approximating 500 words in each. No part of these test documents had been used in the creation of any collocation dictionary. Each of the documents was processed by the confusion program to produce simulated recognition output. For each test document, the overlap program was run once using the general collocation dictionary, and once using the appropriate domain-specific dictionary.

Superfield	Dict. Size	Subdomains Investigated
Applied Science:	4,056	Computing, Energy, Engineering
Commerce:	3,960	Business, Employment, Finance
Pure Science:	4,248	Biology, Chemistry, Maths
Social Science:	7,748	Education, Medicine, Sociology
World Affairs:	7,714	Economics, History, Politics

Table 4.6: Dictionary Size and Test Data for each Superfield

Results: The breakdown of scores in terms of percentage correct is as shown in Table 4.7.

	GENERAL	SPECIFIC
Computing	84.7	82.9
Energy	76.3	66.7
Engineering	70.3	68.4
Business	79.5	75.3
Employment	73.4	61.5
Finance	73.2	63.6
Biology	75.2	77.3
Chemistry	83.8	83.0
Maths	70.5	63.9
Education	68.7	88.7
Medicine	69.1	83.6
Sociology	64.1	73.1
Economics	83.6	94.4
History	70.8	80.0
Politics	77.4	88.6
Mean	74.8	76.8
Std. Dev.	5.95	9.95

Table 4.7: Performance of each Dictionary by Domain

Discussion: The average performances of the general and the domain-specific dictionaries are extremely close (they differ by only two per cent). This is somewhat surprising, since it would be reasonable to assume that domain-specific dictionaries would contain the most appropriate collocations for domain-specific documents. However, for 8 of the 15 documents, the general dictionary is more effective (by as much as 11.9% in one case).

Explanations for this inevitably concern (a) the content of the textual material used as data, and (b) the content of the collocation dictionaries. Evidently, any given document will consist of a variety of language structures, some of which will be general (i.e. not exclusively associated with any particular domain) and some domain-specific (i.e. with restrictions on word senses, etc.). This ratio of "general" to "specific" material will vary between documents and domains, such that a high proportion of "general" material may render the use of a domain-specific collocation dictionary less appropriate.

Moreover, the specific dictionaries were derived from smaller corpora than the GCD and therefore contained fewer entries: 5,545 (on average) compared to 12,475

in the GCD. Furthermore, although the domain-specific corpora were all the same length, due to variations in the type:token ratio the resultant dictionaries varied greatly in size (from 3,960 entries to 7,748 entries). Indeed, this variation in size very closely matches their performance: those larger than average tend to do better than the GCD, and those smaller tend to do worse. This variation in performance is further reflected by the higher standard deviation of the specific dictionaries.

Although the domain-specific dictionary outperforms the general dictionary on average, there are good reasons why the general dictionary would still be preferred in the majority of situations. The first is a practical issue: domain dictionaries are only of use if the domain has been accurately identified in the first place. It is not necessarily the case that this will have happened, nor can it be assumed that the document in question belongs to a given domain at all (it may be some sort of hybrid, or simply too ambiguous to fit neatly in one domain).

Another reason is that of coverage. A domain-specific dictionary may provide good performance within its own particular domain, but outside this its performance is brittle and inflexible. As we have seen from the above results many common language structures are domain-independent, and to provide comprehensive coverage of these a collocation dictionary must be based on as varied a corpus as possible. Additionally, good coverage is required to process all the *alternative* candidates produced by a recognition system.

A further reason is that of consistency. The specific dictionaries have a superior average, but their performance is inconsistent. The specific dictionaries show a standard deviation of 9.95%, whereas for the general dictionaries this figure is only 5.95%. An analysis of the ranges confirms this: for the general, scores vary from 64.1 to 84.7; for the specific, they vary from 61.5 to 94.4. The ranges are therefore 20.6% for the general and 32.9% for the specific. Considering this, the domain specific dictionaries can be said to be less reliable, and based on assumptions about the accurate identification of the domain that may not always be applicable. For this reason, the general collocation dictionary is suggested as being the more appropriate.

4.6 Summary

There are certain classes of English word combinations that cannot be explained using existing syntactic or semantic theories. These predisposed patterns are known as

co-occurrences or *collocations*, and account for many English word combinations. Collocations (and the concordances from which they are derived) have been successfully used for a variety of linguistic purposes. For example, they represent a valuable resource to the lexicographer in the dictionary building process, as they provide empirical information concerning word usage. Strictly speaking, collocations represent *syntagmatic* and *paradigmatic* knowledge rather than *semantic*, but it is argued that they represent the implicit application of syntactic, semantic and pragmatic knowledge [Sharman, 1990], and for reasons of simplicity have been referred to as a source of semantic information.

Collocations are similar to dictionary definitions inasmuch as they can be expressed as a headword followed by a list of semantically related words. However, unlike definitions, there is no convenient repository from which they can be instantly extracted. Instead, they must be compiled, as a product of corpus analysis. To this end, a number of algorithms were investigated, and one selected to be coded as an AWK program. Initially, this "collocation building program" was applied to a small corpus of financial documents. The small *collocation dictionary* so produced seemed highly plausible, so a previously unseen document from the domain of *banking* was passed through the confusion program to produce suitable test data. The result was that the "*collocational overlap*" technique made a significant contribution toward the identification of the correct words from the alternative candidates.

The next step was to "scale up" this pilot study. The collocation-building program was run on the 1 million-word LOB corpus, to produce a "general collocation dictionary" (or "*GCD*") of some 7,130 entries. This was tested using the same unseen *banking* document, and again gave a significant result, which suggested that collocation information compiled from a general corpus could be effective within a specific domain. However, to fully test this hypothesis, it was necessary to investigate more than one domain. To this end, three domains were selected, and test data for these domains gathered. As part of this study, it was additionally possible to investigate (a) a variety of domain-specific dictionaries and compare their performance with the general dictionary, and (b) the extent to which collocations compiled from one domain could contribute to the recognition of text from another. However, the results proved inconclusive. The domain-specific dictionaries made a significant contribution to each respective domain-specific text, but so did the general dictionary. The need for domain-specific dictionaries had therefore not been clearly proven. Cross-domain collocations did not appear to be effective, as was expected,

which underlined the importance of accurate domain identification when using domain-specific dictionaries.

The acquisition of the Longman Corpus enabled investigations to proceed in a more rigorous fashion using much increased test data sample sizes. A balanced corpus of 5 million words was extracted from this to produce another general collocation dictionary; this time of 12,475 entries. Five superfields of the Longman Corpus were selected for investigation, and 15 documents, representing three subdomains from each of the five superfields were extracted. These documents were all at least 500 words in length and had NOT been used in the compilation of any dictionary. Additionally, it was possible to create 5 domain-specific dictionaries for the domains under investigation, each based on at least 500,000 words of domain-specific text. The performance of these specific dictionaries was then compared with that of the new GCD.

For some domains the specific dictionary far outperformed the general, but the overall margin, on average, was only 2.0%. In fact, for many domains the general dictionary outperformed the specific. This suggests that there are many language structures that are not exclusively attached to any one domain, and that the only way to provide a collocation dictionary that is sufficiently flexible and comprehensive is to process as large and varied a corpus as possible. The inadequacy of the domain-specific dictionaries on these occasions reflects an attempt to constrain the highly unpredictable phenomenon of language by using too narrow a framework.

Furthermore, there were other reasons why a general dictionary may be preferable to a domain-specific dictionary. Firstly, specific dictionaries are only effective if the domain of the test data can be accurately identified in the first place. This may not always be the case. Besides, the text in question may not fit neatly within a single domain anyway. Secondly, the requirement of coverage suggests that many specific dictionaries will be inadequate due to their small size and inflexibility when used on text that strays from domain-based patterns. Additionally, good coverage is required to process all the *alternative* candidates produced by a recognition system. Thirdly, specific dictionaries are less reliable due to their inconsistency. Across the 15 domains they show a standard deviation of 9.95%, as compared to 5.95% for the general dictionary. In sum, the domain specific dictionaries are less reliable, lack sufficient coverage and are based on optimistic

assumptions about domain identification. For this reason, the general collocation dictionary is suggested as the more appropriate.

Evidently, there are a number of limitations to the collocation analysis technique. Firstly, it is based on lemmatised (root) forms rather than inflections. However, it is clear that some collocations only exist in particular inflected forms [Schuetze, 1993]. Consequently, it is intended to acquire inflected versions of the above collocation dictionaries and compare these with their lemmatised equivalents (using the same text recognition data). Secondly, the current technique makes no use of function words. However, these are an essential part of a number of important linguistic phenomena such as phrasal verbs [Sinclair, 1987]. It is intended therefore to incorporate such information into future acquisition methods, and compare the results with their "content-word only" predecessors. Thirdly, no use is made of word order information. However, linear precedence has been shown to be a significant factor affecting the manner in which words associate with each other [Church & Hanks, 1989]. Indeed, this is particularly relevant to a run-time recognition application, since data is usually input in one direction anyway (i.e. left-to-right). Consequently, the next phase of collocation acquisition will be to create a set of uni-directional collocations and compare them with their bi-directional equivalent. Finally, the current technique makes no use of distance information. Clearly, there are some collocations that are independent of distance, but there are others whose behaviour is highly distance dependent [Jones & Sinclair, 1974]. It is appropriate that future system development should exploit this constraint.

The acquisition of collocational information is still somewhat problematic. Whereas definitional information can be obtained from LDOCE for some 55,000 headwords, the acquisition of a similar number of collocational entries requires the processing of an immense corpus. The 5 million-word subset of the Longman Corpus can only reasonably provide a collocation dictionary of some 12,331 entries, allowing for reduction, repetitions and the low frequency of occurrence of some words. To provide anything like the coverage given by the LDOCE, a corpus of much greater than 5 million words is necessary [Jelinek, 1985]. It is suspected that the issue of lexical acquisition will form the basis of further studies.

Chapter Five

System Integration

5.1 Introduction

The most successful text recognition system to date is that of the human information processing system. Its principal strengths lie in the ability to (a) make selective use of available visual cues (for fluent readers much of the visual stimulus remains unattended [Just & Carpenter, 1987]) and (b) utilise an *understanding* of the text that can guide the reading process and compensate for any degradation or ambiguity within the visual stimulus. This is possible because word images occur within a meaningful context, and we are able to exploit the syntactic and semantic constraints of the textual material [Rayner, 1983]. Analogously, computerised handwriting recognition can be enhanced by using such higher level knowledge: for both printed and handwritten input, the stimulus alone is not enough to unambiguously identify the text. This is not to say that adequate recognition cannot be achieved *without* understanding, but rather an appreciation of the processes involved in understanding may facilitate the design of more efficient recognition algorithms and systems. The conspicuous gap between the reading performance of people and that of algorithms may reflect the fact that few text recognition systems utilise the many knowledge sources or recognition strategy of the human reader [Hull, 1987]. Evidently, if the design of text recognition systems is to be at all inspired, it may as well be by the best natural example available.

Ramsay [1987] argues that two types of knowledge are used in the process of understanding: *linguistic knowledge* (i.e. knowledge about language itself), and *world knowledge* (i.e. knowledge about the world in general). To some extent, this represents the current state of NLP system development: there are several examples of programs that demonstrate the effective application of some aspect of linguistic knowledge, but very few practical theories regarding the use of world knowledge.

This dichotomy becomes further evident upon closer analysis of the components of language processing systems.

5.1.1 The Components

To design a natural language processing (NLP) system, whether for text recognition, machine translation or some other application, it is necessary firstly to identify the requisite components, and then to specify how they must interact. Table 5.1 shows the levels of knowledge required by a typical system (in this case for understanding and interpreting English [Ramsay, 1987]). These levels need not necessarily correspond to autonomous modules within any computational implementation, but rather organised according to whatever configuration is most appropriate for the particular application. Indeed, the issue of autonomy remains highly contentious [Cairns, 1984].

Level	Subject Matter
world knowledge	what can be assumed
discourse rules	what can be said when
semantics	relationships and identities
syntax	rules about word order
morphological analysis	the significance of word endings
lexical analysis	words and word endings

Table 5.1: Levels of Knowledge for NLP (after Ramsay)

Let us consider each level of knowledge individually. The lower three levels (lexical analysis, morphological processing and syntactic analysis) have been investigated during the present project, and are described in detail elsewhere, e.g. Wells [1992] and Keenan [1992]. Similarly, the use of semantic knowledge is described in other sections of this thesis. Together, these levels represent the *linguistic* knowledge sources defined in the earlier dichotomy. Clearly, the work on these lower levels is far from complete, but the contrast between these and the higher levels (in which no investigations have yet been made) is conspicuous. Some would argue that above semantics lies a level concerned with the use of language in its social context. This level, referred to as *pragmatics*, takes into account the purpose of language in achieving pragmatic ends, such as persuading or requesting information [Greene,

1986]. However, for reasons of clarity and simplicity, this level will be regarded as being implicit within discourse rules and (to a lesser extent) world knowledge.

No attempt has so far been made to incorporate the two higher levels (discourse rules and world knowledge) within the present project. This reflects both the purpose of the current system (i.e. recognition rather than understanding), and the current state of NLP system development - i.e., the lack of practical theories regarding the use of discourse and world knowledge. For these reasons, it is appropriate to address these areas specifically in this section.

5.1.1.1 Discourse Rules

The meaning of a connected set of sentences is greater than the sum of their individual meanings. Consider the following discourse:

Harriet was hungry.

She walked over to the fridge.

Human readers have little difficulty in following the focus of this text, due to their ability to make elaborative inferences and recognise the thematic links that bind the sentences together into a cohesive whole. The presence of these links is often signalled by explicit cues within the text. For example, when a human reader sees a word such as "hence", "therefore" or "thus", they interpret it as a signal that the next sentence will express some consequence of what has just been said [Brooks & Warren, 1970]. Similarly, when they see words like "however" or "but", they interpret it as a signal that the next sentence will express something opposing what has just been said. These words, and other connectives like them, constitute lexical cues within text that help the human reader to maintain the coherence of a discourse (although inferences can still be made without such cues). Additionally, human readers are able to make a multitude of inferences about the sentences within a discourse. For example, backward inferences enable a reader to refer some new information in a sentence to something implied by an earlier sentence, and forward inferences allow the reader to embellish the representation of the currently read text, and create expectations about what is to follow [Clark, 1975].

This draws attention to another aspect of human discourse processing: the ability to construct the referential representation of a text or sentence. At the sentence level, this can involve tasks such as pronominal reference, which exploits linguistic

information such as gender, number and case. At the level of connected sentences, it involves the identification of focused elements (i.e. those which are more thematically central to a text) [Chafe, 1972]. This enables the reader to constrain their inferences and expectations to those that are the most relevant. At the level of a complete text, the main theme may be extracted by constructing high-level generalisations or abstractions [Kintsch & Van Dijk, 1983].

Discourse rules constrain the structure of a particular text genre (descriptive, narrative, exposition, etc.) and the purpose of that text (informative, entertaining, persuasive, aesthetic, etc.) [Brewer, 1980]. They enable the reader to activate the relevant template for the text and create expectations of how the text should progress. Apart from perhaps the use of story grammars [Mandler & Johnson, 1977] and the concept of macrostructure formation [Kintsch & van Dijk, 1983], very little progress has been made towards the implementation of discourse rules in NLP systems.

5.1.1.2 World Knowledge

Much of the information that makes a text coherent is not included in the text at all but resides in the world knowledge shared by the author and most of the readers. The importance of such knowledge becomes apparent when attempts are made to produce computer programs that can understand text [Schank & Abelson, 1977]. This knowledge has many facets: knowledge of people (their needs, wants, attitudes, values, plans, etc.), knowledge of physical laws, knowledge of cultural and social laws, etc. The "Harriet was hungry" example may be only nine words long, but it nevertheless demonstrates many aspects of knowledge that need to be made explicit to a computer before it could be said to understand. For example:

Human needs: the computer has no implicit notion of Harriet's status as a sentient being, nor her concomitant need for sustenance;
Physical laws: the computer has no knowledge of the biology or chemistry involved in the digestive process, and hence no conception of the way in which the intake of food satiates hunger;
Cultural laws: the computer has no knowledge of the cultural norms that would identify the fridge as being a likely repository for food.

There is also knowledge of specific content domains, such as the events and objects involved in attending a lecture or visiting a restaurant. In the latter case, the knowledge may include likely events such as sitting at a table, reading the menu,

ordering, waiting, being served, eating, paying the bill and so on. Attempts have been made to represent such knowledge using a structure known as a *script* [Schank & Abelson, 1977]. Scripts can be thought of as slot-and-filler structures, in which the slots have default values so that events can be inferred even when they are not mentioned explicitly in a text. They aid understanding by imposing an organisation on the information in the text, and providing any extra information required to maintain its coherence.

The acquisition, representation and use of world knowledge are all highly problematic issues, involving many (yet) unanswered questions. For example, how can such a vast quantity of knowledge be acquired? Which pieces of knowledge are relevant to a particular system? How should knowledge be represented, and within what framework should the inferences be made? How should expectations be passed between levels, and how strongly should they influence other processes? Until these questions are answered, no computer can be said to *understand* language. Consequently, recognition systems may not show the speed, adaptability and flexibility of the human system until they do.

5.1.2 Interaction between Components

The interaction between the various components involved in recognition (or understanding) is one of the major problems facing the NLP system designer. Steps towards its solution often begin with considerations of the I/O of the individual modules. In the present project, the semantic analyser has been developed to take input in the form of word candidates, and output those words with their associated scores. Consequently, the semantic analyser can be applied to any system within which word candidates are produced: handwriting, OCR, possibly even speech systems (see Appendices D & E). The same applies to the syntax analyser, so these two modules can run independently, in parallel if necessary, producing their own sets of results. The lexical analyser (which includes morphological processing) has been designed to accept input in the form of a character lattice (see Appendix C), so this can work with any recogniser that produces output in this format.

Consider the use of these modules in the design of an integrated OCR system. Given a TIFF file as the starting point, the data flow and processes could be organised as in Figure 5.1. The "voter" constitutes a module in which results are combined and a unique solution identified. This design has actually been implemented using a

network of transputers [Sherkat et al, 1993]. The architecture is such that processing begins in each module as soon as data becomes available, i.e. partial results flow along pipelines between the modules so that all may work simultaneously whenever possible.

It is arguable that the design of the system should be dictated by the optimal information flow, in which case the simplest approach would be to use these sources of knowledge in a serial fashion, with a uni-directional flow of information. In the case of text recognition, this would mean a bottom-up flow of data and results, working from the character recognition stage to the application of world knowledge. However, this approach has severe limitations, since at any level there are alternative interpretations (i.e. ambiguity) within the data for which the appropriate information is instantly available at some other level.

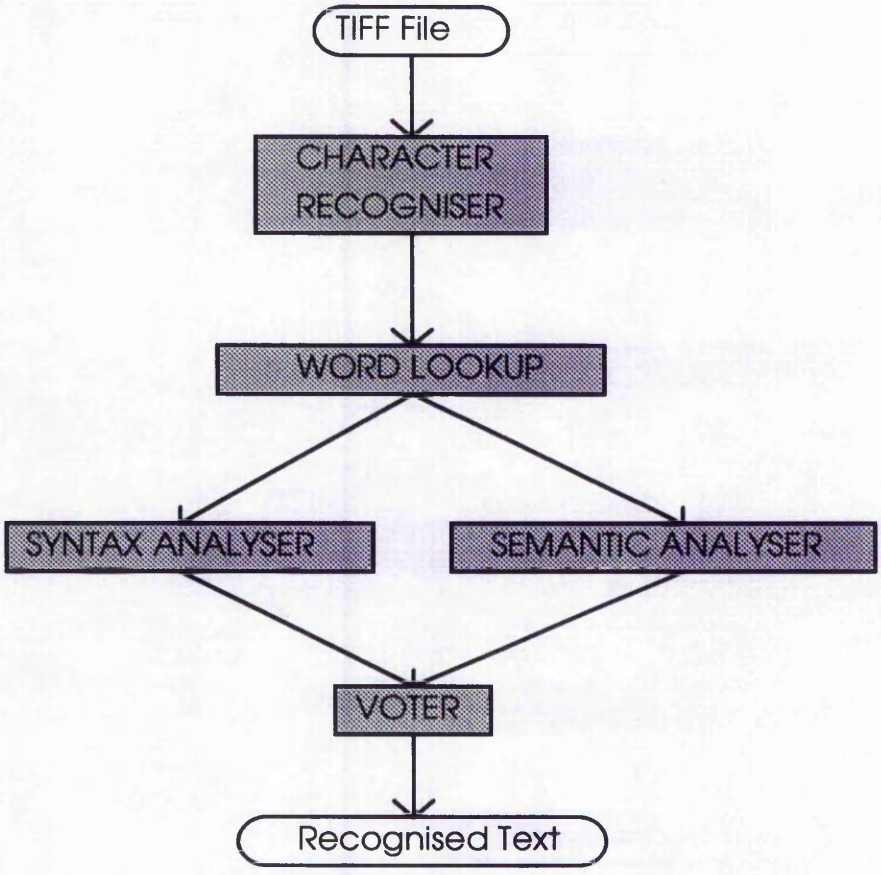


Figure 5.1: An OCR system design

There are two main alternatives to this design: the *blackboard approach* and the *constraint satisfaction approach*. In the former, the "blackboard" refers to a neutral

working area in which partial results may be stored and hypotheses offered to higher-level components. The HEARSAY project [Erman et al, 1980] used this architecture and has proved to be highly influential (albeit less than completely successful) as an example of collaboration between different levels of processing. The system can understand spoken utterances by simultaneously analysing them at different levels (including syntax and semantics), and then combining the results. However, such an approach incurs serious implementational difficulties concerning (a) the format of entries on the blackboard, (b) its coherence as results are added and deleted, and (c) control over the resources available to each module. Moreover, since the present project employs totally independent stages of syntactic and semantic analysis, major re-coding would be necessary to enable these modules to communicate in a meaningful manner.

The latter approach attempts to carry these ambiguities around in the form of constraints. In so doing, multiple interpretations are held simultaneously, and resolved when further data restricts those alternatives to just one unique solution [Stefik, 1981]. Neither approach seems entirely satisfactory, although the latter may have the advantage of facilitating a simple architecture, with a uni-directional flow of information from the bottom upwards. The problem is how to represent the knowledge at each level as multiple constraints that may be mutually conflicting. The OCR system illustrated above incorporates aspects of both approaches, since the information flow is uni-directional, but includes an element of parallelism and a neutral area (the voter) in which results are collected and resolved.

An important point concerning all modules apart from the character recogniser is that they are effectively acting as *filters* in this context, i.e. they cannot contribute *further* interpretations of the data. The character recogniser is the only module that actually suggests possible interpretations of the input - the other modules merely work on these suggestions, eliminating various possibilities or modifying their plausibility each time. In a genuine top-down system, hypotheses would be made concerning the expected input, and part of this process would be the contribution of possible interpretations from the higher levels. This issue constitutes an important aspect of the design of integrated systems, and one that is discussed more fully in Chapter Seven.

Even if the issue of architecture is resolved satisfactorily, or determined by some other overriding consideration, there still remains another problem: the

combination of results. How should the importance of each knowledge level (relative to the others) be determined? One possible solution would be to represent the relative influences of each analyser as a set of numerical weightings. For example, lexical knowledge may be twice as "important" as syntactic knowledge, which may in turn be twice as "important" as semantic knowledge (evidently, these magnitudes will vary according to the specific application and particular data). In this case, the relative weighting of lexical:syntax:semantics could be 4:2:1. These weightings may then be adjusted relative to the pattern recogniser. Indeed, these weightings could possibly be adjusted "on the fly", according to the degree of confidence associated with each analyser. However, the assignment of confidence ratings to the output of each analyser remains a highly contentious issue, and one that is discussed at greater length in Chapter Seven. An empirical investigation could provide some answers to this question, by testing a variety of permutations, running the system using a given input, and measuring the overall recognition rate for each permutation.

Evidently, there are many aspects to the question of integration. A comprehensive study (although highly desirable) would involve much time and effort, and is outside the considerations of this thesis. However, the importance of the semantic analyser relative to the other modules constitutes a narrower issue and one that can be investigated in a reasonable period of time. One of the simpler aspects of this is the relative importance of syntax versus semantics. To investigate this, both modules were integrated in the same program and relative weightings were adjusted using a variety of permutations.

5.2 Experimental Work

5.2.1 Syntax vs. Semantics

Introduction: Psychologists have often argued about the degree of autonomy between syntax and semantics [Forster, 1979]. Studies of language pathology (such as Broca's aphasia) seem to suggest that some aspects of syntactic processing can be the subject of independent dysfunction, and are therefore autonomous. However, this does not imply that such autonomy exists in the case of people whose abilities have not been impaired. Furthermore, there is evidence to suggest that people can understand sentences that are grammatically incorrect by using their world knowledge and semantic knowledge. This suggests that normal comprehension continues because

syntactic analysis operates in conjunction with other analyses [Just & Carpenter, 1987]. Either way, the manner in which they are combined by the present project can be seen as an empirical question - the syntactic and semantic analysers may be run independently, but their outputs must be combined as a single result. The problem is to determine the relative weightings of each analyser.

Method: A data sample of 807 words, complete with alternative candidates, was obtained from a suitable recogniser (in this case an OCR system with a word-lookup post-processor). This data was input to each analyser a number of times, varying the relative weightings of the syntax and semantics each time. The output from each analyser is usually normalised over some standard scale (typically 0-100), so to effect a change in relative weighting all that was required was to change the normalisation scale for one whilst keeping the other constant. For example, normalising both syntax and semantics over the scale 0-100 means that both have equal weighting. Normalising syntax over 0-100 and semantics over 0-50 means that syntax has a weighting equal to twice that of semantics. The permutations investigated can be summarised by the ranges shown in Table 5.2. The first trial corresponds to semantics alone; the second to syntax alone. The remaining five show a constant scale for syntax with semantics progressively increasing from one fifth to an equal weighting.

Trial	Syntax	Semantics
1	0	100
2	100	0
3	100	20
4	100	40
5	100	60
6	100	80
7	100	100

Table 5.2: Normalisation Ranges Investigated

Results: For each permutation, the performance of the system was evaluated in terms of (a) percentage correct versus tied versus incorrect, and (b) the t-score, which is a quantitative measure of how often and by how much the correct word was chosen in preference to other candidates. The results are shown in Table 5.3. The data chosen for statistical analysis consisted ONLY of word positions of semantic interest; i.e. those in which the correct word was a content word competing with a number of alternative candidates. The percentage correct therefore does NOT reflect the

recognition rate of the *overall* system - only the performance of syntax and semantics *on those word positions of semantic interest* (see Conclusion).

RATIO (syn:sem)	%correct/tied/incorrect	t-score
0:100 (all semantics)	59.74 / 7.79 / 32.47	2.01
100:0 (all syntax)	42.86 / 29.87 / 27.27	1.18
100:20	71.43 / 0.00 / 28.57	2.74
100:40	71.43 / 0.00 / 28.57	3.66
100:60	70.78 / 0.00 / 29.22	3.82
100:80	70.78 / 0.65 / 28.57	3.74
100:100	70.78 / 0.00 / 29.22	3.59

Table 5.3: System Performance for each Permutation

Discussion: The first point to note is that in other investigations the semantic analyser was making identifications that were approximately 75% correct. In this investigation, however, this figure is only 59.74%. The major difference in this investigation is that the data is real (i.e. taken from the output from a genuine recogniser) whereas in other investigations it had been simulated. This has inevitable effects on the nature of the input to the semantic analyser, and the manner in which this affects performance is discussed in the conclusion.

Syntax alone produces many ties (29.87%). When it does give a conclusive result, it makes a correct choice 42.86% of cases and an incorrect choice 27.27%. An example from the data shows the manner in which this large number of ties comes about. In this sample, the correct word is always in the first position:

```
a 2004
task 2165
consisting 2234
of 2180
giving 2235 erring 2235
prices 2314 pikes 0 juices 2314 pies 2314 ices 2314 Ices 2314
and 2241
advice 2223 add 1935 ad 2154
and 2229
taking 1993 faking 1993 tiring 1993 firing 2115 tang 2115 fang 2115
orders, 1654
```

In the many cases where the correct word has the same syntactic category as the alternative candidate, syntax cannot make a distinction between them. For example,

"*giving*" and "*erring*" are equally syntactically plausible in the fifth line, both being verbs used in the present participle. Similarly, "*prices*", "*juices*", etc. in line 6 are all plural nouns, so cannot be differentiated, and "*taking*", "*faking*", etc. are all equally plausible in line 10. It is the multiplicity of situations like these that lead to the large number of tied results in the output of the syntactic analyser, and it is these cases that need a further source of information to resolve them. Semantics alone produces fewer ties than syntax (7.79%). When it does give a conclusive result, it makes a correct choice in 59.74 % of cases and an incorrect choice in 32.47%. Semantics, in isolation, is therefore more decisive than syntax (it gives fewer ties) but less reliable (the error rate is higher).

Now let us consider the way in which altering the weightings affects the overall performance. At what level is the performance optimised, and what are relative weightings at this level? For weightings of 100:20 to 100:40, the overall performance is 71.43% correct, with 28.57% incorrect. For weightings of 100:40 to 100:100, the overall performance in percentage terms is slightly poorer (70.78% correct) but the highest t-score is obtained when the ratio is 100:60. So the optimum appears to be approximately 100:50, or something like 2:1 in favour of syntax.

It is also desirable to consider the qualitative differences between the two analysers: How often do they agree or disagree? Is there any pattern to the way in which they agree or disagree? There were 154 positions of semantic interest (as defined earlier). For each of these, the syntax analyser could be either correct, tied or incorrect, and the semantic analyser could be either correct, tied or incorrect. This constitutes a 3*3 array of permutations, as shown in Table 5.4.

	SEMANTICS		
SYNTAX	Correct	Tied	Incorrect
Correct	42	8	16
Tied	31	0	15
Incorrect	19	4	19

Table 5.4: Correlation between Syntax and Semantics

The results in this table may be examined in a number of ways. Firstly, we can add up the individual rows and columns and cross reference the result with Table 5.3 (e.g. adding up each column shows that the semantic analyser got 92 correct, 12 tied and 50 incorrect, which in percentage terms is 59.74% / 7.79% / 32.47%, as above).

Secondly, and what is more important, we can examine the way the two analysers interact within individual word positions. For example, the first row represents the 66 positions that syntax got correct. Of these, 42 were (also) correct by semantics, 8 were tied and 16 incorrect. This implies that where syntax is correct, there is a consensus from semantics. The second row shows that of the 46 ties produced by syntax, 31 were correct by semantics and the remaining 15 incorrect. This statistic implies that many of the ties produced by syntax may be favourably resolved by the semantic analyser (i.e. of the 46 syntax ties, 67.39% would be made correct by applying semantics). The third row shows the 42 cases where syntax is incorrect. Of these, semantics gets 19 correct, 4 tied and 19 incorrect, which implies that when syntax gets it wrong semantics can do little to help.

Analysis of the columns shows a similar pattern of results. When the semantic analyser makes a correct choice, it is usually shared by the syntax analyser. Of the ties produced by the semantic analyser, 66.66% could be resolved by the application of syntax. When the semantic analyser makes an incorrect choice, it can rarely be rectified by the application of syntax (of the 50 cases above, only 16 can be corrected by syntax).

Conclusions: Although semantics can detect more correct words than syntax (giving less tied results) it produces more incorrect results. The syntax analyser is therefore the "safer" (more conservative) of the two due to its lower error rate. It is possible that the use of a different data sample (e.g. text taken from a more "concrete" domain) could produce a different pattern of results, with semantics being the more reliable process. Conversely, a less semantically constrained text could produce a stronger bias towards the use of syntax. Such context sensitivity issues are suggested as the subject for further experimentation.

One important point to note concerns the quality of the input to the semantic analyser, and the possibility of a "garbage in, garbage out" situation. The input data consisted of 807 word positions, as identified by the recogniser (presumably the original text contained the same number). Of these 807 words, 413 were content words, and the remainder either function words or numerics. Of the 413 content words, 229 had been identified uniquely, and the remaining 184 had been identified with alternative candidates. These alternatives were *unranked*, so the word positions were effectively *tied*. The percentage correct/tied/incorrect for this data is therefore 55.45/44.55/0.00. This is the statistic upon which semantics (and syntax) must

improve. When semantic analysis is applied to this data, the ties become either correct or incorrect, or stay tied. Table 5.5 shows how the recognition rate changes according to the application of each analyser: starting with just the recogniser and lexical analyser (i.e. the "Basic" processing) and then adding syntax, semantics, and then both (weighted optimally).

System	Recognition Rate
Basic	55.45 / 44.55 / 0.00
Basic + Syntax	74.54 / 13.31 / 12.15
Basic + Semantics	82.06 / 3.47 / 14.47
Basic + Both	87.27 / 0.00 / 12.73

Table 5.5: System Performance for each Combination

There are other important aspects of the input data besides just the recognition rate. Firstly, there is the number of words found by the recogniser. This need not be the same as the *actual* number of words known to be in the original text. Segmentation errors in the recogniser, for example, may lead to a number of smaller words being concatenated to form one large one, or vice versa.

Secondly, there is the number of cases where input is detected but no valid interpretation can be made, i.e. the candidate list is empty and the correct word is therefore missing from the data. This may be referred to as the "Number of Holes". For the current data this statistic is zero, because the correct word is always present in each position (even if accompanied by alternative candidates). When this figure is high, it suggests that many of the content words upon which the semantic analyser relies are missing from the data, and the likelihood of identifying the correct word in the neighbouring positions is therefore reduced.

Thirdly, there is the number of alternative candidates in each position. Although these alternative candidates may be related graphemically to the correct word (i.e. possess a similar physical appearance), they are often completely semantically unrelated. This is, in effect, "semantic noise". This factor is therefore referred to as the "Noise Level", and expressed as the average size of the candidate list. For example, if each word in the data has been recognised uniquely (i.e. with no alternative candidates), the noise level is 1.00. A high noise level increases the likelihood of spurious overlaps and collocations, resulting in an increased probability of errors by the semantic analyser.

Table 5.6 shows the quality of the data used in this investigation. When comparing the results of different investigations it is important to consider each of these statistics, as they describe the quality of the input data. It follows that there may be some threshold for the quality of input to the semantic analyser, below which it becomes the "garbage in, garbage out" situation described above. The next investigation attempts to quantify this threshold.

Words Found	807
Number of Holes	0
Noise Level	6.50

Table 5.6: Quality of Data

5.2.2 Performance Thresholds

Introduction: Consider the case where the recognition rate passed on from the recogniser and lookup is very low. As far as the semantic analyser is concerned, there may be so much "noise" in the data that the incidence of spurious overlaps and collocations approaches that of the genuine overlaps and collocations. Furthermore, there may be so many "holes" in the data that many of the content words on which semantics depends are absent. In effect, a threshold has been reached beyond which semantics can do nothing to improve the recognition rate, and the application of such analysis would be a fruitless exercise. The detection of this threshold is an empirical issue. To this end, an investigation was set up in which the semantic analyser was applied to three different sets of data with different recognition rates.

Method: A number of handwriting samples representing text from three different domains was obtained. These domains were Business, Employment and Finance, and the length of each text was 521 words, 513 words and 520 words respectively. These samples were input to three different recogniser configurations, to provide a range of performance characteristics. The particular permutations investigated were as shown in Table 5.7. "Untrained" refers to a writer whose handwriting characteristics only featured minimally in the recogniser database. A "trained" writer is one whose handwriting features fully in the database. Version 2 of the recogniser is a modified implementation of Version 1, with generally improved recognition rates.

Combination	Recogniser	Writer
1	Version 1	untrained
2	Version 2	untrained
3	Version 2	trained

Table 5.7: Recogniser/Writer Combinations Investigated

Results: Table 5.8 shows both the quality of the input (as described in the previous investigation) and the effect of the semantic analyser on the overall recognition rate for Combination 1. The first four rows refer to the quality of the data: "Words Found" shows the number of words identified by the recogniser and lexical analyser; the "No. of Holes" shows the number of times the candidate list was empty; "Noise Level" is measured as the average length of the candidate list; and Top Ten shows (as a percentage) the number of times the correct word was present anywhere in the candidate list (for this investigation a maximum of ten candidates was allowed in any one position). The row labelled "Basic" shows the percentage correct/tied/incorrect for top candidate after pattern recognition and lexical analysis. The "+Semantics" column shows how this result is modified by the application of the semantic analyser.

	Business	**Employment**	**Finance**	**AVERAGE**
No. of Words	521	513	520	**518**
Words Found	575	516	534	**541.7**
No. of Holes	61	57	45	**54.3**
Noise Level	6.46	6.54	6.66	**6.55**
Top Ten	34.0	40.1	33.0	**35.8**
Basic	24.5/0.0/75.5	23.7/0.5/75.8	19.6/0.5/79.9	**22.6/0.3/77.1**
+Semantics	24.5/0.4/75.1	24.7/0.0/75.3	19.6/0.0/80.4	**23.0/0.1/76.9**

Table 5.8: Results for Combination 1

Evidently, the semantic analyser makes very little difference to the recognition rate of any of the texts. One major reason for this concerns the *availability* of the correct words. The percentage correct/tied/incorrect results in the table above are based on top candidate only - i.e. a choice is deemed correct only if the correct word is top of the candidate list (i.e. with the highest score). For the semantic analyser to improve performance, it must therefore turn an incorrect choice into a correct choice, i.e. *replace an incorrect top candidate with the correct word.* This can only take place if the correct word is *available* in the candidate list in the first place, i.e. somewhere

in the top ten. The semantic analyser cannot "create" the correct word out of nothing. This is why the row labelled as "Top Ten" is particularly relevant. It shows that on average, the correct word was in the candidate list 35.8% of the time. So even if the semantic analyser performs at 100% (i.e. promoting all *available* correct words to the top of the candidate list) then the overall system performance could still only be 35.8% correct. Evidently, with such "little room for improvement", it is perhaps less surprising that the semantic analyser failed to improve the overall recognition rate.

There are other aspects of the data that undermine the effectiveness of the semantic analyser. The first of these is the Number of Holes: in this case over 10% of the original words are missing. The second is the consistently high noise level: an average of 6.55 candidates in each position. The consistency of the noise level across the three domains at least shows that the performance of the recogniser is repeatable if not accurate!

	Business	**Employment**	**Finance**	**AVERAGE**
No. of Words	521	513	520	**518**
Words Found	518	500	511	**509.7**
No. of Holes	44	45	42	**43.7**
Noise Level	7.84	7.68	8.12	**7.88**
Top Ten	46.0	34.7	38.2	**39.6**
Basic	34.2/0.0/65.8	28.7/0.4/70.9	26.9/0.4/72.7	**29.9/0.3/69.8**
+Semantics	33.8/0.0/66.2	27.2/0.0/72.8	25.2/0.4/74.4	**28.7/0.1/71.1**

Table 5.9: Results for Combination 2

Table 5.9 shows the results for Combination 2. There are a number of differences concerning the quality of the data in this combination. Firstly, the Words Found more closely matches the actual number of words in the text, which suggests the use of a more accurate segmentation algorithm in the recogniser. The Number of Holes has decreased to 8.57% of the original text, but the Noise Level has increased to 7.88. There is a slight increase (3.8%) in the Top Ten rate over Combination 1, but since the top candidate ("Basic") recognition rates have increased by a greater amount (7.3%), the "room for improvement" has in fact decreased. It is suspected that this final factor may explain the inability of the semantic analyser to improve the overall recognition rate.

Table 5.10 shows the results for Combination 3. In this investigation, due to shortages of suitable handwriting samples it was only possible to investigate one domain. The number of Words Found is consistent with the example above. The Number of Holes has fallen sharply (to 3.47% of the original text), but this does not appear to have enabled the semantic analyser to change the overall recognition rate. Possible explanations could involve (i) the Noise Level, which having risen to 8.24% may be causing a greater number of spurious collocations; or more probably (ii) the Top Ten rate, which being only 67% again leaves little room for improvement.

	Business
No. of Words	521
Words Found	518
No. of Holes	18
Noise Level	8.24
Top Ten	67.0
Basic	56.7 / 1.1 / 42.2
+Semantics	57.0 / 0.4 / 42.6

Table 5.10: Results for Combination 3

Discussion: It is clear from this investigation that a threshold exists below which the quality of the input data is too poor to benefit from semantic analysis. This threshold has a number of factors: (a) the overall recognition rate (in terms of both top candidate and Top Ten); (b) the Number of Holes in the data; and (c) the Noise Level (measured as the average length of the candidate list). The number of Words Found is also of interest, but this may be considered as a further reflection of (a).

These three factors have been listed in what appears to be their order of importance. Regarding factor (a) it appears that in all the examples above the recognition rate was too low. This includes Combination 3, in which the Top Ten rate was 67%. It is still unclear to what extent the remaining factors affected the performance of the semantic analyser. To quantify the effect of factor (b), a further investigation was set up which is described in the next section. Some insight into the influence of factor (c) has been gained from the fourth investigation, described at the end of this chapter.

Evidently, the semantic analysis performs better with the artificially confused test data than with that of the systems used above. However, since the simulator

program [Keenan, 1990] produces confusions that are related to the input only by virtue of physical appearance (i.e., graphemically), the difference in performance can only be due to the quality of the data, measured using the terms described above. Essentially, the confusions produced by the simulator are less "disruptive" than the confusions produced by the recogniser since (a) there are fewer of them, (b) there are fewer holes in the data and (c) the Top Ten recognition rate is higher. To illustrate, consider the data used in Tables 3.10 and 4.7. This had an average Top Ten rate of 96.33%, an average Number of Holes of 3.8% (i.e. less than 4 holes for every 100 words) and a Noise Level of 3.13. Consequently, this data was above the "quality threshold", and the semantic analyser was thus able to perform effectively.

Regarding the use of semantic and syntactic analysis in general, it is important to consider some of the wider aspects. Low input data quality is not the only relevant factor. There are situations in which the use of higher-level processing may simply be inappropriate. For example, some applications (such as note-taking) may involve only weak syntactic constraints, and similarly there are situations in which the semantic constraints may be weak (e.g. processing names and addresses on letterheads). To employ syntactic or semantic analysis in such cases would incur a computational overhead for little or no performance increase. Therefore, to know when the use of syntax and semantics is beneficial, it is necessary to examine both the particular application and the quality of the output from the recogniser and the lexical analyser.

Conclusions: The above results have demonstrated three main findings:

(1) There is a threshold for the quality of data input to the semantic analyser, below which it can make no improvement to the overall recognition rate.

(2) Semantics can only "promote" words to top candidate *if they are present in the first place in the candidate list.* If the correct word is absent it cannot be "created". The Top Ten recognition rate is therefore of particular importance, as it shows the maximum overall rate that could be attained, were the semantic analyser to perform perfectly.

(3) The remaining important aspects of the input data are (i) the Number of Holes in the data and (ii) the Noise Level.

Following point 3(i), it was decided to examine how the performance of the semantic analyser changes as more words become "available for promotion" in the candidate list.

5.2.3 Candidate Availability

Method: The output from the recogniser and lexical analyser in Combination 2 in the investigation above was manually edited in the following manner:

1. For each word position, the mean of all the candidate scores was calculated;
2. Where the correct word was not present in this list, it was added, in the middle, with the mean score.

The data therefore consisted of the same top candidate recognition rate, but the Top Ten would now be 100% (i.e. if the correct word was not top candidate it was guaranteed to be elsewhere in the candidate list). This data was then input to the semantic analyser.

Results: The correct word is present in every position, as shown in Table 5.11. Before semantics, it was top candidate (on average) 29.9% of the time. After semantics, it was top candidate 32.9% of the time, an improvement of 3.0%.

	Business	Employment	Finance	AVERAGE
No. of Words	521	513	520	**518**
Words Found	518	500	511	**509.7**
No. of Holes	0	0	0	**0**
Noise Level	8.05	7.95	8.37	**8.12**
Top Ten	100.0	100.0	100.0	**100.0**
Basic	34.2/0.0/65.8	28.7/0.4/70.9	26.9/0.4/72.7	**29.9/0.3/69.8**
+Semantics	39.1/0.8/60.1	29.8/0.8/69.4	29.8/1.7/68.5	**32.9/1.1/66.0**

Table 5.11: Results using Data with High Candidate Availability

Discussion: Considering that in principle the overall recognition rate could now have risen to 100%, this is a fairly modest improvement. However, the "noise level", at 8.12, is still high. It may transpire that the semantic analyser is still being "swamped" by spurious overlaps and chance collocations. Even though the correct words in each position may sometimes be detected (as reflected by the 3.0% improvement) there are so many other relationships between the candidates that very often the strength of association between the correct words is outweighed by a chance relation between two others.

Conclusions: Evidently, the Top Ten recognition rate is important, as it describes the maximum overall recognition rate that could be obtained were semantics to perform perfectly. However, it is not the whole story. There is a compromise between ensuring the availability of the correct words in the candidate list and avoiding impossibly high noise levels. By adjusting parameters on the recogniser, it is possible to modify the output such that the correct word will always be present in an indefinitely long candidate list. The drawback is, of course, the unreasonable noise level that this would incur. With so many alternatives, finding the correct words would be extremely difficult. It was decided therefore to set up another investigation, in which the noise level was drastically reduced.

5.2.4 Reduced Noise Levels

Introduction: The chronological significance of this investigation was such that it was able to meet two objectives. The first of these was the need to obtain data in which the noise levels were reduced, i.e. the candidate lists were smaller. The second of these was the need to test the semantic analyser on other recognition systems and applications.

It was stated earlier that the semantic analyser could be applied to the output of any recognition system in which alternative word candidates were produced. This is because it operates on word candidates, irrespective of the source from which they were generated. As such, it can be applied to a variety of media, including handwriting, text or even speech. It transpired that as the above investigations were taking place, an OCR system was being developed which produced word candidates as its output. It was desirable therefore, to test the semantic analyser using output from this system, and assess the extent to which recognition rates could be improved.

Furthermore, the nature of the output from the OCR system was of particular interest. For reasons partially related to characteristics of the recogniser and to the nature of the medium itself, the output from this system was such that it would tend to perform either extremely well or else fail completely. In the former case, it would produce results in which almost every word was recognised uniquely, and where this was not the case, the correct word would usually still be present but with just one or two alternatives. As mentioned above, this was precisely the type of data that was needed to investigate the third aspect of data quality: the Noise Level. In effect, the noise level of this data was extremely low (and the number of holes practically zero).

It was decided therefore to scan a number of documents that had been degraded in a number of ways (e.g. photocopied light, dark, n^{th} generation, etc.) and present the resultant TIFF files as input to the OCR system. The output would then be subjected to semantic analysis, to see if recognition rates could be improved.

Method: A number of passages of text were obtained from a variety of sources (e.g. newspapers, journals, corpus extracts, etc.) and reproduced in a variety of fonts and point sizes. These paper copies were then photocopied under a range of conditions (e.g. light, dark, n^{th} generation, etc.). The degraded versions were then scanned to produce TIFF files of each, which were input to the OCR system. The output from this was presented as input to the semantic analyser.

Results: It would be inappropriate to present results for this investigation in the same way as those above. This is for a number of reasons. Firstly, it is extremely difficult to show an improvement in performance in the manner used above. This is because not only is the result for top candidate very close to 100% in the first place, but also because even where there are alternative candidates, the correct word is still ranked top of the list (unlike the OCR system used in the first investigation, this system ranks its alternatives). Therefore any scores assigned by the semantic analyser would have to be very great to change the rankings given by the recogniser. Instead, it would be more useful to evaluate the performance of the semantic analyser on its own, i.e. to consider solely those positions where there are competing candidates, and to measure the extent to which it identifies the correct word (regardless of any scores or ranks obtained elsewhere).

Another difference between the analysis of these results and those above is that the measures for the quality of data are no longer fixed. Many of the documents that were input to the OCR system produced 100% correct results, so there was no need for further analysis. In this respect, the "sample size" is no longer a fixed quantity, since the documents chosen for semantic analysis are an unrepresentative subset of the original sample. For this reason, the statistic "Words Found" is no longer relevant, and what is more important, the noise level and the number of holes can no longer be calculated accurately (unless all 65 output files are examined in great detail)! In short, the Number of Holes is negligible (practically zero), and the noise level is just slightly greater than 1.00 (since almost all words have been recognised uniquely).

From all this data, it was possible to find just 38 word positions in which the correct word had alternative candidates. The relevant files containing these positions

were identified and subjected to both semantic and syntactic analysis (as a comparison). The ambiguities were resolved as shown in Table 5.12.

	Semantics	Syntax
Correct	31 (81.6%)	15 (39.5%)
Tied	3 (7.9%)	20 (52.6%)
Incorrect	4 (10.5%)	3 (7.9%)

Table 5.12: Results using Data with Reduced Noise Levels

Discussion: The semantic analyser identifies the correct word in almost 82% of cases. It chooses an incorrect alternative for 10.5% of cases and leaves the remaining 7.9% unresolved (as ties). As mentioned above, it is of no value to translate these performance figures into overall performance figures, since the alternatives are already ranked and the correct word is nearly always top. In this respect, the overall performance stays at nearly 100% regardless of whether semantic analysis is applied or not, so the objective of improving the output of the OCR system has become slightly less relevant with this data.

However, regarding the second objective, it appears that lowering the Noise Level may have been the decisive factor in improving the performance of the semantic analyser. Clearly, when the results for syntax are analysed, it compares most favourably. The syntactic analyser may have a slightly lower error rate, but its propensity for producing ties (over 50%) is a serious disadvantage. The results for percentage correct strongly favour the use of semantics: 81.6% versus 39.5%. Admittedly, the sample size for this comparison is small (38 data points) but it does reflect the pattern of results seen in the first investigation.

The issue of interaction between syntax and semantics remains problematic. There were four cases where semantics chose an incorrect alternative. Of these four, syntax chose the correct word once and left the others tied. Of the three cases where syntax chose incorrectly, semantics chose the correct word twice and left the other tied. This reflects the lack of overall consensus between the two analysers seen in the first investigation. However, this is to be expected (and is indeed desirable) since syntax and semantics are different sources of knowledge and should therefore be independent. Were they to always agree, it would imply that one could be replaced by

the other with no loss of ability or degradation in performance of the overall system. This, for the human information processing system, is clearly not the case.

Conclusions: The performance levels shown by the semantic analyser are encouraging. Evidently, the lowering of the noise level seems to have been a major factor in the increase in performance. This result takes on a particular significance when the equivalent result for syntax is considered: despite the higher error rate, the semantic analyser has made the more valuable contribution for this set of data. If the recogniser was such that the alternatives were left unranked (as in the first investigation), the effect of the semantic analysis would undoubtedly have been to significantly improve the overall system recognition rate.

5.3 Summary

The most successful text recognition system to date is that of the human information processing system. Its ability to make selective use of available visual cues and utilise an *understanding* of the text enables it to compensate for any degradation or ambiguity within the visual stimulus. Word images occur within a meaningful context, and we are able to exploit the syntactic and semantic constraints of the textual material [Rayner, 1983]. It follows that computerised text recognition could be enhanced by using such higher level knowledge. However, attempts to define, represent and utilise such knowledge have met with little success. Of particular difficulty is the task of representing and using *world knowledge*, which plays such a crucial part in human comprehension and by definition also the reading process. Similarly, the rules of human discourse, which are known to shape language in all its forms have likewise eluded rigorous analysis.

A summary of the levels of knowledge used by language processing systems has been provided. The current system employs knowledge from four of the levels shown (lexical, morphological, syntactic and semantic), and this reflects the current state of NLP system development. Since attempts to represent world knowledge and discourse rules have proven extremely problematic, attention has turned to the existing levels, and in particular how these knowledge sources should be combined.

The issue of interaction between components remains unresolved. Various architectures for text recognition systems have been investigated, and a possible architecture for an OCR system has been illustrated. Two types of architecture have

been described: the blackboard approach and the constraint satisfaction approach. The strengths and limitations of each have been discussed. The role of each of the modules has been reviewed - in particular, should the higher levels be represented as filters, eliminating implausible suggestions from the lower levels, or should they actually suggest independent alternatives, as interpretations of the *expected* input? In either case, how should the results from each of the modules be combined? What should be the relative weighting of lexical, syntax and semantic information?

The first investigation investigated a variety of relative weightings between the syntax and semantic analysers, and demonstrated the lack of consensus between the two: they often chose different candidates as being the correct word. The performance of each analyser was assessed in terms of the effect on overall system recognition rates. This investigation also drew attention to the quality of the input data, with specific reference to the possibility of a "garbage in, garbage out" situation. Attempts were made to define measures for the quality of data to which the input to the semantic analyser must adhere. A number of metrics were identified, such as "Top Ten" (recognition rate), "Noise Level" and "Number of Holes". Were the quality of the input data to fall below a certain threshold (as defined using these metrics), then the semantic analyser could not be expected to make any effective contribution.

The second investigation attempted to quantify this threshold with respect to the recognition rate metric, by investigating a variety of recogniser/writer combinations. It was found that semantics could only improve recognition if words were *available for promotion* in the candidate list (i.e. the Top Ten recognition rate was considerably higher than the top candidate recognition rate, thus leaving "room for improvement"). In each of the combinations investigated, the semantic analyser failed to improve the overall recognition rate, due mainly to the lack of words available for promotion and poor quality as defined by the other metrics.

The third investigation explored one of those particular metrics: the Number of Holes in the data. It was found that even when this was reduced to zero, semantic analysis produced only a very modest improvement in recognition rate. Explanations centred on the one metric that so far remained unstudied: the Noise Level. The fourth investigation took data from an OCR system that was known to produce low noise levels in its output (this was primarily because it identified most words uniquely). The data was analysed to find word positions where there was one or more alternatives, and the texts containing these ambiguous positions were processed by the semantic

analyser. Of the 37 ambiguities found, in 31 it chose the correct word, in 4 it chose an incorrect alternative, and the remaining 3 it left unresolved. The syntactic analyser, by comparison, made 15 correct decisions, 3 incorrect decisions, and left the remaining 20 unresolved. This reflects the pattern of results of the first investigation, with the syntactic analyser producing the more conservative results. Again, there was a lack of consensus between the two analysers. This however, is seen as a desirable aspect: were they to always agree, it would imply that one could be removed with no effect to overall system performance. This, as far as the human information processing system is concerned, is clearly not the case.

Chapter Six

Domain Coding and Other Techniques

6.1 Introduction

It can be seen from the preceding chapters that both dictionary definitions and collocations can make a contribution to the text recognition process. There are, however, a number of other techniques worthy of investigation that have so far received only brief mention. These ideas involve less well researched types of semantic information, namely: *domain coherence*, *document structure* and *semantic classes*. They are less well researched in the sense that compared to dictionary definitions and collocations, the number of natural language applications for which they have been used is extremely limited. Brief outlines of each technique are as follows:

Domain Coherence - This involves the use of a coding system to identify the subject area with which a given word is normally associated. Such codes may then be used to represent the constraint of *subject continuity* throughout a passage of text.

Document Structure - Knowledge of the structure of typical documents can aid the location of discourse-based cues (e.g. headings, titles, etc.) that can in turn provide useful semantic information.

Semantic Classes - This involves the use of distributional statistics to identify groups of words that fulfil a similar semantic role. Such groups can then be used to specify a semantic grammar, and provide a further level of constraint to be exploited by a text recognition system.

It is difficult to draw fine distinctions between these techniques - they overlap to a certain degree and hence their enforced division is somewhat artificial. For example, document structure varies according to domain, the domain-based information obtained from a document can be influenced by its structure, and so on. For this reason, the theoretical background behind the use of both domain information and document structure will be considered concurrently under the general topic of discourse processing.

6.1.1 Discourse Processing

A document, or indeed any piece of text, is essentially just a finite series of sentences. Documents may be long or short, simple or complex, but ultimately they are all a collection of discrete linguistic units. However, there is something more to a whole text than separate units such as these. There is an "extra quantity" that distinguishes a whole, meaningful, coherent passage of text from a random selection of unrelated sentences. Moreover, it is the reconstruction and processing of these "extra meanings" that is vital to the process of comprehension and hence reading [Carpenter & Just, 1987]. Consider the example from Chapter Five:

Harriet was hungry.

She walked over to the fridge.

Human readers can understand the relationship between these two sentences, and hence the coherence of this discourse, through the activation of relevant knowledge sources and elaborative inferences. In this case, knowledge of human plans to relieve hunger and of locations of typical food stores generates the required coherence between the two sentences. The identification of the "extra meanings" can only take place once the initial sentences have been in themselves understood. This is a creative, active process, and one that relies upon the translation of input sentences into an internal knowledge representation. During the process of reading, these extra meanings bind the individual units into a complete text. The result is a cohesive meaningful whole, which in many ways represents more than the sum of the individual parts. The process by which superficially separate linguistic units contribute toward the creation of a cohesive whole is described as *discourse processing*. The knowledge structure used to represent these meanings is often referred to as a *schema* [Bartlett, 1932].

6.1.2 Schemata

6.1.2.1 Introduction

Schemata are used to aid the selection, interpretation and distortion of available information as well as in the retrieval, reconstruction and editing of stored information [Schwarz & Flammer, 1981]. For example, an author uses a schema in the creation of a text, and a reader uses a schema in understanding that text. Ideally, these two schemata should coincide; i.e. the reader reconstructs the writer's intentions perfectly from the available information (i.e. the text). However, in practice this is not always the case: the overlap between the two schemata may be less than perfect, resulting in a mismatch between the writer's intentions and the reader's interpretations.

The original idea of the schema has inspired a number of computational implementations of such knowledge structures, such as *frames* [Minsky, 1975] and *scripts* [Schank & Abelson, 1977]. These knowledge structures would be activated, like human knowledge sources, when deemed relevant to the context of the input data. Unfortunately, there is no reliable algorithm for the identification of relevant scripts, or how they should be structured, or what they should contain, or how detailed they should be, or indeed how they could efficiently be acquired.

The process of understanding is therefore dependent on the reader's accurate and reliable identification of the schema that embraces the complete text. Good organisation of a text should normally facilitate this process, enabling the reader to easily identify the general schema. However, if the text is badly organised, and the appropriate general schema is not easily identifiable, then the reader may have to modify his/her schema accordingly to minimise the discrepancy. This process takes time, and with short-term memory being limited, essential information may be lost. It is important, therefore, that this time period and the subsequent information loss are minimised.

6.1.2.2 Schema Identification

There are a number of techniques that a writer can employ to reduce the information loss on the part of the reader. One such technique is the use of a *thematic title* [Schwarz and Flammer, 1981]. The objective of a thematic title is to select and

activate schemata from an existing repertory of schemata, and thus provide the reader with the correct general schema as early as possible. This reduces the possibility of relevant material being dismissed before its contribution to the text as a whole is understood, and enables the reader to concentrate on the encoding of the important ideas. In other words, passages of text should be "labelled" according to their contents.

Another means by which a writer can guide the reader to the appropriate schema is by *initial mention*. It has been shown that in memory for prose, information near the beginning of a passage is recalled better than information appearing later in a passage [de Villiers, 1974]. Meyer [1977] explains this relationship by recourse to the "levels" effect, arguing that since the important information usually appears first in a passage, this information is recalled well due to its importance. This may be true, but perhaps the most interesting aspect of this argument is the underlying assumption that the most important information will be found at the beginning of a passage in the first place. In other words, it presupposes the existence of a linguistic convention that dictates where writers should place the most important information within a text [Kieras, 1980]. Initial mention, by this argument, is in itself a cue to importance. Another explanation is that writing styles have evolved to reflect primacy & recency effects [Baddeley & Hitch, 1974], which dictate that the beginning and end of any list of items will be better recalled than the middle, regardless of the nature of those items. The prevalent use of summaries at the end of expository texts may be a further reflection of this phenomenon.

Kieras states that the composer of a passage normally places the important information at the beginning of a passage to ensure that the reader immediately identifies it as important upon beginning to read the text. He argues that throughout our training in writing we are taught that a paragraph should usually begin with a "topic sentence". Similarly, Carpenter and Just [1977] postulate the use of a *discourse pointer* in text comprehension, whose initial state is determined by the initial portion of the passage. As the interpretation of the passage proceeds, the state of this structure is modified accordingly. Thorndyke's [1979] theories of prose comprehension suggest that the main subject should appear early in the passage to enable the reader to activate the relevant knowledge and select the initial schema. Clements [1979] has described some of the properties of a passage that signal thematic content, and these include surface level features such as initial mention and the topic-comment structure of individual sentences.

There is considerable evidence therefore that the important information carried in a text is presented at the start; i.e. in the title and first sentence(s). These discourse-based characteristics can be considered as aspects of document structure, and as such exploited by text recognition systems.

6.1.3 Discourse Processing and Text Recognition

The human information processing system is the most successful text recognition system to date, and as such it has provided an appropriate model for many computer-based systems. In the present project, the processes of lexical, syntactic and semantic analysis have all to varying degrees been influenced by models of human language processing. It seems desirable, therefore, to extend the capabilities of the current system to include aspects of human discourse processing.

However, there are important differences between the human and computational systems. Human readers, in understanding text, use their pre-existing knowledge sources in an active manner to translate the text into an internal representation, activate the appropriate schema and make whatever elaborative inferences are necessary to reconstruct the "extra meanings" implicit in the text. Computers, by contrast, have little or no pre-existing knowledge, no suitable internal knowledge representation, and no algorithm for the management of inferences.

Such significant differences suggest that the human model is too abstract and therefore inappropriate to a computer-based system. However, there are some theories of human discourse processing that are sufficiently well defined to be relevant to the development of text recognition systems. An example is Kintsch & van Dijk's [1978] theory of text comprehension and the concept of *macrostructure*. According to their theory, the reader constructs "macropropositions" that represent the gist or point of a passage at a global level. The resulting macrostructure is considered to be the global topic or theme of the passage, with irrelevant details omitted. This process model states that the first propositions to be read are held in a limited working memory, whose later contents are influenced by the relation of subsequent input to what is already being held. In other words, new data is interpreted partially according to its relevance to old data. The nature of this initial information is therefore very important, which reflects the linguistic convention that important information should be placed at the start of a text.

6.1.4 Domain Coherence

When a passage of text is written, the writer usually has some definite objective or intention in mind, rather than to just string random sentences together in an aimless series. This intention is normally associated with communicating information about some particular subject area, and consequently the text can be expected to demonstrate some degree of subject coherence. For example, a passage entitled "Particle Physics" that discusses particle physics in the opening few sentences can hardly be expected to suddenly switch to "music" or "cookery" or some other unrelated area. Furthermore, if the text has been well organised, this continuity of subject area should be reflected in the manner in which it is interpreted by the reader.

According to Kintsch & van Dijk's theory, when people read text they interpret new data partially according to its relevance to old data. In this way, the continuity of sentences is constantly maintained: one schema may be used to represent the global interpretation of the text, and other sub-level schemata may be invoked if necessary and relevant to earlier schemata. Although it provides a highly plausible model of this and other aspects of human discourse comprehension, Kintsch & van Dijk's theory uses a knowledge representation formalism that so far has no computational equivalent. Moreover, the absence of a practical algorithm would seem to preclude the development of an effective computational implementation. However, there is a source of semantic knowledge that can be used as a starting point. This source may be crude, static and limited, but it can nevertheless be used to reflect the domain coherence aspect of discourse processing. This knowledge source is referred to as *domain coding*.

Domain codes are essentially labels that may be associated with words to describe the domain or subject area with which they normally associated. For example, the word "stethoscope" may bear the code "MD" to represent *medicine*; "baritone" may bear the code "MS" to represent *music*; "neutron" may bear the code "PS" to represent *physics*, and so on. The form of association between a word and its code is derived according to need. For example, within a dictionary a word can be associated with its code by inclusion as a special field within each sense definition; computational implementations could use whatever data structure is most appropriate.

The codes themselves can be derived as a simple series of subject areas, or into a hierarchy whereby more specific domain codes could imply inheritance of a more

general subject area. For example, "neutron" could be given the code "PPPS" to show that it belongs specifically to *particle physics*, and also to the domain of *physics* in general. These codes could then be seen as belonging to a shallow tree, with general areas at the top and more specific subjects at the bottom.

Walker and Amsler [1986] used the subject category codes present in the LDOCE as a source of domain information for two separate tasks: (a) word sense disambiguation and (b) textual content-assessment. Their input data consisted of text taken from the New York Times Newswire Service. Their technique attempted the simultaneous solution of both of these problems, and consisted of the following steps:

(1) Assign to each word within the data all of its possible subject categories;
(2) Add up the totals of all the different subject codes - the category most frequently represented over the whole passage is deemed to be the subject area of the text.
(3) Identify the polysemous words within the text, and select the senses that carry the domain code chosen in **(2)** in preference to alternative senses not bearing this code.

Although they describe their work as "*far from the stage where we can expect to report definitive conclusions*", they claim that their progress towards meeting the two objectives "*definitely merits further development*". There is no attempt to provide a quantitative evaluation of their results. Slator [1989] also used the domain codes within the LDOCE for a similar purpose, but only after having restructured the hierarchy of the coding system. He suggests that the reorganisation gives a better intuitive ordering of the important concepts in each text, and enables a knowledge-based and context dependent strategy for making word sense selections. Other researchers to have successfully used the LDOCE codes for sense disambiguation include Jost & Atwell [1993] and Guthrie et al [1991]. The latter combined the codes with definition-based techniques by treating them as part of the defining word-set.

There are two ways in which the use of domain codes could contribute to the present project: firstly, for automatic domain identification; and secondly, as an aid to recognition. This chapter is concerned with both of these tasks. However, before examining them in detail, let us consider the means by which such codes may be acquired.

6.2 Domain Coding

A system of domain codes can be either created from scratch, or obtained from a pre-compiled source, such as a machine-readable dictionary (e.g. LDOCE). However, both these methods have their drawbacks. The first is impractical due to the sheer size of the task - a domain code lexicon of any realistic size would involve much repetitive labour. The second, although essentially a simple extraction exercise, is derivative and subjective since it produces a domain coding system that was originally compiled by human lexicographers. Furthermore, since the codes were designed for the human reader, they may not possess a structure or form that is suitable for use by NLP systems. By contrast, a third method has been developed that does not suffer from the above drawbacks and automatically produces domain codes from text corpora.

6.2.1 Domain Code Acquisition

6.2.1.1 The Corpus Technique

Domain codes are essentially a means by which words may be associated with subject areas. For example, the words "*mortgage*", "*loan*", and "*cheque*" are all related to the domain of *finance*, and a dictionary may reflect this by including the code for *finance* (and possibly other domains) within the entries for each of these words. If a list of all the words possessing this code was extracted, it may appear as:

cheque	FI
loan	FI
mortgage	FI
,	,
etc.	etc.

If the codes for other domains are ignored, this may be seen as a mapping between many words and one domain. Now consider a corpus of text from the domain of *finance*. A word-frequency distribution may be extracted, and the frequencies compared with those found in a general (undifferentiated) corpus to produce a relative word-frequency distribution, e.g.:

mortgage	53.50
loan	21.33
cheque	15.00
,	,
etc.	etc.

This is, in effect, a list of words and their *distinctiveness* within that domain, i.e. how strongly associated they are with that domain. Moreover, it constitutes a mapping between one domain and many words. It is therefore analogous to the domain code information described above, except that it is *quantitative* rather than *qualitative*. Instead of just labelling words with a code to say whether they belong to a given domain or not (such distinctions are not always clear-cut), the second method also provides a measure of the strength of this association.

6.2.1.2 The Acquisition Algorithm

The acquisition of domain codes proceeds on a domain-by-domain (i.e. corpus-by-corpus) basis. The procedure is identical for each domain, and is described by the following algorithm:

(1) Take the raw corpus and reduce it (by lemmatisation, removal of punctuation, proper nouns and function words, etc.) to its basic root forms (types);
(2) Produce a type-frequency distribution for this domain;
(3) Obtain the corresponding frequencies for each type from an undifferentiated (general) corpus - any types not found may be assigned a frequency of 1;
(4) Normalise these frequency distributions so that each type's frequency is expressed as a percentage of the total number of tokens within that distribution (without such normalisation, the general frequencies would be higher than the domain frequencies simply due to the general corpus being a larger sample of text);
(5) Compare the frequencies of each type within the domain and in the general corpus (i.e. divide the former by the latter);
(6) Output the list of types with their comparative frequencies (which provides a measure of their "distinctiveness");
(7) Select those words which have a distinctiveness of 3 or above - i.e. their frequency is at least three times greater in the specific corpus than in the general corpus (this threshold has been selected arbitrarily and should be subjected to empirical investigation);

(8) Normalise these frequencies by expressing them as natural logarithms (this effectively translates the exponential distribution into a linear relationship). The resultant file now contains those words distinctive to the domain, and a measure of their distinctiveness within that domain;
(9) Repeat steps (1)-(8) for all domains for which corpora are available.
(10) Merge the domain codes from each domain into a single file. This file now contains that section of the lexicon that displays specialised domain-based behaviour, and identifies the domains with which each word is associated, with a measure of the strength of that association.

This list can never be exhaustive: the acquisition of domain codes is another example of the lexical acquisition problem. The coverage provided by this list of codes can only be as complete as the corpora from which they are derived.

6.2.2 The Use of Domain Codes

Walker and Amsler used the LDOCE domain codes to identify the subject matter of stories on the New York Times Newswire Service, and to aid the process of word-sense disambiguation. Within a text recognition system, domain codes could be used for similar ends: firstly, to identify the domain of a particular text, and secondly, as an aid to recognition. Let us consider firstly the process of automatic topic identification.

6.2.2.1 Domain Codes for Topic Identification

6.2.2.1.1 Pilot Study

Method: Five corpora were acquired from a range of sources (e.g. manual input, email, commercial contributions, etc.), each consisting of approximately 10,000 words after reduction. Domain codes were acquired in the manner described above for each corpus (i.e. *Finance*, *Estate Agents*, *Music*, *Industry* and *IKBS*). Programs were written to read in text files of sample data, identify the domain codes of each word within the text, and then sum the value of these codes over each sentence. For example, consider the sentence fragment "*new style of savings account*" as test data and the following as a typical extract from the domain codes lexicon:

```
account      fi2.753
savings      fi2.656     es1.955
style        es2.019     in1.795
```

Each word in the lexicon is followed by a code ("fi" = *finance*, "es" = *estate agents*, etc.) and its strength of association. So, in this extract, "*savings*" and "*account*" are associated with the domain of *finance* with strengths of 2.656 and 2.753 respectively, whilst "*style*" and "*savings*" are associated with *estate agents* (strength = 2.019 and 1.955 respectively). "*Style*" is also associated with *industry*, at a strength of 1.795. In summing the domain codes for this fragment, the program would output the following totals:

```
Finance:          5.41
Estate Agents:    3.97
Industry:         1.795
```

Finance would therefore be identified as the domain of this fragment of text. The test data for this investigation consisted of 3 separate documents from the domains of *finance*, *estate agents* and *music*. Each test document comprised 200 words or more of naturally occurring text.

Results: After processing each sentence, the program would calculate which domain had scored highest for that particular sentence. When all sentences in a document had been processed, the program would output the total score for the whole text. Tables 6.1, 6.2 and 6.3 show the results for each test document. The first column shows the number of sentences assigned to each domain, and the second column shows the total score for each domain (for the complete text).

DOMAIN	Each Sentence	Complete Text
Finance	15	163.41
Estate Agents	2	72.94
Music	0	46.61
Industry	1	42.71
IKBS	0	37.73

Table 6.1: Results for Finance Text

DOMAIN	Each Sentence	Complete Text
Finance	2	71.29
Estate Agents	10	125.25
Music	1	28.99
Industry	4	56.92
IKBS	1	37.27

Table 6.2: Results for Estate Agents Text

DOMAIN	Each Sentence	Complete Text
Finance	4	42.77
Estate Agents	0	55.34
Music	9	125.26
Industry	2	74.22
IKBS	3	82.79

Table 6.3: Results for Music Text

Discussion: When analysed on a sentence-by-sentence basis, the reliability of the process varies dramatically according to domain. For *finance*, almost all sentences (15 from 18) are correctly identified as belonging to that domain. However, for *music*, there are just as many incorrect assignments as correct. *Estate agents* is somewhere in between these two extremes, with roughly twice as many correct assignments as incorrect.

The process is much more reliable when measured in terms of the score for the document as a whole. For all domains, there is a clear margin between the correct domain and the second placed. This suggests that the process could be used by a text recognition system to identify the correct words from the alternative candidates. In this respect, domain codes are a further source of semantic information, to be used alongside collocations and dictionary definitions.

This result also has consequences concerning the relationship between domain codes and document structure. In particular, would the use of domain codes applied to the title of a document result in reliable identification of the subject matter? The results obtained from single-sentence analysis would suggest not. However, it may be the case that titles warrant a particularly judicious choice of words by the writer, and as such can be expected to show a closer association to the domain than single sentences taken from the body of the text (psychological evidence would suggest that this is the case [Schwarz & Flammer, 1981]). The same can be said of initial mention, i.e. opening sentence or paragraph, etc. [Kieras, 1980]

One major limitation of the current implementation is the absence of any morphological processing (this is due to its nature as a brief exploratory pilot study). This means that the assignment of domain codes is done on a strict pattern-matching basis, with lemmas being ignored if they appear in any form except the uninflected root. The next stage in the development of this technique will be to create the intermediate stage that allows the program to recognise inflected forms of the types

for which domain codes are available, and to assign the scores accordingly. This will enable a greater number of domain codes to be brought to bear on any one sentence, and should thus capture some of the generalisations that are currently being missed by an inflexible pattern-matching process.

Another useful investigation would be to compare the domain codes acquired from the above technique with those acquired from LDOCE. So far, domain codes have been acquired for 5 different domains. When combined, the total number of types possessing codes is 2059 - just over one third of the complete lexicon for these 5 corpora. Each of these types has an average of 1.29 codes per word. This ratio corresponds well with that of the LDOCE, in which 18,000 words from 55,000 entries are marked as having specialised subject senses, with an average of 1.3 subject codes per word.

This investigation also highlights the immediate need for larger corpora - for studies to be credible they must be based on a representative (i.e. large enough) sample of language. The 1 million-word LOB corpus yields data on only 11,757 word types after reduction, although re-lemmatisation using a larger wordlist could increase this figure. (The use of the Longman Corpus in later investigations goes some way towards addressing this need.)

One final consideration concerns the organisation of the corpora. There are many issues to be decided empirically, e.g. should the domains be structured into a hierarchy? If so, how deep should it be? Should the corpora representing higher level domains be composed exclusively of the corpora immediately below them? In determining where one domain begins and another ends, what is the relative importance of human judgement versus statistical measures? It is expected that such issues will be resolved according to the results of further investigations and the perceived needs of the eventual system.

Conclusion: It is suggested that the corpus method of domain code acquisition is superior to the other methods for the following reasons:

- It is quantitative rather than qualitative, i.e. if a word belongs to two or more domains, it is possible to determine which is the most likely domain;
- It is application-specific, i.e. exactly those domains (and only those domains) required by the system may be processed and thus incorporated into the system;

- The hierarchy of domains can be designed to reflect specific structure of the perceived application areas;
- It requires no manual intervention or subjective judgement.

This study has shown that domain codes can be used to identify the subject matter of a complete document. This process is less reliable when performed on a sentence-by-sentence basis. Suggested extensions include:

1. The inclusion of morphology to enable inflections to be identified;
2. Creation and subsequent investigation of test data for the other domains;
3. A study of the relationship between document structure and domain codes;
4. Investigation of the relationship between corpus size and domain code acquisition.

A number of improvements have been incorporated into the following series of investigations. In particular, the algorithm has been extended to include item 1 above, and the acquisition of Longman's Corpus has enabled item 2 to be implemented. Let us now consider this corpus in closer detail.

6.2.2.1.2 Longman's English Language Corpus

The Longman English Language Corpus is composed of 10 major domains or *superfields*. Each of these is subdivided into a number of smaller topic areas or *subdomains*. For example, the domain of "*Commerce*" contains the subdomains "*Business*", "*Employment*", "*Finance*", "*Industry*" and "*Occupations*". The other nine domains are similarly divided into a number of subdomains, as described in Table 4.5 (Chapter Four). It was possible to derive codes to represent domains at either the superfield and subdomain level. However, for reasons of simplicity it was decided initially to test codes based on the superfields only, since this was more likely to produce a positive result. If this technique showed promise, then the extension of the lexicon to include subdomains would constitute a logical progression.

It was evident that the test data should not be extracted from texts that had been used in the derivation of the codes. The most appropriate superfields to investigate were "*Pure Science*", "*Applied Science*", "*Social Science*", "*World Affairs*" and "*Commerce*", since these superfields contained the subdomains that had performed *least* well in the earlier studies of collocations and dictionary definitions, and therefore offered the most "room for improvement". Furthermore, the other, more

arts-based superfields such as "*Arts*", "*Beliefs and Thought*", "*Leisure*", "*Fiction*" and "*Non-fiction*" exhibit a less constrained nature to their discourse, and are less likely to be directly relevant to foreseeable text recognition applications.

6.2.2.1.3 The LELC Codes

Method: Domain codes were derived from the Longman Corpus in the manner described above. The output from this process was a lexicon of some 10,623 coded indices (after lemmatisation & indexing). This lexicon is of the form:

```
11276 fi2.00 nf1.76
11275 ps2.34
11278 ss2.17
11279 be2.11 nf2.52
11280 ar1.79 wa1.98
11284 ar1.95
11288 ps2.27
11289 ar2.11 as1.91
... etc.
```

Each lemma is referenced by a unique index, and this has associated with it a number of domains and their relative strengths of association. For example, the index 11276 is associated with "fi" (*Finance*) at a strength of 2.00, and to "nf" (*Non-fiction*) at a strength of 1.76.

Sample data was extracted from the corpus, comprising 15 separate documents that had *not* been used in the compilation of the domain code lexicon. These 15 documents represented different subdomains from the 5 superfields identified above (3 from each). The test documents were each approximately 500 words in length. Programs were written to read in text files of sample data, identify the domain codes of each word within the text, and then sum the value of these codes over the whole text.

Results: For each of the 15 test documents, the result may be expressed in terms of whether the correct domain was identified as first, second or third choice, as shown in Table 6.4. The "winning margin" represents the difference between the score for the correct code and the score for the highest other code.

Choice	Number of Docs	Average Score	Winning Margin
First	12	82.7	51.1
Second	2	39.2	-24.5
Third	1	10.6	-8.7

Table 6.4: Performance of LELC Codes

Discussion: The correct domain is identified as first choice on 12 out of the 15 trials. The average score for these trials is 82.7, with a margin of 51.1 over the second placed domain. On the occasions where the correct domain is not first choice, it was second choice twice and third choice once. Of the second choice trials, one of them was *Applied Science*, which was mis-recognised as *Pure Science*, by a margin of 40. The other second choice trial was *Social Science*, which was mis-recognised as *Commerce*, by a margin of 9 (the "winning margin" in this case is therefore negative: -24.5). The third choice trial was a mis-recognition of the *Pure Science* document as Beliefs by a margin of 8.7 points. Unfortunately, it is not possible to quantitatively compare these results with those of Walker and Amsler. They report testing their technique on more than 100 Newswire stories, but their evaluation is qualitative: "*The results were remarkably good, considering that the system has not had any fine tuning... It works well over a variety of subjects... and a number of different formats - text tables, two-line abstracts...*"

Conclusion: The LELC codes made a correct domain identification on 12 out of 15 occasions. On the other three trials, the correct domain was identified as second choice on two occasions and third choice on the remaining trial. This would suggest that there is some room for improvement. As mentioned above, the threshold for distinctiveness above which words are included in the domain code lexicon is an arbitrary value, and is the subject of empirical investigation. At the moment, it is set at 3 occurrences. If this is reduced to 1 (so that any word that is at least as frequent in the domain corpus as in the general corpus will be included) then the resultant code lexicon will be much larger. The difference is as follows:

Original Lexicon = 10,623 coded lemmas
Extended Lexicon = 15,047 coded lemmas

It is possible that the extended lexicon, with its greater coverage, will be a more comprehensive source of information for automatic domain identification. To resolve this, the previous investigation was repeated but using the extended lexicon.

6.2.2.1.4 The Extended LELC Codes

Results:

Choice	Number of Docs	Average Score	Winning Margin
First	4	338.0	115.5
Second	9	254.2	-30.1
Third	1	164.5	-87.0
Fourth	1	238.0	-55.0

Table 6.5: Performance of Extended LELC Codes

Discussion: Extending the code lexicon in the manner described above appears to have degraded their efficacy as a source of knowledge for automatic domain identification. The correct domain is identified as first choice for only 4 of the 15 documents. Evidently, a threshold of 1.0 is simply too low to adequately identify those words that display domain-specific behaviour. The failure mode appears to demonstrate a bias towards *Applied Science*, with as many as 10 of the other documents being mis-recognised as belonging to this domain. Since the corpora from which the domain codes were derived were all roughly the same size (500,000 words) there seems to be no immediate explanation for this. A further investigation was set up to see if the LDOCE codes could make a more reliable identification.

6.2.2.1.5 LDOCE Codes

Method: Longman's Dictionary of Contemporary English contains many types of semantic information apart from the usual definitions. In particular, this dictionary has been compiled making extensive use of domain codes. Some 23,000 coded words can be extracted, which after lemmatisation, indexing & the removal of compounds & repetitions becomes a lexicon of some 12,000 coded indices. This compares favourably with the domain code lexicon extracted from LELC (10623 coded indices). Test data was identical to that used in the previous investigation.

Results: After processing each document, the program would output a list of scores corresponding to the totals of all the codes for each domain. However, on this occasion, there were two important differences. Firstly, the codes were qualitative rather than quantitative, so each score was a whole number based on the number of

times each code occurred, rather than a measure of the strength of individual associations.

Secondly, the criteria for success are different. In the previous investigation, the test data and codes had both been derived from the same coding system of 10 superfields, with 5 or 6 subdomains in each and so on. Therefore it was self-evident whether the program had selected the correct code for each test document. However, now that the data are from one coding system and the codes from another, there is a mismatch: a manual decision has to be made as to whether the code selected by the program adequately represents the domain of the test document. For this reason, the results of this investigation are presented in greater detail, as a series of the top three codes associated with each test document (Table 6.6).

Document	Top Three Codes (and Scores)
business	economics (44), business (41), law (29)
employment	business (54), medicine (51), economics (40)
finance	economics (45), business (43), military (29)
computing	data proc. (32), politics (26), military (23)
energy	science (60), engineering (38), military (23)
engineering	maths (40), business (40), engineering (38)
biology	medicine (60), science (34), politics (21)
chemistry	science (73), maths (36), linguistics (35)
maths	maths (46), education (29), sport (22)
education	education (42), politics (29), law (25)
medicine	medicine (56), business (17), maths (15)
sociology	politics (42), business (23), medicine (20)
economics	business (31), economics (30), law(28)
history	medicine (32), politics (22), history (21)
politics	politics (82), law (47), medicine (39)

Table 6.6: Performance of LDOCE Codes

Discussion: It is necessary to go through each of the trials and determine whether a correct identification took place. Given that there is no direct mapping between the coding system of the test documents and the codes used in LDOCE, a manual decision has to be made as to whether the first choice domain can be deemed the correct one. The results can be broken down as follows:

A direct mapping exists for 10 out of the 15 domains. Of these direct mapping cases, it picked the correct one as first choice 5 times, if one accepts that "*computing*"

and "*data processing*" are synonymous (a contentious point in itself). Of the other 5 direct mapping cases, it chose the correct domain as second choice in two cases (*business* and *economics*), third choice in the next two (*history* and *engineering*), and fifth choice in the other case (*sociology*). The remaining five cases have no direct mapping between document code and LDOCE code. Considered in turn, these are:

1. *Employment* - This has no direct equivalent in the LDOCE coding system, either as a subdomain or superfield;
2. *Finance* - This is designated as a subdomain of *Economics*, so in this respect the choice of *economics* as top is perhaps a correct one;
3. *Energy* - No direct equivalent in the LDOCE coding system, either as a subdomain or superfield;
4. *Biology* - This is designated as a subdomain of *Medicine*, so the choice of *Medicine* as top is perhaps also correct;
5. *Chemistry* - This is designated as a subdomain of *Science*, so the choice of *Science* as top is perhaps also correct.

So if we accept that certain domains have been "re-designated" by LDOCE (i.e. seen as fitting elsewhere in the hierarchy, and labelled accordingly), and that the identification of the correct domain should reflect this re-designation, then 2 & 4 from the list above fall into the correct first choice category. The overall result is now as in Table 6.7.

Choice	Number of Docs	Av. Winning Margin
First	8	21.9
Second	2	-2.0
Third	2	-6.5
Fifth	1	-25.0

Table 6.7: Performance of Re-designated LDOCE Codes

Conclusion: The LDOCE codes made a correct domain identification on 8 out of 15 occasions. Of the other seven trials, the correct domain was identified as second choice on two occasions, third choice on two trials, and fifth choice on one trial. In the remaining two trials the correct domain had no direct equivalent in the LDOCE coding system. It can be seen therefore, that the mapping issue considerably complicates the problem of the assessment of the performance of the LDOCE codes.

It is essential that the coding system used by an automatic domain identification program should reflect the classifications of the data to which it will eventually be exposed. If data exists for which it has no suitable category, then that amounts to an omission in its lexical database.

Another reason why the LDOCE codes perform so poorly when compared to the LELC codes involves the number of domains they cover. The LDOCE uses 120 two-letter field codes to denote "basic" subject areas, and 212 subdomain categories to constitute divisions of the basic field codes. Although the above study involved only the basic field codes, this still covers 12 times as many domains as the LELC codes. The chance of a correct identification is thus proportionately smaller. Furthermore, the distribution of codes between the various domains is much less uniform than that of the LELC. These problems become even more significant when the codes are used as an aid to recognition. A further, more detailed discussion in thus provided in the following sections.

Even when allowances are made for the mapping problem, the LDOCE codes still do not perform as well as the LELC codes. It may be concluded therefore that the LELC codes, in their original (unextended) form, are the most suitable for the task of automatic domain identification.

6.2.2.2 Domain Codes as an Aid to Recognition

As mentioned in the Introduction, there is a second process to which domain codes can contribute. This involves their use as an aid to on-line recognition, and requires the domain of the text to have already been accurately identified. This may be achieved by automatic means or otherwise.

6.2.2.2.1 The LELC Codes

Method: Test data was the same as the documents used in the previous investigations, but passed through a confusion program to simulate typical output from a text recognition system. The semantic analyser was modified to read in the domain codes (stored as a binary file) and to place them in memory alongside the usual dictionary definitions or collocations. It would then use this information in the following manner:

(1) For each sample text, a variable representing the domain of that text would be initialised to that of the correct domain (manually and in advance);
(2) The semantic analyser would identify the domain codes associated with each of the candidate words;
(3) Candidate words possessing the correct domain code would have their semantic scores incremented by the strength of their association with that code, multiplied by an arbitrary weighting factor. (This weighting factor would also be varied between trials, as an independent variable, to identify the optimum trade-off between domain and collocation information);
(4) The remaining element of the semantic score would be calculated as normal. In this case, the general collocation dictionary ("GCD") was used;
(5) The semantic analyser would then output the text as a list of candidates and associated semantic scores, which would then be subjected to the usual statistical analysis.

Results: The two independent variables were the domain type and the weighting factor. This produced the results shown in Table 6.8, with each cell containing the percentage of correct choices made. When the weighting factor is 0, scores are composed purely of the semantic score obtained through collocation using the general collocation dictionary. This can be compared to the remaining columns, which show the same collocation score combined with the domain code score multiplied by each weighting factor. Scores based solely on domain code information show as many as 90% tied results, and therefore have not been included in the above table.

	WEIGHTING FACTOR						
DOMAIN	0	1	10	20	30	40	50
Engineering	70.3	71.1	73.6	75.2	72.7	71.9	71.1
Maths	70.5	71.4	72.3	68.8	67.9	66.9	66.1
Sociology	64.1	64.1	64.9	65.8	65.8	65.8	65.8
History	70.8	70.8	71.9	72.9	72.9	72.9	72.9
Finance	73.2	73.2	72.4	73.2	73.2	72.6	71.2
AVERAGES:							
%correct	69.8	70.1	71.0	71.2	70.5	70.0	69.4
Ratio	2.74	2.82	2.93	2.94	2.83	2.76	2.67

Table 6.8: Contribution of LELC Codes to Recognition

Discussion: It can be seen that the optimum weighting factor for domain code information is around 20. The increase in performance obtained at this optimum is 1.6% (the average performance of the GCD across the five worst domains was 69.8% correct, and this increased to 71.2% for a domain code weighting of 20). However, the effect was inconsistent, with the magnitude of the optimum weighting and the effect on performance varying across domains (e.g. -1.7% in the case of *maths* to +4.9% in the case of *engineering*).

In isolation, domain code information produces as many as 90% tied results, due to their sparse lexical coverage. Domain codes therefore can only be used as part of a larger system, i.e. in conjunction with collocations or dictionary definitions.

Conclusion: Domain code information can be of use if given a limited influence within the semantic analyser. The following effects can be observed:

- A low weighting factor for domain code information increases performance. It can be used to identify the correct word between candidates that have equal scores from the collocation analysis. It therefore must be capable of identifying the correct candidate more than 50% of the time.
- When domain code information is heavily weighted, as in the right hand side of the above table, the overall performance goes down. This is because the performance of the collocation analysis is being outweighed or "diluted" by the poorer performance of the domain codes. In this context, therefore, the performance of the domain code analysis is less than 75% (which is the average performance of the GCD).

This investigation provides an analogy for the integration of the semantic processor into the system as a whole. There will be optimum weightings for the scores from each module, and these weightings will reflect the reliability of each process. It also raises questions concerning the derivation of the codes: would larger corpora have produced more reliable codes? Would a finer grain-size (i.e. number of domains covered) have given more accurate codes? These issues are suggested as areas for further experimentation.

There was little point in trying to use the extended codes to aid the recognition of the above test documents. The reason for this is as follows: If a document from a particular domain (e.g. *Commerce*) shows a code distribution with *Commerce* ("CO") firmly at the top, then looking for this code in the recognition phase and incrementing

candidate scores accordingly will be an effective way of differentiating the correct word from the alternatives. However, if the code distribution shows another code as top (e.g. "AS"), then looking for "CO" will be pointless, since a greater proportion of the correct words have the tag "AS" than the tag "CO". In fact, the lower down the distribution "CO" is, the less effective will be the strategy of looking for this code.

Using the extended code lexicon the correct domain is identified as first choice only four out of 15 times. This means that in each of the 15 code distributions, the correct one was placed second or lower 11 times. So if this code is used as the "sought for" code in the recognition phase, then only four times would it be the optimum. Of course, it would be possible to re-designate the documents and look for a different (re-designated) code in the recognition phase, but this would be a post-hoc modification based on very unsound principles.

6.2.2.2.2 The LDOCE Codes

Method: As above, except that the LDOCE codes were used in place of the LELC codes.

Results:

	WEIGHTING FACTOR						
DOMAIN	0	1	10	20	30	40	50
Engineering	70.3	70.3	71.1	70.3	69.4	69.4	68.6
Maths	70.5	70.5	73.2	74.1	74.1	74.1	74.1
Sociology	64.1	64.1	64.1	64.1	64.1	64.9	64.9
History	70.8	70.8	70.8	70.8	70.8	70.8	70.8
Finance	73.2	73.2	73.2	73.2	73.2	72.4	72.4
AVERAGES:							
%correct	69.8	69.8	70.5	70.5	70.3	70.3	70.2
Ratio	2.74	2.74	2.82	2.80	2.78	2.78	2.77

Table 6.9: Contribution of LDOCE Codes to Recognition

Discussion: In this investigation, the optimum setting for the weighting is a factor of 10 or 20 (although 10 has a slightly higher Ratio figure). The increase in performance obtained at this optimum is 0.7% (the aggregate performance was 69.8% correct, and this increased to 70.5% for a domain code weighting of 10). This is half the increase in performance obtained using the LELC codes. Again, the results were inconsistent, with both the magnitude of the optimum weighting and the resultant performance

increase varying across domains (e.g. 0.0% in the case of history to 2.7% in the case of engineering).

These codes appear to have performed poorly for both topic identification and as an aid to recognition. The reasons for this may be related to the origin or *derivational history* of each system of domain codes. Consider the LELC domain code lexicon. This represents a mere 10 domains, and provides reasonably even coverage across all of them. The LDOCE codes, however, were derived from a categorisation developed by Merriam-Webster, and have a background that differs in three important ways: (a) they have been designed for human rather than computational use, (b) they represent a greater number of domains, using a 2-level hierarchy, and (c) the codes are highly unevenly spread across those domains.

Rather than covering a mere 10 superfields, the LDOCE covers 120 "basic" domains, with 212 subdivisions to be found therein. So the sheer number of domains covered by LDOCE means that the chance of a successful match is proportionately smaller. Furthermore, since many of these LDOCE domains are highly specialised, they may be regarded as "noise" within the system, adding to the problem of finding a reliable match between a candidate word's code and the correct code. However, there are solutions to this problem. For example, Jost & Atwell [1993] built a collocation table of domain codes to represent those that co-occur in a sentence in a training corpus, as part of a technique for word-sense disambiguation. The table could be used to assign "absolute preference" to an exact match, and also secondary preference to related (i.e. collocating) domain codes. It is envisaged that this method will form an area for further research.

A further consideration involves the test data itself. This is chosen by hand to represent the experimenter's intuitions about what represents a typical sample from a domain. So given a choice from the 10 LELC superfields, most human observers could make an accurate domain identification. The LDOCE scheme, however, provides so many different codes that a number of these may be applicable to any given text. For example, for the *Commerce* (i.e. *finance*) text there was no directly equivalent code so this had to be represented by either "EC" (*economics*) or "BZ" (*business*). The lack of a coherent mapping between the designated domains of the test data and the designated domains in the domain code lexicon inevitably creates a further source of inaccuracy and poor performance.

It also raises issues about the definition of domains in the first place. Evidently, some sort of standard is needed, so that the classification process applied to the data reflects the same hierarchy of domains as used in the construction of the domain code lexicon in the first place. This classification process must define all areas to be covered (arts, sciences, both, everything?) and how these are to be subdivided such that all essential subject areas are accounted for. The composition of Longman's Corpus (like many others) has been determined by human experts, and therefore represents a subjective quantity. The assignment of documents to domains has also been determined by human judgement. It follows therefore, that its structure may not be the most suitable for a given application. In short, it may be possibly to restructure or re-classify any corpus to more effectively meet the needs of a specific application such as text recognition. Indeed, there may be no such thing as a perfect structure, but the presence of an agreed standard could eliminate some of the mapping problems that contribute to the poor performance of the LDOCE codes.

The spread of the codes also means that the matching process will be biased towards certain domains. For example, "MD" (*medicine*), with 2,153 occurrences is the most frequently occurring code in LDOCE. The next most frequent is "PL" (*politics*), with 1,179 occurrences. "MD" will therefore inevitably come out top given a random sample of text. Since these codes were derived from a categorisation developed by Merriam-Webster, they constitute a human artefact and as such represent the intuitions of individual lexicographers. Any bias towards certain domains, therefore, reflects the subjective perceptions of their creators. Why there should apparently be more medical terms in the English language than any other type is at this stage not clear - but presents an interesting question nonetheless.

A possible explanation for the lack of success of the LELC codes is that the acquisition technique has not been optimised. Within the constraints of time it is therefore suggested that a further investigation be designed to analyse an alternative code gathering algorithm, perhaps with revised frequency thresholds or aimed at the level of subdomains rather than superfields.

Conclusion: Domain code information can be of use if given a limited influence within the semantic analyser. It chooses the correct candidate more than 50% of the time but less than 75% (when domain code information is heavily weighted the overall performance goes down, as the benefit from the collocations is outweighed by the poorer performance of the domain codes).

There may be highly specialised situations where the domain is highly restricted, and collocations are difficult to gather or for some reason less reliable, in which case domain codes may have a useful contribution to make. Apart from this however, it is unlikely that the aggregate 1-2% improvement given by their presence would make their derivation and use a universally worthwhile exercise.

6.3 Other Methods

6.3.1 Document Structure

The linguistic convention of placing important information at the start of a text can be used to focus the semantic analyser on areas likely to invoke the correct schema. Such areas form an initial source of semantic information and provide a starting point for the process of semantic analysis. In this way, the structure of a document can be used to direct certain aspects of the semantic processing, since document *structure* provides constraints on how the document *content* should be processed.

Document structure, like many other aspects of naturally occurring text, is an amorphous, ambiguous quantity. It represents a further level of abstraction, and as such is subject to the same degree of ambiguity that permeates the other levels (syntax, semantics, etc.). It has been suggested that standards for document structure (such as ODA, a document architecture standard, and SGML, a document mark-up language) could be used to aid the recognition process. However, such standards can only be put to their full use on a completely recognised document, as they are based more upon the logical structure of a document than its layout. The appearance of a document may be a reflection of its logical structure, but it is ultimately the logical structure that dictates the selection of SGML tags or ODA formats. Such information could only be reliably provided by the writer of a text, and for this reason these standards are of limited use in the recognition of "unseen" text.

It is possible, however, to use such standards in a limited capacity. Firstly, pre-defined document structures could be used as templates to constrain the input of text in a dynamic system, e.g. a form-filling application or similar. (Although, since such an application would no longer be based on running English text, corpus-based semantic information would be of limited use. For more constrained applications, a

simpler approach to semantics may be appropriate.) Secondly, they could be used to aid the identification of significant document elements such as headings or titles. It is envisaged that further uses for document standards will be identified as development on the underlying techniques progresses.

6.3.2 Semantic Classes

6.3.2.1 Introduction

Semantic classes are essentially groups of words that may be categorised according to their semantic behaviour. For example, within the domain of banking, the words "*deposit*", "*current*" and "*savings*" all behave in a similar manner since they can all be used to modify the word "*account*". Therefore, they can be said to belong to the same semantic class. Although there may be no convenient name for such groups (e.g. "account-modifiers" in the above case) this is not the issue; what is important is the membership of the groups, not the names given to those groups.

Semantic behaviour is not a reflection of syntactic class. The above words all belong to a variety of syntactic classes ("*deposit*" can be noun, verb or adjective; "*current*" can be a noun or adjective; and "*savings*" is a noun but can be used as an adjective within this domain). Instead, what unites them is their semantic behaviour. It should therefore be possible to identify groups of words that display common semantic behaviour, and place them within a discrete semantic class. Some words fall easily into a semantic class; with others the group so formed may seem somewhat artificial. However, it should be possible to define classes for most content words in most domains.

Various attempts have been made to automate the generation of semantic classes (e.g., Hirschman, Grishman & Sager [1975]) and much important background work on classification and *clump-finding* was performed by the Cambridge Language Research Unit in the 1960's (e.g. Sparck-Jones & Jackson, [1967]). This work has tended to focus on specialised text-types referred to as *sublanguages* (see next section). The basic algorithm for such a process could be as follows:

(1) Obtain sample of sublanguage text;
(2) Identify co-occurrence behaviour of constituent words;

(3) Compute the matrix of similarity coefficients between words, based on co-occurrence behaviour;
(4) Perform cluster analysis to identify related clusters of words;
(5) Merge clusters into appropriate number of groups.

This algorithm is necessarily brief. There are many types of similarity coefficient that can be used, numerous cluster analysis algorithms, and many ways to merge clusters into meaningful semantic classes. The most appropriate algorithm for the present application is therefore an empirical issue, and as such is dependent on further investigation.

More recently, researchers have attempted to identify word-classes from raw text (as opposed to a sublanguage), e.g. Hughes & Atwell [1993], Finch & Chater [1991], Atwell & Drakos [1987]. The classes learnt from a corpus such as the LOB tend to reflect both syntactic and semantic constraints: small, closed function classes (e.g. prepositions) cluster on syntactic grounds, but nouns and verbs cluster according to semantic constraints. It is possible that information derived in this manner could contribute to text recognition, replacing the existing sources of syntactic and semantic information. Indeed, such an approach has the advantage of not needing a tagged corpus or MRD, and is applicable to languages other than English. Further investigation of this possibility is therefore highly desirable.

6.3.2.2 The Use of Semantic Classes

Semantic classes could be used by a text recognition system in a manner similar to that of syntactic classes. Just as a syntactic grammar can be used to describe the syntactic restrictions that constrain grammatical text, a semantic grammar can describe the restrictions that constrain meaningful text. The derivation of a semantic grammar is therefore no more of an abstraction than the derivation of a syntactic grammar. There is however, a slight problem with this technique. This problem is best described by first outlining the background linguistic perspective.

The semantic behaviour described above is a quantity that must be measured in order to identify meaningful groups. The methods of analysis by which such quantities are determined are essentially those of distributional linguistics, i.e. patterns of co-occurrence. So to determine the semantic classes within a particular domain, the collocational behaviour of all the constituent words is analysed, and then those words showing similar collocational patterns are grouped together. For

example, "*deposit*", "*savings*" and "*current*" may all show a strong collocation with "*account*", and have in common a number of weaker collocations with certain other words. Their collocational "profiles" are therefore similar, and this similarity forms the criterion by which they are grouped together.

However, previous research has shown that such distinct distributional behaviour is prevalent only within narrowly defined subject areas, in which the main purpose is to record factual information [Sager, 1981]. Beyond these narrow constraints, the distributional behaviour may be too diverse and amorphous, precluding the derivation of meaningful semantic classes. These narrowly defined subject areas are known as *sublanguages*. Hirschman, Grishman & Sager [1975] define sublanguage as "*the specialised use of English within a particular subfield, using a distinguished subset of the language*" (such as medical reports).

This constraint may appear at first sight to be prohibitive. It is true that much of the input to a text recognition system could be constrained to a specific domain, but to what extent the writers will restrict themselves to a "distinguished subset of the language" is unclear. It may transpire that the type of text found in business letters and documents is simply too general and imprecise to meet the requirements of a sublanguage, and any attempt to use this technique will produce poorly defined, inconclusive groupings. However, it may be possible to identify application areas and domains that are sufficiently specific to render the derivation and use of semantic classes a viable proposition. Although Sager's work suggests that the former conclusion may be the case, further investigation is highly desirable.

6.4 Summary

Domain codes are a further source of semantic information. The corpus method of domain code acquisition is preferable to other methods, for the following reasons:

- It is quantitative rather than qualitative;
- It is application-specific;
- The domains can be structured according to a particular hierarchy;
- It requires no manual intervention or subjective judgement.

The LELC codes made a correct domain identification on 12 out of 15 occasions. On the other three trials, the correct domain was identified as second

choice on two occasions and third choice on the remaining trial. This would suggest that there is some room for improvement. By adjusting the acquisition algorithm, an extended set of codes was created. However, the second investigation shows that extending the code lexicon in the described manner degrades their ability to identify the domain of a number of test documents.

The LDOCE codes made a correct domain identification for 8 trials from 15. On the other seven trials, the correct domain was identified as second choice on two occasions, third choice on two trials and fifth choice on one trial. In the remaining two trials the correct domain had no direct equivalent in the LDOCE coding system. The mapping issue considerably complicates the problem of the assessment of the performance of the LDOCE codes. It is essential that the coding system used by domain identification systems should reflect the classifications of the data to which it will eventually be exposed. If data exists for which there is no suitable category, then this amounts to an omission in its lexical database.

Domain code information can be used as an aid to recognition if given a limited influence within the semantic analyser. It chooses the correct candidate more than 50% of the time but less than 75%. There may be highly specialised situations where the domain is highly restricted, in which case domain codes may have a useful contribution to make. The example of 4.9% improvement using the LELC codes with the *Engineering* document is particularly encouraging. Apart from this, however, it is unlikely that an aggregate 1-2% improvement would render their derivation and use a universally worthwhile exercise.

There are a number of aspects of discourse processing that can be incorporated into the design of a text recognition system. Document structure can provide discourse-based cues (e.g. headings, initial mention, etc.) for the extraction of semantically significant information. A number of document structure standards exist that can be used to direct specific recognition applications. Semantic classes of words can be derived for sufficiently constrained sublanguages. These classes of words form a semantic grammar and provide a further recognition constraint. A potential algorithm for such a process has been outlined.

Chapter Seven

Discussion and Summary

7.1 Introduction

The main achievement of this research has been to demonstrate the contribution of a number of sources of semantic information toward the solution of the text recognition problem. These sources, although referred to as "semantic", represent many different kinds of information, such as syntagmatic, paradigmatic, encyclopaedic, or even discourse-based information. The use of such information contrasts with previous work on text recognition, which has tended to focus on the lower level processes (e.g. pattern recognition). Moreover, the limited research that has been done on the higher level processes has usually been concerned with language *understanding* rather than *recognition*.

Still very little is known about the actual processes that take place within human language processing. Indeed, it is arguably the case that the application of semantic knowledge is one of the least well-understood aspects of this process. Nevertheless, a number of attempts have been made to develop a theory of natural language semantics (see Chapter Two). Many of these theories have been shown to work within artificial domains that are both small and concrete. However, extending them for large, real world vocabularies has proved extremely difficult (if not impossible), for a number of reasons. Firstly, there is the problem of acquisition: the hand-crafting of semantic information for a large vocabulary is a highly complex and time-consuming job. Secondly, while some theories may work well with concrete subjects, they are not as effective with abstract concepts, such as "*justice*", "*insurance*" or "*business*". Thirdly, such theories can easily become unwieldy and inefficient when applied to larger domains [Bookman, 1987].

Lastly, and from a slightly different perspective, the standard semantic theories generally have no method of quantitative evaluation. This means that for a given

semantic theory, there is no way of measuring the "percentage correct" against a set of test sentences. Consequently, a further achievement of the present project has been to introduce quantitative evaluation metrics as an important component of each technique.

The techniques used by the current project bear little resemblance to the "established" semantic theories. The adopted techniques may originally have been *inspired* by psychological or linguistic concepts, but they have since been modified according to empirical studies and the requirements of a practical implementation. It would therefore be unwise to make any claims concerning their psychological plausibility or linguistic integrity. There is, however, some independent evidence supporting the use of the definitional overlap technique. The result obtained from the semantic priming investigation shows that the process will select semantically related word pairs in favour of unrelated word pairs. This sensitivity to the semantic relations between words is reflection of one of the characteristics of human word recognition, and provides a justification for the technique that is independent of any particular application.

The process of definitional overlap has been investigated in a variety of ways. Firstly, there has been the issue of the choice of dictionary. This has been investigated using text taken from a large number of domains (15). Since each text sample consisted of at least 500 words, the total number of word positions investigated was over 7,500. Although the results for each dictionary varied greatly within individual domains, the *average* performance was extremely close (OALD = 67.3% correct, CED = 68.3% correct, LDOCE = 69.6% correct). One possible explanation for the slight superiority of LDOCE involves its use of a *core vocabulary* of some 2,000 words. This constraint increases the likelihood of meaningful strong overlaps, since the probability of two semantically related words being defined using common terms is proportionately increased. Another factor in favour of LDOCE is its superiority within the domain of Commerce, which could become a prevalent application area. LDOCE consistently produces results in the 70-80% range for this domain, whilst the CED and OALD are consistently in the 60-70% range.

Attempts have been made to define ways by which the content of dictionary definitions could be refined; for example, by making them domain-specific. To this end, an algorithm for "filtering" dictionary definitions has been outlined, although it remains yet to be implemented on a large scale. The creation of semantic "networks"

from dictionary definitions has also been investigated, and the usefulness of information from a number of levels investigated. It was found that beyond the first level (i.e. the definition) information from MRDs rapidly descended into generality.

Investigations have also been made concerning the optimisation of the overlap algorithm. The performance of this algorithm is particularly important, since it affects the application of both definitions and collocations. A number of significant parameters have been identified, and where possible optimal values have been calculated (e.g. a window size of four words, the use of strong overlap only, the use of definition length compensation and the use of the simpler of the two algorithms). In addition, the presence of semantic relationships between words in ordinary sentences has been validated in an independent investigation.

Two types of collocation have been investigated: domain-specific and general. A General Collocation Dictionary of some 12,331 entries has been compiled from 5 million words of the Longman Corpus (500,000 words from each of the 10 superfields). Additionally, 10 domain-specific dictionaries have been compiled from each of the domain-specific corpora. These have been tested with the data used in the definitional overlap investigations described above. As before, the results for each dictionary varied greatly within individual domains, but the *average* performance was extremely similar (domain specific = 76.8% correct, general = 74.8% correct). Although the domain-specific is superior, there are a number of reasons why the general dictionary (GCD) is to be preferred. Firstly, domain-specific dictionaries are only effective if the domain of the data has already been identified. Secondly, their coverage is inferior to the GCD; and thirdly, their performance is more inconsistent across domains than the GCD.

A number of refinements to the collocation analysis technique have been suggested. These include: (a) the use of inflected rather than root forms; (b) the inclusion of function words; (c) analysis of linear precedence; and (d) the separation of distance-dependent (lexical) collocations from those that are distance-independent (conceptual). On a more general note, there are other aspects of the semantic analyser that could be refined - for example, no use is yet made of syntactic category (even though this has been shown to be related to the usefulness of dictionary definitions). Furthermore, no analysis of the information contained in thesauri has yet been undertaken. These issues are suggested as areas for further research.

An original method for the compilation of domain codes has been developed. This method is based on corpus analysis and has been used to create a basic lexicon of 10,623 coded lemmata, and an extended lexicon of 15,047 coded lemmata. In addition, a lexicon of 12,078 coded lemmata has been extracted from LDOCE. All of these lexicons have been investigated as sources of information for automatic topic identification. It was found that the basic lexicon produced by the corpus analysis method was superior, although its performance was less than perfect (12 correct identifications from 15 trials). A number of issues became apparent during this investigation. Firstly, there was a mapping problem between different sets of codes - the LDOCE codes had been assigned using a different system, and this affected both their performance and the suitability of the evaluation procedures. Secondly, it became apparent that since the extended corpus-based codes were ineffective for topic identification, they would also be ineffective as an aid to recognition.

The issue of how to assign documents to domains and how to select domains for code generation remains largely unresolved. To a certain extent, in using a structured corpus such as the LELC, these questions have already been answered by the corpus compilers. However, this does not imply that the contents could not be reorganised to suit the needs of the present project. For example, could some texts reasonably be said to belong to more than one domain? Is a two-level hierarchy necessarily the ideal structure? From what level(s) should domain codes be derived? The codes produced using the corpus method have thus far included only superfields (i.e. major domains) - it is possible that analysis of the subdomains (i.e. the second level) may provide more satisfactory results.

When the remaining two sets of codes were investigated as an aid to recognition, it was found that their performances were very similar: neither made a significant difference to the performance of the semantic analyser. The corpus based codes were slightly superior, increasing the average recognition rate by 1.6% (the LDOCE codes increased the rate by 0.8%). Possible reasons for this involve the origin of the LDOCE codes. Firstly, they have been designed for human rather than computational use; secondly, they represent a far greater number of domains, using a 2-level hierarchy; and thirdly, the codes are unevenly spread across those domains.

Document structure and semantic classes have also been suggested as sources of semantic information that could be exploited by text recognition systems. However, due to constraints on time and resources it has not been possible to investigate these

techniques to the same depth as dictionaries, collocations or domain codes. They are thus suggested as areas for further work.

7.2 Sources of Knowledge

Two major sources of semantic information have been identified: machine-readable dictionaries and text corpora. Aspects of the design of machine-readable dictionaries have been considered, and a modified set of design criteria has been proposed. For example, it may be the concentration of information that is important in a dictionary, rather than its overall size. Furthermore, the style of definition is important: simple, concrete definitions with numerous example sentences are of greater use than fragmented, abstract definitions with no examples. This is because they are more representative of the type of text to be recognised, and contain more relevant semantic information. The use of a core vocabulary is highly desirable, as this reduces the entropy within definitions (e.g. LDOCE).

The issue of indexing is also important. Investigations in Chapter Three show how the use of a larger wordlist in indexing the CED leads to a greater coverage of English and hence more accurate representations of the original definitions. This in turn produces more successful overlaps. The issue of lemmatisation, or "grain-size" remains unclear. It may transpire that a smaller number of lemmas are more likely to capture the semantic relationships found in some sentences. For example, using the CED, the words "payment" and "account" will only show a strong overlap if "payment" is assigned the same index as "pay".

The case for individual domain dictionaries also remains unclear. Early investigations indicated a great degree of variability in the performance of the definitional overlap technique, implying a need for some sort of domain dictionary in the more specialist or abstract domains. However, later investigations have tended to refute this; in particular the success of the General Collocation Dictionary would seem to suggest that a more general knowledge source is to be preferred. If domain-dictionaries did prove necessary, there would be a variety of ways in which the domains could be organised. A tree-like hierarchy may be appropriate, with specific domains such as "*banking*" near the bottom, and more general domains such as "*commerce*" nearer the top. However, the theoretical basis for constructing such a hierarchy with a depth greater than two levels is extremely weak (the fact that the

Longman Corpus only uses two levels would apparently endorse this standpoint). Furthermore, there is no accepted test for the existence of separate domains in the first place - at what point does "*business*" become "*commerce*", and "*commerce*" become "*trade*"?

Although such decisions may be made arbitrarily, it may be preferable to use statistical metrics. In so doing, related documents may be grouped together within a suitable domain, and similar domains grouped together within a superordinate domain, until the "root" of the hierarchy is reached. Given such a structure, the semantic analyser could then be designed to exploit the tightest possible domain-based constraints. This would involve a process of domain identification, through the interpretation of the document structure cues (e.g. headings, titles, opening sentences, etc.) or possibly user prompts.

There was considerable evidence to suggest that dictionaries contained further useful information beyond that which was immediately available at the first level (i.e. the definition) [Jensen & Binot, 1988]. Consequently, it was argued that the use of *expanded* definitions could provide further useful semantic information. However, this was shown not to be the case for the present project. Progressive expansion of definitions taken from the CED resulted in increasing generality and irrelevance to the context of the original word, thus reducing the ratio of useful information to redundant information. (N.B. - A useful test would be to confirm this hypothesis using definitions taken from the OALD or LDOCE.)

The relationship between syntactic category and overlap potential suggests a further possible refinement in the use of the technique. Applying the technique to every content word in the data regardless of its nature entails a certain amount of redundancy. It is possible that the overlap process would be more effective if applied selectively. Syntactic category may be a suitable parameter with which to direct the application of this process.

Dictionary definitions, as a lexical resource, have received considerable research attention in recent years [Alshawi, 1988]. Collocations have been less well exploited, possibly due to a lack of suitable corpora and techniques for their large-scale compilation. Indeed, for many years the idea of using probabilistic information within an NLP system was viewed with some disdain by the linguistic community. Text corpora were, it was felt, more a *description* of language rather than an *explanation*, and as such they could offer no insight to the "real" question of how

people process language. Stylistic analysis was one of the few tasks for which such statistical information was deemed appropriate [Ellegard, 1962].

However, in recent years there has been a significant revival of interest in corpus-based systems. This is for a number of reasons. Firstly, the lack of success achieved by rule-based systems has led to renewed efforts to find alternative techniques. Secondly, during this period both the processing power and storage capacity of computers have increased dramatically. Furthermore, a great many more textual resources are now available in electronic form. This background has provided researchers with both the incentive and the means to use corpus-based techniques. The trend is further reflected in the commercial sector: only one of the currently available MT (Machine Translation) systems uses rule-based techniques.

A number of corpora have been analysed during the present project, and a variety of collocation dictionaries thus created. The General Collocation Dictionary has been shown to be effective across a wide range of domains; its average number of correct identifications being only slightly inferior to that of the domain-specific collocation dictionaries. However, the performance of the domain-specific dictionaries was highly inconsistent, and for this reason the GCD is to be preferred. This contrasts with the studies using MRDs, in which it appeared that specialist domain-dictionaries may be required for certain situations. This result emphasises the regularity of language, inasmuch as the collocations compiled from a single large corpus can be found repeatedly in domains that superficially appear to use a restricted vocabulary and specialist language structures.

Algorithms for the acquisition and use of domain codes have been discussed. In particular, two types of domain code have been investigated: those extracted from an MRD (LDOCE) and those derived from text corpora. The corpus-based codes have been shown to be of greater value, although the improvement gained with such information is minimal. A number of more obscure sources of semantic information have also been identified. One source is concerned with aspects of discourse processing, and involves document structure information. Another involves the processing of semantic classes (analogous to the parsing of syntactic classes) and offers research potential in the longer term.

7.3 System Integration

The issue of autonomy versus interaction in human language processing is still highly contentious. On one side, there are those who suggest an unstructured, fully interactive model of comprehension (e.g. Marslen-Wilson, [1975]) and on the other those who advocate total autonomy of distinct syntactic and semantic components with no communication between them (e.g. Forster, [1979]). Additionally, there are those who adopt an intermediate viewpoint (e.g. Rayner et al, [1983]) who argue that syntactic choice is initially made independently, but semantic and pragmatic influences are used if the result remains ambiguous.

In an ideal situation, the knowledge sources within the current system would combine constructively, with each process "supporting" the scores given by the others. When one knowledge source "disagrees", the other knowledge sources should have identified the correct interpretation, so that their combined scores would outweigh the dissenting process. However, this ideal situation remains fictitious. It is often the case that the correct interpretation is identified by just one or even none of the analysers. In such an event, it is critical to know the performance characteristics of each analyser. Chapter Five describes some progress made towards this goal; concerning in this case the relation between syntax and semantics. The same comparisons need to be made among all aspects of the recognition process: pattern recognition, lexical, syntax and semantics.

In the current system, syntax and semantics operate as separate modules, with no communication between them. The output from each is merely a measure of the plausibility of each candidate within its immediate sentential context, judged according to syntactic or semantic considerations respectively. Given such an arrangement, it is evident that the output from one processor is of little use to the other. In fact, the level of detail available from such systems is very limited. It is possible, however, to conceive of systems in which data is passed between the analysers to their mutual benefit. For example, when a word like "*give*" is encountered, syntax could inform semantics that this is a three-place predicate, so that semantics may then build a representation of who gave what and to whom. If the case slots cannot be filled adequately [Fillmore, 1968] then an alternative syntactic interpretation may be sought.

Similar investigations have been made concerning the interaction between the syntax analyser and lexical analyser [Wells, 1991]. Consider the situation in which the lexical analyser can find no word that exactly fits the information from the pattern recogniser. In such a case it is desirable to attempt a less precise approach, searching the lexicon on partial information such as initial letter, word shape, word length or syntactic category. The first three of these sources of information (i.e. word shape, word length and initial letter) relate to the *orthography* of the missing word, and may thus be obtained from the pattern recogniser. Syntactic category, however, is not related to orthography. To obtain such information, the syntax analyser must be used in a *predictive* capacity, in which the transition matrices are used to "suggest" the most likely syntactic category of a missing word, given the categories of the preceding words. This category may be added to the list of constraints upon which the lexicon is searched, reducing both the search time and the length of the resultant candidate list. To illustrate, Wells [1991] describes sample distributions of words from a lexicon of some 60,000 items, according to length, first letter and grammatical category. Investigations have shown some combinations of such partial information to be extremely effective - for example, a short adjective beginning with z could be almost uniquely identified.

Errors involving the pattern recogniser can take many forms, and there are two types that are particularly relevant to the issue of interaction. These are: (a) the case where no allowable strings have been found by the lexical analyser, and (b) the case where the correct word is absent from the candidate list. Errors of type (a) are simple inasmuch as the candidate list is empty, therefore they can be located precisely within the data and their presence can be signalled to error-handling routines as necessary. Errors of type (b) are unfortunately much more difficult to handle, since there is at present no reliable algorithm by which their presence can be detected. Whereas human readers may detect a misread word by its syntactic or semantic incompatibility, the current syntactic and semantic analysers simply process the candidate lists they are given, and have no capacity for detecting the presence or absence of the correct word. In this respect, there seems little point in developing sophisticated error-handling routines if the majority of errors cannot be detected in the first place. Evidently, an error cannot be *corrected* unless it has first been *detected*. Consequently, it may be useful to investigate the means by which commercial grammar/style checkers are able to detect and suggest corrections for errors in word-processed documents. (By analogy, Atwell [1987] suggests the use of word-processor error detection techniques in speech recognition.)

Given that errors of type (b) cannot reliably be detected, let us turn our attention again to errors of type (a). Since these errors can easily be located, it would seem that the error-handling routines described above (including predictions based on syntactic category) could be directly applicable. However, there is a problem. Since errors of type (b) are virtually undetectable, there is no logic to dictate that the syntactic categories upon which predictions are based will be accurate. The candidates preceding an error of type (a) could easily contain errors of type (b), and any predictions thus created would be similarly erroneous. Furthermore, if predictions are based upon *all* candidates in the preceding positions, this will inevitably include *incorrect* candidates, hence a number of spurious predictions will be generated. It is of course possible to generate predictions from only the top candidate in preceding positions, but there is no guarantee that the top candidate will be the correct word. Further study is required to measure the incidence of type (a) and type (b) errors, and the way in which these metrics vary for different recognisers and writers.

Like syntax, the semantic analyser can be used in a predictive capacity. Also like syntax, its use is constrained by the incidence of type (b) errors and the generation of predictions from incorrect candidates. However, there is an important difference between the syntactic and semantic analysers concerning the *form* of their predictions. The syntax analyser uses a coding system of some 109 grammatical categories, and any predictions that are made must take the form of one (or more) of these categories. The semantic analyser, by contrast, uses whole words as its basic unit of representation, with no intermediate form. So when used in a predictive capacity, the semantic analyser would suggest *actual words* rather than some intermediate representation. These words could comprise a list of those collocations that are most strongly associated with the preceding four words (it is unlikely that dictionary definitions could provide effective predictions). If a collocate occurs more than once in this list it could be highlighted as a particularly likely candidate. In the case of off-line recognition (e.g. OCR), both preceding and subsequent words could contribute to the suggestion of candidates. Empirical studies need to be made regarding the extent to which predictions passed from semantics are effective, and this is suggested as an area for further work.

All the above techniques make assumptions concerning the nature of the data. Moreover, when a practical implementation is discussed, further assumptions must be made concerning the algorithms and data structures. For example, it must be assumed that the necessary feedback loops can actually be implemented, and the generation of

predictions can be efficiently managed. This is a considerable task in itself, and as such remains outside the scope of this thesis. However, directions for further work can be indicated. A useful study would be to investigate the effectiveness of such error-handling routines using a variety of data qualities, ranging from the simple to the complex. The former could consist of data in which errors of type (a) have been located, and the words preceding them having been recognised accurately and uniquely. The latter would be more representative of genuine data, in which errors of both types are present, but their locations are unspecified. It may transpire that since semantic predictions take the form of actual words, further searching of the lexicon may be unnecessary. However, this would seem somewhat optimistic, and in any case it still leaves the hardest problem (i.e. error detection) unsolved.

There is another situation in which the semantic analyser could contribute valuable information to another analyser. Consider the process of definitional overlap. This was originally used for the task of automatic sense disambiguation [Lesk, 1987]. If the semantic analyser could use this process to determine the sense of word candidates, then this information could then be passed to the syntax analyser and used to restrict the number of syntactic categories that each candidate could possess. This would reduce the number of spurious transitions that the syntax analyser had to process, and thus improve its ability both to recognise the correct candidate and to generate plausible expectations for type (a) errors.

The issue of information passing raises another important question: the direction of data flow. At present, the system operates solely in a bottom-up fashion: the pattern recogniser suggests possible characters, from which the lexical analyser eliminates certain combinations. The syntax and semantic analysers then rank those combinations according to their sentential context. So the character recogniser is the only module that actually suggests *independent* interpretations of the input - the other modules merely work on these suggested interpretations, eliminating various possibilities or modifying their plausibility each time. In effect, the lexical, syntax and semantic analysers are acting just as passive "filters" in this context, as they do not contribute *further* interpretations of the data. In a genuine top-down system, hypotheses would be made concerning the expected input, and part of this process would be the contribution of interpretations that were independent of those produced by the pattern recogniser.

The extent to which top-down processing facilitates human language processing is well documented [Carpenter & Just, 1987]. Furthermore, there are many researchers who believe that the conspicuous gap between the reading performance of people and that of machines is a reflection of the fact that few text recognition systems utilise the many knowledge sources or recognition strategy of the human reader [Hull, 1987]. Human readers are able to use common-sense knowledge of the world in a seemingly effortless way, and in so doing restrict the activation of expectations to those that are most relevant to the immediate context. Unless this knowledge can be acquired and expressed in a computational manner that is both rigorous and efficient, the creation of expectations is in danger of disintegrating into a combinatorial explosion of irrelevant possibilities. Repeated attempts to solve this problem in the 1970's (e.g. Minsky [1975], Schank [1977] and Charniak [1977]) met with little success, and in recent years there have been few researchers prepared to take on this apparently intractable problem.

At present, the higher-level knowledge sources of lexical, syntax and semantics are combined using an arbitrary weighting system, in which semantic information has 50% of the weight of syntax information, and syntax has 50% of lexical. These weightings still need to be justified by empirical data. Moreover, it is likely that these weightings will need to be context-sensitive. For example, if the system is required to perform in a note-taking situation, the syntactic constraints may be weaker. In such an instance, it may be desirable to either reduce the weighting given to syntactic information or to activate alternative syntactic processes. Similarly, the importance of semantic information will vary according to domain and application. In such circumstances, the weight of the semantic score may be proportionately altered.

It may prove desirable to modify the weightings further, in an interactive fashion. Given feedback from the user (on performance, etc.), the weights could be "tuned" until they best suit the operating conditions. As discussed below (see Lexical Acquisition), the system would need to be interactive to allow "tailoring" of the lexicon to an individual user, and this could be extended to include tailoring of the weightings.

Regardless of the weightings, there will inevitably be situations in which the system makes errors. Unfortunately, at present, there is no reliable method for error detection without resorting to human intervention. It is desirable, therefore, to identify characteristics that can indicate the *likelihood* of an error having occurred.

One such characteristic is the degree of conflict between the various analysers. When lexical, syntax and semantics agree, this may imply that the correct word is present in the data (and is the subject of that consensus). When they disagree, it is possibly because the word that would "unite" them is absent from the data. In such cases, it may be desirable to signal this fact to the pattern recogniser and instigate further pattern recognition, from which the correct interpretation will hopefully be identified. Whether this technique is practicable or not depends to a large extent on the collective reliability of the higher-level processes. If disagreement between these processes occurs often, and not just when the correct word is absent from the data, such feedback loops may prove counter-productive. Further study of this area is required.

The same analogy can be extended to error detection within the semantic processor itself. It may be possible to measure the likelihood of an erroneous choice by applying more than one source of semantic information. For example, if definitional overlap favours one candidate, collocations favour a second, and domain codes a third then it is possible that the correct word is absent from the candidates. The lack of consensus may indicate a less reliable choice. In view of this, it may be useful to attach a "credibility rating" to the semantic score, based on the degree of consensus. However, to a certain extent, the case for reliability or credibility ratings is a circular argument, since the information that is used to measure the credibility of a score is the same information that is used to calculate that score in the first place! Evidently, such "credibility" information is no more than that which is implicit in the raw scores for each sentence position.

7.4 Lexical Acquisition

For many years the process of lexical acquisition was seen as a rather mundane task that could take place once the more "interesting" aspects of NLP system development were complete. Consequently, many of the NLP systems that were developed possessed only "sample" lexicons of perhaps a few hundred words [Whitelock, 1987]. Not surprisingly, attempts to expand these lexicons through extensive hand-crafting usually proved futile. Furthermore, the lack of an agreed standard for lexical representation led to the development of highly disparate systems; even for those sharing a common theoretical foundation. Attempts to create larger, more comprehensive systems through the combination of smaller systems were therefore often precluded by incompatible representational formats.

By contrast, contemporary linguistic theory places a great importance on the development of the lexicon, and several current research projects reflect this change of attitude (e.g. GENELEX [Normier & Nossin, 1990] and MULTILEX [McNaught, 1990]). Indeed, the present project sees the issue of lexical acquisition as fundamental to the design of robust NLP systems. Sources of semantic information have been selected and investigated specifically according to their availability. However, the acquisition of collocational information is somewhat problematic. Definitional information can be obtained from the LDOCE for some 55,000 headwords, but the acquisition of a similar number of collocational entries would require the processing of an immense corpus. This may be illustrated by the example of the GCD. To create this, a 5 million-word subset was extracted from the Longman Corpus and subjected to the analysis described in Chapter Four. Due to the statistical instability of low frequency types, some threshold has to be identified below which types are too infrequent for collocational information to be compiled [Beale, 1987]. This threshold was set at 3 occurrences, i.e. only types occurring 3 times or more in the corpus would be analysed. After the deletion of empty entries and repetitions, a collocation dictionary of some 12,475 entries was produced, which is considerably smaller than LDOCE with its 55,000 definitions. If the threshold was reduced to 1 (to include all types) this would still only provide a collocational dictionary of some 15,000 entries. To provide anything like the coverage given by LDOCE, a much larger corpus is necessary.

Other researchers have faced the same lexical acquisition problem. Jelinek [1985] demonstrated the inadequacy of small lexicons in speech recognition systems. He showed that a 5000-word lexicon based on high frequency types covered only 92.5% of the words used in a corpus of business and technical correspondence. This means that the error rate of the recogniser will exceed 7% regardless of how good it is otherwise. He also calculated the coverage given by larger lexicons:

Vocabulary Size	Text Coverage(%)
5000	92.5
10,000	95.9
15,000	97.0
20,000	97.6

Table 7.1: Text Coverage and Vocabulary Size (after Jelinek)

It can be seen that this asymptotic curve will give diminishing returns for each increase in vocabulary size. Furthermore, the construction of a massive fixed vocabulary involving technical terms may be possible only if such technical corpora are already available. Jelinek concludes that *personalisation* of the vocabulary is the only alternative. He suggests that this may be achieved by dynamically varying the vocabulary to consist at any moment of the last N different words used. The coverage so produced is substantially higher (a vocabulary of 5000 words now gives 95.5% coverage) but it requires the processing of the last 56,000 words to assemble this vocabulary. Indeed, contemporary system developers have not been slow to exploit the idea of a dynamically updated lexicon. For example, the Dragon Dictate system uses a standard initial lexicon that is gradually personalised, so improvements in accuracy gradually accumulate and are noticeable immediately. Furthermore, there are others who share the view that lexical acquisition must be an ongoing process (rather than something that can be set out in precise detail in advance). For example, Levinson and Liberman [1981] argue that the best design strategy for NLP systems is to give them the "*basic set of expectations and abilities needed to learn a language*" rather than "*a wealth of descriptive detail*".

There are other problems of lexical acquisition that are of a more hidden nature. Firstly, words and units of semantic information do not always have a one-to-one correspondence. The creation of a semantic lexicon cannot proceed on a simple word-by-word basis. Compound words and idioms often have a semantic content that is unrelated to their constituent parts. For example, when people use the phrase "*red herring*", it usually has little to do with either colours or fish. However, it undoubtedly has a recognised semantic value of its own, and must therefore be included within the lexicon. The issue of where such units of semantic information should be stored (as part of "*red*", part of "*herring*", or as an entry on its own?) remains unresolved.

Another problem is that some concepts may be represented by a number of different word forms. For example, "*slaughterhouse*" and "*abattoir*" are two words that refer to the same concept. Similarly, there can be cases where one word is a more specific instance of another, e.g. "*cat*" and "*mammal*", etc. A problem may arise if an NLP system fails to recognise that the semantic relations within a given sentence may be preserved even when particular words have been substituted for their synonyms or hyponyms. A lexicon that gives no consideration to such relations would be unable to deal adequately with such transformations. As Amsler [1989] puts it, "*The systems*

attempting to process language material lacked a complete lexicon of the language they were attempting to manipulate intelligently, and had no rules for understanding how to recognise these lexical concepts when they appeared in text".

Some linguists may argue that a complete lexicon requires a complete set of semantic entries, therefore any machine-readable dictionary being used as the source of semantic knowledge should accommodate this requirement. However, since the goal of the present project is recognition and not understanding, the need for a unique semantic representation for every input is less significant. To some extent, the problem described above is transcended by using collocations rather than dictionary definitions. "*Red herring*" is itself a collocation, and a very strong one at that, so it is unlikely that alternative candidates in these two word positions would be favoured by the semantic analyser. In such a case, the absence of a unique semantic representation for this compound is not necessarily a problem for the current system. Indeed, Sharman [1989] claims that the use of such probabilistic word-word associations constitutes the implicit application of syntactic, semantic and pragmatic knowledge.

The issue of lexical acquisition remains a serious issue for the system as whole (even if less so for the semantic analyser). Several studies have illustrated the inadequacy of machine-readable dictionaries, notably that of Walker & Amsler [1986] which showed that of 119,630 word forms represented in a corpus of New York Times Newswire text almost two-thirds (64%) were missing from the Merriam-Webster Seventh Collegiate Dictionary (see Chapter Two). It transpires that a quarter of these were proper nouns, many of which appeared in a different form in the text to that of the dictionary (e.g. hyphenated, abbreviated, etc.). Amsler argues that the creation of a comprehensive lexicon of proper nouns may be impractical if not impossible, and that such lexical items need to be effectively *parsed* (at run-time) to identify the semantic concept they represent.

7.5 Further Work

The efficient acquisition of lexical knowledge is a problem common to many NLP systems, and as such represents a continuous theme running throughout the present project. Fortunately, a number of new, larger corpora will soon be available, and their use should go some way towards alleviating this problem (e.g. the British National Corpus, which will eventually consist of 100 million words). However, as

Jelinek [1985] points out, the creation of a comprehensive lexicon is an elusive goal, and steps must be taken towards permitting the interactive extension of the lexicon according to specific user needs. Procedures and techniques to allow such individual "tailoring" need to be developed and evaluated.

Evidently, the domain of a document can be determined from its contents, as described in Chapter Six. However, there are explicit lexical cues within documents that may provide this information in a more efficient manner. To this end, a systematic examination of document structure is suggested, giving particular attention to the areas of initial mention, headings and thematic titles. An empirical study is required, inasmuch as it is necessary to determine the extent to which such lexical cues can be relied upon to provide accurate domain information. It is also desirable to determine the extent to which specific document types exhibit sublanguage characteristics, since this constitutes a prerequisite for the derivation of semantic classes. Although they have received only brief mention in this thesis, techniques involving the use of semantic classes have shown efficacy for other NLP applications (e.g. Sager [1981]) and as such may offer potential in the longer term.

The integration of the semantic analyser into the complete text recognition system remains a highly problematic issue. It is extremely difficult to devise a coherent architecture whilst the input and output characteristics of each module remain indeterminate. Much more needs to be known about the form and content of the information each module needs and the predictions it can create. Only when such quantities have been specified can models of system interaction be empirically investigated. It may transpire that a variety of architectures needs to be considered, possibly involving the use of feedback loops, blackboard approaches or constraint satisfaction procedures.

The semantic analyser has been used as a "filter" system throughout the present project. However, ideas concerning its use as a predictive system have been outlined. Further work is needed to develop these ideas and produce a robust implementation, such that an empirical study can be carried out to determine which of these two approaches is the most suitable.

The treatment of errors also remains highly problematic. The process of error correction cannot proceed until error-detection has taken place. The development of efficient error-detection routines is therefore of prime importance. In particular, an empirical study is required to measure the incidence of type (a) and type (b) errors,

and to determine the way in which these quantities vary for different recognisers and writers. It has been suggested that a lack of consensus between the various analysers could be used to signal the presence of possible errors. Further study of this and the possibility of other error-dependent phenomena is required.

As a suggestion, it may be desirable to investigate the effectiveness of error-handling routines across a variety of data types, ranging from the simple to the complex. The former could consist of data in which errors of type (a) have been located, and the words preceding them having been recognised accurately and uniquely. The latter would be more representative of genuine data, in which errors of both types are present, but their locations are unspecified.

7.6 Conclusions

Language, in all its forms, is inherently ambiguous. Moreover, the linguistic rules that may be used to constrain that ambiguity become weaker as the knowledge level becomes higher. In effect, the application of those rules goes from the deterministic to the probabilistic. Consider the case of the lower levels, i.e. those of the pattern recogniser and the lexical analyser. The pattern recogniser will always try to form alphanumerics out of the "squiggles" it perceives, and the lexical analyser will always try to find genuine English words. So if a user wrote the letter string "goocl", the system would assume the user had not intended to write a non-word and therefore probably suggest the word "good" as the most likely candidate. Evidently, the possibility that the user actually wrote a non-word is discounted, i.e. its probability is deemed to be always zero. This constitutes a deterministic choice, as non-words are always eliminated from consideration.

Higher up the knowledge levels, however, the situation changes. No longer can deterministic choices be made - all that can be said is that one interpretation is more *likely* than another. Nothing can be ruled out categorically. The rules that constrain language become weaker, and now act more as guidelines or heuristics. Some generative linguists may argue that there is a universal grammar that underpins human linguistic competence (if not performance). Conversely, there are those who argue that the distinction between grammatical and ungrammatical is more of a gradient than a black-and-white dichotomy [Sampson, 1987] and the fact that human readers are still capable of understanding (and hence recognising) syntactically ill-

formed sentences would seem to support this. Besides, there are many applications in which linguistic output would be *intentionally* ungrammatical (e.g. note-taking).

Moving upwards from syntax the linguistic constraints become even weaker. Semantics, pragmatics, discourse rules, etc. all provide restrictions on what is acceptable, but the rules are so weak that for each one it is easy to conceive of examples in which they cease to apply, e.g.:

Domain Information: This refers to the use of specialist subject area information. For example, supposing the user wrote "*I took my clogs to Cruft's*". Domain information would suggest that the fourth word in this sentence should probably be "*dogs*". However, if the user was Dutch, the first interpretation may have been correct.

World Knowledge: This refers to the use of knowledge of the real world. For example, supposing the user wrote "*The cod sat on the mat*". Knowledge of the practicalities of the real world tells us that the second word could not really be "*cod*", but it could possibly be "*cat*". However, if the writer was Lewis Carroll, and the text was "*Alice in Wonderland Part Two*", the original interpretation may have been correct.

Semantics: This refers to the identities of words and the way they relate to each other. Supposing the user wrote "*colourless green ideas sleep furiously*". Semantic information would suggest that this combination of words is invalid. However, if the writer had been Noam Chomsky, a semantically anomalous sentence may have been the intended result.

Discourse Information: This refers to the order and manner in which utterances can be made, and may be demonstrated by a simple dialogue: (Person A): "*Do you have the right time?*" (Person B): "*Yes thanks*". This exchange appears to break the rules of discourse, as Person B appears to have ignored Person A's request for information. However, the Person B may have intended to express sarcasm or flippancy, in which case the exchange becomes more plausible. This example also illustrates the use of Pragmatics, which refers to the use of language to achieve practical ends. In this case Person B may desire to irritate Person A.

In addition to the theoretical considerations above, it is necessary to address some of the practicalities of text recognition systems. The recent development of

notepad-style computers has extended the availability of computing resources to those whose needs had been hitherto constrained by a requirement for mobility. Laptop computers may have been sold as portable systems, but a major limitation is that they cannot be used from a standing position. By contrast, typical notepad computers are lightweight, A4-size, can be carried in one hand and operated with the other. Consequently, there are further groups of people for whom such technology becomes convenient (e.g. doctors, salespeople, field service personnel), and the applications to which such machines can be put are growing (e.g. form-filling, order taking, stock control, etc.). In each case, the recognition of handwriting and text constitutes a highly desirable aspect of their functionality. Educational applications represent a further possibility: there is a valuable role for such machines to play in teaching children how to form characters properly, how to write legibly, and how to spell. Developments in the field of interactive gesture-based "front-ends" constitute a further enhancement to the usual range of input, allowing features such as pen-driven word processing, the integration of diagram editing with text, and the inclusion of sketches and hand-drawn figures.

A consideration of the needs of a complete system or "commercial product" often reveals more questions than answers. Many of the issues are strategic in nature, for example: How domain-specific should the system be? How user-specific should it be? What facility should there be for training? Regarding the latter issue, efforts must be made to ensure that this process as user-friendly as possible. The system that expects a naive user to get involved in the minutae of segmentation algorithms will not be received favourably by its potential buyers.

A large part of the development effort must address the issue of interface design. For example, how should the presence of alternative candidates be signalled to the user? With dynamic handwriting recognition systems, when should the script disappear and the ASCII text appear? To change script into text immediately after each word may prove disconcerting to the user - it may be preferable to wait till the end of a sentence, or wait for a specific cue from the user. Moreover, the semantic analyser works in a window of four words behind the text being written, so its influence will always lag sometime behind the current input.

To conclude, a basic text recognition system has been demonstrated. The semantic constraints inherent in natural language have been investigated, and sources of semantic knowledge have been identified and assessed. Attempts have been made

to develop semantic analysis techniques based on these sources of knowledge, and where appropriate these techniques have been investigated experimentally and subsequently incorporated into a "semantic analyser". This module has been integrated with the other elements of a complete text recognition system, and its effects on overall system performance have been investigated and quantified.

Bibliography

T. Ahlswede & M. Evens (1988) 'Generating a relational lexicon from a machine-readable dictionary', *Int. Jnl. of Lexicography,* **1**, OUP

H. Alshawi (1988) 'Analysing the dictionary definitions', in B. Boguraev & E. Briscoe (Eds.) '*Computational Lexicography for Natural Language Processing*', Longman, London, pp.153-169

R.A. Amsler (1989) 'Research towards the development of a lexical knowledge base for natural language processing', *Proc. 1989 SIGIR Conf. Assoc. for Computing Machinery*

J. R. Anderson & G. H. Bower (1973) '*Human Associative Memory*', Winston, Washington DC

E.S. Atwell (1987) 'How to detect grammatical errors in a text without parsing it', in B. Maegaard (Ed.), *Proc. of Third Conf. of the European Chapter of the ACL*, New Jersey, pp.38-45

E.S. Atwell & N. Drakos (1987) 'Pattern recognition applied to the acquisition of a grammatical classification system from unrestricted English text', in B. Maegaard (Ed.), *Proc. of Third Conf. of the European Chapter of the ACL*, New Jersey, pp.56-63

E.S. Atwell, S. Arnfield, G. Demetriou, S. Hanlon, J. Hughes, U. Jost, R. Pocock, C. Souter & J. Ueberla (1993) 'Multi-level disambiguation grammar inferred from English corpus, treebank and dictionary', *Proc. of IEE Colloquium on Grammatical Inference*, Essex University, Colchester

A.D. Baddeley & G. Hitch (1974) 'Working memory', *8*, in G.H. Bower (Ed.) '*The Psychology of Learning and Motivation*', Academic Press, New York

Y. Bar-Hillel (1967) 'Dictionaries and meaning rules', *Foundations of Language,* **3**, pp.409-414

F.C. Bartlett (1932) '*Remembering*', Cambridge Univ. Press, Cambridge, Mass.

A. Beale (1987) 'Towards a distributional lexicon', in R. Garside, G. Leech & G. Sampson (Eds.) '*The Computational Analysis of English*', Longman, pp.149-162

G. Berry-Rogghe (1970) 'Collocations - Their Computation and Semantic Significance', *Unpublished Ph.D. Thesis,* Univ. of Manchester

E. Black (1988) 'An experiment in computational discrimination of English word senses', *IBM Journal of Research and Development,* **32**

B. Boguraev & E. Briscoe (1989) (Eds.) '*Computational Lexicography for Natural Language Processing*', Longman, London

D. Bolinger & D.A. Sears (1981) '*Aspects of Language*', Harcourt Brace Jovanovich, Inc., New York

L.A. Bookman (1987) 'A Microfeature-based scheme for modelling semantics', *Proc. 10th Int. Joint Conf. on AI,* Milan

L. Braden-Harder & W. Zadrozny (1989) 'Lexicons for broad coverage semantics', *Proc. 1st International Lexical Acquisition Workshop,* Detroit, Michigan, pp.85-92

W.F. Brewer (1980) 'Literary theory, rhetoric and stylistics: Implications for psychology', in R.J. Spiro, B.C. Bruce & W.F. Brewer (Eds.) '*Theoretical Issues in Reading Comprehension*', Erlbaum, Hillsdale, NJ

C. Brooks & R.P. Warren (1970) '*Modern Rhetoric*', Harcourt, Brace & World, New York

P. Brown, J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, J. Lafferty R. Mercer & P. Roosin (1989) 'A Statistical Approach to Machine Translation', *IBM Research Report #14773,* Yorktown Heights, NY

R.J. Byrd, N. Calzolari, M.S. Chodorow, J.L. Klavans, M.S. Neff & O.A. Rizk (1987) 'Tools and methods for computational lexicology', *Computational Linguistics,* **13**

R.J. Byrd (1989) 'Discovering relationships among word senses', *Proc. 5th Annual Conference of the UWC for the New Oxford English Dictionary,* Oxford, pp.67-79

H.S. Cairns (1984) 'Current issues in research in language comprehension', in R. Naremore (Ed.) '*Recent Advances in Language Sciences*', College Hill Press, San Diego

N. Calzolari (1984) 'Detecting patterns in a lexical database', *Proc of the 10th Int. Conf. on Computational Linguistics*, Stanford, pp.170-173

P.A. Carpenter & M.A. Just (1977) 'Integrative processes in comprehension', In D. LaBerge & S.J. Samuels (Eds.) '*Basic Processes in Reading: Perception and Comprehension*', Erlbaum, Hillsdale, NJ

P.A. Carpenter & M.A. Just (1983) 'What your eyes do while your mind is reading', in K. Rayner (Ed.) '*Eye Movements in Reading: Perceptual and Language Processes*', Academic Press, New York

J.M. Cattell (1885) 'The inertia of eye and brain', *Brain*, **8**

W.L. Chafe (1972) 'Discourse structure and human knowledge', in R.O. Freedle & J.B. Carroll (Eds.) '*Language Comprehension and the Acquisition of Knowledge*', Winston & Sons, Washington, DC

E. Charniak (1987) 'Connectionism and explanation', *Proceedings of the 3rd Workshop on Theoretical Issues in Natural Language Processing*, Las Cruces, New Mexico, pp.71-74

M. S. Chodorow, R. J. Byrd & G. E. Heidorn (1985) 'Extracting semantic hierarchies from a large on-line dictionary', *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, Chicago, Illinois, pp.299-304

N. Chomsky (1957) '*Syntactic Structures*', The Hague: Mouton

N. Chomsky (1965) '*Aspects of the Theory of Syntax*', MIT Press, Cambridge, MA

Y. Choueka (1988) 'Looking for needles in a haystack', *Proc. of RIAO 88*, **14**

K. Church & P. Hanks (1989) "*Word association norms, mutual information and lexicography*", Proc. 27th Meeting of the ACL, pp.76-83.

H.H. Clark (1975) 'Bridging', in B. Nash-Webber (Ed.) '*Theoretical Issues in Natural Language Processing*', Cambridge, MA

P. Clements (1979) 'The effects of staging on recall from prose', in R.O. Freedle (Ed.) '*New Directions in Discourse Processing*', Ablex, Norwood, NJ

A. M. Collins & M. R. Quillian (1969) 'Retrieval time from semantic memory', *Journal of Verbal Learning and Verbal Behaviour*, **8**, pp.240-247

J.T. Critz (1982) 'Frame based recognition of theme continuity', *Proceedings of the 5th International Conference on Computational Linguistics*, North-Holland Publishing Company, pp.71-75

S. Crowdy (1992) Personal communication

K. Dahlgren & J. McDowell (1986) 'Using commonsense knowledge to disambiguate prepositional phrase modifiers', *Proceedings of the 1986 Conference of the AAAI*, pp.589-593

F. Debili (1982) 'Analyse syntaxico-semantique fondee sur une acquisition automatique de relations lexicales-semantiques', *Unpublished Ph.D. Thesis*, University of Paris

G. Demetriou (1993) 'Lexical disambiguation using CHIP (Constraint Handling In Prolog)', *Proceedings of 1993 European Conference of the ACL*, Utrecht

T. A. van Dijk & W. Kintsch (1983) '*Strategies of Discourse Comprehension*', Academic Press, Orlando

S.C. Dik (1987) '*Stepwise Lexical Decomposition*', Peter de Ridder Press, Lisse

M. Dyer (1989) 'Lexical acquisition through symbol recirculation in distributed connectionist networks', *Proc. 1st International Lexical Acquisition Workshop*, Detroit, Michigan pp.205-214

L.D. Earnest (1962) 'Machine recognition of cursive writing', *Information Processing*, Butterworths, London pp.462-466

M. Eden (1964) 'Handwriting and pattern recognition', *IRE transactions on Information Theory*, pp.160-166

A. Ellegard (1962) 'A statistical method for determining authorship: The Junius letters 1769-1772', *Gothenburg Studies in English No. 13*, University of Gothenburg

A. Ellegard (1978) 'The syntactic structure of English texts: a computer-based study of four kinds of text in the Brown University Corpus', *Gothenburg Studies in English,* **43**

L.D. Erman & V.R. Lesser (1975) 'A multi-level organisation for problem solving using many diverse cooperating sources of knowledge', *Proc. 4th Int. Jnt. Conf. on Artificial Intelligence,* Tbilisi, pp.483-490

L.D. Erman, F. Hayes-Roth, V.R. Lesser & D.R. Reddy (1980) 'The Hearsay-II speech understanding system: integrating knowledge to resolve uncertainty', *Computing Surveys,* **12,** pp.213-253

L.J. Evett & G.W. Humphreys (1981) 'The use of abstract graphemic information in lexical access', *Quarterly Journal of Experimental Psychology,* **33A,** pp.325-350

C.J. Fillmore (1968) 'The case for case', in E. Bach & R.T. Harms (Eds.) '*Universals in Linguistic Theory*', Holt, Rinehart & Winston, New York

S. Finch & N. Chater (1991) 'A hybrid approach to the automatic learning of linguistic categories', *AISB Quarterly,* **78,** pp.16-24

K.I. Forster (1979) 'Levels of processing and the structure of the language processing', in W. Cooper & E. Walker (Eds.) '*Sentence Processing: Psycholinguistic Studies*', Erlbaum, Hillsdale, NJ

A. S. Fraenkel, D. Raab & E. Spitz (1980) 'Semi-automatic construction of semantic concordances', *Data Bases in the Humanities and the Social Sciences,* North-Holland Publishing Company, pp.101-103

J. Greene (1986) '*Language Understanding: A Cognitive Approach*', OUP

R. Grishman (1986) '*Computational linguistics: an introduction*', Cambridge University Press, Cambridge

M. Gross (1985) 'Projecting the lexicon-grammar on texts', *Proc. 1st International Roman Jakobsen Conference,* New York University

B.J. Grosz (1986) 'The representation and use of focus in a system for understanding dialogues', in B.J. Grosz, K. Sparck Jones & B.L. Webber (Eds.) '*Readings in Natural Language Processing*', Morgan Kaufman

C. Guo (1989) 'Building a machine-tractable dictionary from LDOCE', in B. Boguraev & E. Briscoe (Eds.) '*Computational Lexicography for Natural Language Processing*', Longman, London, pp.211-217

J. Guthrie, L. Guthrie, Y. Wilks & H. Aidinejad (1991) 'Subject-dependent co-occurrence and word sense disambiguation', *Proceedings of 29th Annual Meeting of the ACL*, Berkeley, California, USA, pp.146-152

J. Guthrie, (1993) 'A note on lexical disambiguation', in C. Souter & E. Atwell (Eds.) *Corpus-Based Computational Linguistics*, Rodopi Press, Amsterdam, pp.217-238

P. Hanks (1986) (Ed.) '*The Collins English Dictionary*', Second Edition, William Collins Sons & Co., Glasgow

G. G. Hendrix (1979) 'Encoding knowledge in partitioned networks', in N. V. Findler (Ed.) '*Associative Networks: Representation and Use of Knowledge by Computers*', Academic Press, New York

L. Hirschman, R. Grishman & N. Sager (1975) 'Grammatically-based automatic word class formation', *Information Processing and Management,* **11**, pp.39-57

A.S. Hornby (1974) '*Oxford Advanced Learner's Dictionary of Current English*', Third Edition, Oxford University Press

J. Hughes & E.S. Atwell (1993) 'Automatically acquiring and evaluating a classification of words', *Proc. of IEE Colloquium on Grammatical Inference*, Essex University, Colchester

Y. Huizhong (1986) 'A new technique for identifying scientific/technical terms and describing science texts', *Literary and Linguistic Computing,* **1**, Oxford University Press, pp.93-103

J.J. Hull (1987) 'A computational theory and algorithm for fluent reading', *Proc. 3rd. IEEE Conf. on AI*

J.R. Hurford & B. Heasley (1983) '*Semantics: a coursebook*', Cambridge University Press

P. Jacobs (1989) 'Making sense of lexical acquisition', *Proc. 1st International Lexical Acquisition Workshop,* Detroit, Michigan, pp.63-69

E. Javal (1879) 'Essai sur la physiologie de la lecture', *Annales D'Occulistique,* **82,** pp.242-253

F. Jelinek, R.L. Mercer & L.R. Bahl (1983) 'Continuous speech recognition: statistical methods', *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. PAM-5

F. Jelinek (1985) 'The Development of an experimental discrete dictation recognizer', *Proc. of the IEEE,* **73,** pp.1616-1623

K. Jensen & J. Binot (1988) 'Dictionary text entries as a source of knowledge for syntactic and other disambiguations', *Proceedings of the 2nd Conference on Applied Natural Language Processing,* Austin, Texas

S. Johansson (1980) 'The LOB corpus of British-English texts: presentation and comments', *ALLC Journal,* **1**

S. Johansson, R. Garside, K. Hofland & G. Leech (1986) '*The Tagged LOB Corpus: Vertical/Horizontal Version*', Norwegian Computing Centre for the Humanities, Bergen University

S. Jones & J. Sinclair (1974) "*English Lexical Collocations*", Cahiers de Lexicologie, **24,** pp.15-61.

U. Jost & E.S. Atwell (1993) 'Deriving a probabilistic grammar of semantic markers from unrestricted English text', *Proc. of IEE Colloquium on Grammatical Inference,* Essex University, Colchester

M.A. Just & P.A. Carpenter (1978) 'Inference processes during reading: Reflections from eye fixations', in J.W. Senders, D.F. Fisher & R.A. Monty (Eds.) '*Eye Movements and the Higher Psychological Functions*', Erlbaum, Hillsdale, NJ

M.A. Just & P.A. Carpenter (1987) '*The Psychology of Reading and Language Comprehension*', Allyn & Bacon Inc., Boston

J.J. Katz & J.A. Fodor (1963) 'The structure of a semantic theory', *Language,* **39,** pp.170-210

F.G. Keenan (1990) 'The Use of Linguistic Knowledge in a Handwriting Recognition System', *Unpublished Transfer Report for CNAA*

F.G. Keenan (1992) 'Large Vocabulary Syntactic Analysis for Text Recognition', *Unpublished PhD Thesis,* Nottingham Trent University

E.F. Kelly & P.J. Stone (1975) '*Computer Recognition of English Word Senses*' North-Holland Publishing Company, Amsterdam

D.E. Kieras (1980) 'Initial mention as a signal to thematic content', *Memory and Cognition,* **8,** pp.345-353

W. Kintsch T.A. van Dijk (1978) 'Toward a model of discourse comprehension and production', *Psychological Review,* **85,** pp.363-394

R. Krovetz & W.B. Croft (1987) 'Word-sense disambiguation using machine-readable dictionaries', *Proc. 3rd Waterloo Conference of the UWC for the New Oxford English Dictionary,* pp.127-135

I. Lancashire (1987) 'Using a Textbase for English-language Research', *Proc. 3rd. Ann. Conf. of UWC for New Oxford English Dictionary,* Waterloo

G. Leech (1993) '100 million words of English: The British National Corpus (BNC) Project', *English Today*

G. Leedham (1989) 'Pitman's handwritten shorthand: Machine recognition and transcription', *Proc. Fourth IGS Conference,* Trondheim, Norway

M. Lesk (1986) 'Why I want the OED on my computer and when I'm likely to have it', *Sigcue Outlook,* **19,** pp.62-66

M. Lesk (1987) 'Automatic sense disambiguation using machine readable dictionaries', *Proc. 4th SIGDOC Conf. of Assoc. for Computing Machinery*

V.D. Lesser, R.D. Fennel, L.D. Erman & D.R. Reddy (1977) 'Organisation of the HEARSAY-II speech understanding system', *IEEE Trans. ASSP,* **23,** pp.11-23

S.E. Levinson & M.Y. Liberman (1980) 'Speech recognition by computer' *Scientific American*

W.A. Lea (1980) (Ed.) '*Trends in speech recognition*', Prentice-Hall

R. Mackin (1978) 'On collocations: 'words shall be known by the company they keep'', *"In honour of A.S. Hornby"*, OUP, pp.149-165

J.M. Mandler & N.S. Johnson (1977) 'Remembrance of things parsed: Story structure and recall', *Cognitive Psychology,* **9**, pp.111-151

J. Markowitz, T. Ahlswede & M. Evans (1986) 'Semantically significant patterns in dictionary definitions', *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics,* Columbia, pp.112-119

A. McKinnon (1975) 'Aberrant frequencies as a basis for clustering the works of a corpus', *CIRPHO Review,* **3**, pp.33-52

J. McNaught (1988) 'Computational lexicography and computational linguistics', *Lexicographica,* **4**

J. McNaught (1990) 'Reusability of lexical and terminological resources: steps towards independence', *Proc. International Workshop on Electronic Dictionaries,* OISO, Kanagawa, Japan, pp.97-107

Merriam (1963) '*Webster's Seventh New Collegiate Dictionary*', G. & C. Merriam, Springfield, Massachusetts

B.J.F. Meyer (1977) 'What is remembered from prose: A function of passage structure', in R.O. Freedle (Ed.) '*Discourse Production and Comprehension: Advances in Research and Theory*', Ablex, Norwood, NJ

D.E. Meyer & R.W. Schvaneveldt (1971) 'Facilitation in recognising pairs of words: Evidence of a dependence between retrieval operations', *Journal of Experimental Psychology,* **90**, pp.227-234

G.A. Miller (1985) 'WordNet: a dictionary browser', *Proceedings of the First Conference of the UW Centre for the New Oxford English Dictionary,* University of Waterloo, Canada

M. Minsky (1975) 'A framework for representing knowledge', in P. H. Winston (Ed.) '*The Psychology of Computer Vision*', McGraw-Hill, New York

R. Mitton (1986) '*The Machine Usable Form of the Oxford Advanced Learners Dictionary*', Oxford Text Archives

R. Montague (1970) 'Pragmatics and intensional logic', *Synthese,* **22**, pp.68-94

C. Morris (1938) '*Foundations of the Theory of Signs*', Chicago University Press, Chicago, Illinois

J. Murray (1928) (Ed.) '*The Oxford English Dictionary*', Oxford University Press, Oxford

B. Normier & M. Nossin (1990) 'GENELEX project: EUREKA for linguistic engineering', *Proc. International Workshop on Electronic Dictionaries,* OISO, Kanagawa, Japan, pp.63-70

C. K. Ogden & I. A. Richards (1923) '*The Meaning of Meaning*', Routledge & Kegan Paul, London

M. Phillips (1985) '*Aspects of Text Structure*', North-Holland, Amsterdam

T. Plate (1989) 'Obtaining and using co-occurrence statistics from LDOCE', in B. Boguraev & E. Briscoe (Eds.) '*Computational Lexicography for Natural Language Processing*', Longman, London, pp.202-210

L. Postman & G. Keppel (1970) '*Norms of Word Association*', Academic Press, New York

P. Procter (1978) (Ed.) '*Longman Dictionary of Contemporary English*', Longman Group Ltd., London

K.E. Pugh, S. Harness, N. Sherkat & R.J. Whitrow (1992) 'Icon recognition for providing access to WIMP interfaces for the blind and partially sighted', *Proc. of the First Workshop on Iconic Communication,* Brighton, UK.

A. Ramsay (1987) 'What we say and what we mean', *AI Review,* **1**

A. Raw, B. Vandecapelle & F. Van Eynde (1988) 'EUROTRA: an overview', *Interface,* **3**, pp.5-32

K. Rayner, M. Carlson & L. Frazier (1983) 'The interaction of syntax and semantics during sentence processing: Eye movements in the analysis of semantically biased sentences', *Journal of Verbal Learning and Verbal Behavior,* **22**, pp.358-374

G.M. Reicher (1969) 'Perceptual recognition as a function of meaningfulness of stimulus material', *Journal of Experimental Psychology,* pp.274-280

A. Renouf (1987) 'Corpus Development', in J. Sinclair (Ed.), *Looking up: An Account of the COBUILD Project in Lexical Computing*, Collins, Glasgow

E. Rosch (1975) 'Cognitive representation of semantic categories', *Journal of Experimental Psychology*, **104**, pp.192-233

T.G. Rose & L.J. Evett (1992) 'A large vocabulary semantic analyser for handwriting recognition', *AISB Quarterly*, **80**, pp.34-39

T.G. Rose & L.J. Evett (1993a) 'Handwriting recognition using semantic information', *Proc. Sixth Int. Conference on Handwriting and Drawing*, Paris, France

T.G. Rose & L.J. Evett (1993b) 'Text recognition using collocations and domain codes', *Proc. of the Workshop on Very Large Corpora*, Columbus, Ohio, pp.65-73

T.G. Rose & L.J. Evett (1993c) 'Semantic analysis for large vocabulary cursive script recognition', *Proc. Second IAPR Conference on Document Analysis and Recognition*, Tsukuba Science City, Japan

D. E. Rumelhart, P. H. Lindsay & D. A. Norman (1972) 'A process model for long-term memory', in E. Tulving & W. Donaldson (Eds.) '*Organization of Memory*', Academic Press, New York

N. Sager (1981) '*Natural language information processing: A computer grammar of English and its applications*', Addison-Wesley

G. Sampson (1987) 'Evidence against the grammatical/ungrammatical distinction', in W. Meijs (Ed.) '*Corpus Linguistics and Beyond*', Rodopi, Amsterdam, pp.219-226

G. Sampson (1989) 'How fully does a machine-usable dictionary cover English text?', *Literary and Linguistic Computing*, **4**, pp.29-35

K.M. Sayre (1973) 'Machine recognition of hand-printed words: a project report', *Pattern Recognition*, **5**, pp.213-228

R. C. Schank (1972) 'Conceptual dependency: A theory of natural language understanding', *Cognitive Psychology*, **3**, pp.552-631

R.C. Schank & R.P. Abelson (1977) '*Scripts, Plans, Goals and Understanding*', Lawrence Erlbaum Associates, Hillsdale, NJ

H. Schuetze (1993) *"Word space"*, in S. Hanson, J. Cowan & C. Giles (Eds.) "Advances in Neural Information Processing Systems", San Mateo CA, Morgan Kaufman.

M. Schwartz & A. Flammer (1981) 'Text structure and title - effects on comprehension and recall', *Journal of Verbal Learning and Verbal Behaviour,* **20,** pp.61-66

R.A. Sharman (1989) 'Observational evidence for a statistical model of language', *IBM Research Report, UKSC 205*

R.A. Sharman (1990) 'The development and use of corpus-derived probabilistic language models', *Paper for Speech and Language Technology Workshop on Corpus Resources,* Oxford, pp.16-20

N. Sherkat, R.K. Powalka & R.J. Whitrow (1993) 'A parallel engine for real-time handwriting and optical character recognition', *Proc. JET POSTE 93 - The 1st European Conference on Postal Technology,* Nantes, France

J. Sinclair (1970) *'English Lexical Studies: Report to OSTI on Project C/LP/08'*, Dept. of English, University of Birmingham

J. Sinclair (1987) *'Looking up: An Account of the COBUILD Project in Lexical Computing'*, Collins, Glasgow

B. Slator (1989) 'Using context for sense preference', *Proc. 1st International Lexical Acquisition Workshop,* Detroit, Michigan, pp.77-84

F. Smadja (1989) 'Macrocoding the lexicon with co-occurrence knowledge', *Proc. 1st International Lexical Acquisition Workshop,* Detroit, Michigan, pp.197-204

C. Souter (1990) 'Systemic-functional grammars and corpora', in J. Aarts & W. Meijs (Eds.) *Theory and Practice in Corpus Linguistics,* Rodopi Press, Amsterdam

K. Sparck Jones & D. Jackson (1967) 'Current approaches to classification and clump-finding at the Cambridge Language Research Unit', *Computer Journal,* pp29-37

M. Stefik (1981) 'MOLGEN I and II', *Artificial Intelligence,* **16,** pp.111-170

G.J. Sussman & G.L. Steele (1980) 'CONSTRAINTS - a language for expressing almost hierarchical descriptions', *Artificial Intelligence,* **14**, pp.1-39

C. Tappert, C. Suen & T. Wakahara (1990) 'The state of the art in on-line handwriting recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence,* **12**

C.T. Tappert (1984) 'Adaptive on-line handwriting recognition', *Proc. 7th International Conference on Pattern Recognition,* pp.1004-1007

A. Tarski (1931) '*Logic, Semantics, Metamathematics*', OUP, Oxford

P.W. Thorndyke (1979) 'Knowledge acquisition from newspaper stories', *Discourse Processes,* **2**, pp.95-112

P. Toma (1977) 'SYSTRAN as a multi-lingual machine translation system', *Commission of the European Community: Overcoming the Language Barrier,* Verlag Dokumentation, Munich

J. Vaissiere (1985) 'Speech recognition: a tutorial', in F. Fallside & W.A. Woods (Eds.) '*Computer Speech Processing*', Prentice Hall

P.A. de Villiers (1974) 'Imagery and theme in recall of connected discourse', *Journal of Experimental Psychology,* **54**, pp.180-187

P. Vossen, W. Meijs & M. den Broeder (1988) 'Meaning and structure in dictionary definitions', in B. Boguraev & E. Briscoe (Eds.) '*Computational Lexicography for Natural Language Processing*', Longman, London, pp.171-192

D.E. Walker (1980) 'SRI research on speech understanding', in W.A. Lea (Ed.) '*Trends in Speech Recognition*', Prentice-Hall

D. E. Walker & R. A. Amsler (1986) 'The use of machine-readable dictionaries in sublanguage analysis', in R. Grishman & R. Kittredge (Eds.) '*Analyzing language in restricted domains*', LEA, Hillsdale, NJ, pp.69-83

D.E. Walker (1989) 'Developing lexical resources', *Proc. 5th Annual Conference of the UWC for the New OED,* Oxford, pp.1-22

D.L. Waltz & J.B. Pollack (1985) 'Massively parallel parsing: a strongly interactive model of natural language interpretation', *Cognitive Science,* **9**, pp.51-74

C.J. Wells, L.J. Evett & R.J. Whitrow (1989) 'The use of orthographic information for script recognition', *Proc. of the Fourth IGS Conference,* Trondheim Norway

C.J. Wells, L.J. Evett & R.J. Whitrow (1991) 'Word look-up for script recognition - Choosing a candidate', *Proceedings of the First Int. Conference on Document Analysis and Recognition,* St. Malo, France

C.J. Wells (1992) 'The Use of Orthographic and Lexical Information for Handwriting Recognition', *Unpublished PhD Thesis,* Nottingham Trent University

C. Wheddon (1990) 'Speech Communication', in C. Wheddon & R. Linggard (Eds.) '*Speech and Language Processing*', Chapman & Hall, London, pp.1-28

P. Whitelock, M. Wood, H. Somers, R. Johnson & P. Bennett (1987) (Eds.) '*Linguistic Theory and Computer Applications*', Academic Press, New York

R.J. Whitrow & C. Higgins (1987) 'The application of n-grams for script recognition', *Proceedings of the Third Int. Symposium on Handwriting and Computer Applications*

Y. Wilks (1973) 'An artificial intelligence approach to machine translation', in R.C. Schank & K.M. Colby (Eds.) '*Computer Models of Thought and Language*', W.H. Freeman, San Francisco

M. Wilson (1984) 'The Composition of the Mental Lexicon', *Unpublished PhD Thesis,* Cambridge University

T. Winograd (1983) '*Language as a Cognitive Process*', **1:** Syntax, Addison-Wesley

L. Wittgenstein (1953) '*Philosophical Investigations*', Blackwell, Oxford

J.J. Wolf & W.A. Woods (1980) 'The HWIM speech understanding system', in W.A. Lea (Ed.) '*Trends in Speech Recognition*', *Prentice-Hall*

P.T. Wright (1989) 'Algorithms for the recognition of handwriting in real-time', *Unpublished Ph.D. Thesis,* Nottingham Polytechnic

U. Zernik (1989) 'Paradigms in lexical acquisition', *Proc. 1st International Lexical Acquisition Workshop,* Detroit, Michigan

Appendix A

Calculation of the z-score

(a) Mean Difference

$$md = \frac{\sum D}{N}$$

(*b*) Sum of Squares for D

$$\sum d^2 = \sum D^2 - \frac{(\sum D)^2}{N}$$

(c) Standard Deviation of D

$$S_D = \sqrt{\frac{\sum d^2}{N}}$$

(*d*) Standard Error of Mean Difference

$$S_{\bar{D}} = \frac{S_D}{\sqrt{(N-1)}}$$

(*e*) z - score

$$z = \frac{md}{S_{\bar{D}}}$$

where N = number of pairs

D = difference between the correct word and highest other

When the sample size is large, the standard score (*z*) is used to measure the ratio of the difference between the means to the standard error of this difference. The z-score is interpreted using normal probability tables. When the sample size is small (e.g. < 30) the *t* Ratio or Student's *t* is used instead of the normal probability tables.

Appendix B

The Collocation Algorithm

The words co-occurring with each lemma are treated as if that lemma were the node of a set of linguistic collocations [Lancashire, 1987]. To find the collocates for a given lemma, the following algorithm is used:

(1) Obtain a concordance of the lemma, i.e.
(a) find all of its occurrences in the corpus, then
(b) edit the context of each occurrence so that it extends no more than four words either side of the lemma.

(2) The concordances are then treated as a "mini-text", and subjected to a word frequency analysis to discover the collocates of the node. This involves the use of a statistical procedure known as the *z-score* to measure the strength of association between a lemma and its collocates. This "distance relation" score is based on the *actual* frequency of a collocate in the mini-text compared with the *expected* frequency of the collocate *were all words distributed randomly throughout the text.* A collocate may seem to occur frequently in the mini-text when in fact it occurs fewer times than it should on average. The formula used is shown overleaf, whereby:

- **P** is the probability that a word selected from the rest of the corpus is the target word or collocate;
- **E** is the expected probability that the collocate will appear in the mini-text;
- $\mathbf{S_D}$ is the standard deviation;
- **z** is the z-score.

L_{mini} = length of minitext

F_{mini} = frequency of collocate in minitext

L_{rest} = length of rest of corpus

F_{rest} = frequency of collocate in the rest of corpus

$$P = \frac{F_{rest}}{L_{rest}}$$

$$E = P \times (L_{mini})$$

$$S_D = \sqrt{(L_{mini} \times (P \times (1-P)))}$$

$$z = \frac{(F_{mini} - E)}{S_D}$$

Appendix C

Character Lattice Format

The character recogniser may propose a selection of possible characters as interpretations of the input. These alternative character sequences may be conceived of as a lattice structure, whereby each segment of the input string is represented as a set of possible character combinations along with a probability score to indicate how close the candidate character matched its template stored in the database. Table 8.1 shows the contents of a typical lattice for the word "*its*". A graphical representation of this is shown in Figure 8.1.

The lexical analyser traverses this lattice to determine which of the possible character sequences represent allowable English words. In the example shown, the allowable words are "*its*", "*ox*" and "*us*". Once these words have been identified, their syntactic and semantic information is made available to the relevant analysers for further processing.

Node Number	Characters and their Probability	Next Node(s)
0	(start of word)	1, 3, 6
1	i:94 v:60 r:50 o:45 l:25	2, 5
2	t:80 f:60 l:50 i:40 x:40	4
3	h:81 u:75 k:71 b:66 n:31	4
4	s:53	7
5	x:85 u:76 b:67 v:42	7
6	m:75	7
7	(end of word)	-

Table 8.1: Graph Representation of a Character Lattice

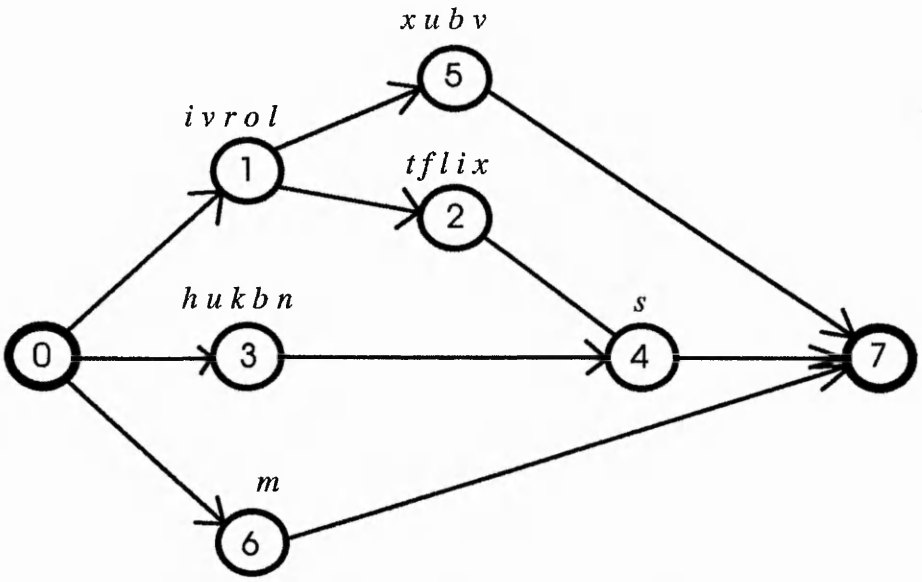


Figure 8.1: Graph Representation of a Character Lattice

Appendix D

Handwritten Business Text

The data in Section 2 below represents the results of semantic analysis on a 522-word business document. The text of the original document is given in Section 1. This text was handwritten and recognised (on-line) by a developmental cursive handwriting recognition system, incorporating a lexical analyser. The word candidates are therefore a product of the lexical analysis, whereas the scores, normalised over the range 0-25, are the product of semantic analysis. Each word position is shown on a separate line, with the correct word being shown in the first position wherever possible. In cases where the recogniser could not identify any allowable words this is indicated by the use of the "????" marker.

1. Text of the Original Document

The present decade has already been christened the Caring Nineties, and this attitude is certainly reflected in the political headlines by the dominance of debates on health, education and local government. However, the need to apply this strategy to business has so far been overlooked. Following the Enterprise Eighties it is time for business as well to address the Caring Nineties. After the rough and tumble of the last twelve years when minds were focussed on seizing opportunities, surviving the challenges of competition and achieving success, now is the time to take a breath and consider the future. Though the competitive spirit must not be dampened, the approach needs to change. No more the flash in the pan idea which brings a quick reward and then dies - in order to be successful over the next twelve years a longer term emphasis involving thought and planning is required. A disposition to continuous advancement will be vital for success. Without constant development of processes and products, in line with technological progress and changes in consumer demands, competitors will start

to impinge on market share and profits will fall. Such progress will only be possible if every company recognizes the importance of R and D, innovation and technology transfer and is willing to engage in such activities. It is these factors that will be the key to competitive advantage in the nineties. For those companies presently fighting to maintain their existing levels of business such statements may seem idealistic. However, now is the ideal time for businesses to examine their planning policies and shift their focus from short term results to long term performance. Unless companies are prepared to take the risk with investment in their futures, production will stagnate, demand will fall and any hope of improving our export position will disappear. Private spending on research and development and innovation in Britain is amongst the lowest in Europe at the moment. Though the government is largely responsible for this situation, it is also the consequence of British companies traditionally regarding investment in these areas as a luxury. This is an attitude that must change. In a speech at the University of Warwick last week, Peter Lilley called for a cultural revolution in industry, science and finance. This he said was needed to overcome a bias against practical skills and industrial occupations which has been in existence in Britain since the industrial revolution. Such a massive swing in attitude will take a long time to materialize and will require change from educational grass roots throughout every form of business, commercial and scientific interest. Investment for progress and development must become an instinctive way of life for British business. The government will have a vital role to play if this change in attitude is to be effected. Though an increase in public spending on R & D is always useful alone it will not produce the required outcome. The focus must be on raising the status of innovation, making firms recognize its influence and importance for success, encouraging industry and science to work effectively together and altering financial pressures to facilitate this shift in emphasis.

2. *Results of Semantic Analysis*

the 11 tre 0 tie 25 tell 22 tic 1 till 11 tile 23 tit 3 tilt 9
present 25 pursuit 4 pulsate 0 pursed 4 prelate 0 prised 1 proved 8 priced 4 putted 8
decade 18 tillage 0 teetotal 0 village 25 titlark 0 dilate 0 oblate 0
has 6 liar 3 hm 0 lion 25 lint 0
already 6 alveolars 0 lineally 0 unfolds 9 reveilles 0 critically 9 cleverly 25 calculus 12 unlearns 0 cruelties 3
been 6 keen 21 teen 0 boon 0 eon 0 felon 0 local 25
christened 3 clustered 25 blustered 0 glistened 3 glistered 0 glittered 5 unlettered 0 conceited 2 contrived 11 fructified 0

the 6 tell 24 tre 0 till 6 tic 1 tie 10 he 6 tile 14 lie 25
caring 6 curing 11 coning 15 coring 13 luring 3 toning 12 lolling 25 lining 12
nineties 3 unities 7 unites 4 uncles 4 lulling 4 lilting 25 nudes 0 nuclei 6 lunches 6 ringlet 1
and 11 curd 7 cud 0 avid 1 end 25 arid 7 aid 12 find 6
this 13 tin 16 lilts 25 tills 0 lolls 14 lull 7 lint 0 lilt 25 till 13 loll 14
attitude 24 altitude 8 lettuce 5 retrieval 13 neutral 25 athletic 12 whittle 3
is 18 it 18 s 18 j 18 i 18 t 18 ti 25
certainly 25 certainty 25 electrify 2 torrential 8 creations 17 tortillas 0 creditors 4 treaties 20 carollers 8 clearway 0
reflected 25 inflected 3 inflicted 4 infected 20 rejected 20 refuted 4 refitted 0 revolted 13 injected 8 filleted 2
ill 25 in 8 it 8 nt 0 lit 9 tit 12 m 8 hi 2
the 9 tre 0 tell 25 tc 0 lie 16 he 9 tie 21 lilt 1
political 25 poetical 14 preterite 0 arterial 2 libertine 0 tideline 0 poultice 0 inertial 0 secretive 17
headlines 4 healthier 25 healthily 25 volatilities 4 illegally 15 lichgates 0 rendition 0 realities 0 localities 17
by 8 ill 25 oil 11 lit 11 bit 14 tit 15 its 8
the 5 tre 0 tee 1 tell 25 tec 4 tie 9 till 5 tile 5
dominance 25 doctrinaire 0
of 11 col 0 if 11 or 11 lei 0 let 25 lot 25
debates 19 deflates 12 derates 25 donates 4 deputes 6 deviates 7 dictates 17 telltales 0 dilates 1 tunnels 5
on 10 er 0 or 10 sir 0 cell 7 oil 16 tell 25 sit 10
health 18 heath 0 matter 11 heater 25 molten 4 leant 7 helmet 4 millet 1
education 8 educator 8 titillation 1 levitation 0 centurion 0 couturier 0 flirtation 0 contriver 3 cantilever 25
and 25 curd 8 arid 4 rind 1 cud 0 unit 10 avid 1 null 1 vivid 4
collie 0 helot 0 trefoil 25
????
rued 1 need 8 reed 25 wed 2 rood 0 ivied 0 nod 2 iced 3 vied 1 void 2
to 23 tc 0 ti 25 it 23
apply 3 appal 2 alley 2 abbey 0 uphill 2 apron 25 apple 8
this 25 tills 0 trill 0 lilt 0 tells 11 tin 11 till 25 tilts 6 flit 0
strategy 17 tetchily 0 servility 1 terrorists 2 tireless 25
to 16 tr 0 tv 0 ti 25 tb 0 h 16 t 16 it 16 w 16
fullness 21 rustless 11 bristliest 13 business 8 fulness 0 buttress 3 stillness 25 listless 1 frilliest 1 silliness 16
has 25 leis 0 hers 25
so 25 tb 0 to 25 tv 0 ti 17 it 25 tr 0
fill 8 far 4 fur 16 for 25 fa 1 foil 2 fin 5 flu 0 loll 0 fir 3
bear 9 boar 3 sear 0 scar 6 been 25 soar 7 lear 0 beer 3 boon 0 seer 1
????
foretelling 14 retelling 9 selectivity 17 selecting 17 fleecing 10 ululating 0 filtrating 6 relieving 25 solitarily 21 volubility 0
the 24 tc 0 tell 15 tre 0 tie 4 tr 0 tic 0 ti 25 till 24
enterprise 25 interstice 1 lutanist 0 lutenist 0
colitis 0 notifies 2 uglifies 7 eighties 2 eights 0 utilities 8 civilities 9 nostrils 2 listens 25 lights 8
its 21 ti 25 tb 0 us 21 is 21 tit 4 ill 11 tr 0
is 16 oi 0 it 16 y 16 s 16 ti 25 g 16 i 16
time 7 twill 0 tune 5 tine 0 tire 9 tulle 0 trill 0 till 25 twirl 1 true 9
loll 0 he 25 to 25 roll 24 re 25
business 7 bristliest 3 justness 25 listless 1 lustier 1 lustiest 1 bristlier 3 bristles 3 bistros 0 pistils 0
as 25 us 25 err 2 ai 23 lit 21 or 25 es 0 er 0
null 1 mill 11 need 25 melt 19 milt 0 reed 6 roll 24 ivied 1
tv 0 tr 0 ti 23 t 25 h 25 w 25 it 25

address 25 goalless 0 littleness 20 coldness 19 holiness 4 niceness 23 erectness 10 alcoholic 12 triteness 0
the 19 tre 0 tic 4 tell 13 tie 19 till 19 tee 5 tec 6 tile 25
caring 13 lowing 25 coring 11 towing 0 cowing 23 cairns 3 lairds 0 carries 18 lorries 6 cabins 3
nineties 20 invites 25 utilities 24 rivets 17 lintels 3 illicitly 0 riots 8 lintel 3 trinities 10 illumes. 0
after 25 atte 0 alter 12 lift 10 utter 23 litter 23 offer 9 otter 0 aster 0
the 23 tre 0 tie 25 tic 6 tell 17 till 23 tc 0 tile 25 tee 6
rough 18 vouch 0 ouch 0 nigh 23 inch 25 vich 0 lough 0 rich 11 oriel 2
and 25 curd 14 cruel 10 arid 5 cud 0 avid 1 cruet 0 end 15 unit 9
tumble 25 tremble 18 tense 11 indite 0 rinse 3 revise 18 nurse 7 temple 0 lunette 0
of 25 or 25 oi 0
the 25 tre 0 lie 11 tell 11 tie 4 lit 5 tic 0 till 25 lilt 3
last 14 lust 2 lout 0 lent 3 lost 9 lint 0 lest 25 tart 2 fast 11
twelve 25 truffle 5 culture 8 rueful 5 rubric 1 lunette 0 innate 6 entente 0
sculls 0 scowls 0 scows 0 scouts 11 levels 7 secrets 25 lentil 2 tenets 1
literal 15 lichen 0 littler 25 tiller 0 victor 0
rinds 1 winds 8 winch 1 finds 9 writs 6 muds 7 finch 0 much 25 funds 11 finis 0
mere 7 were 25 wore 4 nerve 5 more 25 mire 7 rune 1 rove 1 wove 1
focussed 25 realised 0 realistic 15 trellised 0
on 25 car 8 sir 0 or 25 err 1 er 0 cell 7
seizing 22 busing 25
????
infilling 0 shrilling 25 instilling 1 shrining 1 jelling 0 thrilling 6 stunning 3 skinning 4 skittling 0 lulling 1
tell 14 tre 0 tee 1 till 25 tile 18 tie 4 lie 4 tilt 5 lit 12
challenges 10 converses 25 converges 5 traverses 6 correctives 10 favourites 6 currencies 6 detectives 3 frivolities 2 crinolines 0
of 25 or 25 g 25 if 25 y 25 f 25 oil 16 er 0 col 0
????
and 25 curd 4 cud 0 arid 4 avid 1 end 17 aid 5 null 2 rid 12
achieving 16 accruing 9 attuning 8 relieving 25 relining 0 abutting 1 coveting 1 conning 11
success 16 stillest 25 silliest 20 littlest 20 incest 4
roll 25 iron 7 loll 1 ion 7 doll 5 troll 1
is 25 ti 24 it 25 i 25 r 25 s 25 tr 0
the 25 tell 12 tee 1 till 25 tec 1 tile 19 tae 0 tilt 6 felt 24 flit 1
trill 0 tulle 0 tire 25 tile 22 hint 7 title 4 hurl 0
to 20 tv 0 ti 25 tr 0 tb 0 it 20
toff 25
tr 0 u 25 oi 0 a 25 ti 19 n 25 h 25 it 25
breath 7 swath 0 sleuth 0 sleetier 0 swelter 1 stretch 13 stiller 25 stencil 1
????
????
the 25 tell 19 till 25 tie 13 tile 22 tic 6 lie 22 tee 1 he 25 tilt 8
literal 25 ritual 11 fettle 0 triune 0 initial 19 trivial 22 intuit 0 revolt 10 feline 0 fetal. 4
though 25 trough 4 trench 5 touch 18 tough 5 tench 2 torch 11 lough 1
the 19 tre 0 tell 22 tie 15 tic 5 till 19 lie 25 lit 14 tile 15
constitute 25 constrict 2
spirit 25 sprit 0 spirt 0 split 15 invite 7 sprite 2 trifle 10 thrill 13 unlit 0 glint 3
must 25 rust 4 unit 7 rest 14 unfit 9 mist 20 nest 16 next 25 mutt 0 mute 17
not 25 riot 4 rot 16 viol 0 lute 0 nor 25 dolt 0 title 10 hot 6
lie 25 he 19 be 19 tie 17 tic 5 vie 0 re 19
enlivened 2 delivered 25 fluttered 14 frittered 5 countered 18
the 25 tre 0 tell 13 lie 25 tie 21 lit 6 he 25 till 25 tic 6
approval 25 aristocrat 16 territorial 12 diploid 0 attribute 13
needs 14 reeds 5 weeds 25 weds 11 nods 14 roods 0 voids 5 rods 9 vouch 0

to 25 tv 0 ti 7 tb 0 tr 0 it 25
change 15 charge 12 chance 10 dance 25 deluge 1 deuce 0 douce 0 cringe 1 clavicle 0 coerce. 4
no 25 to 25 ho 14
wove 2 wore 6 mere 18 uncut 0 move 14 wont 1 rout 1 more 25 rent 7 inert 6
the 15 tell 7 tee 1 tre 0 tec 25 tae 0 till 15 tile 6 tie 7 tire 7
flush 8 flash 17 flail 2 leash 2 ease 25 rase 0 fence 15 leach 0 flair 1 ruse 2
in 15 ill 6 lit 6 tit 2 nt 0 hi 25 id 1 it 15 ti 2
the 25 tell 12 tre 0 lie 8 tc 0 tie 16 tic 0 till 25 lit 14
pun 0 pan 5 par 6 pair 17 pull 25 lull 2 pall 1 sun 24 flan 0 run 6
idea 17 din 2 cha 0 den 25 lain 4 dill 0 ain 0
which 25 lowlier 6 collier 1 courier 1 littler 8 vitriol 0 illicit 0
livings 24 sings 19 snugs 11 lings 0 lungs 7 tings 2 slings 2 pillar 25 stings 2
er 0 oi 0 u 25 n 25 a 25 tr 0 h 25 r 25 ti 11
guide 4 guile 1 quite 25 crude 4 quell 0 gullet 0 dwell 8 glide 4 quill 0 dulcet 0
reward 3 inward 25 lunatic 1 herald 1 toward 25 uncivil 0
and 25 curd 2 hull 0 arid 3 avid 0 cud 0 lull 1 null 1 unit 3
then 25 thou 25 thin 4 their 25 lien 17 lieu 0 tour 3
dies 25 dill 0 din 1 flies 17 flier 0 fill 13 vies 14 diy 0
oil 2 ill 5 in 25 ai 8 on 25 id 3 oi 0 or 25
order 14 cider 1 older 25 elder 11 bidet 0 clan 8 croon 1 tiger 3
to 25 tv 0 ti 16 tr 0 tb 0 h 25 if 25 t 25 it 25
be 25 lie 5 he 25 tie 11 tic 0 re 25 vie 3
successful 25 successive 16
one 25 cue 1 over 25 crier 0 own 25 ore 5 iron 5
the 25 tell 6 tee 2 tec 3 toll 1 till 25 toe 1 tile 5 tre 0 tie 7
next 25 reft 0 nett 0 writ 6 refit 0 eft 0 wit 1 rift 2 red 7
twelve 25 tulle 0 truffle 4 trifle 1 erudite 1 willful 7 tweet 0
years 25 leafs 25 tzars 0 tears 15
a 25 u 25 d 25 tr 0 n 25 it 25 ti 5
longa 0 longer 25 linger 7 tinges 3 tourer 0 lough 1 truces 2 conger 0
term 25 turn 18 tarn 4 tern 0 torn 10 burn 11 larn 0 bum 6 loom 11
emphasis 20 illiberally 18 limbless 0 curtails 6 artillery 12 entrails 1 enuresis 0 cuirass 0 airless 25 limitless 17
involving 25 intuiting 0 lunching 12 inverting 19 hireling 0 intently 22 lulling 3 hitching 1 hulling 14 tunefully 18
????
curd 2 and 25 cud 0 arid 3 avid 0 turd 0 wid 0 end 10
planning 20 plaining 9 blinding 13 plunging 19 picturing 25 plurality 19 blurring 6 plaudits 0 branding 10 braiding 0
is 25 s 25 y 25 j 25
required 10 recruited 17 reclined 1 leisured 9 requited 0 recurred 25 licence 4 rectified 14 filleted 2 nitwitted. 0
oi 0 u 25 n 25 a 25 tr 0 ti 8 d 25 it 25
????
tr 0 to 25 tv 0 ti 7 tb 0 t 25 h 25 it 25 o 25
continuous 25 centurions 2 continuity 11 cotillions 0 carillons 0
advancement 25
will 25 rill 0 wilt 3 cull 1 riff 10 niff 0 cult 11
be 25 re 25 ze 0 bf 0 pi 2
vital 5 idol 1 little 8 dirt 1 trial 3 did 25 trill 2 tile 3
for 25 fer 0 loll 0 roll 2 fa 0 fill 4 rill 0 toll 0
success 20 gullets 0 gutless 10 stillest 15 cutlets 0 circlets 0 silliest 8 cuticle 4 littlest 25 cutest. 2
titillate 2 virulent 1 titular 0 trilled 8 tittivate 0 imitate 25 initiate 22 lifeline 0 lulled 2
constant 8 construe 1 continue 25 telltale 0 catlinite 0 conclave 0 conserve 17 halliard 0
development 25 concurrent 3 curtailment 5

of 25 if 25 bf 0 or 25 oi 0 tr 0 tv 0 it 25 ti 11
processes 25
curd 5 and 25 cud 0 arid 2 avid 0 vivid 3 aid 5 wid 0 null 1
products 20 flotillas 0 produces 25 strands 6 trounces 0 provinces 24 sittings 21 litanies 2 rivalries 7
in 25 ill 5 tr 0 lit 4 nt 0 ti 10 tit 0 it 25
lire 0 frill 2 tire 18 ill 19 fire 25 hill 0 furl 0
with 25 litter 5 vital 3 liter 0 wilt 1 titter 0 trill 0 will 8 virtu 0
technological 25 ethnological 0 tautological 1
progress 25 frostiest 0 prosiest 0 poshest 1
and 25 curd 2 avid 0 cud 0 wid 0 unit 5 writ 9 end 5
changes 25 clanger 0 charges 22 charger 1 chances 12 clavicles 0 divulges 0 granges 0 flanges 3
in 25 ill 6 id 5 lit 8 tit 0 it 25 ti 3 nt 0
connexion 25 construct 12 conjurer 3 constrict 2 contriver 9
demands 25 lowlands 1 dullards 0 almonds 5 councillors 4 alliances 11 actuarial 12 volitional 1
inheritors 0 outfitters 25 interiors 13
will 4 rill 0 cill 0 wilt 1 ill 7 cull 0 all 25 cult 3
stunt 5 start 25 stud 6 staid 0 slant 1 strut 2 stout 8 stool 6 clout 4 fart 4
to 25 tc 0 tv 0 ti 8 tb 0 t 25 it 25 b 25
impinge 10 unhinge 0 impulse 25 misrule 0
car 6 on 25 oar 1 cell 2 or 25 er 0 ai 6 tell 7
terrible 25
fall 14 full 7 fare 3 shall 25 shale 0 fore 2 share 8 snarl 1 franc 1
and 25 cud 0 curd 3 arid 5 avid 0 cull 0 hull 0 turd 0 cuff 1 aid 2
profits 25 profit 25 flouts 1 pitches 10 hitches 1 pilots 6 pieties 1 flours 4 violet 0 fronts 13
will 25 wilt 7 rill 0 niff 0 wile 5 riff 3 lull 2
tall 4 toll 2 fall 25 full 14 tale 4 tail 8 tuft 3 title 9 tool. 5
such 25 lintel 3 littler 5 stroll 2 tiller 0 sliver 0 silver 2 sneer 1
progress 25
will 25 rill 0 wilt 8 niff 0 riff 4 wile 6 lief 0
overlie 14 entitle 22 bristle 11 evolve 25 entire 7 outsize 0 snottie 0 bugle 2
possible 15 fossilize 25 solstice 0 fissure 1
i 25 if 25 f 25 ti 2 r 25 v 25 it 25 d 25 l 25 t 25
every 25 aery 0 wiry 3 clews 0 lulls 1 clears 4 vilely 1 creels 4 culls 0 oilers 0
tonsillitis 0 centrally 25 constants 18 transiently 0 contrarily 17 whistling 11 narrowly 11 neutrality 11 contrary 17 whittling 2
????
the 25 tell 6 tee 0 tec 0 tre 0 till 25 tile 2 tie 3 tic 0
importance 25
of 25 col 0 or 25 lei 0 let 4 lot 5 er 0
l 25 v 25 r 25 f 25 i 25 t 25
cud 0 curd 17 and 25 arid 1 avid 10 trill 0 livid 0 turd 0 vivid 1 hull 0
oi 0 d 25 e 25 it 25 if 25 a 25 tc 0 f 25 c 25 ti 4
????
curd 25 and 19 cud 0 civil 2 civet 0 turd 0 avid 7 dud 0 end 2 unit 1
tautology 25
transfer 25
curd 17 lend 14 and 25 omit 1 turd 0 hull 0 arid 1
is 25 it 25 i 25 n 25 ti 13 u 25
idling 2 willing 4 rifling 2 wilting 25 riding 3 luffing 0 lulling 1 lilting 0 tilling 0 lining 7
tv 0 to 25 tr 0 tb 0 ti 14 it 25 h 25 t 25 w 25
criticise 25 tillage 25
in 25 ill 3 hi 1 ai 22 lit 2 id 1 tr 0
such 25 cud 0 and 25 suet 8 inch 4 suit 7 arch 2 avid 0
activities 8 civilities 9 attrition 1 cotillion 0 loiterer 0 athletes 3 laities 0 vitrifies 0 clarifies 25 flatterer. 1

it 25 if 25 d 25 ti 14 e 25 r 25 f 25 i 25
is 25 it 25 i 25 s 25 ti 14 c 25 tc 0
these 25 trice 0 tense 4 tulle 0 truce 1 lust 1 lose 4 fuse 3 trust 4
factors 19 throve 0 victors 25 tailors 14 thrive 15 furore 0
dint 1 hint 4 tint 1 lint 0 tent 2 tulle 0 there 25 thole 0 tune 4 tutti 0
will 25 rill 0 wilt 2 wid 0 hill 0 rid 14 niff 0
he 15 lie 7 be 15 tre 0 hi 25 tie 7 tic 2 ire 0
the 15 tre 0 tell 7 lie 4 tec 25 tee 0 tc 0 till 15 tie 5 tic 2
lay 5 key 25 bey 0 buy 10 tory 0 cry 9 lolls 1 ay 2
tr 0 to 25 tv 0 ti 9 tb 0 it 25 h 25 u 25 w 25
infertile 25
cultivate 25
in 25 ill 8 lit 6 tr 0 ti 12 nt 0 tit 2 it 25
the 25 tell 9 tre 0 till 25 tee 1 tile 4 tec 3 tilt 2 tc 0 tire 5
nineties 8 lintels 3 unites 25 utilities 22 linctus 0 unities 25 trinities 6 units 9 undies 0 lunches. 13
for 25 fir 2 ill 9 of 25 lot 9 tor 0 ho 2
those 25 thole 0 hose 1 troll 1 froze 11 most 25 mote 0 toll 1 frost 5 titbit 0
companies 25 compares 23 complines 0 compatriot 1 confining 11 contraries 18 compiling 20 rerunning 0 retribution 2
presently 18 preserves 25
fighting 7 filleting 24 lighting 16 filtering 21 tigerflies 0 littering 23 fighters 7 tittering 0 lighters 15 listing 25
to 25 tc 0 ti 3 it 25
j 0
thin 10 trill 0 tiled 4 lull 1 then 25 null 2 till 25 lieu 0
existing 25 exiting 12 exiling 22
levels 7 lends 4 lures 2 tends 3 feuds 6 tenets 1 turds 0 lives 17 fends 1 lines 25
of 25 ill 5 col 0 if 25 lei 0 let 8
business 8 silliness 8 justness 22 stillness 25 listless 2 buttress 7
such 25 inch 6 stroll 6 tiller 0
selections 17 stevedores 0 seventeens 12 jewellers 25
very 25 lulls 0 hilly 6 vilely 1 rely 3 hells 10 runs 7 ley 4
scorn 4 scour 0 seem 25 zoom 6 loom 11 silent 17
idealistic. 25
honour 3 however 25 waverer 1 renown 1 vainer 3 herein 0 leveller 2 hairier 4
new 5 view 17 lien 1 hell 25 lieu 1
j 25 s 25 y 25
the 25 tre 0 tell 6 till 25 tile 1 tc 0 tee 1 tilt 2
ideal 10 dial 4 idiot 23 cleat 0 dint 1 devil 25 lactic 1 dent 4
tine 1 tune 25 twirl 0 tilde 0 true 15 tulle 0 title 13 twill 0 tittle 1
loll 1 roll 19 fin 5 fa 2 fill 25 toll 2 foil 5
businesses 25
tr 0 to 25 tv 0 ti 6 tb 0 h 25 it 25 u 25
examine 25 tideline 0
their 25 twill 0 twirl 0 trill 0 elicit 3 blur 2 vital 8 liter 0
planning 16 slurring 0 plaining 16 stunning 6 planing 8 staining 25 starving 13 starring 13 stoning 17 staring 11
polices 12 series 17 policies 25 pierces 8 polities 1 literals 17 politics 18 fiercer 16 socials 12
cruel 25 unit 22 cruet 0 null 3 tulle 0 hull 0 allot 15 oriel 3 alice 0
shift 23 stuff 9 stiff 7 shirt 6 short 25 swift 10 sift 25 stilt 0 snort 1 strife 1
their 25 thin 12 twill 0 then 25 thou 25 trill 1 toil 5
focus 7 lolly 0 folly 2 foals 2 rolls 4 lolls 0 fouls 25 locus 2
four 14 flout 1 foul 25 from 14 flour 10 flint 0 fill 3 font 6 hour 5
shout 6 short 25 stunt 1 stout 2 smelt 1 blunt 2 grunt 2 glint 1 stroll 5
teun 0 term 25 town 6 tarn 1 turn 5 tour 7 tern 0 torn 3 tun 1

results 8 lectures 9 vultures 25 nestles 4 insults 1 neutral 6 fillets 1 rustles 1 ulsters 2 rustlers 1
to 25 tv 0 ti 9 tb 0 it 25
long 25 bug 7 hag 3 brig 0 lag 8 lolly 0 big 11 lolls 1
term 25 trill 0 illicit 0 virtu 0 brat 3 tutti 0 elicit 4 brill 0
performance 25 pullulate 0 relevance 23 revolutionist 0 felonious 0 relevant 23 perchance 0 strenuous 3 reflexions. 13
unless 25 illness 3 lidless 1 littlest 3 rudest 1 unrest 1 livens 9 tunics 1
contraries 25 contraltos 0
are 25 arc 0 will 9 rue 2 cue 1 one 25 ore 1 rill 0 wilt 1
prepared 25 propound 3 profaned 1 profound 19 fulfilled 25 pigtailed 0 purloined 0 shoplifted 0 pillared 0 puissance 0
to 25 tr 0 ti 3 tb 0 t 25 it 25 h 25 b 25 u 25
tube 12 tulle 0 table 24 tape 25 abc 0 tune 21 turtle 2 tousle 0 twill 0
the 25 tell 9 tie 2 tic 0 till 25 tile 7 tc 0 lie 5 tit 5
risk 25 visit 24
with 25 tilth 0 litter 6 lull 1 wilt 1 trill 0 tiller 0
interment 0 nutrient 12 intervene 25 interwove 7 intertwine 5 nutritive 2
ill 6 in 25 lit 4 tr 0 tit 7 nt 0 it 25 hi 1
thin 1 then 25 lilt 0 loll 0 thou 25 lien 0
futures 23 tellies 0 rennet 5 lentils 2 felines 0 retires 25
production 25
will 25 wilt 14 lull 6 rill 0 ruff 0 niff 0
stagnate 25 engraft 0
dullard 0 fenland 0 devilling 21 alluring 1 lecturing 25 curricula 12 thrilling 12
will 12 rill 0 wilt 4 trill 2 rice 5 nice 4 vice 25 wife 5
tall 13 toll 2 talc 4 toff 0 tuft 6 lull 25 trill 7 tail 21 troll 1
and 25 rind 1 curd 5 auld 0 wind 8 wild 3 hula 0
any 25 lily 0 wry 1 wiry 4 airy 1 lulls 5 ally 2 culls 0
hope 22 lope 0 trope 1 lose 25 hose 6 tripe 1 tope 0 rope 16 ripe 21
of 25 if 25 bf 0 or 25 oi 0 tr 0
impurities 19 nunneries 0 villainy 25 villeins 0 implicitly 10 virulently 5 mulleins 0 villains 25 ruffianly 0 unbuckles 0
our 25 can 25 cur 0 err 1 call 2 van 4
????
position 25 postern 0
will 25 rill 0 wilt 3 hill 0 trill 1 wid 0 lull 3
disappear 14 discussion 25 interprets 10 disasters. 7
innate 12 private 25 titillate 1 filtrate 8 tittivate 0 titivate 0 lunette 0
spending 10 spelding 0 spelling 14 trending 25 splitting 7 silencing 10 sterling 2 spilling 3 spurting 1 splicing 0
on 25 err 1 sir 0 er 0 feu 0 fen 0 cell 6 fell 3
research 25 reservoir 15
curd 19 and 25 avid 10 cruel 2 arid 2 cud 0 omit 2 mid 25 cruet 0 null 1
development 25 derailment 0 containment 9 deterrent 1 concurrent 2 determine 16
and 25 curd 19 avid 10 cud 0 villa 0
innovation 16 uneaten 0 titillation 1 ululation 0 involution 2 direction 8 initiation 12 nutrition 25
in 25 ill 2 lit 4 nt 0 tit 2 tr 0
pitfall 3 sirloin 4 strain 25 tinfoil 0 billow 1 pillow 4 stroll 6
is 25 ti 2 s 25 it 25 g 25
????
the 25 tell 7 till 25 tre 0 tile 8 tic 0 tie 2 hie 0 one 25
lowest 4 fewest 0 foulest 1 tallest 2 frailest 1 flattest 1 latest 3 whilst 25 wiliest 0
in 25 ill 5 id 1 tit 4 lit 7
cruise 14 civilize 13 cuticle 16 cutest 2 entitle 25
it 25 of 25 re 25 tr 0 if 25 ti 3 or 25 oi 0

the 25 tell 10 till 25 tee 3 tile 13 tre 0 tilt 3 ire 0
truculent 0 inoculate 1 intervene 25 roulette 1 routine 11 uterine 0 unquote 0
though 25 trough 2 tough 1 lough 0 court 3 tonal 1
the 25 tre 0 tell 6 tee 3 tae 0 till 25 lie 3 tie 2 tile 8 tec 1
government 25
is 25 n 25 it 25 i 25 u 25 ti 4 tr 0 t 25 y 25
congeal 3 calyces 0 congers 0 coulters 0 lingers 7 lagers 0 targets 25 ringers 0 tinsels 1 forgers 6
????
fer 0 loll 0 for 25 fa 0 he 25 foil 0 roll 6 flu 0
tills 0 this 25 trill 2 flit 0 lilt 0 till 25 tells 4 dirt 1 tilts 1
titillation 0 adulation 0 situation 17 iteration 1 invitation 10 titration 0 irritation 25 initiation 19 sedation 2 structural 22
it 25 ti 9 if 25 e 25 u 25 r 25 f 25 n 25
is 25 n 25 it 25 u 25 i 25 ti 10 tr 0 t 25
also 25 hell 5 cell 3 crest 3 trill 2
the 25 tell 6 tre 0 tie 2 till 25 tee 1 tile 2 tec 1
consequence 24 consistence 25
of 25 if 25 or 25 bf 0 col 0 lei 0 let 6
brittle 25 sitteth 0 billet 6 pintle 0 pithier 2 stitch 12 sitter 0 brutish 15
censures 1 cranberries 0 translation 17 contraries 12 territories 7 constitutes 10 continues 25 controller 12 hailstones 1 utilisation 0
traditionally 25
rightfully 25
investiture 25
in 25 ill 6 it 25 lit 7 m 25 tit 5 nt 0 ti 14
true 25 tulle 0 thole 0 tune 20 hue 5 tittle 2 hoe 6 little 25 froze 12
cull 0 lull 1 cue 1 one 25 are 25 ore 1 owe 3 rue 2 vine 3
as 25 ai 17 ill 8 is 25 hi 1 lit 5 tit 7 its 25
a 25 u 25 tr 0 ti 19 n 25 oi 0 h 25 it 25
luxury 6 tilling 0 lilting 25 filling 10 taxing 7 tilting 17 elitists 0 filthily 2 fitting. 12
this 25 tills 0 trill 0 till 25 tells 6 toils 2 tall 2 tolls 1 fill 3
is 25 it 25 s 25 ti 13
cur 1 our 25 an 25 air 7 on 25 till 25 all 25 owl 2 lilt 8
attitude 25 altitude 11 certitude 0 noontide 0 derelict 2 architect 5 trounce 0 nutritive 0
that 25 trot 1 drat 1 tat 0 lint 0 licit 0 tot 0 teat 3 tnt 0 dint 0
must 25 aunt 2 crust 2 runt 0 mint 2 mutt 0 nest 3 wrest 1 orient 6 west 2
change 25 deluge 1 flange 3 range 22 clause 7 altruist 1 culture 7 derive 10 lecture. 6
ill 7 ti 9 lit 8 tit 7 nt 0 it 25
er 0 a 25 tr 0 ti 9 h 25 u 25 n 25 it 25
speed 25 spied 24 speech 10 speller 9 splicer 0 sliced 8 specie 0 splice 0
at 25 nt 0 lit 7 tit 7 tre 0 oil 10 lie 13
the 25 tre 0 tell 8 till 25 tee 1 tile 8 tie 2 lie 10 tilt 5
university 0 littorals 13 itinerary 25 irritants 14 intrusts 0 internes 0 tutelary 0
of 25 or 25 col 0 g 25 oil 12 d 25 cor 0 oi 0 er 0
latitude 25 ventricle 0
last 23 lout 0 lost 7 lint 0 tout 0 list 25 tint 2 tail 16 east 21
mode 18 mole 9 vole 0 node 8 virile 13 rode 12 niece 1 voile 0 incite 2 invite 25
peter 0 situ 0 betel 0 steel 25 beta 2 sleet 1 peer 17 seer 3 steer 14
hilly 25 vilely 4 frilly 2 inky 0 filled 22 ireful 1
called 12 culled 0 cabled 2 ended 25 coded 13 coiled 3 cubed 8 lulled 1 collect 5 inflect 2
loll 0 for 25 he 25 roll 9 fer 0 fa 0 fir 1 fit 12
tr 0 ti 5 oi 0 u 25 h 25 n 25 it 25
cultural 25 arterial 2 interval 9 inertial 0
involution 5 revolution 25 evolution 20 elocution 0 rendition 0 invention 15
ill 7 nt 0 lit 4 it 25 tit 0 u 25

leftists 0 tensity 0 relicts 0 utensils 25 utility 4 locusts 1 neuritis 0 tonsils 0 rentiers 4 tensely 8
science 9 sconce 0 scenic 1 salute 1 scarce 6 silence 10 sunlit 4 tactile 1 sterile 25
curd 2 cud 0 and 25 end 9 civil 5 aid 5 turd 0 hull 0 wid 0
finance 25 fiance 0 finalist 23 chalice 1 titanic 0 vitalist. 13
this 25 tills 0 till 25 blur 2 tells 5 fill 12 tolls 1 toils 2 lilt 1 tall 3
lie 7 he 25 tre 0 tie 6 re 25 lit 5 tic 0 hi 1
laid 16 said 0 livid 0 scud 0 send 11 solid 25 lard 2 lend 13 loud 8
was 25 wal 0 vies 0
needed 7 nailed 8 voided 25 railed 5 recited 3 heeded 0 raced 7 trebled 0 hailed 6 hulled 3
to 25 tc 0 ti 8 tb 0 it 25 t 25 k 25 b 25 c 25
irresolute 9 overtone 25
tr 0 oi 0 ti 7 a 25 it 25 r 25
seas 25 seal 9 bras 3 bias 8 leas 1 zeal 2 seat 14
cyclist 25 tideline 0
practice 25 preterite 0 prattle 2
shrills 19 shifts 23 shiny 25 shrill 19 shins 4 lifting 23 sculls 0 sluts 0 suing 4
rad 0 curd 2 nail 2 cud 0 and 25 rail 5 cad 4 unit 3 null 0 acid 3
industrial 25 inductive 6 incertitude 0 interstice 1 transitive 0 trinitrini 0
occupations 25
rid 5 wid 0 rich 7 wild 25 hillier 8 wilier 0
lieu 0 liar 0 lien 9 lion 7 hen 1 has 25
been 25 seen 2 blur 1 bear 6 heal 2 leal 0 zen 0 boar 10
in 25 ill 6 lit 6 tit 5 nt 0 ti 17
irritate 25
in 25 ill 8 ai 7 it 25 lit 8 tit 7 at 25 nt 0
valiant 1 bittern 3 variant 18 tinfoil 0 stroll 18 villain 25 strain 21
since 25 sine 0 sire 0 nice 8 gillie 0 grille 0 lintel 0 little 2
the 25 tre 0 tell 6 lie 4 tie 1 toll 0 lit 5 bloc 0
transitive 25 trinitrini 0
revolution. 25
gullet 0 stroll 6 gull 3 grill 3 still 25 snivel 0 street 0
tr 0 a 25 ti 5 oi 0 u 25 n 25 d 25 it 25
massive 25 reissue 11 whistle 19 thistle 5
smug 3 suing 6 swing 15 snug 3 siring 4 swig 2 slung 0 swills 1 stung 0 string 25
lit 2 in 25 ill 5 tit 1 nt 0 it 25 tr 0
attitude 25 altitude 10 clientele 3 littoral 1 retrieval 22 architect 15 outrival 0 whittle 4 alluvial 2
will 21 rill 0 wilt 2 cill 0 ill 25 cull 0 wid 0 riff 3
????
tr 0 ti 8 u 25 a 25 n 25 it 25
long 25 lag 11 lacy 0 lorry 3 hag 2 tag 6 lolly 0 lolls 2
tine 1 trill 3 tire 17 true 25 tune 14 tulle 0 tile 7 hull 0 title 9
to 25 tv 0 tr 0 ti 8 tb 0 t 25 it 25 b 25
materialist 11 naturalist 10 wateriest 4 machinist 22 wifeliest 21 luxuriance 1 medallist 2 winteriest 25
and 25 cud 0 avid 0 arid 5 null 1 aid 4 dull 3 unit 6 rid 6 lull 5
will 7 rill 0 wilt 2 lull 5 niff 0 lilt 0 till 25
require 25 filature 0 leonine 0 requite 0 lignite 0 recline 0 equine 0 latrine 8 figure 8
chance 25 dance 18 douce 0 clavicle 0 clause 13 chalice 1 divulge 0
from 25 flour 4 lull 5 flow 5 flew 2 friar 1 full 6 flout 1
educational 25
grass 25 glass 11 grills 9 truss 0 trills 0 stall 4 slats 4 stills 22 gluts 1 grail 1
roots 25 riots 5 rots 11 rote 3 foots 12 loots 2 roods 0 rods 8 rode 9
throughout 25 illustrate 4
erring 5 envy 11 crag 25 ovary 3 lung 9 wing 16
four 25 fern 7 fear 3 fen 0 feu 0 lea 0

of 25 ill 3 col 0 lei 0 let 7 lot 6 lit 5
business 12 silliness 12 justness 25 stillness 18 pitiless 13 sultriest 0 listless 1 shirtiest 2 nuttiest 0 lustiest 1
connubial 0 blindfold 25
curd 2 cud 0 and 25 turd 0 aid 2 rid 5 hull 0 tulle 0
satellite 25 shellfire 0 stricture 10 tincture 2 subvert 22
lutenist 0 littlest 25 intrust 0 interest 13 liveliest 15 littler 25 interior 4 infertile 0 infinite 7 litter. 7
investment 25
loll 1 roll 25 fa 1 fill 18 fin 21 rill 0
progress 13 prioress 0 plovers 0 strollers 4 priestess 8 brothels 2 blethers 0 brothers 25 frovilous 0 broilers 8
tulle 0 unit 11 little 25 alee 0 wile 2 trill 0
development 25 containment 19 deterrent 1
must 25 wriest 1 wrest 0 nest 4 wrist 1 rust 1 mist 5 rest 5 west 4 inert 3
serene 4 belittle 3 second 8 feature 25
cur 0 an 25 all 25 air 6 err 1 cill 0 lilt 0 till 25 hit 3
instinctive 6 instructive 8 distinctive 25 incertitude 0 distraint 0 disincline 1
locus 25 lochs 13
of 25 col 0 or 25 cor 0 oot 0 lot 7 if 25 tor 1 id 2
life 16 hie 0 ire 1 lie 10 fife 0 lift 14 tic 9 tie 25
for 25 loll 0 fer 0 fill 2 roll 13 rill 0 till 25 toll 1
pithier 0 illicit 0 ritual 9 rivulet 0 pitch 25 stitch 2
business 13 justness 25 listless 2 pugilist 0 lustiest 1 piglets 10 busiest 15 ringlets 0 rustiest 8 rustless. 8
the 25 tell 10 tre 0 tie 12 till 25 tile 2 tic 3 tee 1
????
will 18 rill 0 wilt 2 ill 25 wile 2 line 12 rift 4
have 25 lane 6 lave 0 hone 0 rune 0 rave 0 lone 12 love 7 haul 4 hove 0
u 25 oi 0 a 25 tr 0 n 25 ti 8 h 25 it 25
vital 11 little 25 lithe 5 litre 3 trill 4 title 5 trial 4 tittle 1
role 25 vole 0 rode 25 ole 0 rote 2 vote 8 ride 19 ode 1 loll 1
to 25 tr 0 tv 0 ti 14 tb 0 it 25 h 25 if 25 b 25
play 17 ploy 0 slay 2 stay 4 belly 4 peals 1 plus 25 billy 0 pills 5
if 25 j 25 ti 14 f 25 is 25 y 25 it 25
this 25 tills 0 till 25 trill 2 fill 2 tells 8 lilt 0 toils 2 tolls 2 bill 6
charge 8 change 25 chance 15 device 8 crevice 3 clavicle 0 chalice 0 deluge 2 grange 0
in 25 ill 5 lit 3 tit 2 m 25 hi 1 it 25 ti 19
attitude 25 altitude 5 clientele 1 lettuce 1 outwore 0 outride 0 intellect 3 intrude 7 overture 0 culture 7
is 25 n 25 it 25 u 25 ti 14
to 25 tc 0 ti 14 it 25 t 25 c 25 o 25
be 25 lie 7 ze 0 ye 3 vie 1 tie 3
affected 25 effected 25 collected 10 corrected 19 directed 24 articled 6 diverted 3 telltale 0 altered. 19
though 25 trough 4 tough 2 frivol 1 lough 0 thrush 1 trillion 1
an 25 cur 0 all 25 our 25 air 8 lit 8 on 25 in 25 hit 4
increase 25 unease 1 dialectic 2 reveille 0 intricate 6 indicate 22 inviolate 1 violence 24
ill 7 in 25 ai 3 it 25 lit 10 tit 2 u 25
public 16 fluvial 0 pubic 1 shrill 25 rubric 8 fulfil 18 thrill 13 filbert 0
spelding 0 spending 5 spurning 0 spinning 18 spurring 3 spirally 0 spiriting 25 sterility 11
on 25 cell 2 oil 4 tell 9 lilt 0 err 1 er 0 till 25 sill 0
l 25 r 25 j 25 i 25 t 25 f 25
and 25 cud 0 cull 0 tulle 0
d 25 oi 0 it 25 e 25 if 25 g 25 a 25 ti 9
is 25 y 25 it 25 ti 13 i 25 n 25
alleys 7 allays 7 relays 12 retells 6 heretic 23 reverts 23 rivals 25

useful 25 ireful 2 weft 0 lucre 0 nerve 10 riffle 0 trifle 8 litchi 0
alone 19 afoul 0 flour 17 allure 1 flare 4 atone 6 odour 25
it 25 ti 10 tr 0 r 25 tc 0 e 25 u 25
will 25 rill 0 cill 0 wilt 3 cull 1 cult 8 niff 0
not 25 wot 0 riot 2 rot 6 wet 6 ret 0 net 9 viol 0 lute 0
produce 25 product 24 flotilla 0 pretence 3 precinct 0 procure 6 succinct 4 pollute 8 violence 8
the 25 tre 0 tell 8 till 25 lie 9 tee 3 tile 13 tie 7 tic 0
required 17 fulfilling 14 included 25 intruded 5 requited 0 inquired 6 recruited 6 recurring 6 reclining 1 revivified 0
outcome 25 stricture 4 titillate 2 brilliant 16 tittivate. 0
the 25 tre 0 tell 14 tee 3 tec 7 till 25 tile 13 tae 0 tc 0
form 5 four 25 fowl 3 fours 0 lolls 1 lolly 0 firm 7 folly 1 rolls 6
wrist 3 rust 5 unlit 0 limit 21 lust 3 twilit 2 trust 11 twist 25 unite 12
be 25 lie 13 tic 0 tie 11 re 25 ze 0
on 25 er 0 cell 3 tell 19 oil 11
????
the 25 tre 0 tell 14 till 25 lie 11 tie 9 tile 8 tee 1
status 7 stars 25 tritely 1 slaty 3 neatly 13 lentils 2 lectern 0 trawls 3 tilths 1
of 25 ill 7 col 0 oil 10 lot 7 lit 9 bf 0 cor 0 oot 0
innovation 25 innovator 25 utilisation 0
robing 5 raking 4 wilfully 9 linking 20 rating 13 licking 4 lazing 1 mulling 1 walling 25 virtually 13
trills 0 tills 0 lilts 5 trill 0 tilts 25 rills 0 flits 0 flit 0 furl 0 title 25
????
its 25 ti 3 is 25 let 1 tit 0 lit 4 lei 0
infertile 25
and 25 cud 0 curd 1 allot 1 hull 0
importance 25 importune 0
loll 0 for 25 roll 3 fer 0 lie 4 fill 4 toll 1 rill 0 foil 1
gullets 0 circlets 0 cutlets 1 inlets 25 elicits 17
????
industry 25 intensely 25 loiters 1 tonsils 0 intensity 11 utensils 6 incivility 0 nullity 1 integers 16 integrity 6
curd 2 and 25 cud 0 arid 3 avid 0 cull 0 vivid 3 aid 3
silence 8 science 17 scenic 4 circuit 12 sunlit 25 sciatic 0 cornice 2 coolie 0
to 25 ti 2 tr 0 tb 0 t 25 if 25 it 25 b 25
drift 25
effectively 25
together 25 octette 0 resettle 1 torturer 13 rotate 2 belittle 3 foretell 1 retailer 6
and 25 curd 11 cruel 14 avid 0 unit 6 arid 4 rind 0 vivid 2 cruet 0 rivet 1
altering 6 nettling 2 clothing 5 clotting 16 alerting 5 uttering 8 attiring 0 clouting 1 cruelties 2 cutting 25
financial 16 circulate 25 ironical 9 elucidate 8 titillate 14 cranial 3 cultivate 25 fluctuate 19 intuitive 22 filtrate 8
pressures 25 philistines 0 proclivities 8 prostitutes 13 bullshits 3 intestines 3 belittles 11
tr 0 to 25 ti 10 tb 0 h 25 it 25 t 25
facilitate 25 vacillate 2 laureate 0 lacerate 2
till 25 tell 6 toil 3 toll 1 tilt 2 loll 0 ills 5 lilt 0 fit 5 bit 9
shift 4 swift 4 shirt 3 strife 1 flirt 0 elite 0 suite 10 little 25 twill 0
ill 6 in 25 lit 1 tit 2 it 25 ti 6 tr 0
emphasis 25 annalists 1 illiteracy 18 lunatics 10 triviality. 19

Appendix E

OCR Business Text

The following data represents the results of semantic analysis on an 807-word business document. This text was recognised using an OCR system developed by Hewlett-Packard that incorporated lexical analysis (i.e. word-lookup). The word candidates are therefore a product of the recogniser, whereas the scores are the product of the semantic analyser. This particular example shows the data before normalisation, so function words have scores of zero and there is no upper limit to the semantic score (compare with Appendix D). Each word position is shown on a separate line, with the correct word being shown in the first position. Note that the system treats punctuation as part of a word and not as a separate word, e.g. the candidate set: " a} 0 2} 0 a 0".

I 0
began 16
my 0
career 17
in 0
commodity 5
broking 5
as 0
a} 0 2} 0 a 0
secretary 14
with 0
a 0
brokerage 45
company 49 Company 49
then 0 men 31 the} 0 that 0 met 56 diet 18 me} 0 fit 26 Or 0 Of 0
became 16 Became 16 because 0 flame 11 frame 36
a 0
Personal 21 mortal 54
Assistant 38
to 0
four 0
brokers. 8 brokers, 8 brokers: 8 brokers; 8 brokers 8
I 0

am 0 arm 4 aim 8 airs 1 arts 47 mix 2 Hit 2 it 0 if 0 is 0
now 0
working 60
as 0 2 0
a 0
Commodity 7
Broker 6
in 0 its 0 it! 0 fit 2 it 0 us 0 if 0 is 0 Us 0
the 0
City 3
of 0
London 0
for 0
one 0
of 0
the 0
largest 23 large 23
American 0
Brokerage 5
houses 40 horses 3 homes 3 hot 4
having 0
been 0
offered 62 of 0
the 0
opportunity 57
to 0
study 13
for 0
the 0
Commodity 23
Brokers 12 brokers 12 Broker's 12 broker's 12 Broken 11 broken 11
Registration 158
exam 60 exam} 60 man} 6 man 6 mad} 7 Man 6 am 0 an 0 met 11 exam, 60
which 0
it 0 if 0
is 0 Is 0 8 0
obligatory 31
to 0
pass 26
according 47
to 0
American 0
law, 95
before 0 bore 30
one 0 fine 36 {me 0 rifle 5 (me 0 rule 85 me 0 on 0 tire 10 he 0 fire 18 be 0 tin 33 fin 23 {in 0 no 0 Go 9
may 40 my 0 in 0 In 0 fit 23
trade 31 made 32
or 0
accept 23 art 112 an 0 at 0 am 0 a 0
orders 77
from 0 four 0 fin 23 him 0 in} 0 it 0 if 0 a 0
clients. 12 clients 12
Basically, 65 Basically; 65
as 0

a 0
Commodity 67
Broker, 7 Broker; 7
my 0
duties 9 dines 53 dries 3 dies 4
are 0
to 0
service 39
existing 30 ending 41
clients, 32
a 0
task 14
consisting 12
of 0
giving 32
prices 13 juices 39
and 0
advice 61 add 45
and 0
taking 33 tiring 37 firing 8
orders, 61
Keep 108 keep 108
trading 76
and 0
commission 98
records, 114 records; 114
deal 196
with 0 writ 47 grim 6 unit 17
client's 183
problems, 95 problems; 95 problems 95
and 0
attempt 104
to 0
sign 14 sigh 13 sir 22 sit 18 sin 16
up 0 trip 23 tip 66 in 0 2 0
more 0
clients. 46 clients 46
Not 0 Aim 25 lot 19 Kin 2 {of 0 AIM 25 Lot 19 At 0 I} 0
only 0
am 0 art 13 an} 0 air 28 am} 0
I 0
paid 0
a 0
salary, 15 salary; 15 salary 15
but 0 put 3 bin 3 pin 8 rain 5 run 8 fun 7 tin 7 {am 0 fin 4 {in 0 him 0 ran 8 in 0
I 0
also 0 so 0
reap 78 map 25 Map 25 ten 0
the 0 {he 0
benefits 120
from 0 front 10 from} 0 hour 69 hut 9 Hut 9 us 0 at 0
commissions. 58
Work 94
usually 41
starts 31

at 0
around 0
10 0
am 0
each 0
day, 47
the 0
first 44 fire 13
job 9
of 0 fit 6 {if 0 {it 0 (if 0 fat 6 {of 0 {at 0 eat 45 (it 0 if 0 it 0
the 0
day 49
being 0
to 0
check 5
how 0
all 0
the 0
markets 7
closed 3
and 0
then 0
compare 56
prices 116
with 0 win! 2 win: 2 win; 2 win 2 With 0 will 28 {sum 23 grim 2
how 0
markets 201 marks 13
in 0 if} 0
London 0
are 0
trading, 33 trading. 33 trading; 33 trading 33 trailing, 2 trailing 2
as 0
this 0
is 0 3 0 5 0
a 0
fairly 0 fairy 23 Any 0 fair 28 my 0 far 1 fly 2 fail 3 Air 1 An 0 by 0 My 0 Am 0
good 56
indication 4
of 0
how 0
markets 26
in 0
New 27 few 0
York 0 aim 9 Yet 0 Yes 0 at 0 or 0 as 0 of 0
and 0
Chicago 0
will 74
behave 11 brave 4 have 0
later. 0 later 0
If 0 if 0
necessary, 29 necessary; 29 necessary 29
I 0
ring 48 leg 23 fire 32 me 0 if 0 in 0 it 0 an 0 at 0 a} 0
clients 68
to 0

tell 130 fell 37 felt 27 ten 0 ME 0 12 0 I 0
them 0 then} 0 then 0
what 0 win 37 fat 13 {at 0
is 0
happening. 54 happening 54
I 0
then 0 men 37 met} 21 tie} 16 me:} 0 me}} 0 he} 0 in 0 tin 191 fin 155
check 16
that 0 flat 39 tin 169 fin 153 Him 0 Am 0 in 0 An 0 AM 0
yesterday's 18
trading 42 trailing 18 tracing 23
has 0
gone 27
through 0 tough 28 rough 207
the 0
computer 88
correctly, 54 correctly; 54 Correctly, 54 correctly 54 Correctly 54 correct, 54 correct; 54 Correct, 54
and 0 aid 68 mad 30 are 0 arm 73 am} 0 aim 46 am! 0 Hid 139 an} 0 Had 0 an! 0 me 0 He 0 Met 19
Me} 0 Me! 0 at 0 a} 0 a! 0 a 0
write 72
up 0
the 0 me 0 die 85 lie 136 tie 117 He 0 Me 0 do 0 no 0 So 0 3 0
trading 81 trailing 89 darling 22 nailing 57
and 0 air 10 arm 44 art 17 aim 29 mad 24 mud 67 am 0 Hid 39 an 0 Mad 24 Mud 67 Had 0 at 0 a 0
commission 51 Commission 51
records 47 records. 47
My 0
next 0
task 7 {ask 34
is 0
to 0
go 31
through 0
the 0
overnight 36
comments. 12 comments 12
The 0 file 79 Tie 13 Me 0 He 0 he 0 I 0 a 0 if 0
various 93
analysts 58 analysis 35
in 0
New 49 New} 49 New! 49
York 0
write 81
daily 48 flatly 11 {lady 66 duty 44 flat 11
comments 68 Comments 68
after 0
the 0 tire 34 die 52 me 0 tie 112 Me 0 We 0 he 0 He 0 us 0
markets 19
close, 34 close; 34 close 34 dose, 3 dose; 3 dose 3 miss 0
which 0 {rich 12
are 0
received 15
here 0
by 0 fly 17 try 17 Try 17 by! 0 I 0
the 0 {he 0 {be 0 file 14 fine 79 tire 27 fire 18 me 0 tie 13 Me 0 He 0 he 0 Us 0 Be 0 SO 0 06 0 I 0

following 106 flowing 18
morning. 66 morning, 66 morning 66
Their 0 their 0 Then 0 Then: 0 Then; 0 Them 0 then 0
views 71
are 0 me 0 am 0 an 0 no 0 Me 0 He 0 a 0
not 0
necessarily 66
accurate, 35 accurate; 35
but 0
reading 13 teaming 13 reading} 13 tearing 4 rearing 7
the 0
relevant 118
information 146
certainly 35 certain 35
helps 132 taps 32 tops 48 has 0 lies 28 ties 24 he} 0 he! 0 he) 0 he] 0 be} 0 be! 0 lie} 28 lie! 28 be) 0 lie) 28 tie} 24 be] 0 to 0 fin 4 I 0
to 0
give 109 {give 109 {five 0 gave 126 am 0 {am 0 a 0 go 130
some 0 sorts 91 smile 77
insight 156
into 0 him 0 in 0 if 0
the 0 file 38 die 122 Me 0 tie 117 He 0 GO 108 SO 0
direction 92
of 0 lit 54 (it 0 (if 0 Of 0 fit 60 {it 0 {if 0
the 0 {he 0 {be 0 {lie 135 tire 107 die 196 me 0 tie 124 Me 0 us 0 He 0 Us 0 he 0 SO 0
markets, 65 markets; 65
and 0 am} 0 aid 22 are 0 arm 102 rid 93 aim 82 hid 49 mad 34 an} 0 bid 99 had 0 Hid 49 Mad 34 bad 106 mud 20 me 0 he 0 Me 0 be 0 He 0 at 0 a 0
possible 80
levels 88 feels 101 Men 84 ten 0 I 0
at 0
which 0 will 104 writ 47
to 0
take 51
a 0
long 108
or 0 of 0
short 90
position 25
(i.e. 0
to 0
buy 120 tiny 10 {buy 120 lady 8 fitly 28 bury 30 {my 0 [my 0 May 35 by 0 lit 10 fit 28 {it 0 lit. 10 bit 14 {if 0 bit. 14 fat 11 lit, 10 fit. 28 bit, 14 hit 12 {it. 0 hit. 12 {if. 0
or 0
sell 314 salt 65 sad 79 set 33 I 0
a 0
particular 400 {particular 400
commodity). 107 commodity}. 107 commodity) 107 commodity), 107 commodity} 107 commodity}, 107 commodity): 107
I 0
should 0
point 115 print 113
out 0 {out 0 (out 0 off 0 on 0 run 56 sit 94 art 101 fun 91 eat 76 fin 166 tin 199 {in 0 (in 0 aim 38 or 0 of 0 an 0 oh 71 am 0 a 0 if 0 our 0
that 0 Him 0

I 0
trade 139 {fade 48 made 60 fade 48 {rare 20 rare 20 hats 30 Hats 30 hard 43 hate 23 fare 78 bars 67
Hard 43 Hate 23 fins 178 tins 161 time 70 fine 101 me 0 he 0 He 0
in 0 if} 0 it) 0 if) 0 my 0 it} 0
Commodity 144
Futures 124 {futures 124
markets, 133 markets; 133
which 0 wine 98 wire 20 we 0 lie 75 tie 22 are 0 he 0
means 20
that 0 {hat 43 {bar 96 flat 37 {fin 14 tin 75 War 74 fin 14 him 0 Him 0 Man 46 if 0 it 0 {I 0
you 0 if 0 in 0 {a 0
can 0 ran 84 Can 0
either 0 her 0 Her 0 he 0 lie 121 be 0 tie 35 He 0
buy 50 Buy 50 {buy 50 bury 62 may 56 {my 0 cry 59 May 56 be 0 ill 42 ill. 42 in. 0 in 0 ill, 42 in, 0
fit 32 us 0 in: 0 Be 0 a 0 if 0 9; 0 5; 0 9! 0 5! 0 9 0 5 0 {5 0 {3 0 {9 0
a 0
contract 85 Contract 85
[for 0 (for 0 {for 0
delivery 83
in 0 if} 0 if: 0 if; 0 it} 0 is} 0 it: 0 is: 0 its 0 it; 0 is; 0 us 0
the 0 tire 27 me 0 die 14 tin 42 tin: 42 tin; 42 tin. 42 fin 7 On 0 0 0
future), 80 future}, 80 future). 80 future} 80 future}. 80 future); 80 future}; 80 future) 80 future] 80
hoping 396
to 0
sell 340 still 124 Mill 25 911 0 MY 0 611 0
it 0 if 0
at 0
a 0
higher 123 brief 23
price, 121 price 121 {nice, 5 [nice, 5 (nice, 5 pure, 15 juice, 4 {nice 5 [nice 5 (nice 5 {true, 4 pure 15
juice 4 {true 4 grid, 4 {me, 0 [me, 0 (me, 0 me, 0 Me, 0 me 0 Me 0 us 0 is, 0 Is 0
or 0 of 0
sell 165
a 0
contract 215
(to 0 {to 0 [to 0
receive 296 receive: 296 receive; 296 receive. 296
in 0
the 0 die 57 did 0 do 0 me 0 20 0
future), 179 future}, 179 future); 179 future} 179 future}; 179 future) 179 future] 179 slum), 3 slum},
3 slum} 3 dine} 5 fun} 11 fun) 11 fun] 11 sin 78 Win 67
hoping 349
to 0
buy 330 fitly 84 {my 0 {no 0 if 0 0 0 {5 0 {9 0 {3 0
it 0 if 0
back 0 track 179 {lack 103 tea 170 in 0
cheaper. 289 cheaper 289 cheaper, 289 cheapen 289
Thus 0 {bus 273
you 0
can 0 Can 0 car} 174 car: 174 car! 174 Car} 174 ear} 78 fail 114 far} 19 Car: 174 fair 59 ear: 78 Car!
174 ear! 78 me 0 mix 61 no 0 Me 0
make 185 male 22 me 0 die 30 one 0 file 38 rid 42
a 0 3 0 8 0
profit 229
or 0 (of 0 {of 0 (fit 95 {fit 95 [or 0 [of 0 (in 0 {in 0 On 0 on 0 Or 0 Of 0 Oh 163 oh 163 (or 0 {or 0

a 0
loss) 149 loss} 149 Kiss) 23 kiss) 23 Kiss} 23 kiss} 23 Miss) 0 Miss} 0 is) 0 is} 0 Is) 0 Is} 0 if) 0 if} 0 it) 0 If) 0 it} 0 If} 0 It) 0 It} 0 10 0 12 0 {2 0
either 0 her 0 nor 0 ear 85 car 20 bar 18 our 0 air 22 am 0 fin 8 on 0 an 0 in 0 a 0 0 0 6 0
way. 0 way 0 way, 0 way: 0 {my. 0 {my 0 {my, 0 {my: 0 May. 43 May 43 May, 43 May: 43 {20 0 we. 0 we 0 we, 0 we: 0 {2. 0 {2, 0 {2 0 {2: 0 0 0
At 0
midday, 11 midday; 11
I 0
usually 17 usual 17
go 60 {go 60 {do 0 {a} 0 2} 0 {a) 0 2) 0 25 0 6} 0 9} 0 4 0
to 0 My 0 in 0 {0 0
lunch, 84 lunch; 84 fund}, 17
but 0
am 0 arm 146 an} 0 aim 19 am} 0 a 0
back 0 {lack 34 link 70 fin 455 03 0 on 0 {in 0 us 0 08 0 or 0 of 0 in 0
by 0 fly 26 try 78
1.30 0 1.3! 0 13! 0 1.3} 0 13} 0 131 0 139 0 130 0 I 0 I. 0 I, 0 I: 0 I; 0
pm 26 but 0 pin 73 bin 2 put 68 {fin 306 {bin 2 [fin 306 (fin 306 {lift 40 fun 18 it 0 if 0 in 0 {a 0 a 0
as 0 us 0 is 0
the 0 {he 0 {be 0 {lie 39 tire 25 tin 17 fin 453 {in 0 do 0 20 0 0 0 I 0 4 0
Currency 44
markets 68
in 0 lit 25 it; 0 ill 46
Chicago 0
open 257
at 0 a! 0 8! 0 81 0
that 0 diet 154 met 119 film 119 War 105 Met 119 Him 0 Wet 233 him 0 net 104 it 0 if 0
to. 0 time 58 time, 58 line 140 fine 96 time: 58 tune 49 tune. 49 tune, 49 fund 34 {me 0 find 21 fins 313 tins 213 nine 0 mine 0 Mine 0 sins 71 kind 70 mud 68 mile 101 mile. 101 mile, 101 mud. 68 male 27 male. 27 mud, 68 male, 27 mild 60 mad 78 {in. 0 {in, 0 fin. 313 fin, 313 tin. 213 {in 0 tin, 213 {in: 0 fin 313 tin 213 fin: 313 {in; 0 sin. 71 kin. 7 tin: 213 sin, 71 fin; 313 kin, 7 tin; 213 sin 71
Quite 0 life 181 due 104 me 0 no 0 We 0 Me 0
often 0 off 0 lift 198 of! 0 run 135 fun 110 fin 1091 {in 0 (in 0 rid 81 on 0 oil 305 oh 138 [in 0
we 0 lit 173
have 0 Wave 184 Have 0 he 0 lie 134 be 0 fin 1088 {in 0 tie 93 We 0 no 0 He 0 [in 0 (in 0 tin 934 {an 0 me 0 Me 0 in 0 03 0
various 59
economic 51
figures 16
released 43
at 0
1.30 0 Mill 24 30 0 50 0 if 0 it 0
(all 0 {all 0 (ill 62 [all 0 {ill 62 (fill 36 {fill 36 (is 0 {is 0 [is 0 OR 0 Oh 7 a 0 0 0 I 0
pertaining 6
to 0
the 0 {he 0 me 0 die 74 did 0 Me 0 tin 52 no 0 0 0
United 1198 lifted 145
States), 300 States}, 300 States} 300 Staffs} 37 Staffs), 37 Staffs}, 37 State, 300 Staff, 37
which 0
can 0 Can 0 ran 38 {an 0 car} 105 {am 0 fail 64 rail 132 far} 64
affect 92 Wed 17 lift 126 red 28 led 57 rod 72 fed 62 it 0 if 0
these 0 dine 115 the 0 {he 0 die 147 did 0 {so 0 [so 0 We 0 we 0 I 0
markets 103 markets. 103

For 0
example 70
if 0 {lie 12 {tie 12 {no 0 die 109 the 0 me 0 Hut 12 no 0 We 0 {12 0 Me 0 it 0
the 0
Unemployment 16
figure 44
comes 39 crimes 46
out 0 of 0 in 0 or 0 fit 138
and 0 end 57 arm} 33 arm! 33 arm 33 aim 9 am} 0 am 0 an} 0 am! 0 an 0 an! 0 in} 0 in 0 it 0 if 0 {I
0 a 0
is 0 if 0
much 0 mud} 14
higher 11
than 0
we 0
expected, 3
it 0
weakens 24
the 0
US 0 US. 0 US, 0 US: 0 US; 0 Us 0 Us. 0 Us, 0 {15 0 Us: 0 Us; 0 U.S. 0
Dollar, 24
and 0
therefore, 0
the 0
other 0
currencies 1
rally 1 rally. 1
It 0 I! 0 If 0
is 0 15 0 {5 0
important 32
for 0
me 0 true 64 tie 4 us 0 he 0 He 0
to 0
know 45 Know 45
what 0
economic 78
data 15
is 0 3 0
due 29
and 0 aid 43 mad 9 Aid 43 mud 10 Mad 9 Mud 10 a 0
how 0
it 0 if 0
will 61 win 20 11 0
affect 114 effect 35 feet 16
the 0
markets. 36 markets, 36 markets: 36
At 0
2 0
pm 6
the 0
metal 3 met 5
markets 25
in 0 it} 0 if} 0 it 0 if 0
New 59 No 0
York 0

and 0
Treasury 55
Bonds 38 Bores 19 Boxes 31 Bends 6 Binds 6 Bands 16 Ends 30 Boil 27 Box} 31
in 0
Chicago 0
open. 90
Gold 112 Cold 86
and 0
Silver 99 Sir 35
are 0 me 0
the 0 me 0 tire 28 die 52 fire 50 file 15 due 18 use 27 tie 18 us 0 He 0 he 0
most 0 host 18 nod 77 rod 38 it 0 if 0
widely 50 wide 50
traded 145 trailed 20 trade} 145 tried 39
Commodities 72
Futures 88 {futures 88
contracts, 142 contracts 142 contacts, 52 contacts 52
and 0
at 0
this 0 {his 0 tins 10 fins 56 tills 0 tips 12 bus 21 tin 10 fin 56 {in 0 8 0
time 16
I 0
usually 88 usual 88
ring 42
several 0
clients 143 rents 44 cents 47
to 0
let 93
them 0 then} 0 then 0 merit 34 men} 59 men! 59
know 214 Know 214 {now 0 Bow 13 low 69 row 15 now 0 few 0 lent 7 line 9 {me 0
what 0
is 0
happening. 100 happening, 100 happening 100 happening: 100
If 0 if 0
a 0
client 75 merit 42 Hint 147
gives 69 {gives 69 eyes 22 giver 69 given 41 yes 0 {yes 0
an 0 art 40 air 22 At 0 us 0 As 0 am 0
order 115
to 0
buy 153 My 0 be 0
or 0
sell 165
gold 185
or 0
silver, 67 silver; 67
the 0
entry 5
of 0
the 0 {he 0
order 5
is 0 3 0
done 0 (lone 9 time 12 do 0 tie 13 he 0 me 0 be 0
over 0
the 0

telephone 32 telephone: 32 telephone. 32 telephone; 32 telephones 32
as 0
my 0
company 125
has 0 {us 0 fin 36 tin 23 {in 0 fun 24 in 0 In 0 I 0
direct 40
lines 25 fines 13 Lines 25 times 9 fires 12 tires 15 Dies 6 uses 15 Lies 43 {is 0 us 0 is 0 as 0 8 0 2 0 I 0
to 0 so 0
the 0 {he 0
trading 37 trailing 33 nailing 43
floor 112
in 0
New 9
York. 0 York 0
This 0
enables 8
us 0 {is 0
to 0
enter 12 utter 15
orders 18
quickly 93 quick}} 93 quick}! 93 quick)} 93 quick)! 93 quick]} 93 quick]! 93 quick} 93 quick! 93 quick 93
before 0
the 0 tire 207 me 0 die 37
markets 333 marks 16
move 373 mere 21 more 0 have 0 mine 0 nine 0 me 0
and 0
then 0 their 0 men 30 met} 12
receive 10 AM 0 8 0 2 0
executions 31
back 0
for 0
the 0 {he 0
clients 28
in 0 At 0
a 0
very 0
efficient 35
manner. 4
Speed 30
in 0
entering 4
and 0
the 0
execution 59
of, 0 of; 0
market 117 {market 117
orders 42 firm 203 ten 0 run 117 fun 18
(i.e. 0
no 0
price 156 {nice 38 {once 0 [nice 38 (nice 38 juice 35 {like 0 pure 16 fine 11 {him 0 fee 49 we 0 me 0 Me 0 He 0
limit) 128 limit} 128
is 0 3 0

extremely 123 extreme 123
important, 80
as 0
these 0 them 0 the 0 me 0 Me 0 We 0
markets 12 marks 6
are 0
very 0
volatile, 7
and 0 aid 9 Aid 9 Hid 6 a 0
a 0
large 27
profit 97 off 0 pit 20
or 0
loss 92 toss 4
can 0
be 0
made 27 me 0 fire 20 in 0
very 0
quickly. 7 quickly, 7 quickly: 7 quickly; 7 {quickly. 7 quick. 7 quick, 7 quick: 7 quick; 7
At 0
other 0 outer 37 offer 13 Grim 10 off 0 on 0 of} 0 of! 0 of 0 Or 0 Of 0
times, 19 times; 19
when, 0 when; 0
for 0
example, 36 example. 36 example; 36 sample, 15
a 0
client 12
gives 109
an 0 a 0 art 14 air 11
order 36
with 0
a 0
limit 29 {unit 54
price 60
on 0 of} 0 fin 1 off 0 {in 0 ran 3 ton 27 (in 0 ion 18 {an 0 {on 0 can 0 [in 0 oh 1 or} 0 oil 24 tin 4
it 0 if 0
(i.e. 0
he 0
is 0
not 0
prepared 8
to 0
pay 35 bay 3 tray 5 {ray 2 {lay 5 pray 51 pity 3 {my 0 if 0 {5 0 {3 0 {9 0
more, 0 more; 0 more 0 nine, 0 nine; 0 name, 24 dine, 10 mine, 0
or 0 of 0
receive 77
less) 0 less} 0
and 0 aid 21 at 0 a 0
that 0
price 8 pure 17 nice 7 is 0
is 0
not 0
close 4 dose 2 dine 22 else 0 die 4 do 0
to 0
where 0 when 0 are 0 ten 0 fee 24 am 0 an 0 on 0 a 0

the 0
market 220
is 0
currently 73 Currently 73
trading, 50 trading. 50 trading; 50 trading: 50
we 0
can 0 Can 0 car} 19 earn 92 ear} 15 Car} 19 cut 48 out 0 cry 13 dry 38
enter 19 eater 41 end 34 mud 43
orders 49
over 0
the 0 {he 0 [he 0 {be 0 {tie 73 [be 0 Me 0
wire 58
system. 26 system 26 stern. 11 stern 11
In 0 lit 20 If 0
either 0
case, 35 ease, 13 case; 35 case 35 ease; 13 ease 13 {as 0 do 0 Go 8 so 0 {a 0
it 0 if 0
is 0
vitally 44
important 139
to 0
complete 50
the 0
correct 11 Correct 11 toned 6 coped 7 cost 36 foot 6 ton 7 for 0 cow 5 God 0 Cow 5 set 34 Get 0 sit 88 sir 6 On 0 Or 0 Oh 2 OH 2 2 0 6 0
paperwork. 51 paperwork, 51 paperwork: 51
If 0 if 0
tickets 47
are 0 am 0 me 0 an 0 He 0 Me 0 a 0
not 0 he 0 rid 13 me 0 tie 111 it 0 if 0 It 0 If 0 I 0 hot 52
written 143 Written 143
up 0 lip 78 {2 0 [2 0 (2 0 tip 95
at 0
the 0 me 0 file 134 time 246 tie 136 He 0 no 0 he 0 do 0
time, 147 tune, 17 time; 147 nine, 0 tune; 17 nine; 0 dine, 7 mine, 0 fine, 37 turns 42 runs 13 ring, 15 find, 15 tins 10 ring; 15 find; 15
or 0
are 0 me 0 am 0 an 0 Me 0 He 0 a 0
written 143
incorrectly, 106
it 0 if 0
can 0 ran 22 car} 47 rail 27 car] 47 rain 25 win 21 man 18 Sin 138 men 18
lead 202 lead} 202 lead! 202 had 0 Mad 28 mad 28 Wed 16 Had 0 wed 16 Red 29 hut 21 let 74 a} 0 of 0 at 0 a! 0
to 0
order 137
errors. 185 errors 185 errors, 185 errors: 185 errors; 185
By 0 8} 0 By: 0 By; 0 By! 0 fly 189 {is 0 {8 0
3.30 0 330 0
pm 16 pin 28 bin 28 {am 0 {bin 28 firm 60 [am 0 on 0 fun 23 On 0 oh 102 off 0 oil 77 or} 0 of} 0 Oh 102 it 0 if 0
the 0 {he 0 {lie 30 die 22 me 0 did 0 tie 82 Me 0 no 0 He 0 he 0 us 0 a 0 0 0 I 0
other 0 outer 62 offer 30 omit 20 off 0 on 0 let 279 net 86 in 0
markets 180
are 0 at; 0 air 223 as: 0 at 0 am 0 as; 0 at! 0 as! 0 fin 187 far 99 ear 195 tin 336 in 0 an 0 a 0 at: 0

all 0 a} 0 At 0 I 0
open. 275 open, 275 open 275 open: 275 open; 275 opera. 81 opera, 81 opera 81 opera: 81 opera; 81 ten. 0 ten, 0 ten 0 ten: 0 ten; 0 let 260
Throughout 0
the 0 {he 0 [he 0 tire 103 {be 0 me 0 {tie 60 file 146 fine 69 die 191 fire 272 line 228 [be 0 life 72 tie 60 tin 298 lie 64 fin 172 {in 0 do 0 Me 0 We 0 He 0 Go 194 go 194
afternoon 249
and 0 aid 100 are 0 rid 84 arm 162 Hid 252 aim 58 me 0 mud 77 Me 0 He 0 a} 0 at 0 a! 0 a 0 if 0 {I 0 it 0
early 146 ear 84 car 117 Or 0 Of 0
evening, 209 evening 209 owning, 0
it 0 if 0
is 0
vital 34
to 0
keep 32
all 0 at! 0 an 0 a!! 0 a}! 0 fill 35 till 0 rid 14 aid 27 fin 3 ill 13 I 0 a 0
tire 40 file 44 the 0
clients 40 Clients 40
informed 104
of 0
fluctuations 107
in 0
prices 136 ion 20 on 0
of 0
the 0 {he 0 tire 53 file 18 fire 44 me 0 die 435 {tie 92 due 72 tin 13 fin 9
commodities 49
they 0 the} 0 the}: 0 the}! 0 the}; 0 {he} 0 she} 0 me} 0 me}: 0 me}! 0 me}; 0 die} 116 die}: 116 die}! 116 the 0 me! 0 {he 0 she 0 {be 0 did 0 me 0 die 116 Mad 31 Hid 68 Sad 39
have 0 Wave 50 he 0 lie 76 be 0 tie 121 We 0 Me 0 in 0
a 0
position 33 portion 26
in 0 if}, 0 if} 0 if}; 0 in, 0 in; 0
or 0
are 0 am 0 an 0 Me 0 He 0 a 0
looking 28
to 0 {0 0 {6 0
take 48 fair 7 {air 28 mile 9 {an 0 {am 0 {a 0
a 0
position 58
in. 0 in 0 in, 0 in: 0
During 0 Dining 13 firing 39 tiring 34
this 0 {his 0 fins 30 tins 9 fits 30 {fin 30 fin 30 win 12 if 0
time 42 tune 15 nine 0 mile 13
I 0
usually 107
receive 39
more 0 nine 0 from 0 me 0 in 0
printed 10 {hinted 12 {united 35 {mined 0
comments 39
on 0
the 0 tire 81 me 0 die 54 tin 22 fin 7
markets 99
which 0
help 70 help} 70 bed 73 ties 39 fed 28 lies 16 tied 16 fee 54 {fed 28 {led 27 led 27 tea 39

me 0 the 0 The 0 file 27 me: 0 me; 0 me. 0 me, 0 its 0 us 0 he 0 He 0 far 18 is: 0 to: 0 I 0 if 0 it 0
to 0
give 111
advice 154 nice 74 arts 48
where 0 Where 0
necessary. 32 necessary 32
I 0
continue 9
to 0
enter 6
orders 12
until 0
the 0 die 88 me 0 Are 0 We 0 Me 0 He 0 be 0 he 0 As 0
markets 33
start 18 star} 19 star! 19 stem 37 mad 29 had 0 man 53 Mad 29 men 53 Man 53 hut 17 51 0 5} 0 5! 0
a 0
closing. 76 closing 76 Closing. 76 Closing 76
The 0 Tire 61 life 100 {he 0 the 0 {be 0 Due 41 Tie 153 lie 178 We 0 we 0 I 0 If 0 if 0 6 0 I} 0
ryhthm 51 thin 130
of 0
my 0 in} 0 in}: 0 my: 0 in) 0 my! 0 my; 0 in] 0 {my 0 ray 45 no 0 may 56 {is 0 he 0 us 0 so 0 in 0 in:
0 me 0 do 0 in! 0 in; 0 if 0
afternoon 33 after: 0 after} 0 after; 0 am 0 art 46 an 0 a 0
depends 160 {lends 30 depart 45
on 0 On 0
the 0 me 0 tire 40 {tie 104 file 23 fire 13 {he 0 {me 0 die 28 {be 0 tie 104 its 0 us 0 He 0 we 0 We 0
{8 0 {2 0
relative 77 relative. 77 relative, 77 relative: 77
business 93
of 0 Of 0
the 0 {be 0 {he 0 its 0 tie 208 Me 0 He 0 We 0 So 0 Be 0 0 0
markets 185
which 0
in 0
turn 7 ruin 9 firm 60 film 50 hint 22 {am 0 him 0
depends 98
on 0 off 0 oil 41 of} 0 or} 0 or:} 0 oh 12 of 0 or 0 a} 0 an 0 at 0 we 0
various 165
factors 115 {actors 30
including 72
the 0 {he 0 {be 0 {lie 27 me 0 die 66
world 179
news 106 news, 106
economic 73
data, 69 data; 69 date, 56 date; 56 {a, 0 if, 0 in, 0 {I 0
world 164 wore 27 write 121 Mind 16 Wind 45 wine 39 Wire 22 Wife 25 Who 0 wed 15 we 0 we} 0
we] 0
leaders' 83 leaders 83
speeches, 88
or 0 lit 58 0; 0 fit 124 in 0 an 0 of 0 {if 0 {it 0 air 115
purely 51
on 0 Oil 40 off 0 Oh 9 tin 9 fin 7 {in 0 run 63 (in 0 till 0 in 0 On 0 oil 40 oh 9
rumours 147
circulating 158
concerning 0

adverse 75
weather 268
conditions 128
or 0 of 0 Or 0 Of 0
bad 43 {lad 14 load 46 {bad 43 Load 46
crops. 35 crops 35 crops, 35 Crops. 35 Crops 35 Crops, 35 tips. 12 tips 12 tips, 12 lips. 14 lips 14
To 0 Is 0 16 0 10 0 15 0 I 0
summarise, 0
the 0
job 15 {oh 18 gift 30 it 0 if 0
is 0 {5 0 15 0 {3 0 13 0 if 0
extremely 12
enjoyable 33
but 0
can 0 ran 7 Can 0 {an 0 rail 10 fail 10 fair 10 {am 0 mix 11
never 0 newer 20 nor 0 her 0 my 0 tin 12 My 0 if 0 it 0 in 0
be 0 of 0 Of 0 tie 32 Do 0 if 0 it 0
mastered 60 master 60
- 0
who 0 ill 21
can 0 {an 0 {in 0 van 7 tin 6 {am 0 fin 3 {fin 3 fun 17 tail 8 fail 22 him 0 man 52 arm 9 in 0 {my 0 mix 7 an 0 am 0 it 0 at 0 a 0
say 63
where 0
the 0
markets 35
are 0 air 6 aid 9 am 0 an 0 me 0 Me 0 no 0 in 0 a 0 3 0
going 43 from 0 film 10
next 0
week 112 work 15
or 0
next 0 nod 11 fit 32 {it 0 {if 0 6 0
month? 86
For 0 in 0 or 0
this 0 us 0 in 0 its 0 ill 41 if} 0 it} 0
reason 12
every 0 sir} 51 Get} 0 or} 0 Or} 0 or) 0 or] 0 Or) 0 Or] 0 it} 0 at} 0 oil 6 Oil 6 it) 0 it] 0 at) 0 do 0 Go 34 so 0 60 0 0 0 9 0 a 0
day 130 day! 130 in 0 it 0 if 0 {I 0
is 0
different. 23
One 0
day 196 {lay 33 tiny 55 {my 0 any 0 if 0 it 0 a} 0 at 0
can 0 Can 0 {an 0 cap 18 car} 32 fat} 18 far} 26 fail 11 {am 0 Cap 18 gap 22 Gap 22 on 0 or 0 oh 26 a 0 6 0
be 0 he 0 lie 86 {if 0 lip 76
very 0 if} 0
exciting, 34 exciting 34
the 0 me 0 Hid 84 Kid 36 me: 0 Me 0 no 0 0 0
next 0 text 6 red 30 of 0 at 0 ill 33 or 0 a! 0 01 0 0! 0 2 0
quite 0
the 0 {he 0 {lie 27 {be 0 tie 6 Me 0 He 0 he 0 Us 0 We 0 0 0
opposite. 4