1. Introduction

> *Human Language is like a cracked kettle on which we beat out tunes for bears to dance to, when all the time we are longing to move the stars to pity.*
>
> *Gustave Flaubert (1821−1880)*

Machine-readable dictionaries and thesauri are used as tools in many areas of linguistic and lexicographical research. The work done for this thesis was in two main areas: the mechanics of using a machine-readable dictionary and thesaurus on a computer, and the ways in which these tools can be used in sense disambiguation and intelligent text retrieval.

The Macquarie Dictionary[1] and The Macquarie Thesaurus[2] were supplied on a magnetic tape from the publishers. The tape containing the dictionary was a typesetting tape and contained a large amount of typesetting information which was either of no interest from a lexicographical point of view or had to be used to derive structural information about the dictionary. The type of tape received also meant that the process of converting the tape into a database was a non-trivial task. The thesaurus was received in a much more structured (for the purposes of this thesis) format. Each word was preceded by a code which identified its position in the thesaurus. The structure of the data made deciphering the information in the thesaurus straightforward so that the only decision that had to be made was the form of the database in which the data would be stored. Appendix B describes in detail the problems outlined above and how they were solved.

Sense disambiguation is the process of deciding in which dictionary sense a word in a particular context is being used. Lesk and other researchers have used machine-readable dictionaries in an attempt at automating this process. The ‘Literature review’ of this thesis examines some of the methods experimented with previously while the chapter entitled ‘Sense disambiguation’ presents implementation and further development of these ideas.

A thesaurus contains sets of words that are synonyms. A mapping from the words in the thesaurus to their corresponding dictionary sense is not supplied in any thesaurus. The chapter entitled ‘Thesaurus to dictionary mapping’ gives a method by which a computer can mechanically determine the dictionary sense in which a word is used in a thesaurus. The reverse procedure is also of interest; mapping a word sense in a dictionary to a list of synonyms in a thesaurus.

The algorithms introduced above can be used to improve the performance of retrieval by keyword in intelligent text retrieval systems. They make it possible to specify word senses, rather than just words, as keys and a piece of text can be retrieved if it contains a word that is the same as a key or is a synonym of it. The chapter ‘Text retrieval’ presents these ideas in detail.

Some of the work performed for this thesis has commercial applications for dictionary and thesaurus publishers who can use the algorithms to check the consistency of their publications and in the use of newswire retrieval where more sophisticated retrieval techniques are required to allow selective retrieval from the vast amount of newswire data produced each day. These applications are expanded upon in the chapter called ‘Applications’.

2. Literature review

> *The History of every major Galactic Civilization tends to pass through three distinct and recognizable phases, those of Survival, Inquiry and Sophistication, otherwise known as the How, Why and Where phases.*
>
> *For instance, the first phase is characterized by the question* How can we eat? *the second by the question* Why do we eat? *and the third by the question* Where shall we have lunch?
>
> *Douglas Adams*
> *The Hitch Hiker's Guide to the Galaxy*

2.1 Automatic sense disambiguation using machine-readable dictionaries

This paper by Michael Lesk[16] sets out a new method by which the sense of a word in a piece of text can be determined by machine. Overlaps between the words in the definition of the target word and the definitions of nearby words are counted and the sense of the target word with the greatest number of overlaps is guessed to be the correct sense.

Lesk discusses the example *pine cone*. Using the Oxford Advanced Learner's Dictionary of Current English there are two major senses for *pine*: 'kind of evergreen tree with needle-shaped leaves…' and 'waste away through sorrow or illness…' And *cone* has three separate definitions: 'solid body which narrows to a point…', 'something of this shape whether solid or hollow…', and 'fruit of certain evergreen trees…' For this example *evergreen* and *tree* are common to two of the sense definitions. So a program that counts word overlaps would find that the senses of the tree and its fruit are the likely senses for *pine* and *cone* when they appear together in text. This is a cheap solution to the discrimination problem; expensive methods are (hypothetical) expert systems or complete models of the world.

Another point discussed by Lesk was the selection of the machine-readable dictionary to be used to supply word definitions. The results of tests performed on the Webster's 7th Collegiate, the Collins English Dictionary, and the Oxford Advanced Learner's Dictionary of Current English were all comparable. The dominating characteristic is expected to be the length of entry, which is about the same for all of these dictionaries. Table 1 is taken verbatim from Lesk's paper except that it has been augmented with data concerning The Macquarie Dictionary.

Size of Dictionaries					
	OALDCE	**W7**	**CED**	**OED**	**Macquarie**
Bytes	6.6 MB	15.6 MB	21.3 MB	350.0 MB	18.7 MB
Headwords	21000	69000	85000	304000	77993
Senses	36000	140000	159000	587000	178269
Bytes/headword	290	226	251	1200	240

OALDCE	Oxford Advanced Learner's Dictionary of Current English
W7	Merriam-Webster 7th New Collegiate
CED	Collins English Dictionary
OED	Oxford English Dictionary (estimated)
Macquarie	The Macquarie Dictionary

TABLE 1. Dictionary sizes

Lesk reports that after some 'very brief' classification experiments accuracies of 50−70% were obtained with short samples of *Pride and Prejudice* and an Associated Press news story.

2.2 Why use words to label ideas: the uses of dictionaries and thesauri in information retrieval

This paper by Michael Lesk[14] discusses the design and use of a thesaurus that may be used in an information retrieval system. Such a thesaurus aims to combine words into a set of categories such that the words are synonymous, given the usage of the words in the literature. The important point made is that a lexicographer tries to *split* words: it is important to separate shades of meaning, while a retrieval thesaurus compiler tries to *lump* words: it is important not to have two names for the same thing (p 3).

Thesauri can be used in automatic indexing systems which allow the mapping of words to concepts. 'Salton and his associates did many studies of such thesauri some years ago, and found measurable and substantial although not dramatic improvement over simple keyword counting' (p 7). Salton did some experiments in which several test collections were used to evaluate automatic keyword retrieval systems. These tests used thesauri containing perhaps 500 to a few thousand entry terms, mapped into a few hundred categories. In general, about half of the words occurring in the documents were significant words for the purposes of retrieval, and were assigned to at least one category. Deciding which words were significant was a major part of the work in making the thesaurus. Salton's results were that performance was about 10% better using the thesaurus. An important point that Lesk makes is that the thesaurus had to be especially constructed for the field of study; a standard thesaurus could not be used. It is also apparent that building this special thesaurus was not a trivial task, and these thesauri do not resemble conventional thesauri. 'For example, if the word *kerosene* appears in a scientific abstract collection only as jet fuel, and thus appears only in documents about airplanes, it may be perfectly reasonable to say that it is a synonym of airplane' (p 8). Thus *kerosene* will be the same as *airplane* for retrieval purposes. Lesk notes that this type of system will not generalise beyond the exact documents with which it was built.

A discussion of the use of ‘frames’ in information retrieval states that these systems have only been used in experimental form to date and have not been applied to any significant subject area. Until a usage of frames covering a large subject area is produced frames are unlikely to be used in many retrieval systems.

Lesk concludes by stating the desired attributes of a thesaurus which is to be used in information retrieval and surmising that a dictionary and a thesaurus could be combined into a single hierarchical structure, but he admits he does not know how to do this.

2.3 What use are machine-readable dictionaries? A summary of the “Automating the lexicon” workshop

This paper by Lesk[12] is a report of the activities and papers presented at a workshop of creators and users of machine-readable dictionaries held in Italy in May 1986. Of interest in this workshop was a description of how Robert Amsler was finding lexical information automatically in news wires. Winfried Lenders reported that Amsler was finding definitions in appositive phrases and also making lists of phrases that repeat. He outlined ways by which one might accumulate dictionary information by fully mechanised procedures, and talked about how to represent the results. This was followed by a discussion of the availability of computer tapes and dictionaries, and the fears of publishers versus the desires of the researchers. The publishers wanted to ensure that if any money was made from the tapes they would get some. So far they haven’t seen much money, if any, coming from use of machine-readable dictionaries, nor any improvements to their dictionaries.

Probably the most important section of the workshop for the purposes of this thesis is where Lesk reports on his own paper about how to do sense disambiguation by counting overlaps of word definitions. Following that report it was claimed that a similar process had been used before by Margaret Masterman but using a thesaurus rather than a dictionary. However she counted overlaps of category numbers rather than words and had not implemented the procedure on a computer.

The point was made, as has been made many times before, that in order to build an automatic lexicon one should have the database of information from which a dictionary is generated rather than just a typesetting tape. The work done for this thesis would heartily support such a comment. It is a task of some difficulty, fraught with the possibility of error, to build a database containing all the information in a dictionary from a typesetting tape. The problem is greatly compounded when the information on the typesetting tape has been prepared by hand, allowing human inconsistencies to get in the way of automated extraction of information.

2.4 Typesetting from a dictionary database

This brief document by Robert Amsler[7] contains a short description of work in progress as part of a Bellcore ‘Interdictionary Project’ whose goal is to determine in what format a

dictionary should be entered into a computer. The ideal format allows changes to be made to the content, layout or topography of the dictionary without affecting any of the other components. Amsler states that this is believed to be an essential step if dictionary database publishing is to advance to the stage where it is suitable for a publisher's needs and at the same time facilitates the activities of computational linguists and information scientists.

2.5 Information in data: using the Oxford English Dictionary on a computer

This paper by Michael Lesk[13] is a report on a conference held at the University of Waterloo's Centre for the New OED, in November 1984, entitled 'Information in Data'. It is not of particular interest except for some comments made about the problems of converting a historical text such as the Oxford English Dictionary into a computer database. The comment was made that 'computer types kept thinking that they could rationalize some of the inconsistencies in the OED, and the lexicographers kept pointing out that these were not inconsistencies, but carefully thought-out choices (e.g. whether an etymological note appears before the sense definitions or between two of them)' (p 5). This debate led one Waterloo staffer to say that 'there are no inconsistencies, only rules we haven't yet discovered'.

2.6 The use of machine-readable dictionaries in sublanguage analysis

This paper by Donald Walker and Robert Amsler[20] discusses the theory behind, and design of, a program called FORCE4 which is a procedure for *full-text content assessment*. This program uses a machine-readable version of the Longman Dictionary of Contemporary English (LDOCE), which contains semantic codes not present in the printed version. The LDOCE is a medium-sized dictionary designed primarily to be used by people for whom English is not the native language. It contains one particularly useful feature; the inclusion of two sets of semantic codes. The second set, which is used in FORCE4, identifies word senses distinctive of particular subject fields. These subject codes are used in Walker and Amsler's research on content assessment. Words in the LDOCE are described by a two-letter subject code that marks a major area such as Medicine (*MD*) or Political Science (*PL*). Words can be described more precisely by the addition of a subfield category that indicates the division within a basic field code, the combination of two field codes, or the addition of a locality code. There are 212 subfield categories that indicate division of the basic field types; for example Physiology is described as *MDZP* and Diplomacy is *PLZD*, the 'Z' being used in the third position exclusively as an indicator of sub-categorization. An example of two combined field codes is that of the entry *lightning conductor*. The code for this word is *MLCO*, that is, the combination of Meteorology (*ML*) and Building (*CO*). There are 38 locality codes that identify major geographical areas and countries, and distinguish areas within them. Thus, 'U' represents Europe and 'F' represents France. Combined with Meteorology, the code *MLUF* is applied to *Mistral*, a wind that is characteristic of Southern France. This dictionary includes over 2600 combinations of two, three or four letter codes. Of the

55000 entries in the dictionary, 18000 are marked with a specialized subject sense with an average of 1.3 subject codes per word. Examples from page 76 are given in Table 2.

heavy	*FO*	food
	ML	meteorology
	TH	theater
rainfall	*ML*	meteorology
high	*SN*	sounds
	FO	food
	DGXX	drugs and drug experiences
	RLXX	religion
	ML	meteorology
	AU	motor vehicles
wind	*ML*	meteorology
	MDZP	physiology
	MU	music
	NA	nautical
	HFZH	hunting

TABLE 2. Subject codes

In the paper, FORCE4 is demonstrated on some news stories from the New York Times News Service (NYTNS) which have the important characteristic that they are generally about only one topic. The aim of FORCE4 is to classify the news stories into one of the subject classifications of the LDOCE. This is done by looking up each word of the news story in the LDOCE and keeping a running total of the number of words that fall under each subject code. Once the entire piece of text has been scanned the subject code with the greatest number of words is deemed to be the topic of the news story.

Walker and Amsler tested FORCE4 on a set of more than 100 NYTNS stories. They state that 'The results were remarkably good, considering that the system has not had any fine tuning or specialization with regard to the text at hand' (p 77). While the paper does not present any figures regarding what percentage FORCE4 correctly classified the authors state that it works well over a variety of subjects including law, defence, sports, radio and television.

A number of conditions were identified as important for FORCE4 to work well. The content-bearing words of the text must have entries in the dictionary being used and there must be a subject code for the sense of the word as it is being used in the text. A common function word such as *in* must not also be a content-bearing word with a subject code which may be incorrectly included in many pieces of text. It is also important that a sufficient quantity of text is examined for the topmost subject code to stabilize. According to Walker and Amsler this is typically more that a sentence, but less than a paragraph, about a single topic. For this reason news wire stories are perfect examples.

The important point to be drawn from this paper is that a dictionary, or for that matter a thesaurus, containing meaningful subject categories can be very useful in classifying pieces of text.

2.7 Machine-readable dictionaries

This chapter of the *Annual Review of Information Science and Technology* written by Robert Amsler[5] is, as the title suggests, a summary of the work that has been performed using machine-readable dictionaries. Amsler raises some points that are of interest here. Content analysis by Walker and Amsler [20], as presented previously, is discussed as well as work on thesaurus construction. He mentions that Amsler and White describe a technique for building a thesaurus-like structure from the definitions in an ordinary machine-readable dictionary. The technique described makes it possible to check that the definitions of words have the same meaning as used in the thesaurus. The authors also offer the possibility of inverting the process to build a dictionary from a thesaurus (p 171).

In addition Amsler discusses the methods of dictionary database design. Fredericksen has outlined a database structure for dictionary definitions that permits the user convenient access to all words that share the same sense, all senses of a given word and all phrases in which a given word occurs. Also Amsler gives references to works that describe the compact storage of words and near-optimal hash functions in the context of natural language work.

A short history of the current methods of performing word disambiguation follows. Several computational mechanisms for performing word disambiguation have been proposed. In one mechanism experiences are represented as frames and scripts and are cleverly probed to produce the correct interpretation of a word in context. This strategy has a major deficiency in that there have only been a few examples constructed to illustrate the techniques involved and there is no source of the required experiences in a form suitable for entry into a computer. The other mechanisms presented have a similar flavour and rely on some fairly deep understanding of the text being examined. The strategies contrast markedly to those proposed by Lesk[16] which require only a superficial understanding of the text being processed.

Some work has been done in the area of the automatic generation of dictionary entries. In 1972 Grove wrote that ‘no one says that a machine may someday be able to define’. Little progress has been made in efforts to come up with a working program to automatically generate dictionary entries. Granger has shown how a program that is analysing text can attempt to determine the meaning of a word from its context. Keirsey produces evidence that an ISA hierarchy can be used to learn new words from their textual context. However it appears that none of these ideas have been translated into a finished working program.

2.8 Deriving lexical knowledge base entries from existing machine-readable information sources

In this paper Amsler[6] asserts that machine-readable text can be exploited far more than has been done to date in the derivation of lexical information for natural language

processing. Today's word-processing and printing equipment is almost completely computer-based. He suggests that the enormous quantities of machine-readable text being produced in this way should not be overlooked by lexicographers. The main problem that he sees in using this information is the method in which the text is handled. Most machine-readable text is produced for a specific purpose and is therefore formatted in such a way that it is useful for that purpose only. This makes it extremely difficult for the information to be used for other purposes. He quotes as an example the *World Almanac and Book of Facts* which is a ten million character book published in the United States. It contains a great deal of information that could usually be handled in a computer database, yet the machine-readable text of this book is available only as a by-product of its computer formatting. The task of converting this typesetting information into a database format is very nearly a prohibitively expensive effort. Amsler makes the point that the entire body of information has already been typed into a computer, yet because its format is so different from that which could be used by a database program, the information cannot be used in an information retrieval system. The problem is that the designers of typesetting languages and database systems have not been able to come up with a system that allows both their aims to be satisfied. The required solution is a combination of a typesetting language and a database program.

The second section of Amsler's paper deals with *NewsWire Lexicography* (p 2). Bell Communications Research has a link to a newswire service that allows it to collect a very large quantity of text each day. The first use of this text is to build citations for the lexical knowledge base. These citations can be used in the construction of dictionaries for use in computer programs attempting to deal with everyday language.

The newswire stories frequently contain proper nouns in an appositive syntactic structure, for example 'Dr. Apple, an optometrist from Boca Raton, Florida, was among the visitors at this week's seminar series in Rome.' These types of sentences allow proper nouns, 'Dr Apple' in this example, to be defined in-line. Current natural language technology is sufficiently advanced to allow automated extraction of these constructs and to list them as possible new lexical information. These lists can then be perused by lexicographers for possible dictionary inclusion.

Large bodies of text, such as a collection of newswire stories, have often been used to make word lists and frequency counts. A more difficult problem is to identify lexical units rather than blank separated words. It is much more interesting to know that the phrase 'cross a cheque' occurs a number of times than it is to have the separate counts for 'cross', 'a' and 'cheque'. However isolating a multi-word lexeme is a non-trivial problem. Amsler suggests the following simple heuristic: a lexical unit is any sequence of non-function words. The results reported suggest that this heuristic method is fallible

3. Thesaurus to dictionary sense mapping

POLONIUS: What do you read, my lord?
HAMLET: Words, words, words.
William Shakespeare (1564−1616)
Hamlet (Act II, scene ii)

This chapter is a discussion of a method of mapping entries in The Macquarie Thesaurus to the correct sense in The Macquarie Dictionary. This algorithm, once fully developed, can be used to identify the dictionary sense of each word in the thesaurus. The information obtained using the method described here can be used to implement a thesaurus browser which, when asked for the definition of a word, can supply the definition corresponding to the sense in which the word is being used.

The ideas presented here are an application and development of the work of Lesk[16] who first put forward this method of sense disambiguation.

3.1 The purpose of the algorithm

The aim of the algorithm being described here is to map an occurrence of a word in the thesaurus to its correct sense in the dictionary. An example of what is meant is perhaps appropriate here. A page of the thesaurus may contain the following fragment of text:

electricity
n. **electricity**, juice, power, supply; **faradism**, galvanism, hydro-electricity, magneto-electricity, piezoelectricity, pyroelectricity, static electricity, thermoelectricity, triboelectricity, voltaism; **electric current**, Foucault current, alternating current, amperage, charge, current density, direct current, eddy current, grid current, impulse, ripple current, thermionic current; **spark**, arc, carbon arc; **voltage**, electric potential, electromotive force, grid bias, potential, potential difference, tension; **capacitance**, absolute permittivity, admittance, capacity, commutation, conductance, dielectric strength, elastance, electric field strength, flux, impedance, inductance, induction, load, power factor, reactance, relative permittivity, resistance, resistivity, superconductivity, susceptance, susceptibility, wattage; **static**, atmospherics, interference; **discharge**, brush discharge, corona, disruptive discharge, drain, flashover, gas discharge, glow discharge, leakage current, shot noise.

For this example the word of interest is *juice*. The definition of *juice* is:

juice *v.*, **juiced**, **juicing**.
–*n.* **1.** the liquid part of plant or animal substance.
2. any natural fluid secreted by an animal body.
3. any extracted liquid, esp. from a fruit.
4. essence; strength.
5. *Colloq.* **a.** electric power.
b. petrol, fuel oil, etc., used to run an engine.
6. *Colloq.* any alcoholic beverage.
–*v.t.* **7.** to extract juice from (fruit or vegetables).
–**juiceless**, *adj.*

For the purposes of this program senses are numbered from 1 to *n* without any alphabetic suffixes. So the above definition has senses 1 to 8. The correct sense is the one that gives the meaning of the word as it is used in the relevant section of the thesaurus. Here *juice* is being used as another word for electricity, and so, given this numbering scheme, it can be seen that the correct definition is number 5, 'electric power'. The purpose of this is to identify sense 5 as the one which corresponds to this use of the word *juice* in the thesaurus.

3.2 Testing

The first two algorithms discussed in this section were tested using a data set prepared with the assistance of Joanne Lynton. This data set consisted of 39 words and their thesaurus classifications with the corresponding dictionary senses. A list of the contents of the data set is given in Table 3:

keyword	part of speech	superquark	quark	word	correct sense
electricity	n	electricity	electricity	juice	5
discourtesy	n	discourteous person	discourteous person	boor	1
pain	n	ache	ache	qualm	3
emblem	n	flag	flag	standard	11
fauna	n	sheep	goat	kid	1
deception	v	deceive	deceive	trick	15
obviousness	adj	obvious	obvious	obvious	1
fertility	v	be fertile	produce	crop	23
ill health	n	cold	cold	cold	26
sign	v	gesture	gesture	wave	26
pleasantness	adj	pleasant	sweet	sweet	4
foreignness	adj	foreign	foreign	alien	7
fastening	v	fasten	bind	tie	1
essay	n	essay	introduction	introduction	4
strangeness	adj	strange	eccentric	funny	2
extraction	v	extract	extract	draw	5
closeness	adj	close	adjacent	contiguous	2
effort	adj	effortful	effortful	hard	6
fine arts	v	depict	pencil	charcoal	4
generality	adj	general	prevalent	widespread	2
fauna	n	PREHISTORIC ANIMALS	PREHISTORIC ANIMALS	mammoth	2
jealousy	v	be jealous of	covet	covet	1
ascent	n	lift	pulley	pulley	1
killing	v	kill	asphyxiate	smother	1
representation	n	portrait	caricature	caricature	1
excess	adv	excessively	too much	ad nauseam	1
impropriety	adj	improper	base	low	25
goodness	adv	well	well	excellently	1
intangibility	n	soul	phantom	apparition	1
food	n	drink	soft drink	cordial	5
dwelling	n	cabin	dump	dump	21
closure	n	plug	plug	cork	3
hardness	v	harden	harden	set	48
buying	v	buy	shop	shop	8
showiness	adj	showy	ostentatious	flamboyant	2
time measurement	v	time	time	clock	9
book	n	newspaper	comic	comic	5
entertainment	n	film	film	film	6
payment	n	income	advance	advance	20

TABLE 3. Short test file

The other algorithms were tested using a data set prepared with the assistance of Macquarie Library Pty Ltd. This data set identifies a word in the thesaurus and its corresponding dictionary sense and was produced by selecting a random sample of words from The Macquarie Thesaurus and asking the lexicographers to indicate which dictionary sense corresponds to that usage of the word.

The sample was prepared by randomly selecting 5000 thesaurus entries from those entries whose corresponding dictionary definition has more than one sense. This selection was done using Algorithm R from Knuth [11 §3.4.2] as implemented by Frank O'Carroll. This sample when shuffled using Algorithm P can be used as a source of examples for testing.

The first 300 records from the set were prepared for classification. Each entry has the words surrounding it in the thesaurus printed along with the dictionary definitions of the word. Each of the senses in the definition is numbered and the lexicographer is asked to indicate the correct sense with a tick. This is illustrated below:

Victorian

abstinent, abstemious, abstentious, puritanical, self-denying, steady, temperate, Victorian, wowserish; **ascetic**, ascetical, austere; **celibate**, chaste, clean, continent, honest, moral, pure; **virgin**, intact, vestal, virginal; **frugal**, spare, sparing, Spartan; **teetotal**, dry, off the grog, on the square, on the wagon, on the water wagon, sober, sober as a judge, stone-cold sober.

1 *adj.* **1**. of or pertaining to Queen Victoria (1819–1901, Queen of Great Britain and Ireland, 1837–1901) or her reign or period: *the Victorian age.*

3 2 **2**. *having the characteristics usu. attributed to the Victorians, as prudishness.*

3 **3**. *of or pertaining to the State of Victoria.*

4 *–n.* **4**. *a person living in the Victorian period.*

5 **5**. *a person having the characteristics usu. attributed to the Victorians; a prude.*

6 **6**. *one who was born in Victoria or for whom it has come to be the home State.*

Once all the examples have been classified the correct sense is entered into the example file in Table 4:

Word	Key number	Part of speech number	Paragraph number	Quark number	Dictionary sense
allegation	420	0	0	0	2
anent	525	5	0	0	2
angle	74	3	0	1	11
anticipation	33	0	0	0	2
archaic	753	2	0	2	1
area	705	0	1	0	1
arousal	668	0	0	3	3

TABLE 4. Sample of Macquarie test data

The fields in Table 4 describe the location of a word in the thesaurus in a numeric format. This table provides the same sort of information as in Table 3.

Macquarie Library had a few problems determining the correct sense of some of the supplied words.

- The lexicographer reported that sometimes the word in the thesaurus paragraph appears only as a run-on in the dictionary definition. This is a problem because run-ons have no applicable sense in the definition. An example of this occurs with the word *arousal*:

arousal

sex, female, gender, male, opposite sex; **bisexualism** , androgyny, hermaphroditism, virilism; **sexlessness** , asexuality; **lust**, arousal, concupiscence, excitement, heat, horniness, hotpants, libido, oestrus, passion, pride (*Obs.*).

1 –*v.t.* **1.** to excite into action; stir or put in motion; call into being: *aroused to action, arousing suspicion.*
2 **2.** to wake from sleep.
3 **3.** to awaken sexual excitement and readiness in.
4 –*v.i.* **4.** to become aroused.

Here the lexicographer stated that the noun definitions are missing as *arousal* is a run-on from *arouse* and that the thesaurus paragraph relates to definition 3.

There are two ways to resolve this dilemma. One method is to exclude run-ons from the example file. However this is unrealistic as when classifying real text run-ons will occur and must be handled in some manner. The second solution is to enter as the classification for that example the one suggested by the lexicographer as being closest to the (non-existent) correct sense. This means that in the case of run-ons it is possible that the wrong sense (the part of speech will be incorrect) will be indicated but it will at least contain the correct meaning of the word. This approach was chosen as it more closely represents the desired behaviour of the algorithm.

- In some cases two or more definitions apply to a usage of a word in the thesaurus. This is caused by the fine sense differentiation in the dictionary compared to the thesaurus. An example is:

bland

pleasant, acceptable, agreeable, bland, compatible, enjoyable, inoffensive, nice, offenceless, palatable, piacevole, pleasing, sapid, simpatico, to one's taste, welcome; **amiable**, adorable, benign, courteous, genial, good-natured, good-tempered, kindly, likeable, lovable, sweet-tempered; **charming** , attractive, beautiful, becoming, comely, cute, easy on the eyes, engaging, glam, glamorous, graceful, piquant, pretty, taking, winning, winsome; **cheerful**, cosy, jolly, merry (*Archaic*); **delightful**, delectable, delicious, delightsome (*Archaic*), fragrant, gladsome, glorious, goluptious, gorgeous, heavenly, lovely, luscious; **sweet**, dulcet, mellifluous; **bittersweet**, piquant.

1 *adj.* **1.** (of a person's manner) suave; deliberately agreeable or pleasant but often without real feeling.
2 **2.** soothing or balmy, as air.
3 **3.** mild, as food or medicines: *a bland diet.*
4 **4.** non-stimulating, as medicines.

The thesaurus section for *bland* can be taken to mean either sense 1 or sense 2. The ideal solution in this case would take these multiple senses into account during

testing. However this was not done due to time constraints and as the problem only occurred a relatively small number of times. The solution used was to choose, sometimes arbitrarily, the most applicable sense.

- In a few cases the dictionary did not contain the correct sense of the word as it was used in the thesaurus. For example:

> **voluntary**
>
> **unpaid**, due, not yet payable, outstanding, undischarged; **deferring payment**, moratory; **complimentary**, free, gratis, tax-deductible, tax-free, untaxed; **honorary**, uncustomed, unpaid, unremunerated, unrewarded, voluntary.
>
> 1 –*adj.* **1.** done, made, brought about, undertaken, etc., of one's own accord or by free choice: *a voluntary contribution.*
> 2 **2.** acting of one's own will or choice: *a voluntary substitute.*
> 3 **3.** pertaining to or depending on voluntary action or contribution.
> 4 **4.** *Law.* a. acting or done without compulsion or obligation.
> 5 **b.** done by intention, and not by accident: *voluntary manslaughter.*
> 6 **c.** made without valuable consideration: *a voluntary conveyance or settlement.*
> 7 **5.** *Physiol.* subject to or controlled by the will: *voluntary muscles.*
> 8 **6.** having the power of willing or choosing: *a voluntary agent.*
> 9 **7.** proceeding from a natural impulse; spontaneous: *voluntary faith.*
> 10 –*n.* **8.** something done voluntarily.
> 11 **9.** a piece of music, frequently spontaneous and improvised, performed as a prelude to a larger work, esp. a piece of organ music performed before, during, or after an office of the church.

  In the thesaurus *voluntary* is being used in the sense of 'unpaid'. Definition 6 is the closest but is not quite right. These cases can be regarded as errors in the dictionary (this is confirmed by the publishers of the dictionary).

- The opposite type of error occurred when the thesaurus used a word in an incorrect sense. For example:

monkey

sewing machine, charka, knitting machine, napper, overlocker, scutch; **sewing aid**, bobbin, bodkin, buttonholer, card, clew, comb, crochet hook, darner, darning needle, distaff, dobbie, dobby, filature, godet, hemming foot, knitting needle, knitting wire, monkey, needle, pick, reed, ripple, rippler, sacking needle, spindle, tacking needle, temple, thimble, three-cornered needle, upholsterer's needle; **spinning wheel**, spinning jenny, throstle; **loom**, Jacquard loom, power loom, scribbler; **flax mill**, filature, willower.

1 –*n.* **1.** any member of the mammalian order Primates, including
the guenons, macaques, langurs, capuchins, etc., but excluding
man, the anthropoid apes, and usu., the lemurs.
2 **2.** a person likened to such an animal, as a mischievous child, a
mimic, etc.
3 **3.** the fur of certain species of long-haired monkeys.
4 **4.** *Colloq.* a sheep.
5 **5.** any of various mechanical devices, as the ram of a pile-
driving apparatus, or of a wool press.
6 **6.** *U.S. Colloq.* an addiction to narcotic drugs, seen as a burden or
affliction: *have a monkey on one's back.*
7 **7.** *Colloq.* **a.** (formerly) the sum of 500.
8 **b.** the sum of $500.
9 **8.** *N.Z. Colloq.* → **mortgage.**
10 **9. a monkey on one's back,** any obsession, a compulsion, or
addiction, seen as a burden, as a compulsion to work or an
addition to drugs.
11 **10. get one's monkey up,** *Colloq.* to become angry or enraged.
12 **11. make a monkey of,** to make a fool of.
13 **12. monkey business,** trickery; underhand dealing.
14 **13. monkey tricks,** mischief.
15 –*v.i.* **14.** *Colloq.* to play or trifle idly; fool (oft. fol. by *about with* or
with).
16 –*v.t.* **15.** to imitate as a monkey does; ape; mimic.
17 **16.** to mock.
18 *n.* a looped strap which an inexperienced buckjumper grips with
his right hand.

In this case *monkey* should not be in this list as it does not mean *sewing aid*. These cases are due to errors in the thesaurus.

- Another error in the thesaurus was that some lists contain a mixture of parts of speech. For example:

just

smallest, least, minim, minimum, slightest; **inconsiderable**, fractional, inappreciable, infinitesimal, insignificant, insubstantial, light-weight, marginal, minimal, minor, negligible, vestigial; **measly**, halfpenny, mere, paltry, pelting (*Archaic*), petit, petty, potty, trifling; **mean**, niggardly, stingy; **slight**, certain; **cursory**, superficial, tenuous; **just**, bare, mere, no more than, only.

1 *adj.* **1.** actuated by truth, justice, and lack of bias: *to be just in one's dealings.*
2 **2.** in accordance with true principles; equitable; even-handed: *a just award.*
3 **3.** based on right; rightful; lawful: *a just claim.*
4 **4.** agreeable to truth or fact; true; correct: *a just statement.*
5 **5.** given or awarded rightly, or deserved, as a sentence, punishment, reward, etc.
6 **6.** in accordance with standards, or requirements; proper, or right: *just proportions.*
7 **7.** (esp. in biblical use) righteous.
8 **8.** actual, real, or genuine.
9 –*adv.* **9.** within a brief preceding time, or but a moment before: *they have just gone.*
10 **10.** exactly or precisely: *that is just the point.*
11 **11.** by a narrow margin; barely: *it just missed the mark.*
12 **12.** only or merely: *he is just an ordinary man.*
13 **13.** *Colloq.* actually; truly; positively: *the weather is just glorious.*
14 *n., v.i.* → **joust.**

The correct definition in this case is number 12, an adverb, even though the paragraph is meant to contain adjectives.

- A problem was caused by a loss of information in the dictionary data base. Secondary headwords are stored as keys to the entire definition of the headword under which they fall. If secondary headwords were to be keys only to the part of the main definition that was applicable then searches such as `at the same time` would lead directly to the correct definition. This will be corrected in a future version of the dictionary database. It should be noted however that this will not affect applications such as sense disambiguation because recognising that a sequence of words is a unit and should be treated as such is a decidedly non-trivial problem.

- A more subtle problem is described by Bill Smith from Macquarie Library in the following comment:

> Commonly, a whole sub-category is ambiguous as not only do the words have multiple meaning, but they have the SAME multiple meanings. Especially so in the case of transitive/intransitive.

This problem relates to the case when more than one sense is correct but is slightly

different as the alternative senses have different parts of speech.

Testing of the algorithms described in this section was done by running the algorithm on each of the examples and counting the number of correctly classified words. Some algorithms occasionally return more than one sense as the correct sense. If this list of senses includes the correct sense it is listed as being ‘correct and wrong’. The term ‘close’ is taken to mean either ‘correct’ or ‘correct and wrong’.

The percentage that someone guessing randomly would expect to get correct is useful to compare against the results of the various algorithms. This number is calculated using the following formula:

$$\text{0sfont]I]ize]u'}\frac{1}{n}\sum_{i=1}^{n}\frac{1}{d_i}\text{*100\%}$$

In this equation n is the number of examples and d_i is the number of senses in the definition of word i.

3.3 The algorithm

When an intelligent being (a human) decides which dictionary sense corresponds to a word in the thesaurus it uses its world model to make the best choice. It is not possible with the present technology to build a sufficiently comprehensive world model into a computer, so another simpler and thus less accurate method of sense mapping needs to be used. The methods presented here all use the concept of counting overlaps between the dictionary definitions of words to make the best choice. Each method listed below uses a different means to measure the overlaps between definitions.

3.4 Simplest method

Here the correct sense is chosen by assuming that it is the sense of the word *juice* which has the greatest overlap with the definitions of the synonyms of *juice*. To determine the the number of overlaps the definitions of the words *electricity*, *power*, and *supply* are looked up in the dictionary and combined (the definitions of these words are given in Appendix A). This gives a list of words which are sorted and stripped of all function words and then compared against each of the senses of the word *juice*. A word that appears n times in the definition of a sense and in l definitions of the other words it is being matched against contributes l overlaps to the total. (An earlier implementation, which did not perform as well as this method, had this contribution fixed at one match.)

A function word adds meaning to or shows the relationships between content words in a sentence. Two examples are *the* and *of*. The full list of function words is given in Table 5:

Colloq	by	ones
I	esp	or
a	etc	out
adj	for	so
adv	from	something
all	have	that
an	he	the
and	in	to
any	into	up
are	is	used
as	not	vt
at	of	which
be	on	with

TABLE 5. Function words

3.4.1 Example: *juice*

The results of running this algorithm on the word *juice* which is listed under the section headed by *electricity* are given in Table 6:

Sense number	Overlap with word			Total overlaps
	electricity	power	supply	
1				0
2	body, fluid	body		3
3				0
4		strength		1
5	electric	power		2
6		engine	fuel	2
7				0
8				0

TABLE 6. Results from *juice* example

Each row of Table 6 contains the results of comparing the definitions of *electricity*, *power* and *supply* against a sense of the word *juice*. The first column lists the sense number of the word *juice*, the second, third and fourth columns give the words found in common with this sense and the last column is the total number of common words.

The results in Table 6 indicate that the correct sense is number 2, as it is the sense with the greatest number of word overlaps (three). This is incorrect as in this context the word *juice* does not mean 'any natural fluid secreted by an animal body'. The correct sense, number 5, tied with sense number 6 in second place with two overlaps. It is possible to hypothesise a few reasons why the algorithm did not work in this case:

- The correct sense in the definition of *juice*, 'electric power', has very few words compared to some of the other senses. The second sense, which was the one incorrectly chosen, has five non-function words in its definition. This imbalance in

definition lengths leads to considerable bias towards those senses with the longest definitions.

- The correct sense for the words *electricity* and *power* were both missing from the dictionary. This has been checked with the publishers and confirmed.

The above two reasons both suggest that the word *juice* was not a good first choice so two further words were selected for testing.

3.4.2 Example: *boor*

Another page of the thesaurus contains the following information:

discourtesy
n. **discourteous person**, Goth, bastard, blackguard, boor, bull in a china shop, churl, gremmie, ocker, roughie, vulgarian, yahoo; **minx**, fishwife, hoyden, quean, tactless Tilly; **insulter**, knocker, snubber.

The definition of the word of interest, *boor*, is given below:

boor
n. 1. a rude or unmannerly person.
2. a peasant; a rustic.
3. an illiterate or clownish peasant.
4. a Dutch or German peasant.
5. any foreign peasant.
6. (*cap.*) Boer.

The correct sense of the word *boor* in the fragment of text from the Macquarie Dictionary is number one. The results from running the simple algorithm are given in Table 7:

Sense number	Overlap with word									Total overlaps
	bastard	blackguard	churl	Goth	gremmie	ocker	roughie	vulgarian	yahoo	
1	person	person	person, rude	person, rude				person	person	8
2			peasant, rustic							2
3			peasant							1
4			peasant							1
5			peasant							1
6										0

TABLE 7. Results from *boor* example

For this example the algorithm chose the correct sense by a ratio of four to one over the sense with the next highest number of overlaps. The definitions of the other words from the thesaurus, *bastard*, *blackguard*, etc. are given in Appendix A.

3.4.3 Example: *qualm*

Here is another example, the word *qualm*:

pain

n. **ache**, pain, pang, qualm, throb, throe, twinge, twitch; **headache**, hemicrania (*Obs*), migraine, sick headache, splitting headache; **earache**, toothache; **backache**, Lebanese back, Mediterranean back, shagger's back; **stomach-ache**, colic, collywobbles, gastralgia, gripes, hunger pain, pain in the gut; **afterpains**, growing pains, phantom limb pains, referred pain, teething pains; **cardialgia**, angina; **hyperalgesia**, arthralgia, brachialgia, causalgia, hemialgia, neuralgia, neuritis.

qualm
n. 1. an uneasy feeling or a pang of conscience as to conduct.
2. a sudden misgiving, or feeling of apprehensive uneasiness.
3. a sudden sensation of faintness or illness, esp. of nausea.

The correct sense here is number three.

Sense number	Overlap with word							Total overlaps
	ache	pain	pang	throb	throe	twinge	twitch	
1			feeling		pang			2
2	sudden		feeling, sudden			sudden	sudden	5
3	sudden	sensation	sudden			sudden	sudden	5

TABLE 8. Results from *qualm* example

As the above table shows the correct sense, number three, had an equal number of matches with sense number two. In the case of a tie the algorithm returns both senses as the correct ones.

3.4.4 Results

The results from running this method on the test file containing 39 examples are given in Tables 9 and 10. The last column labelled 'Modified algorithm' lists the results of the method in which if a word appears *n* times in the definition of a sense and *m* times in the definitions of the other words it is being matched against contributes *m* overlaps to the total. These results are encouraging for a first attempt and indicate that the method has some promise.

Keyword	Part of speech	Word	Classification	Modified algorithm
electricity	n	juice	8	3
discourtesy	n	boor	3	3
pain	n	qualm	3 8	3 8
emblem	n	standard	8	8
fauna	n	kid	3 8	3 8
deception	v	trick	8	8
obviousness	adj	obvious	3	3
fertility	v	crop	3	3
ill health	n	cold	3	3 8
sign	v	wave	8	8
pleasantness	adj	sweet	8	8
foreignness	adj	alien	8	8
fastening	v	tie	3	3
essay	n	introduction	3	3
strangeness	adj	funny	3	3
extraction	v	draw	8	8
closeness	adj	contiguous	8	8
effort	adj	hard	8	3
fine arts	v	charcoal	8	8
generality	adj	widespread	8	8
fauna	n	mammoth	8	8
jealousy	v	covet	8	3
ascent	n	pulley	3	3
killing	v	smother	8	8
representation	n	caricature	8	8
excess	adv	ad nauseam	3	3
impropriety	adj	low	8	3
goodness	adv	excellently	3	3
intangibility	n	apparition	3 8	3 8
food	n	cordial	3	3
dwelling	n	dump	3	3
closure	n	cork	3	3
hardness	v	set	8	8
buying	v	shop	8	8
showiness	adj	flamboyant	8	8
time measurement	v	clock	8	8
book	n	comic	3	3
entertainment	n	film	3 8	3 8
payment	n	advance	3 8	3

TABLE 9. First results

	l		*m*	
correct	14 out of 39	36%	18 out of 39	46%
close	19 out of 39	49%	23 out of 39	59%
random	8 out of 39	19%	8 out of 39	19%

TABLE 10. Summary of first results

3.5 Variations

The preceding section showed the difference in performance caused by the two different methods of accounting for words that occur more than once in a definition. It presented the results for *l* and *m* overlap counting. This section shows the results for two other possibilities: *n* and $m \times n$. The results for the file of 39 examples are:

	n		$m \times n$	
correct	19 out of 39	49%	20 out of 39	51%
close	24 out of 39	62%	25 out of 39	64%
random	8 out of 39	19%	8 out of 39	19%

These results do not show any great difference between the results obtained by the two different methods. This not surprising given the limited test set available for this stage of testing. However the best results were obtained for the $m \times n$ overlap counting so this method was used for the rest of the testing (standard method).

3.6 Equal definitions

In the preceding experiments an overlap was defined as a word in the definition of a sense of the target word and a word in the definition of one of the surrounding words being typographically the same. This section of the project was an experiment to see if changing the definition of a word overlap improves the performance of the thesaurus to dictionary matching algorithm. The new definition says that two words are the same if they have a dictionary definition in common. For example the following definition for *condone* means that *condone*, *condoned*, *condoning* and *condoner* all have the same definition.

> **condone** *v.t.*, **–doned, –doning.**
> **1.** to pardon or overlook (an offence).
> **2.** to cause the condonation of.
> **3.** to atone for; make up for.
> **4.** *Law.* to forgive, or act so as to imply forgiveness of (a violation of the marriage vow).
> **–condoner,** *n.*

It was expected that this different method of overlap counting would be an improvement as the matching algorithm would count inflected forms and run-ons as overlaps. These types of words are very closely related to their headwords and so should be regarded as identical for the purposes of this algorithm.

The results obtained by classifying the file of 300 examples prepared with the help of Macquarie Library were:

	standard method		equal definitions	
correct	152 out of 299	50.84%	148 out of 299	49.50%
close	175 out of 299	58.53%	176 out of 299	58.86%
random	73 out of 299	24.46%	73 out of 299	24.59%

These results do not show the expected improvement. This is thought to be because the effect discussed above is only slight and is overwhelmed by implementation problems.

The change in the method of overlap counting is achieved by looking up each word in a definition in the dictionary database. This returns a list of addresses of the definitions of the word in the file of definitions. These addresses are put into a list for each sense of each word. These lists are compared when looking for overlaps; in this method an overlap occurs when two identical addresses occur. Some words such as *all*, *pine* and *ask* have more than one definition in the Macquarie Dictionary. For example the definitions of *pine* are:

> **pine**
> n. **1.** any member of the genus *Pinus,* comprising evergreen coniferous trees varying greatly in size, with long needle-shaped leaves, including many species of economic importance for their timber and as a source of turpentine, tar, pitch, etc.
> **2.** any of various more or less similar coniferous trees.
> **3.** the wood of the pine tree.
> **4.** *Colloq.* → pineapple.
> **–pinelike,** *adj.*
> **pine** *v.,* **pined, pining,** *n.*
> –*v.i.* **1.** to suffer with longing, or long painfully (fol. by *for*).
> **2.** to fail gradually in health or vitality from grief, regret, or longing.
> **3.** to languish, droop, or waste away.
> **4.** to repine or fret.
> –*v.t.* **5.** *Archaic.* to suffer grief or regret over.
> –*n.* **6.** *Obs.* or *Archaic.* painful longing.

Both of these definitions have to be included in the lists of addresses. If, for example, only the first address is included then the word *pined* will not count as an overlap with *pine* because its definition will include number 2 only and so will not match with the address of definition 1. The problem with including all the definitions appears when the word *pine* occurs in the definition of the target word and one of the surrounding words. In this case the number of overlaps counted is doubled because there are two definitions in common. This problem is much worse when more common words with three or more definitions occur in both lists. Solving this problem by recoding the algorithm should improve the effectiveness of the modification outlined in this section. This recoding, however, must go under the heading of further work. For the rest of this project the simple non-dictionary method is used.

3.7 Simultaneous determination

All the methods outlined previously determine the sense of one word in a list of words. This is done by deciding which of the senses of the target word has most in common (the greatest number of word overlaps) with the definitions of all the other words. Another approach is possible, that is to determine the correct sense of all the words in the list simultaneously rather than just one word in the list. For example, if presented with the list:

pulley, block and tackle, capstan, cat, winch, windlass.

the algorithm should decide that the correct definitions are:

> **1.** a wheel with a grooved rim for carrying a line, turning in a frame or block and serving to change the direction of or transmit power, as in pulling at one end of the line to raise a weight at the other end.
>
> *n.* the pulley blocks and ropes used for hoisting.
>
> *n.* a device resembling a windlass but with a vertical axis, commonly turned by a bar or lever, and winding a cable, for raising weights (as an anchor) or drawing things closer (as a ship to its jetty).
>
> **7.** *Naut.* a tackle used in hoisting an anchor to the cathead.
>
> **2.** a windlass turned by a crank, for hoisting, etc.
>
> *n.* **1.** a device for raising weights, etc., usu. consisting of a horizontal cylinder or barrel turned by a crank, lever, or the like, upon which a cable or the like winds, the outer end of the cable being attached directly or indirectly to the weight to be raised or the thing to be hauled or pulled.

The first thing that needs to be commented on is the number of combinations of senses the algorithm will have to consider. If there are four words in the list each with six senses there are 6^4 possibilities. The problem is much worse in some cases where there are ten or more words in the list and some of those words have up to twenty senses.

This variation is implemented by calculating the number of possibilities and then for each possibility, comparing each of the senses with each of the other senses and calculating the total number of overlaps for that possibility. The possibility with the highest number of overlaps gives the correct sense for each of the words in the list.

The limited amount of test data available and the need for consistent results between tests means that the same test set of 300 examples had to be used to test this algorithm. This is done by using the algorithm to determine the correct sense of each of the words found in the indicated section of the thesaurus but only printing out the result for the word for which the data set contains the correct sense.

The following table contains the results for this method and the standard one (the one with $m \times n$ overlap counting). Each of the examples in the test file were run with a cpu time limit of 60 minutes.

	standard method		simultaneous determination	
correct	152 out of 299	50.84%	36 out of 172	20.93%
close	175 out of 299	58.53%	124 out of 172	72.09%
random	73 out of 299	24.46%	44 out of 172	25.78%

There were only 172 examples completed because of the cpu time limit and because in some cases the number of possibilities overflowed the machine precision of a unsigned 32 bit number ($2^{32} = 4\,294\,967\,296$). The loss of these examples reduces confidence in predictions that are drawn from the conclusions as the examples lost were not chosen randomly but rather are those examples with the longest definitions and the longest thesaurus lists. These are exactly the examples which might be expected to perform best. So the above results may tend to underestimate the results that might be obtained with unlimited machine time and arbitrary precision numbers. However it can be argued that such conditions are never available.

Notwithstanding doubts about the exact validity of the results, they do show that simultaneous sense determination is not worth while as the results using this method are worse than those that would be obtained using random guessing. The percentage expected to be correct using random guessing has increased because those examples with the target word having large numbers of senses have been removed by time limits and overflows.

The experiments were repeated using the equal definitions overlap modification discussed previously with the same basic results—simultaneous determination should not be used.

The main reason for the failure of simultaneous sense determination is the small number of words in each sense. When the standard method of sense determination is used each sense of the target word is compared with all the definitions of the surrounding words and this gives a large number of words with which to find overlaps. When using simultaneous determination only the few words in each sense are available for overlaps. As many senses are very closely, related using only one sense at a time loses the contribution these other senses may make to choosing the correct sense. For example, many words have different sense definitions for the noun and verb with the same base meaning and when attempting to find the correct sense of another word by using this definition better results would be obtained by combining these senses.

The increase in the number of trials that produced both correct and incorrect answers is due to the algorithm not being able to distinguish between possibilities. This results from the paucity of words in the definitions being worked with. In this case many possibilities would get zero or one overlaps.

3.8 Length weighting

A final experiment was to measure the effect of a simple weighting of the number of overlaps obtained by each sense by the length of the definition of that sense. The method chosen was to divide the number of overlaps by the number of non-function words in that sense. The result of this is that the sense that has highest proportion of overlaps amongst its non-function words is chosen. This variation was intended to remove the bias towards senses with longer definitions.

The results for this method compared to the standard method are given in the following table.

	standard method		**length weighted**	
correct	152 out of 299	50.84%	144 out of 299	48.16%
close	175 out of 299	58.53%	156 out of 299	52.17%
random	73 out of 299	24.96%	73 out of 299	24.46%

As has been found before this variation had almost no effect on the results of the experiments. It appears that the difference in definition lengths is small and has little effect on the results.

3.9 Conclusions

The experiments in this chapter indicate that the best method for performing thesaurus to dictionary sense matching uses simple overlap counting and $m \times n$ overlap counting.

The results obtained here were at the low end of those reported by Lesk[16] but were applied to a different domain; thesaurus to dictionary sense matching rather than sense disambiguation in natural text. Improved results should be obtained by further experimentation into ideas such as weighting the worth of an overlap by the frequency of occurrence of the word in the dictionary. But for now the results are sufficient to

4. Dictionary to thesaurus sense matching

One thing that literature would be greatly the better for
Would be a more restricted employment by authors of simile and metaphor.
Authors of all races, be they Greeks, Romans, Teutons or Celts,
Can't seem just to say that anything is the thing it is but have to go out of their way to say that it is like something else.

Ogden Nash
Very Like a Whale

As the preceding chapter described, it is possible to find the dictionary sense of a word as used in a particular section of the thesaurus, but there are also many applications which require the reverse operation. This chapter describes how given a word and its dictionary sense it is possible to choose which entry of this word in the thesaurus corresponds to the given dictionary sense. For example the sense 'petrol, fuel oil, etc., used to run an engine' of the word *juice* corresponds to the section of the thesaurus that contains 'fuel, combustible, feed, juice'. This algorithm means that it is possible to convert a word sense into a list of synonyms in thesaurus. Because of the hierarchical structure of the thesaurus the synonyms are graded into those words that are close synonyms and those that are less closely related.

4.1 Example

An example using the word *standard* is now given. The dictionary definition of *standard* is:

standard *n.* **1.** anything taken by general consent as a basis of comparison; an approved model. **2.** the authorised exemplar of a unit of weight or measure. **3.** a certain commodity in which the basic monetary unit is stated, historically usu. either gold or silver (**gold standard, silver standard,** or **single standard**), or both gold and silver in a fixed proportion to each other (**bimetallic standard**). **4.** the legal rate of intrinsic value for coins. **5.** the prescribed degree of fineness for gold or silver. **6.** a grade or level of excellence, achievement, or advancement: *a high standard of living.* **7.** a level of quality which is regarded as normal, adequate, or acceptable. **8.** a fitting or size, as for clothes, which is regarded as normal or average. **9.** (*usu. pl.*) behaviour, beliefs, etc., regarded as socially desirable or acceptable. **10.** a class in certain schools. **11.** a flag, emblematic figure, or other object raised on a pole to indicate the rallying point of an army, fleet, etc. **12.** a flag indicating the presence of a sovereign. **13.** *Mil.* **a.** any of various military or naval flags. **b.** the colours of a mounted unit. **14.** *Her.* a long tapering flag or ensign, as of a king or a nation. **15.** something which stands or is placed upright. **16.** an upright support or supporting part. **17.** an upright timber, bar, or rod. **18.** *Hort.* a tree, shrub, or other plant having a tall, erect stem, and not grown in bush form or trained upon a trellis or other support. **19.** *Bot.* → **vexillum.** **20.** a piece of music or the like of lasting popularity, esp. one often revived with new arrangements. **21.** standard petrol. –*adj.* **22.** serving as a basis of weight, measure, value, comparison, or judgment. **23.** of recognised excellence or established authority: *a standard author.* **24.** normal, adequate, acceptable, or average: *standard goods, a standard fitting.* **25.** (of a variety of a given language, or of usage in the language) characterised by preferred pronunciations, expressions, grammatical constructions, etc., the use of which is considered essential to social or other prestige, failure to conform to them tending to bring the speaker into disfavour.

The word *standard* occurs in the thesaurus under 14 different guide words. These are:

average, classic, control group, ethnic, flag, flower, organ, fossil fuel, musical piece, rank, standard (model), standard (rule), ordinary, standard.

Each of these guide words has an associated list of words which are synonyms of *standard* used in one of its different senses. Some of these lists of synonyms are:

flag, banderol, banner, bannerette, burgee, dojuane ensign, fanion, gonfalon, guidon, hoist, jack, labarum, pennant, pennon, standard, streamer, vexillum.

average, medium, middle, norm, normal, normality, par, standard.

classic, model, standard.

control group, criterion, standard, touchstone, yardstick.

ethic, moral, moral code, moralism, standard.

The user might wish to know which of the above thesaurus entries or synonym lists corresponds to the word *standard* used in the sense:

> **11.** a flag,emblematic figure, or other object raised on a pole to indicate the rallying point of an army, fleet, etc.

The task of the dictionary to thesaurus matching algorithm is to decide that the following synonym list is the correct one:

> **flag**, banderol, banner, bannerette, burgee, dojuane ensign, fanion, gonfalon, guidon, hoist, jack, labarum, pennant, pennon, standard, streamer, vexillum.

4.2 The algorithm

The algorithm for dictionary to thesaurus sense matching is very similar to that for thesaurus to dictionary sense matching. It is based on the observation that the definitions of related word senses tend to have words in common and the more closely related the words the more words in common. This observation is used here to perform the required matching.

The first step of the algorithm is to look up the given word in the dictionary and to extract the definition of the required sense. All function words are then removed from this definition. In the above example concerning the word *standard* this would give the following word list:

> flag, emblematic, figure, other, object, raised, pole, indicate, rallying, point, army, fleet.

The next stage is to look the target word (*standard*) up in the thesaurus. If it does not appear in the thesaurus the algorithm reports an error. Should the target word appear only once there is only one choice for the matching thesaurus paragraph. This choice will be correct if the thesaurus and dictionary are consistent.

When the target word appears more than once in the thesaurus the algorithm must decide which of the lists of synonyms found contains the target word used in the sense required. Each entry in the thesaurus has a list of synonyms associated with it; these lists are retrieved and each word in the list (except for the target word) looked up in the dictionary. The definitions for each of the words in a synonym list are combined together to give a long list of words corresponding to each occurrence of the target word in the thesaurus. The text of the dictionary sense of the target word (obtained in the first step) is then compared with each of the word lists. The thesaurus entry corresponding to the word list with the most overlaps is assumed to contain a list of synonyms of the word sense.

Continuing the *standard* example involves reading from the thesaurus each of the 14 lists of synonyms, removing the word *standard* from each list and looking up each of the remaining words in the dictionary. These definitions are then formed into a list of words for each of the thesaurus entries of *standard*. Each of these 14 lists is then compared against definition number 11 of *standard*; and the word list with the largest number of overlaps is assumed to be associated with the correct thesaurus entry for *standard*.

4.3 Testing

The dictionary to thesaurus matching algorithm was tested on the same data set as the thesaurus to dictionary matching algorithm. While this test set was designed to match a word in a thesaurus section to a dictionary sense, it can also be regarded as matching a dictionary sense to a thesaurus paragraph. This is valid because a dictionary sense will correspond to at most one thesaurus paragraph due to the much greater precision of dictionary senses compared to thesaurus groupings. So this algorithm could be tested on the test set of 300 examples.

It is possible that the test set is biased because the data set does not consist of a random sample of all dictionary senses. A preferable technique would involve choosing such a sample and having it independently classified by thesaurus entry. However this was not possible due to time constraints and so the above procedure had to be used causing an unknown level of bias in the results.

correct	181 out of 299	60.54%
close	186 out of 299	62.21%
random	115 out of 299	38.48%

These results are promising for a first attempt at dictionary to thesaurus matching as they show that the algorithm was able to choose the correct section in over 60% of cases whereas random guessing would achieve less than 40% accuracy. Further experimentation should be able to improve the performance of the algorithm to the 80% level which would be required for the text retrieval methods discussed later. However an important first step is to retest the algorithm using a more carefully designed test set.

5. Sense disambiguation

Proper words in proper places, make the true definition of style.
Jonathan Swift (1667−1745)
Letter to a Young Clergyman

This chapter describes a method for determining the dictionary sense in which a word in a piece of English text is being used. This work is a development and implementation of the algorithm described by Lesk.[16]

The algorithm might be given a piece of text such as 'all hands to reef topsails' and be asked to determine in what sense the word *reef* is being used. The dictionary definition of *reef* is:

> **reef**
> *n.* **1.** a narrow ridge of rocks or sand, often of coral debris, at or near the surface of water.
> **2.** *Mining.* a lode or vein.
> **reef**
> *n.* **1.** a part of a sail which is rolled and tied down to reduce the area exposed to the wind.
> −*v.t.* **2.** to shorten (sail) by tying in one or more reefs.
> **3.** to reduce the length of (a topmast, a bowsprit, etc.), as by lowering, sliding inboard, or the like.
> −*v.i.* **4.** (of a horse) to throw its head up, thereby pulling against the reins.
> **reef**
> *v.t. Colloq.* **1.** to remove, usu. by force (fol. by *out*).
> **2.** to steal (fol. by *off*).

The correct answer would be sense 4:

> −*v.t.* **2.** to shorten (sail) by tying in one or more reefs.

5.1 The algorithm

Sense disambiguation is accomplished by looking up the target word, *reef*, in the dictionary and forming a list of non-function words for each sense. In the *reef* example this would give eight lists of words, one for each of the senses of *reef*.

The next stage is to look up in the dictionary each of the non-function words surrounding the target word in the text. For this example the words would be 'hands, topsails'. (The performance of the algorithm is much improved if before looking the words up in the dictionary the words have plural endings removed. This would mean that many more of the surrounding words would be found in the dictionary.) All the dictionary definitions of the surrounding words are combined into one long list of words.

The list is then compared with each of the senses of the target word. The sense with the greatest number of overlaps is deemed the sense in which the target word is being used in the piece of text. The implementation of this algorithm also incorporates the various

alternative measures of a word overlap, simultaneous determination and sense length weighting discussed in the Chapter ‘Thesuarus to dictionary sense matching’.

5.2 Testing

Testing this algorithm needs a large data set similar to that used to test the thesaurus to dictionary sense matching algorithm. For this algorithm what constitutes a suitable test set will depend on the desired area of application. When testing the performance of the algorithm on the task of determining the correct sense of words in news stories the test set should contain a set of words from news stories and their correct dictionary sense. Such a data set was not available.

The required data set can be produced by randomly selecting passages from a suitable body of text and manually classifying one or more of the non-function words in each of the passages. This data set could then be used to test and compare the various differing implementations of the sense determination algorithm.

Some experiments were done on a few short news stories provided by Australian Associated Press (AAP). These test indicated two main points:

- The number of overlaps attributed to a dictionary sense needs to be weighted by the number of non-function words occurring in the definition of the sense. As before, this was done by dividing the number of overlaps by the number of non-function words.
- The list of function words used previously was inadequate; it was observed that many words that were being used as function words were being counted as overlaps. To overcome this problem a new list of function words was constructed. It consisted of the 64 most commonly occurring words in the text of the Macquarie Dictionary. The list is given in Table 11:

of	on	having	it
the	one	ac303	up
or	ME	Gk	are
to	Colloq	person	part
in	Also	not	out
from	is	like	such
and	any	form	two
as	esp	other	OF
adj	pl	into	its
an	adv	usu	made
by	that	something	state
with	used	small	one’s
etc	at	act	place
for	be	being	See
which	who	OE	body
ac312	pertaining	make	some

TABLE 11. New function word list

Once these changes were made the experiments conducted produced some promising results. The best of these were on the following news item from AAP:

> DOLLAR OPENS TOKYO
> The dollar opened at 1.8155/60 marks against 1.8187/97 in New York. The dollar fell as low as 142.90 yen despite central bank intervention at 143.00 yen, dealers said. Selling pressure was strong from securities houses and institutional investors in hectic and nervous trading on underlying bearish sentiment for the dollar, they said. Most dealers were surpised by the dollar's sharp fall against the yen in New York, although many had expected such a drop to happen eventually.
> AAP-RES

All the following tests were run using the entire article (after removing function words and stripping suffixes) as the list of surrounding words.

The word *yen* has two very different meanings:

> **yen** *n., pl.* **yen.**
> the monetary unit of Japan.
> **yen** *n., v.,* **yenned, yenning.**
> *Colloq.* –*n.* **1.** desire; longing.
> –*v.i.* **2.** to desire.

The sense determination algorithm reports the correct sense as 'the monetary unit of Japan'. This is the correct sense of *yen*.

Another important word is *bearish* as it gives a good clue to the mood of the currency markets as reported in the news item. The definition of *bearish* is:

> **bearish**
> *adj.* **1.** like a bear; rough; burly; morose; rude.
> **2.** *Stock Exchange.* unfavourable and tending to cause a decline in price.
> **–bearishly,** *adv.* **–bearishness,** *n.*

The algorithm chose the correct sense '**2.** *Stock Exchange.* unfavourable and tending to cause a decline in price.'.

Another example tried was the word *bank*. In this case the algorithm got the wrong answer. It returned the definition of the verb:

> **7.** to keep money in, or have an account with, a bank.

when the correct answer is:

> *n.* **1.** an institution for receiving and lending money (in some cases, issuing notes or holding current accounts that serve as money) and transacting other financial business.

It is interesting to note that although the algorithm got the wrong sense it still returned a sense whose meaning is very close to the correct one.

5.3 Conclusion

The results presented in the last section were promising but there are many cases in which the algorithm returns the wrong sense. Testing using a randomly chosen test set is required before making any conclusive predictions about the performance of the algorithm on real data. Further improvements, such as using sentence structure to find the part of speech of the target and surrounding words, also need to be investigated.

6. Text retrieval

> *Knowledge is of two kinds. We know a subject ourselves, or we know where we can find information upon it.*
>
> *Dr Samuel Johnson (1709−1784)*
> *Letter to William Strahan*

Large databases of text containing information such as scientific papers, legal cases, newspaper archives and library catalogues have been common for many years. Such databases are typically too large to allow a user to scan the entire collection of text in search of an interesting item. For this reason more practical and faster methods of retrieving pieces of text have been devised.

One of these methods is called keyword retrieval. In this method a user specifies a keyword (or perhaps a boolean combination of keywords) to be used in retrieval. For example a simple query such as `bridge` would retrieve all articles containing the word ‘bridge’. A boolean combination such as `juice and not (apple or orange)` might be used to find articles that mention ‘juice’ but are not concerned with fruit juices.

The standard method of text retrieval by keyword is fraught with difficulties caused by the imprecise nature of words. Section 6.1 explains and attempts to solve these problems using the information contained in machine-readable dictionaries and thesauri.

Other methods of retrieval from a large database allow the user to retrieve articles that concern topics of interest to the user. For example a user might want all the articles that are concerned with physiology. Previous implementations of this method of text retrieval have required the text to be classified by subject before this process can be used and this normally has to be performed manually. Section 6.2 gives a method by which text classification can be automated.

6.1 Text retrieval by keyword

Existing text retrieval systems allow the user to specify a list of keywords of interest. Each piece of text in the database is checked to see whether or not it contains one of the keywords specified. If it does the piece of text is retrieved.

This method is easy to use and, by using algorithms such as hash tables and inverted indices, efficient to implement. However, keyword retrieval suffers from the restriction that an exact match between the keyword and a word in the text is required for retrieval. If the user is attempting to retrieve text concerned with road works, specifying the keyword *bridge* will only find articles that explicitly mention ‘bridge’. Articles that contain the words ‘walkway’, ‘pontoon’ or ‘footbridge’ will not be retrieved even though they are close synonyms of the keyword.

Keyword retrieval also suffers from the problem of retrieval of irrelevant articles that contain a keyword used in a different sense to that intended by the user. Someone who

wants to know about road works and uses the word *bridge* will not be interested in an article relating to the card game. The new text retrieval system described in this section attempts to use the information contained in dictionaries and thesauri to solve the two problems outlined above. This new method is based on two algorithms described previously: one to perform sense disambiguation in natural text and another to perform dictionary to thesaurus sense matching. The way that these two algorithms can be used in text retrieval is outlined in this and the following sections.

6.1.1 How to retrieve text

The method described here for text retrieval consists of three parts: the selection of the keywords by the user, the processing of the text to be searched, and the selection of relevant pieces of text.

6.1.1.1 Keyword selection

In a standard keyword retrieval program the user enters words to be matched against words in the text. In this new system the user is also prompted to obtain information on the intended sense of each keyword specified. There are two possible ways of obtaining this information. The first and preferred method is to look up the word provided by the user in the thesaurus, displaying the entries for the various senses of the word. The user can then specify which of the senses best captures the intended meaning of the word. For example the entries for *juice* are:

alcohol, electricity, fuel, liquid, nitty-gritty, secretion, vigour.

The user can then indicate whether the sense meaning electricity or the sense meaning fruit juice is required. With this method the can user expand the range of searching by including words at a higher level of the thesaurus by simply choosing a sense and asking for the region to be enlarged to include more remote synonyms of the keyword. Once this is done the system has a thesaurus paragraph number to be used in retrieval.

In the second method the user is presented with the dictionary definition of the keyword and is asked which of the indicated senses is intended. This sense could then converted into a thesaurus paragraph or matched directly against occurences of that keyword in the stored texts.

6.1.1.2 Text Processing

Most text archives available are distributed by some organisation. For example, press agencies distribute news stories to paying subscribers. To use this new method of text retrieval the distributor will have to run a computer program to preprocess the text. It will determine the sense of each non-function word in the text using the sense disambiguation algorithm. As in the ‘Keyword selection’ section this dictionary sense has a corresponding thesaurus paragraph which is determined using the dictionary to thesaurus matching algorithm and has a unique number. Once this process is complete each article

of text has a list of thesaurus paragraph numbers which are stored and transmitted with the article and can then be used for retrieval as discussed below.

6.1.1.3 Retrieval

When a user is retrieving text, either interactively or in a batch procedure, each paragraph number generated by the user is compared against those associated with each article. If a match occurs the article is retrieved and presented to the user. Should this method produce too many incorrectly retrieved articles the user can specify more than one keyword for each topic of interest and require that an article be selected only if some number of paragraph numbers match.

6.1.2 Example

Suppose an ASIO agent wishes to look up or retrieve articles concerned with entrapment. He would enter the keyword *entrap*. The system presents him with the various senses for 'entrap' listed in the thesaurus. He would then choose the sense containing the following synonyms:

> endanger, compromise, entrap, expose, imperil, jeopardise, peril, put the skids under.

The following article would be retrieved:

> Soviet foreign ministry security agents showed off some of the bugging devices they say were discovered in Soviet diplomatic missions throughout the U.S. Ivan Miroshkin of the Soviet foreign ministry security service said that several bugs with connections to radio transmitters were uncovered. The presentation was designed to counter U.S. charges of Soviet spying. The Soviets displayed photographs and devices they called 'Violations of their sovereign territory.' They said the devices were taken out of the Soviet residence in New York City, the mission in Washington and consulate in San Francisco. The Soviets did not say if any of their secrets had been compromised by the presence of the listening devices.

When this article was processed the word 'compromise' had its sense determined using the sense disambiguation algorithm. The correct sense was:

> **7.** *Mil.* to subject (classified material) to the risk of passing to an unauthorised person.

as opposed to the more common meaning of making mutual concessions. The dictionary to thesaurus matching algorithm was then used to convert this word sense into a thesaurus paragraph. This was the same as the one chosen by the agent, so the article was retrieved.

6.2 Text retrieval by classification

Text retrieval by classification is an alternative to text retrieval by keyword. Keyword retrieval searches in the text for individual words that match a particular word or pattern.

One particular word occurring in the text is enough to conclude that an article should be retrieved. Text retrieval by classification takes a more global approach. The entire text of an article is used to determine the subject matter of an article and the user is then able to retrieve articles that are about a subject of interest. As the entire text is used this works best on databases where each article is on one topic. Newswire stories are a good example of such text.

6.2.1 Previous Work

As discussed in the literature review Walker and Amsler have written a program called FORCE4 [20,19]. It uses the subject codes of the Longmans Dictionary of Contemporary English to perform text retrieval by classification. The successful use of this method requires a dictionary with a meaningful set of subject codes. A simple extension to FORCE4 would be to attempt to determine the sense of each word in the text and use only the subject code associated with that sense, rather than recording the subject codes given for all the senses of the word. As the Basser Department of Computer Science does not have access to the LDOCE this modified method could not be tested.

6.2.2 Classification

Classification of text can be accomplished with a standard dictionary without subject codes if a thesaurus is available to supply subject groupings. The resulting subject classifications will depend on the subjects under which the words in the thesaurus are classified. Best performance will be obtained by using a thesaurus that reflects the subject classifications that a user would be expected to employ. Text can be classified by looking up each word of the text in the thesaurus and recording under which thesaurus paragraphs it falls. The subject of the thesaurus paragraph that occurred most frequently is deemed to be the subject of the article.

A standard thesaurus, such as The Macquarie, has several levels of classification. These group words into small sets of synonyms and larger groups of loosely related words. Determination as to which of the four levels of classification is best suited for retrieval purposes has yet to be made.

This method can, in a similar way to FORCE4, be modified to use a dictionary to determine the sense in which a word is being used in text. Once the sense is known the dictionary to thesaurus matching algorithm can be used to map the word into a single thesaurus paragraph. Finding only one paragraph number for each word leads to better accuracy in classification.

Another extension that would help to solve the problem of misclassification is to assume that articles can have more than one subject. This can be achieved by saying that the subjects of a piece of text are those subjects whose frequency of occurrence is within some percentage (specified by the user) of the most common subject.

6.2.3 Retrieval

Once all the pieces of text in a collection have been classified a user can retrieve articles about any desired subject by simply specifying the required subject.

6.3 Conclusion

This chapter has described two new methods of performing retrieval from large bodies of text. These methods offer the promise of more efficient and successful retrieval of text.

Both the methods consist of two parts: the two underlying algorithms that perform sense disambiguation and dictionary to thesaurus sense matching, and a way of using these algorithms effectively. These two parts are both important to the success of the overall system. Improvements in both are necessary to produce a commercially viable system.

7. Applications

'The time has come,' the Walrus said,
'To talk of many things:
Of shoes − and ships − and sealing-wax −
Of cabbages − and kings −
Of why the sea is boiling hot −
And whether pigs have wings.'
Lewis Carroll (1832−1898)
Through the Looking Glass

7.1 Thesaurus browser

Many people use dictionaries and thesauri when writing reports and other text using word processing software. At the moment these dictionaries and thesauri are used in the printed form; the only use for machine-readable dictionaries is in word lists for spelling checking. The databases produced in the section 'Tape formats' can be used in an interactive thesaurus browser which has links to an online dictionary. New storage technologies such as CD-ROMs, which can contain up to 500 megabytes of information, make it possible for the user of a personal computer to have, at a reasonable price, access to the large quantities of data in machine-readable dictionaries.

The thesaurus browser designed for this thesis allows the perusal of the online thesaurus in a way similar to that used with a printed thesaurus. The advantage of the browser is that the copious flicking of pages required in the use of the printed thesaurus is replaced by immediate retrieval. It is also easier to decide that you have chosen the wrong sense of a word and to go back a stage. The dictionary linkage means that the user can quickly get the definition of any word in the thesaurus and, via the use of the thesaurus to dictionary sense matching algorithm, the system can be requested to supply only the sense of the word as used in the section of thesaurus currently being examined. For example if the user is viewing the section of the thesaurus containing:

> **endanger**, compromise, entrap, expose, imperil, jeopardise, peril, put the skids under.

and asks for the definition of word 2, *compromise*, the system will reply with:

> **7.** *Mil.* to subject (classified material) to the risk of passing to an unauthorised person.

Of course, the full definition may also be obtained.

As with many other conversions from the printed word to the computer screen, the browser suffers from the low bandwidth of even the highest quality computer display compared to a printed page of text. It is possible to fit much more information in a more easily read manner onto a printed page. However the browser makes use of the computer's ability to find information more quickly and more accurately to provide a

useful alternative to the printed dictionary and thesaurus.

7.2 News wire retrieval

Australian Associated Press (AAP) recently released a product called *Flak Fury*.[3] Flak Fury is a program that runs on an IBM Personal Computer and allows selective storage of AAP stories as they become available over the news wire. AAP produces more that two million words each day, generating more than 40000 news items in many different news categories. Flak Fury is targeted mainly at business users who would not be able to scan even a small proportion of such a large quantity of information and its aim is to store only those news items that are of interest to the user.

The retrieval system is based on categories and search words. News items can be retrieved because they are in a particular category, such as *shipping* or *trade unions*, or because they contain a keyword. Assuming news stories are correctly classified the category retrieval system will produce stories that are of interest. However in many instances category retrieval will have to be supplemented with keyword-based retrieval in order to reduce the number of uninteresting stories displayed. The problem with keyword retrieval is that an exact match is needed between the key and a word or words in the news item. A preferable system would allow a story to be matched to a keyword if it contained a synonym of the keyword. The desired system would match the sentence 'Property and retail group Hooker Corp looks set for its most profitable year . . .' against the key *real estate* because the word *property* is being used in the same sense as real estate. The algorithms designed for this thesis would allow such a system to be realised.

Implementing such a system would require advance processing to determine synonym lists. Each non-function word in a news item would have its sense determined using a dictionary such as The Macquarie Dictionary. This dictionary sense would then correspond to a paragraph in a thesaurus. Each paragraph in the thesaurus contains a list of synonyms for the word in the news item, *property* in this example. The key list for this news item would contain the unique identifier for this paragraph. When setting up the keyword list for retrieval the user would be asked for the dictionary sense of the keywords and these would then be converted into thesaurus paragraph identification numbers to be used to retrieve news stories. The sense determination that has to be performed on each word in each news story need not be carried out by the user but can be performed once by AAP and transmitted as a set of retrieval codes with each news story.

The user of this system may wish to use the key *building* in the sense of the construction of houses etc. Once this key is entered the retrieval system would display the dictionary definition of building, which is:

> **building**
> *n.* **1.** anything built or constructed
> **2.** the act, business, or art of constructing houses, etc.

In this example the sense of the word *building* that the users means is sense number two;

that is, the building of houses.

When AAP is processing a news story that contains 'The development application placed before council for the construction of 512 home units was rejected' it would be able to determine that *development* is being used as sense number five. The definition of development is:

> **development**
> *n.* **1.** the act, process or result of developing.
> **2.** a developed state, form, or product.
> **3.** evolution, growth, expansion.
> **4.** a fact or circumstance bringing about a new situation.
> **5.** a building project, usu. large, as an office block, housing estate, shopping complex, etc.
> **6.** the preparation of vacant land for building by the provision of roads, sewerage, etc.
> **7.** *Music.* the part of a movement or composition in which a theme or themes are developed.

Building sense two and *development* sense six both fall in the same section of the thesaurus:

> **making**, building, construction, contrivance, crystallisation, development, elaboration, erection, fabrication, facture, fashion (*Obs.*), formation, manufacture, manufacturing, output, prefabrication, preparation, production, synthesis, synthesisation, turning, turnout, twinning (*Crystall.*), working.

Thus the modified version of Flak Fury would retrieve the news article containing the word *development* as the identification numbers of *building* and *development* will match.

Carefully designed, this new retrieval system will be able to find more articles of interest and display fewer articles that are not of interest.

8. Conclusion

The Road goes ever on and on
Out from the door where it began.
Now far ahead the Road has gone,
Let others follow it who can!
Let them a journey new begin,
But I at last with weary feet
Will turn towards the lighted inn,
My evening-rest and sleep to meet.
J. R. R. Tolkien
The Lord of the Rings

The aim of the project was to perform experiments and design algorithms to use the data contained in the Macquarie Dictionary and Thesaurus. These were provided on a tape by Macquarie Library Pty Ltd, the publishers, and read into two databases. The design of the database for the thesaurus was found to be adequate for both the browser, and the sense disambiguation and text retrieval algorithms devised. The structure of the dictionary database has been found to be lacking for our purposes. The printed version of the dictionary contains a great deal of information readily accessible to the human reader. All this information is on the typesetting tape although extracting it is very difficult. More work needs to be done to extract information such as the part of speech of each sense in the dictionary.

Algorithms for sense matching between the thesaurus and the dictionary have been implemented and tested. The results from these tests are promising as they show that a quick and easy method of sense matching is feasible. Further improvement and testing should allow these methods to be used in thesaurus browsers and text retrieval packages.

Sense disambiguation in natural text is traditionally regarded as a very difficult problem (‘What type of ball did Cinderella go to?’). Lesk proposed a simple solution to the problem. Implementation and experimentation with his algorithm shows some promising results but it still awaits proper testing.

Two new methods of test retrieval have been designed. Both these methods use the sense matching and sense disambiguation algorithms described in this thesis. These new methods will be able to used in commercial text retrieval packages. AAP has expressed interest in the methods and a patent application is underway.

The conclusion of this thesis is that machine-readable dictionaries and thesauri can be used in sense disambiguation and in commercially viable text retrieval programs.

9. Bibliography

For knowledge itself is power.
Francis Bacon (1561−1626)
Religious Meditations

1. *The Macquarie Dictionary,* Macquarie Library Pty Ltd, St. Leonards NSW (1981).
2. *The Macquarie Thesaurus*, Macquarie Library Pty Ltd, Dee Why NSW (1986).
3. *What is Flak Fury?,* AAP Information Services, 6th Floor, 364 Sussex Street, Sydney (1987). Advertising Brochure
4. Amsler, Robert A., ‘‘The Structure of The Merriam-Webster Pocket Dictionary,’’ TR-164, University of Texas, Austin TX (December 1980).
5. Amsler, Robert A., ‘‘Machine-Readable Dictionaries,’’ pp. 161-209 in *Annual Review of Information Science and Technology*, ed. Martha E. Williams, American Society for Information Science (1984).
6. Amsler, Robert A., *Deriving Lexical Knowledge Base Entries from Existing Machine-Readable Information Sources,* Bell Communications Research, Morristown NJ (May 1986).
7. Amsler, Robert A., *Typesetting from a Dictionary Database,* Bell Communications Research (July 1986).
8. Choueka, Y., S. T. Klein, and E. Neuwitz, ‘‘Automatic Retrieval of Frequent Idiomatic and Collocational Expressions in a Large Corpus,’’ *ALLC Journal* **4**, pp. 34-38 (1983).
9. Geller, V. J. and M. E. Lesk, *User Interfaces to Information Systems: Choices vs. Commands,* Association for Computing Machinery, Murray Hill NJ (1983).
10. Henshaw, A. S. and R. L. Jones, *The Role of Artificial Intelligence in Online Retrieval,* Computer Power Group, Garran ACT.
11. Knuth, Donald E., *The Art of Computer Programming,* Addison-Wesley (1981).
12. Lesk, Michael E., *What Use Are Machine-Readable Dictionaries? A Summary of the “Automating the Lexicon” Workshop,* Bell Communications Research, Morristown NJ.
13. Lesk, Michael E., *Information in Data: Using the Oxford English Dictionary on a Computer,* Bell Communications Research.
14. Lesk, Michael E., *Why Use Words to Label Ideas: The Uses of Dictionaries and Thesauri in Information Retrieval,* Bell Communications Research, Morristown NJ.

15. Lesk, Michael E., ‘‘Computer Software for Information Management,’’ *Scientific American* **251**(3), pp. 162-172 (September 1984).

16. Lesk, Michael E., *Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone,* Bell Communications Research, Morristown NJ (1986).

17. Peterson, James L., ‘‘Webster’s Seventh New Collegiate Dictionary: A Computer-Readable File Format,’’ TR-196, University of Texas, Austin TX (May 1982).

18. Pook, Stuart L. and Jason Catlett, ‘‘Making sense out of searching,’’ *Proceedings of the Online Information Conference 1988,* Library Association of Australia (January 1988).

19. Walker, Donald E., ‘‘Knowledge Resource Tools for Accessing Large Text Files,’’ TR-85-21233-25, Bell Communications Research, Morristown NJ.

20. Walker, Donald E. and Robert A. Amsler, ‘‘The Use of Machine-Readable Dictionaries in Sublanguage Analysis,’’ pp. 69-83 in *Analyzing Language in Restricted Domains: Sublanguage Description and Processing,* ed. Ralph Grishman and Richard Kittredge, Lawrence Erlbaum Associates, Hillsdale NJ (1986).

10. Appendix A

> *'What is the use of a book,' thought Alice, 'without pictures or conversations?'*
>
> *Lewis Carroll (1832−1898)*
> *Alice in Wonderland*

electricity
n. **1.** an agency producing various physical phenomena, as attraction and repulsion, luminous and heating effects, shock to the body, chemical decomposition, etc., which were originally thought to be caused by a kind of fluid, but are now regarded as being due to the presence and movements of electrons, protons, and other electrically charged particles.
2. the science dealing with this agency.
3. electric current.

power
n. **1.** ability to do or act; capability of doing or effecting something.
2. (*usu. pl.*) a particular faculty of body or mind.
3. political or national strength: *the balance of power in Europe.*
4. great or marked ability to do or act; strength; might; force.
5. the possession of control or command over others; dominion; authority; ascendancy or influence.
6. political ascendancy or control in the government of a country, etc.: *the party in power.*
7. legal ability, capacity, or authority.
8. delegated authority; authority vested in a person or persons in a particular capacity.
9. a written statement, or document, conferring legal authority.
10. one who or that which possesses or exercises authority or influence.
11. a state or nation having international authority or influence: *the great powers of the world.*
12. a military or naval force.
13. (*oft. pl.*) a deity or divinity.
14. (*pl.*) *Theol.* an order of angels.
15. *Colloq.* a large number or amount.
16. *Physics, Elect.* the time rate of transferring or transforming energy; work done, or energy transferred, per unit of time.
17. *Mech.* energy or force available for application to work.
18. mechanical energy as distinguished from hand labour.
19. a particular form of mechanical energy.
20. *Maths.* the product obtained by multiplying a quantity by itself one or more times: *4 is the second, 8 the third, power of 2.*
21. *Optics.* the magnifying capacity of a microscope, telescope, etc., expressed as ratio of diameter of image to object.
22. the powers that be, those in authority.
−*v.t.* **23.** to supply with electricity or other means of power.

24. (of an engine, etc.) to provide the force or motive power to operate (a machine).

supply *v.*, **-plied**, **-plying**; *n., pl.* **-plies**
–*v.t.* **1.** to furnish (a person, establishment, place, etc.) with what is lacking or requisite.
2. to furnish or provide (something wanting or requisite): *to supply electricity to a community*.
3. to make up (a deficiency); make up for (a loss, lack, absence, etc.); satisfy (a need, demand, etc.).
4. to fill (a place, vacancy, etc.); occupy as a substitute.
–*v.i.* **5.** to fill the place of another, temporarily, or as a substitute.
–*n.* **6.** the act of supplying, furnishing, providing, satisfying, etc.
7. that which is supplied.
8. a quantity of something provided or on hand, as for use; a stock or store.
9. (*usu. pl.*) a provision, stock, or store of food or other things necessary for maintenance.
10. a parliamentary grant or provision of money for the expenses of government.
11. *Econ.* the quantity of a commodity, etc., that is in the market and available for purchase, or that is available for purchase at a particular price.
12. *Elect.* a source of electrical energy.
13. (*pl.*) *Mil.* **a.** articles and materials used by an army or navy of types rapidly used up, such as food, clothing, equipment, and fuel.
b. the furnishing of supplies, and the management of supply units and installations.
14. *Obs.* reinforcements.
15. *Obs.* aid.
–*adj.* **16.** *Elect.* denoting or pertaining to a source of electrical energy or its characteristics.
–**supplier**, *n.*

supply
adv. in a supple manner. Also, **supplely**.

Goth
n. a barbarian; rude person.

bastard
n. **1.** an illegitimate child.
2. something irregular, inferior, spurious, or unusual.
3. *Colloq.* an unpleasant or despicable person.
4. *Colloq.* any person (without pejorative sense).
–*adj.* **5.** illegitimate in birth.
6. spurious; not genuine; false.
7. of abnormal or irregular shape or size; of unusual make or proportions.
8. having the appearance of; resembling in some degree: *bastard box*
9. a bastard of a thing, *Colloq.* a terrible thing.
10. happy as a bastard on Father's Day, *Colloq.* extremely pleased.

blackguard
n. **1.** a coarse, despicable person; a scoundrel.
–*v.t.* **2.** to revile in scurrilous language.
–*v.i.* **3.** to behave like a blackguard.
–**blackguardism**, *n.*

churl
n. **1.** a peasant; a rustic.
2. a rude, boorish, or surly person.
3. a niggard; miser.
4. *Eng. Hist.* a freeman of the lowest rank.

gremmie
n. Colloq a surfboard rider whose behaviour in the surf is objectionable. Also, **gremmy**.

ocker
n. Colloq **1.** the archetypal uncultivated Australian working man.
2. a boorish, uncouth, chauvinistic Australian.
3. an Australian male displaying qualities considered to be typically Australian, as good humour, helpfulness, and resourcefulness.
–*adj.* **4.** of or pertaining to an ocker.
5. distinctively Australian: *an ocker sense of humour*. Also, **okker**.
–**ockerish**, *adj.*

roughie
n. → **tommy ruff.**
roughie
n. Colloq. **1.** one who is rough or crude.
2. a shrewd trick; a cunning act.
3. *Horseracing.* a horse with little chance of ever winning a race.

vulgarian
n. a vulgar person, esp. one whose vulgarity is the more conspicuous for his wealth, prominence, or pretensions to good breeding.

yahoo
n. **1.** a rough, coarse, or uncouth person.
–*v.i.* **2.** to behave in a rough, uncouth manner (fol. by *around*).
–*interj.* **3.** (an exclamation expressing enthusiasm or delight).

11. Appendix B

> *Had I been present at the Creation, I would have given some useful hints for the better ordering of the universe.*
>
> *Alfonso The Wise (1221−1284)*

The Basser Department of Computer Science received a tape from Macquarie Library Pty Ltd containing The Macquarie Dictionary and The Macquarie Thesaurus. This section describes the format of the data files provided on the tape, with reference to converting them into a database. Some of the information presented here comes directly from the 'Explanatory Notes' in The Budget Macquarie Dictionary.

11.1 Dictionary

The dictionary was supplied in 26 files, one for each letter of the alphabet. The data in each file consisted of a large number of '|' delimited fields. A sample of the data with each '|' translated to a new-line is given below. Each file ends with '|' followed by a number of spaces.

```
\H allegiance
\P /i[/f/f7'lid57ns/i]
\D1 /in. /1 /nthe obligation of a subject or citizen to his
sovereign or government; duty owed to a sovereign or state.
\D2 /1 /nobservance of obligation; faithfulness to any person or
thing.
\E [ME /ialegeaunce /n(with /ia/n- of obscure orig.), from OF
/iligeance. /nSee /sliege/n]
\H allegiant
\P /i[/f/f7'lid57nt/i]
\D1 /iadj. /nloyal.
\H allegorical
\P /i[/f/f2l7'g6r1k7l/i]
\D1 /iadj. /nconsisting of or pertaining to allegory; of the nature
of or containing allegory; figurative: /ian allegorical poem/n,
/imeaning/n, /ietc. /nAlso, /ballegoric.
\R @/ballegorically, /iadv.
\H allegorise
\P /i[/f/f'2l7g7ra1z/i],
\S /iv., /b-rised, -rising.
\D1 @/iv.t. /1 /nto turn into allegory; narrate in allegory.
\D2 /1 /nto understand in an allegorical sense; interpret
allegorically.
\D3 @/iv.i. /1 /nto use allegory. Also, /ballegorize.
/n@/ballegorisation/n, /i[/f/f.2l7g7ra1'ze147n/i], /in.
@/ballegorise, /in.
```

11.1.1 Record types

A record consists of a line starting with '\' and all following lines that do not start with '\'. The symbols at the beginning of each record are explained in Table 12:

Symbol	Meaning
\H	Headword The headword is the word or words which are being defined in a particular entry. Separate entries are made for all words which, though spelt identically, are of quite distinct derivation; in such cases, each headword is followed by a small superscript number.
	Pronunciation The pronunciation is given in the International Phonetic Alphabet. For some headwords more than one pronunciation is given, the first of these being the one more widely used.
\S	Inflected forms If a headword has irregularly inflected forms (any form not made by the simple addition of the suffix to the main entry) a summary of these forms is given immediately after the pronunciation. The regular forms are given, however, when necessary for clarity or to avoid confusion.
	Definition *n* Definitions are individually numbered; numbers appear in a single sequence which does not begin afresh with each grammatical form. In some cases where two definitions are very closely related, usually within the same field of information, they are marked with bold-face letters of the alphabet under the same definition number. The range of *n* is $1 \leq n \leq 99$. Definitions numbered greater than 99 are handled via the use of split entries. See below for the section on split entries.
\E	Etymology Etymologies appear in square brackets after the definition or definitions of the entry.
	Run-on headwords Words which are derivatives of the headword and are a simple extension of the meaning are run on after the etymology, or (if there is no etymology) after the last definition in the entry. Such headwords appear in secondary bold-face, followed by an indication of their grammatical form.

TABLE 12. Record types

11.1.2 Headwords

Some headwords are followed by a comma. These commas are an accident of history and are removed from the the headword before it is made into a key. However, the comma is not removed from the text of the entry.

11.1.3 Secondary headwords

Idiomatic phrases, prepositional verb phrases, etc. are usually listed in secondary bold face alphabetically under main headwords. Such entries are usually placed under the word thought to be most frequently used to find the entry. Where a secondary headword has more than one meaning, the various meanings are listed after bold-face letters of the alphabet.

The following regular expressions can be used to find most of the secondary headwords (the secondary headword will be matched by the part of the regular expression between the parentheses):

```
\|\\D[1-9][0-9]* /[1A] +([^>/][^,]*)
  $p\|\\D[1-9][0-9]* /b([^>][^,]*)
```

11.1.4 Run-on headwords

Run-on headwords may be found by applying the following regular expression to the run-on field of each entry:

```
(/b[@ ]*|, )([a-z][^[,/.]*[^,/ .])
```

The run-on headword is matched by the part of the regular expression between the second set of brackets. The regular expression is applied repeatedly until no more run-on headwords can be found.

11.1.5 Inflected forms

Finding inflected forms is more difficult than finding run-ons and secondary headwords. In many cases all that is supplied is a suffix to be merged with the headword rather than the whole inflected form. This makes it necessary to determine the complete inflected form using some rule.

11.1.5.1 Finding inflected forms

Inflected forms (or the suffixes used to form them) are found by applying the two following regular expressions:

```
  –?/b([^:;,/]*[^:;.,/ ])
/b[^/]*, ([^:;,/]*[^:;.,/ ])
```

to any inflected form record that starts with '`/i`'. (The inflected form will be matched by that part of the regular expression in parenthesis.) The first regular expression finds the inflected forms that directly follow a change to bold font and the second finds those inflected forms that follow a bold comma (some care needs to be taken here as not all commas are printed in bold, for example *about-face*). These regular expressions fail on the word *corroborate* because no comma separates the two inflected forms listed for that word.

A suffix is distinguished from the complete inflected form because the first letter of the inflected form is a hyphen or, in the case of the first regular expression, the whole piece of text matched starts with a hyphen. The second part of the above rule is necessary because a few definitions have the hyphen printed in normal font rather than bold, for example *accelerate*. The algorithm in the following section produces the complete inflected form from a suffix and the headword.

11.1.5.2 Joining suffixes

The method used to join a suffix to its headword involves overlapping them in such a way that the maximum number of characters match. Should there be a tie in the number of matching characters the rightmost overlap is used. Words with zero matches are regarded as errors and rejected. For example, when joining *disorientate* and *-tated* two positions have non-zero match counts; one has four matches (when the *tated* is over the *tate*), and one has one match (when *tated* is over the *te*). The overlap with four matches is chosen, giving *disorientated*.

This rule is modified by the following: *y* at the end of a headword will match any *i* in the suffix and the first character of the suffix must be matched in the headword. For the first part of this modification to work correctly it is necessary to remove any superscripts from the end of the headword.

A final modification is: matching starts at the second character of the headword, so that the suffix cannot replace the inflected form completely. Without this modification joining *caecum* and *-ca* would result in *ca* rather than *caeca*. Several of the special cases listed in Table 13 result from this modification. However, these cases can be regarded as errors in the dictionary.

The lists of inflected forms using this rule produced some duplicates; some words were keys to the same definition more than once. This occurs because one of the inflected forms is the same as the headword, for example the plural of *aircraft* is *aircraft*, or because the same inflected form is listed twice, for example *backbite* and *-bit*. This problem was solved by removing duplicates from the key list for each definition.

An examination of the first 25% of the words produced by this rule found some inflected forms that were not correctly handled. These words, and some others rejected by the matching algorithm as having zero matches, were placed in a table and handled as special cases.

Headword	Suffix	Inflected form
almsman	-woman	almswoman
almsman	-women	almswomen
cooee	-cooeed	cooeed
cooee	-cooeeing	cooeeing
cyesis	-es	cyeses
forego	-went	forewent
forgo	-went	forwent
gar	-gars	gars
money	-ies	monies
outgo	-went	outwent
rev	-revved	revved
undergo	-went	underwent
use	-used	used
use	-using	using
vexillum	-vexilla	vexilla

TABLE 13. Special cases

The presence of these special cases causes a loss of confidence in the overall success of the entire rule. However the above cases could not be handled without including special cases, such as *went* matches *go*.

A slightly different rule was used as an experiment. This rule was the same as the above except that matches were counted from the left only until the first mismatch. The experiment was abandoned as *basis* and *-ses* became *basises*.

11.1.6 Split entries

Some entries are split into two or more sections. These split entries may be identified by two methods:

1. searching for adjacent sections with exactly the same headword (including any superscripts). These entries are joined as though the second '`\H`' was not present. The definition count in the second section should follow on from the first and no other fields are repeated. An example of this type of continuation, from the definition of *all*, is given below:

```
\D7 /1 /nthe whole number: /iall of us.
\D8 /1 /neverything: /iis that all/n?
\D9 @/in. /1 /na whole; a totality of things or qualities.
\H all
\D10 /1 /none's whole interest, concern, or property: /ito give/n,
/ior lose one/n'/is all.
\D11 /1 /nSome special noun phrases are:$p
\D12 /babove all, /nbefore everything else.$p
\D13 /bafter all, /3/2 /nafter everything has been considered; not
withstanding.
```

2. looking for entries with run-on '`$xxx`' or '`$XXX`'. These indicate that the following entry will have headword '`XXCONTINUED`' and should be joined to the

current entry. The '\P' and '\S' fields may be repeated and should be ignored.

This type of continuation occurs in the definition of the word *run*, a sample of which is given below:

```
\D80 /1 /n(of a newspaper) to publish (a story).
\D81 /1 /iU.S. /nto put up (a candidate) for election.
\D82 /1 /nto melt, fuse, or smelt, as ore.
\D83 /1 /n> /bsmuggle.
\D84 /1 run a book, /iColloq. /nto accept bets.
\R $xxx
\H XXCONTINUED
\P /i[/f/frxn/i]
\S /iv., /bran, run, running, /in., adj.
\D1 @/iv. /1 /nSome special verb phrases are:$p
\D2 /bcut and run, /nto take to flight.$p
\D3 /brun across, /nto meet or find unexpectedly.$p
\D4 /brun after, /nto seek to attract.$p
```

Notice that the definition count starts from 1 again although in the printed dictionary the definition counters are unaffected by this break. Further examples of this type of continuation are to be found in the definitions of *get*, *go* and *set*.

The above two types of continuation may both occur in the same entry as in the definition of the word *run*. In this case after a split entry of the second type the repeated headword for a split entry of the first type will be '`XXCONTINUED`', not '`run`'. For example:

```
\D28 /brun into, /3/2 /nto encounter unexpectedly.
\D29 /2 /nto collide with.
\D30 /2 /nto amount to: /ian income running into five figures.$p
\D31 /brun off, /3/2 /nto depart or retreat quickly.
\H XXCONTINUED
\D32 /2 /nto produce by a printing or similar process.
\D33 /2 /nto write or otherwise create quickly.
\D34 /2 /nto steal (fol. by /iwith/n).
```

11.1.7 Special ASCII characters

Some ASCII characters are used for special purposes in the text; others are printed unaltered. Table 14 lists some of these characters and their uses:

ASCII	Use
@	an en rule ‘−’
>	an arrow ‘→’
#	small space
"	an open quote: ‘
´	a close quote: ’
?	a question mark ‘?’
%	a space, see *catalectic*
$f6%$f1	‘%’ symbol in font 6, used in *agent purple* def. 1
*	an asterisk ‘*’
&	‘&’, used in *amphibious* def. 1 and def. 2
=	only used in *hyperbolic functions* def. 1 as a superscripted minus.
<	a space, see *Chinese red*
$f6<$f1	‘’ symbol in font 6, used in *ape* def. 5.
{	occurs once, in error (should be [), in *sirdar*; this word is not included in the 1981 edition
}	not used
^	not used
_	not used
|	end-of-line marker
\	only appears in column 1 as prefix to record type

TABLE 14. Special ASCII characters

11.1.8 Inline symbols

Various symbols occur throughout the text of the entries to indicate font changes, special characters, etc. These symbols are listed in Table 15. Note that some of these symbols are just typesetting information and can be ignored for lexicographic purposes.

Symbol	Meaning
/n	light roman text
/b	bold roman text
/s	use small capitals
/i	light italic text
/f	phonetic characters follow (often used redundantly)
$p	a paragraph follows
$xh	unknown, for an example see *assets*, def. 4
$xn	unknown, for an example see *fashion*
$[	unknown, for an example see *charge*, def. 29
$dh	unknown, for an example see *assets*
$xxx	only appears in runons, see above under Split Entries
$XXX	only appears in runons, see above under Split Entries
$f1	light roman text, for an example see *agent purple*
$f3	light italic text, for an example see *ab-*
$f6	special font, for an example see *slice* def. 5a
$sn	unknown, for an example see *fashion*
$c]	appears in *metric system* table only
$tml3	appears in *metric system* table only
$gi0]	appears in *metric system* table only
$j	unknown, for an example see *amphibology*
$w.*n*]	unknown, for an example see *amphibology*
$i1*s*]	print the string *s* as a subscript
$I1*s*]	print the string *s* as a subscript
$s1*s*]	print the string *s* as a superscript
$S1*s*]	print the string *s* as a superscript
$ac*ddd*]	accent the preceding character with accent character *ddd*
$dp*ddd*]	special character *ddd*
$DP*ddd*]	special character *ddd*

TABLE 15. Inline symbols

The definition numbers and letters for the dictionary entries are controlled and printed by the following inline symbols:

Symbol	Meaning
/1	Definition Counter: starts at zero for each headword; '/1' increments by 1 and prints the definition number followed by a '.' in bold.
/A	Alphabetic Definition Counter: '/1' resets to 'a'; '/A' increments by 1 and prints the counter as a lower case letter followed by a '.' in bold.
/2	Secondary Definition Counter: '/3' resets to 0; '/2' increments by 1 and prints the definition number followed by a '.' in bold.
/3	See '/2' above. This symbol does not result in any characters being printed.

11.1.9 Special characters

Special characters such as mathematical symbols and the Greek alphabet are stored in the text as '$dp*ddd*]' or as '$DP*ddd*]' where the digits *ddd* identify the particular character. Table 16 contains the codes, the number of occurrences in the text, the closest *troff*(1) equivalent and any ASCII approximation of each of the special characters used.

Code	Count	Char	ASCII	Code	Count	Char	ASCII	Code	Count	Char	ASCII
013	32	½		212	31	•		288	1		
014	11	¼		213	1	○		290	2	Γ	
017	2	⅓		228	32	/	/	291	4	Δ	
019	2	⅛		229	3	/	/	292	1	Φ	
030	4	1⁄16		230	1	{	{	293	2	Λ	
031	1	⅙		231	1	}	}	294	1	Ξ	
035	1	½		249	1	#	#	295	3	Π	
036	1	¼		251	4	†		296	5	Σ	
054	1	3		252	2	‡		297	2	ϒ	
055	3	4		255	15	*	*	299	1	Ψ	
130	33	+		257	1	⊥		300	2	Ω	
131	26	−		258	1	§		303	3	´	´
133	1	=		260	1	α		304	2	`	`
136	24			261	3	β		305	1	ˆ	ˆ
137	16	´		262	1	γ		306	1		
139	9	+	+	263	1	δ		307	204	°	
140	95	×	x	266	1	ζ		308	1		
141	33	–	−	267	1	η		309	1		
142	107	=	=	268	1	θ		310	11		
143	1	±		270	1	ι		312	11	¨	
145	3	÷		272	4	χ		315	1	¨	
147	3	<	<	273	1	λ		335	3	¦	¦
148	4	>	>	274	5	μ		336	1	¦	¦
149	131	·	.	275	1	ν		341	3	‖	
155	2	+	+	276	1	ξ		351	3	-	−
156	2	–	−	277	8	π		384	1		
157	2	×	x	278	2	ρ		416	1	β	
158	1	=	=	279	1	σ		425	1	∫	
164	4	*	*	280	1	ς		430	2		
165	21	*	*	281	1	τ		472	1		
168	27			282	1	υ		486	1		
170	38	´	´	283	1	φ		522	1	‘	
171	10	"	"	284	1			536	1		
174	1			285	2	χ		990	479	æ	
188	11	√		286	1	ψ		991	10	œ	
192	1	¯		287	1	ω		999	1		
211	25	•									

TABLE 16. Special characters

11.1.10 Accents

Accents are indicated by following the character to be accented with ‘`$ac`*ddd*`]`’, where *ddd* identifies the particular accent to use. So ‘`e$ac303]`’ is é. Table 17 lists the code of the accents that occur in the dictionary data files, the number of occurences and the closest *troff*(1) approximation.

Code	Count	Char	Code	Count	Char	Code	Count	Char
073	1	²	228	1	/	310	12	
074	1	³	301	57	¸	311	5	˙
130	2	⁺	303	4186	ˊ	312	12285	˚
131	1	ˍ	304	231	`	315	4	˚
132	1	×	305	460	ˆ	318	20	
142	3	=	306	317	ˍ	335	1	⎮
188	3	√	307	12	°	351	1	-
192	7	¯	308	98	¯	355	1	¯
194	1	ˍ	309	5		523	2	ˮ

TABLE 17. Accents

11.1.11 Tables

There is a single table in the dictionary under the definition of *metric system*. The text for this table is listed below:

```
\H metric system
\P /i[/f/f'm8tr1k s1st7m/i]
Decimal multiples and submultiples of SI units may be formed by
means of the following prefixes:$p
\H metric system
\D2
$gi0]$tml3.6,C3,C3,L3,C3,C3]$c]$f3$w.12]Prefix$c]Symbol$c]Factor$c]Pre
fix$c]Symbol$c]Factor$j$c]$f1$w.12]deka$c]da$c]10$c]%deci$c]d$c]10$s1-
1]$j$c]$w.12]hecto$c]h$c]10$s12]$c]%centi$c]c$c]10$s1-2]$j$c]$w.12]kil
o$c]k$c]10$s13]$c]%milli$c]m$c]10$s1-3]$j$c]$w.12]mega$c]M$c]10$s16]$c
]%micro$c]$dp274]$c]10$s1-6]$j$c]$w.12]giga$c]G$c]10$s19]$c]%nano$c]n$
c]10$s1-9]$j$c]$w.12]tera$c]T$c]10$s112]$c]%pico$c]p$c]10$s1-12]$j$c]$
w.12]peta$c]P$c]10$s115]$c]%femto$c]f$c]10$s1-15]$j$c]$w.12]exa$c]E$c]
10$s118]$c]%atto$c]a$c]10$s1-18]$j
\D3 /nThus 1#000#000 watts (10$s16]W) may be expressed as 1 megawatt
(1 MW). Each SI unit name and each prefix has an internationally
uniform symbol associated with it.
```

11.2 Database structure

The dictionary is in two separate files. The first is a database that uses Bruce Ellis's database software. This database consists of only one type of entry with the following structure:

```
keyword
{
                    kword key all.
                    addr.
}
```

This database contains all the keys in the dictionary; headword, secondary headword, inflected forms and run-ons. The second file contains the dictionary definitions in the following format:

```
definition
{
        length
        headword
        inflected_forms
        {
                definitions
        }
        etymology
        run-ons
}
```

The definition file contains the dictionary definitions one after another in the same order as they appeared in the original 26 files. Each keyword entry in the key database contains in the kword field a key for the definition and in the addr field the location or *seek* address of the corresponding definition. For example the word *condone* will have a record in the key database with kword equal to *condone* and addr the address of the definition of *condone*. *condoned*, which is an inflected form of *condone*, will also have a record in the key database although its addr field will contain the same value as that for *condone*. Each definition is in the file only once with possibly multiple keys pointing to it.

Each definition in the definition file starts with a four byte length field. This is a four digit hexadecimal number which is the length in bytes of the entire definition exclusive of the length field. The other fields in the definition record are all ASCII strings which contain the various fields of the definition. Each of the fields is followed by an ASCII `nul` and each individual definition (sense) is followed by a newline character.

C routines have been written to read these definitions from the definitions file and place the various fields in a C structure for easy manipulation. The sequential nature of the file also makes it possible to process each of the dictionary definitions in alphabetical order, something not possible with the database of keywords as the database software does not allow the sequential access of each of the records in the database. The only possibility is to retrieve all the keyword entries at once but this will exceed most users' memory limits.

The two files that hold The Macquarie Dictionary have the following sizes in bytes:

dict_key	11 235 840
dict_entry	15 955 804

While this database structure is very compact and easy to use for simple queries such as looking up the definition of a word, it does cause problems when put to more sophisticated uses. These problems are not due to the database losing information that is readily accessible in the raw files, but rather due to not extracting all that can be extracted before the typesetting information is removed. The types of information that should be extracted and included in the database are:

- the type of each key; whether headword, run-on etc
- the part of speech of each sense definition
- which senses a secondary headword refers to.

Future implementations of the dictionary database should include this type of information.

11.3 Thesaurus

The thesaurus was supplied in nine files. There are 812 different keywords in the thesaurus, the first ninety nine (0001 to 0099) are in file 1, the next hundred (0100 to 0199) in file 2, etc. A sample of the first file is listed below:

```
0018,n.,02,E,,*delegation,,,
0018,n.,02,E,,deputation,,,
0018,n.,02,E,,mission,,,
0018,n.,02,E,,{AGENT; AMBASSADOR,,,
0018,n.,03,A,,*ambassador,,,
0018,n.,03,A,,ambassador extraordinary,,Y,
0018,n.,03,A,,ambassador plenipotentiary,,Y,
0018,n.,03,A,U.S.,ambassador-at-large,,Y,
0018,n.,03,A,,ambassadress,,,
0018,n.,03,A,,career diplomat,,,
0018,n.,03,A,,charge$ac303] d'affaires,,,
0018,n.,03,A,,diplomat,,,
0018,n.,03,A,,diplomatist,,,
0018,n.,03,A,,high commissioner,,,
0018,n.,03,A,,minister resident,,Y,
0018,n.,03,A,Brit.,resident,,,
0018,n.,03,B,,*attache$ac303],,,
0018,n.,03,B,,commissioner,,,
0018,n.,03,B,,consular agent,,,
0018,n.,03,B,,cultural attache,,,
0018,n.,03,B,,first secretary,,,
```

The comma-separated fields are:

Field Number(s)	Description
1	Keyword number
2	Part of speech
3	Paragraph number
4	Guide word identifier
5	Label
6	Word
7, 8, 9	flags

The *keyword number* is a four digit number that identifies the keyword associated with the particular entry. The first entry with a given keyword number is usually the keyword for the section.

The *part of speech* is the part of speech of the particular entry. The possible values are:

Abbreviation	Part of speech
n	noun
pron	pronoun
adj	adjective
v	verb
adv	adverb
prep	preposition
conj	conjunction
interj	interjection
phr	phrase

The above list is in the order that the entries appear in the data files. Not all entries have all parts of speech; however all entries have a noun as the first entry.

The *paragraph number* is a two digit integer and provides another level of grouping for the entries. It starts at 1 after each new keyword and is incremented by 1 for each new paragraph. Note the count does not restart from one upon a change in the part of speech. The highest paragraph number is 74.

The *guide word identifier* is the lowest level of grouping provided. The identifier is a single character from a set containing the capital alphabetics and the '#' symbol. The actual character is not significant but the change from one character to another indicates that a new group has commenced. The first word of a new group is called the guide word.

The *label* is a string which indicates some special characteristic of the word being listed.

The *word* is the word currently being listed. In some cases a special character may be used as a prefix. These characters and their meanings are:

Character	Meaning
*	A bold word
{	Pocket edition entry

In most cases a guide word is also a bold word, an exception is:

```
0733,interj.,06,F,,you can stick that for a joke,,,
0733,interj.,06,G,,may your chooks turn to emus,,,
0733,interj.,06,G,,and kick your dunny down,,,
0734,n.,01,A,,*sweetness,,,
```

This exception occurs because the phrase '*may your chooks turn to emus and kick your dunny down*' is longer that 50 characters and so could not be stored in Macquarie Library's database.

A pocket edition entry is an entry that is only to be included in the pocket edition of the thesaurus. An entry with a double '{' is a continuation of the previous pocket edition entry, eg:

```
0089,n.,01,M,,sag,,,
0089,n.,01,M,,{ORGANIC CAVITY; NICHE; CAVE; EXCAVATION;,,,
0089,n.,01,M,,{{INDENTATION,,,
0089,n.,02,A,,*organic cavity,,,
```

Various flags may appear in the last three fields. The field a flag occurs in is irrelevant. The possible flags are:

Flag	Meaning
X	This grouping contains a list of related words
Y	Delete this word from the concise edition
Q	Query this word

11.3.1 Lists

A list is a paragraph of terms which are not synonyms but which are nevertheless closely related. For example the words *lyrebird* and *bellbird* are not synonyms—they are not interchangeable—but they are both indigenous Australian birds. The guide word of a list will be in all capitals and this entry will be printed in italics in the text. An example of a list is:

```
0023,n.,13,A,,*WINE,X,,
0023,n.,13,A,,amontillado,X,Y,
0023,n.,13,A,,asti spumante,X,Y,
0023,n.,13,A,,barolo,X,Y,
0023,n.,13,A,,barsac,X,Y,
0023,n.,13,A,,beaujolais,X,Y,
0023,n.,13,A,,Blanquette,X,,
```

Some lists have the name of the list, ‘WINE’ in the above case, and the first element of the list, ‘amontillado’, joined by a colon. For example:

```
0139,n.,06,D,,quipu,,,
0139,n.,07,G,,*ELEMENTS OF COMPUTATION: addend,X,Y,
0139,n.,07,G,,amount,X,,
0139,n.,07,G,,argument,X,Y,
```

Some lists have two colons, for example:

```
0553,n.,04,B,,Triassic,X,,
0553,n.,04,C,,*CAINOZOIC ERA: - TERTIARY: Palaeocene,X,,
0553,n.,04,C,,Eocene,X,,
0553,n.,04,C,,Miocene,X,Y,
```

To allow a search on the first word on the list, for example ‘addend’, it is necessary to split each entry containing a colon followed by a space into two entries by splitting on the last colon followed by a space in the entry. So the above example will become:

```
0553,n.,04,B,,Triassic,X,,
0553,n.,04,C,,*CAINOZOIC ERA: - TERTIARY,X,,
0553,n.,04,C,,Palaeocene,X,,
0553,n.,04,C,,Eocene,X,,
0553,n.,04,C,,Miocene,X,Y,
```

It is not sufficient to split just on a colon as this would destroy the following entry:

```
0267,n.,04,E,,N:P:K ratio,X,,
```

11.3.2 Special characters and accents

The special characters that occur in the thesaurus are:

Code	Count
228	1
307	2

The accents that occur in the thesaurus are:

Code	Count	Code	Count
301	6	305	27
303	230	306	18
304	64	308	11

All these special characters and accents have the same *troff*(1) and ASCII approximations as in the dictionary.

11.3.3 Missing information

The printed version of the thesaurus (1986 edition) sometimes groups two or more keywords together under a combined heading. For example the keywords *the past*, *present* and *future* are all listed under the combined heading *PAST/PRESENT/FUTURE*. The information required for this extra level of grouping is not provided in the supplied data files.

After each keyword a list of related keywords is provided. For example after the keyword *the past* the following line is printed:

Related Keywords: THE OLD 496; EARLINESS 190; BAD TIMING 321

These lists of related keywords are missing from the data.

11.4 Database structure

The thesaurus is stored in a single database with one kind of record which has the following structure:

```
entry
{
element key all.
label.
id key all.
}
```

The element field contains the word that this entry concerns, label contains any label that is associated with this word and id is a numeric key associated with the entry and describes how it fits into the hierarchy of the thesaurus. This number is composed of the following bitfield and stored in the database in modified hexadecimal:

```
typedef struct
{
unsigned int b_keynum   : 10;
unsigned int b_pos      : 4;
unsigned int b_paranum  : 8;
unsigned int b_subnum   : 9;
unsigned int b_guide    : 1;
}
b_fullnum;
```

The b_keynum field is the keyword number of the entry. The b_pos is a number between zero and nine and identifies the part of speech of the entry. The paragraph number of the entry is stored in the b_paranum field. The b_subnum field corresponds to the guide word identifier of the entry. If the entry is a guide word the b_guide field is 1.

This database structure allows any word to be found by either its string or its position in the thesaurus. The only problem of this database structure is the large number of requests needed to retrieve all the entries in a particular section of the thesaurus. However if this becomes a major problem it can be solved, at the price of extra disk space, by adding extra keys that contain a subset of the information in the id field.

The database containing The Macquarie Thesaurus is 21 309 440 bytes long.

Experiments with machine-readable dictionaries

by

Stuart Leonard Pook

A thesis submitted in partial fulfilment
of the requirements for the degree of
Bachelor of Economics (Honours)

Basser Department
of Computer Science

University of Sydney

December 1987

ABSTRACT

Machine-readable dictionaries are just starting to become available to information scientists and lexicographers. Many other databases of machine-readable text such as news wires, library catalogues and collections of scientific papers are also becoming available as well as increasing in size and coverage. New sets of tools are required to efficiently store and access these new sources of information. This thesis explores some of the ways that machine-readable dictionaries and thesauri can be used in the transmission and retrieval of these new databases.

A piece of text must first be understood before it can be processed by an intelligent retrieval system. The most basic level of understanding is at the word level; the computer needs to understand in which sense the words in the text are being used. Machine-readable dictionaries and thesauri can be used in this task provided they are first stored in suitable databases.

Once the pieces of text have been analysed they must then be classified and transmitted. Then the user of a database must be provided with the tools necessary to retrieve interesting items. Algorithms that can be used to implement new methods of text retrieval and the ways in which they can be used in commercial products are presented.

ACKNOWLEDGEMENTS

The work that produced this thesis, and the many years of study beforehand, would not have been possible without the assistance of my parents. To them must go most of the credit for getting me through my years of study at university. I must thank Jason Catlett, my supervisor, for suggesting the area of research of this thesis and for providing continued support and ideas throughout the year. He assisted me in my efforts to turn a vague area of interest into this thesis. Richard Tardif and Bill Smith, from Macquarie Publishing, provided the original tape containing both The Macquarie Dictionary and The Macquarie Thesaurus, as well as many ideas and criticisms during meetings throughout the year. Bob Amsler, from Bell Communications Research, encouraged me with his interest in my work in the early part of the year and provided useful comments by electronic mail from the United States. All the work required for this project was based on two data bases: one containing The Macquarie Thesaurus; the other containing The Macquarie Dictionary. The database software used was written by Bruce Ellis. His assistance in correcting my somewhat hazy ideas on database structure and in the actual use of his database software is much appreciated. John Mackin, a programmer at Basser, gave me encouragement and advice throughout the year for which I am grateful. Last, but not least, I must thank Joanne Lynton for typing and proofreading, as well as making my year an enjoyable experience.

CONTENTS

1. Introduction\f3 .....\fP 1

2. Literature review\f3.....\fP 3
   - 2.1 Automatic sense disambiguation using machine-readable dictionaries\f3 .....\fP 3
   - 2.2 Why use words to label ideas: the uses of dictionaries and thesauri in information retrieval\f3 .....\fP 4
   - 2.3 What use are machine-readable dictionaries? A summary of the ‘‘Automating the lexicon’’ workshop\f3 .....\fP 5
   - 2.4 Typesetting from a dictionary database\f3 .....\fP 5
   - 2.5 Information in data: using the Oxford English Dictionary on a computer\f3 .....\fP 6
   - 2.6 The use of machine-readable dictionaries in sublanguage analysis\f3 .....\fP 6
   - 2.7 Machine-readable dictionaries\f3.....\fP 8
   - 2.8 Deriving lexical knowledge base entries from existing machine-readable information sources\f3 .....\fP 8

3. Thesaurus to dictionary sense mapping\f3.....\fP 10
   - 3.1 The purpose of the algorithm\f3 .....\fP 10
   - 3.2 Testing\f3 .....\fP 11
   - 3.3 The algorithm\f3 .....\fP 18
   - 3.4 Simplest method\f3 .....\fP 18
   - 3.5 Variations\f3.....\fP 23
   - 3.6 Equal definitions\f3.....\fP 23
   - 3.7 Simultaneous determination\f3 .....\fP 25
   - 3.8 Length weighting\f3.....\fP 27
   - 3.9 Conclusions\f3 .....\fP 27

4. Dictionary to thesaurus sense matching\f3 .....\fP 28
   - 4.1 Example\f3.....\fP 28
   - 4.2 The algorithm\f3 .....\fP 30
   - 4.3 Testing\f3 .....\fP 31

5. Sense disambiguation\f3 .....\fP 32
   - 5.1 The algorithm\f3 .....\fP 32
   - 5.2 Testing\f3 .....\fP 33
   - 5.3 Conclusion\f3.....\fP 35

6. Text retrieval\f3.....\fP 36
   - 6.1 Text retrieval by keyword\f3 .....\fP 36
   - 6.2 Text retrieval by classification\f3 .....\fP 38
   - 6.3 Conclusion\f3.....\fP 40

7. Applications\f3 ..... \fP 41
7.1 Thesaurus browser\f3 ..... \fP 41
7.2 News wire retrieval\f3 ..... \fP 42

8. Conclusion\f3 ..... \fP 44

9. Bibliography\f3 ..... \fP 45

10. Appendix A\f3 ..... \fP 47

11. Appendix B\f3 ..... \fP 50
11.1 Dictionary\f3 ..... \fP 50
11.2 Database structure\f3 ..... \fP 59
11.3 Thesaurus\f3 ..... \fP 61
11.4 Database structure\f3 ..... \fP 64

LIST OF TABLES

TABLE 1. Dictionary sizes\f3.....\fP 4
TABLE 2. Subject codes\f3 .....\fP 7
TABLE 3. Short test file\f3 .....\fP 12
TABLE 4. Sample of Macquarie test data\f3.....\fP 13
TABLE 5. Function words\f3.....\fP 19
TABLE 6. Results from *juice* example\f3.....\fP 19
TABLE 7. Results from *boor* example\f3.....\fP 20
TABLE 8. Results from *qualm* example\f3 .....\fP 21
TABLE 9. First results\f3.....\fP 22
TABLE 10. Summary of first results\f3.....\fP 22
TABLE 11. New function word list\f3.....\fP 33
TABLE 12. Record types\f3.....\fP 51
TABLE 13. Special cases\f3 .....\fP 54
TABLE 14. Special ASCII characters\f3.....\fP 56
TABLE 15. Inline symbols\f3.....\fP 57
TABLE 16. Special characters\f3.....\fP 58
TABLE 17. Accents\f3.....\fP 59