A Simple Syntactic Approach for the Generation of Indexing Phrases*

Gerard Salton
Zhongnan Zhao
Chris Buckley

TR 90-1137
July 1990

Department of Computer Science
Cornell University
Ithaca, NY 14853-7501

*Department of Computer Science, Cornell University, Ithaca, NY 14853-7501. This study was supported in part by the National Science Foundation under Grant IRI-87-02735.

A Simple Syntactic Approach for the Generation of Indexing Phrases

Gerard Salton, Zhongnan Zhao, and Chris Buckley *

July 26, 1990

Abstract

A syntactic approach is described for generating indexing phrases usable for the content identification of natural-language texts. The phrase generation method is based on a simple language analysis system that determines the syntactic function of individual text words with a high degree of accuracy, and chooses of indexing phrases based on weights assigned to the phrase components. The proportion of phrases that appear to be acceptable for content identification ranges from 96 to 98 percent.

1 Introduction

Many different language analysis procedures have been proposed over the years to control the assignment of index terms to documents stored in information retrieval systems. A standard method consists in using sets of properly weighted single terms for content representation, the term weights being used to distinguish the more important from the less important terms [1-3]. More refined methods may be based in the use of preconstructed thesauruses designed to recognize synonymous or other similarity relations between terms [4,5], and on the construction of term phrases consisting of combinations of single terms.[6,7] Still other content analysis approaches are based on the construction of so-called knowledge bases which provide complete semantic characterizations of the entities of interest in a particular domain. Using such knowledge bases, an attempt is then made to carry out a deep semantic analysis of a text before assigning content identifiers.[8-14]

The construction of thesauruses and knowledge bases specifying the relevant semantic environment raises very substantial conceptual and practical problems

*Department of Computer Science, Cornell University, Ithaca, NY 14853-7501. This study was supported in part by the National Science Foundation under grant IRI 87 02735

when the subject area of interest is not severely circumscribed. The most immediately usable approach to an enhancement of single term indexing strategies then consists in the construction of *term phrases* to supplement the single term indexing products. Term phrases are sets of single terms that collectively carry meaning and represent more refined entities than the individual term components. For example, "computer science" represents a concept quite apart from that of "computer", or "science", alone.

Term phrases can be generated in many different ways, for example by using statistical term co-occurrence methods where phrases are defined as two or more single terms that occur frequently in close proximity to each other in the texts of a document collection.[6, 7, 15]. Alternatively, a simple syntactic tag assignment may be attempted based on a dictionary search that identifies each text word, as being a noun, an adjective, an adverb, and so on. A phrase may then be defined as a particular word sequence with specified sequences of assigned syntactic tags – for example, noun-noun or adjective-noun sequences.[16, 17] A still more refined process consists in carrying out a complete syntactic analysis of a text, producing one or more syntactic parse trees for each text sentence. Phrases can then be constructed from particular text words that are related within the syntactic tree structures.[18-23]

Most phrases generation system are hampered by the fact that many directly relevant subject phrases cannot be generated by using a shallow text analysis without appropriate semantic controls. Furthermore, the quality of the phrases that can actually be assigned is not easily assured. A statistical term co-occurrence process will generate a large number of potential phrases, some of which will necessarily be semantically improper. A syntactic process may reject certain poorly related statistical term combinations, but other problems are than created due to language ambiguities that cannot be resolved by purely syntactic methods. A recent evaluation of the retrieval effectiveness of a statistical phrase generation process compared with that of a syntactic procedure based on the use of the PLNLP syntactic analysis system [24, 25] found that the accuracy of the syntactic phrases reached only about 92 percent in the most favorable circumstances.[26] Overall, a refined statistical phrase generation method appears to be preferable to the syntactic tree construction method used by the PLNLP parser. [27]

In the remainder of this note, a new syntactic analysis system is introduced, and its use is described for the generation of indexing phrases in information retrieval.

2 The Bell Laboratories Syntactic Analysis System

The PLNLP syntactic analyzer developed at the IBM Research Laboratory produced one or more syntactic parse tree for each available text sentence, or text fragment. [24, 25] A typical parse tree is shown in Fig. 1 for the sentence "cryptographic transformations may provide both privacy as well as authentication in communications and message transmission systems". The sample analysis is questionable on several accounts – for example, the prepositional phrase "in communications and message transmission systems" should probably modify the main verb "may provide", rather than the complement "privacy and authentication". Furthermore, the modifier "in communication and message transmission systems" should probably be interpreted as "in communications systems and message transmission systems" so that "communications" should modify "systems", which it certainly does not do in the analysis of Fig. 1.

Despite such uncertainties, a strategy which defines an indexing phrase as a set of mutually dependent noun and/or adjective structures located in the same syntactic sentence produces indexing phrases such as "cryptographic transformations" and "message transmission systems," as well as single terms such as "privacy", "authentication", and "communications".

Because the syntactic function of many words is highly ambigous in many languages – a word such as "base" may represent a noun, an adjective, or a verb in English – and because syntactic sentence structures are uncertain, syntactic analysis systems such as the IBM PLNLP analyzer may produce multiple syntactic analyses for many input sentences. Only two different parse trees are obtained for the sample sentence of Fig. 1. However, 32 distinct analyses are obtained for the input

> "Furthermore, whereas encryption methods were used primarily for government and military communications in earlier years, secrecy transformations are now often applied to business and commercial information, or to personal data pertaining to individuals that may be stored in computer systems or sent over electronic communications lines."

The availability of multiple syntactic analyses complicates the phrase generation process, because it is impossible to distinguish the correct from the extraneous syntactic structures. In these circumstances, one may have to resort to semantic restrictions that may apply to particular subject domains and specific text environments, and the analysis may need to be extended across the boundaries of individual sentences.

Another possibility for reducing the ambiguity of standard syntactic output, that has been used with increasing success in recent years, consists in using accumulated statistics derived from the analysis of large bodies of text to specify the occurrence probabilities of particular sequences of syntactic tags. When particular word sequences carry a multiplicity of syntactic tags, it is then possible to choose that set of syntactic tag assignments corresponding to the most likely tag sequence. For example, if a particular sequence of three words carries the ambiguous tag assignments [adjective, noun], [adjective, noun, transitive verb], [noun, transitive verb], and the sequence "adjective-adjective-noun" is more frequent in the language than alternative interpretations, such as for example "adjective-verb-verb", then an input such as "gray base board" would receive the "gray (adjective) – base (adjective) – board (noun)" interpretation rather than the alternative adjective-verb-verb assignment. By using such statistical approaches for the disambiguation of syntactic tag assignments, analysis systems have been developed that produce only one syntactic interpretation for each text sample, corresponding to the most likely structural interpretation.[28, 29]

One syntactic parsing system recently developed by K.W. Church at ATT Bell Laboratories uses the statistical methodology to produce syntactic tag assignments for ordinary English text. This analysis system also includes a bracketing process designed to identify phrases consisting of noun and adjective sequences. Only a single interpretation is produced for each input fragment.[30, 31] The Bell Laboratories parser is used in this note for the assignment of content phrases to documents, and for the production of global indexes capable of providing access to complete document collections.

An example of the output obtainable by using the Bell Laboratories parser is shown in Fig. 2 for the sentence "Chapter 5 Text Compression The usefulness and efficiency of text processing systems can often be improved greatly by converting normal natural-language text representations into a new form better adapted to computer manipulation". The chosen syntactic tag assignment is shown on the right side of Fig. 2, and the noun phrases are identified by the square brackets. The sample sentence is correctly analyzed except for the initial structure where the phrases "Chapter 5" and "Text Compression" are merged because of a missing period in the original text after "Chapter 5."

An analysis of the syntactic output obtained with the Bell Laboratories parser, covering 50 text-sentences corresponding to four pages of printed text from a standard textbook (pages 131-135 of [32]), indicates that 60 percent of the sentences are error-free. Twenty errors in syntactic interpretation occur in these four printed pages, but only six of them are serious from the point of view of phrase construction. The sixty percent accuracy rate in sentence interpretation obtained with the Bell Laboratories analyzer compares with a 32 percent accuracy rate for the more complex PLNLP analyzer used in earlier studies.[26]

Fig. 3 contains a list of some typical errors made by the Bell Laboratories grammar, and of the corresponding erroneous phrase constructions. Various false syntactic tag assignments occur in examples 1, 3, 7 and 8 ("today" interpreted as a noun, "deciphers" as a plural noun, "place" as a noun, and "set" as a past tense verb). As a result, questionable phrases are produced such as "today secrecy transformations" (instead of "secrecy transformations"), and "a message" (instead of "a message set"). In examples 2 and 4 of Fig. 3, the idiomatic structures "on the other hand", and "vice versa" are not recognized. In example 5, the conjunctive structure "cryptographic enciphering and deciphering operations" is not properly recognized, and in example 6 the difficult punctuation in "Fig. 6.1(a)" causes problems.

The examples of Figs. 2 and 3 show that the bracketing structure obtained with the Bell Laboratories grammar is not directly usable for the production of reliable indexing phrases. Various refinements in the phrase production system are described in the next section.

3 Indexing Phrase Construction

In principle, the syntactic analyzer can be applied to ordinary text without preprocessing phase. When special-purpose texts are analyzed, it is however necessary to make sure that the text segments are properly punctuated. Thus periods, or other appropriate ending marks, must be present after titles, section headings, figure captions, etc., to insure that the phrase bracketing system does not straddle such self-contained units. The example of Fig. 2 illustrates the problems caused by missing punctuation in the input.

Following the bracketing operation illustrated in Figs. 2 and 3, a number of simple post-processing steps may help in producing improved indexing entries:

a) Deletion from the bracketed structures of phrase compounds identified by the following syntactic tags.

```
  i) articles (AT)
 ii) number (CD)
iii) demonstrative pronouns (DT, DTI, DTS, DTX)
 iv) pronouns of various kinds (PPO, PPS, PPLS, PPSS)
  v) qualifiers (QL)
 vi) WH words (who, whose, which, etc.)
```

b) Deletion of adjectives (JJ tag) contained on a special list of deletable adjectives (actual, additional, available, basic, best, complete, corresponding, different, difficult, distinct, and so on).

c) Deletion of phrases derived from idioms contained on a special list of deletable idioms (in the same sense, in that case, in the case of, in principle, on the other hand, vice versa, etc.)

d) Deletion of phrase components consisting of single characters.

e) Deletion of phrases contained in other longer phrase constructions (thus, if "linguistic text element" is present, "linguistic text" and "text element" are not admitted).

Additional refinements can also be introduced such as a limited recognition system for prepositional phrases designed to replace such phrases by indexing units without prepositions. For example, prepositional phrases with "of" can be inverted in some circumstances to generate "text meaning" from "the meaning of a text", and "data confidentiality" from "confidentiality of the data".[26] In addition, a limited type of conjunction analysis for "and" and "or" might be used to generate the phrases "enciphering operation" and "deciphering operation" from input constructions such as "enciphering and deciphering operations".

Two sample text paragraphs, representing the beginning of chapter 5 of [32], labelled I 254 and I 255, respectively, are shown in Fig. 4. Fig. 5 contains the lists of phrases and single terms obtained for the sample documents by the syntactic bracketing system and the previously mentioned post-processing steps. Each indexing entry is followed by the assigned syntactic tag (JJ for adjective, NN for singular noun), and by a frequency indicator giving the number of occurrences of the entry in the document. The indexing phrases listed in Fig. 5 appear to be reasonably reflective of text content. Some of the single terms, on the other hand, could be dispensed with including, "addition", "example", "time", "year", and so on.

Table 1 shows a summary of the indexing products obtained for the complete texts of chapters 5 and 6 of reference [32]. The statistics in the upper part of the Table reflect word occurrences; multiple occurrences of phrases and single terms are listed separately. The lower part of the Table covers distinct single terms and phrases. The percentage figures are *term precision* measures reflecting the proportion of acceptable table entries, that is, the proportion of entries that are appropriate for document content identification.

As the Table shows, the phrase precision is extraordinarily high, reaching 97 percent for the phrase occurrences, and 96 percent for distinct phrases in both chapters 5 and 6. The term precision is much more modest for the single terms: about 50 percent for the terms in Chapter 5 and about 70 percent for those in Chapter 6. The example of Fig. 5 shows that the quality of the single terms varies greatly. Terms such as "secrecy" in document 254, and "redundancy" in document 255, appear directly germane, whereas "chapter", "time", and "year" may be extraneous. When large sets of potential index entries are generated, as in the lists of Table 1, the best policy may consist in eliminating all single term indexing entries, while using only the phrases entries. This reduces the number of distinct indexing entries by a factor of 2 approximately, to a total of 376 and 365 for the two sample chapters. When the single terms are eliminated, some useful terms may be lost, but the term precision rises dramatically.

Instead of removing single terms as a whole, a better policy for the construction of reduced indexing sets might consist in introducing a term weighting system. When term weights are assigned to all potential index terms, all indexing entries that do not include appropriately weighted terms could be eliminated. Fig 6(a) contains a list of highly weighted terms for the sample documents 254 and 255 of Chapter 5. The weight assignment shown next to each term is obtained as the product of the frequency of each term in the document multiplied by the inverse of the collection frequency of the term (tf $\times$ idf weight). [1-3] Such a weight favors terms that occur frequently in individual documents but rarely on the outside. The center portion of Fig. 6 shows the reduced list of indexing entries – both single terms and phrases – that contain at least one highly weighted component.

A further reduction in the size of the index term set is obtained by insisting that *all* term components be highly weighted. The corresponding index term set is shown in the lower portion of Fig. 6 for the two sample documents. The example of Fig. 6 shows that the desirable single term entries "secrecy" and "redundancy" are preserved on the reduced lists because both words are highly-weighted terms.

The summary statistics for indexing entries containing at least one highly weighted term are given in Table 2. Table 3 shows the same information for terms in which all components are highly weighted. A comparison between Tables 1, 2, and 3 indicates that the size of the chosen indexing sets decreases as the criteria for term membership become more restrictive. At the same time, the index term precision is uniformly high, reaching 98 percent for the 84 and 82 phrases chosen for chapters 5 and 6 of [32] when only highly-weighted components are allowed. When both single terms as well phrases are admitted as index terms, the precision reaches 75 percent and 81 percent, respectively, for the terms containing only highly weighted components.

The appendix contains the full set of 277 distinct indexing phrases obtained for chapter 5 when the phrases contain at least one of the 10 most highly weighted terms for each document in a chapter. The phrase precision is 96 percent for the entries in Table 2; the appendix confirms that very few questionable entries (marked by x) are included.

In the earlier study performed with the PLNLP syntactic analyzer [26], various index term sets were generated, based partly on term weight restrictions, and partly on the syntactic tag assignments specified by the syntax. The total number of syntactic phrases consisting of noun-noun and adjective-noun constructions obtained with the PLNLP grammar was 297 and 325 for the two sample chapters, respectively, and the phrase precision was 87.5 percent and 88.6 percent. The size of these phrase sets is directly comparable with the 277 and 270 distinct phrases shown in the "one-in-top-10" output of Table 2. In the latter case, the simpler Bell Laboratories grammar is used and the phrase precision is 96 percent for both chapters. The comparison between the term sets obtained in the two studies confirms earlier results obtained by previous experiments: when general-purpose texts are processed, the simpler linguistic analysis procedures are normally more effective than the more powerful ones.

[15, 27]

When phrase components are restricted to particular subtrees in the analyzed syntactic output, as they are for the PLNLP analyzer, the added conditions create complications that produce uncertain output more often than not. Fig. 7 shows a list of questionable index entries for the two analysis systems used experimentally. These examples show that the more serious error are obtained by the analysis system with greater complexity. Until sophisticated semantic components can be used as part of a language analysis system, it is safer to remain with the conceptionally simpler approaches that tend to be more forgiving for general-purpose texts.

4 References

1. G. Salton, A Blueprint for Automatic Indexing, *ACM SIGIR Forum,* 16:2, Fall 1981, 22-38.

2. G. Salton and M.J. McGill, *Introduction to Modern Information Retrieval,* McGraw Hill Book Company, New York, 1983,

3. G. Salton, C.S. Yang, and A. Wong, A Vector Space Model for Automatic Indexing, *Communications of the ACM,* 18:11, November 1975, 613-620.

4. K. Sparck Jones, *Automatic Keyword Classification for Information Retrieval,* Butterworths, London, 1971.

5. G. Salton, Experiments in Automatic Thesaurus Construction for Information Retrieval, *Information Processing 71,* North Holland Publishing Company, Amsterdam, 1972, 115-123.

6. G. Salton, On the Role of Words and Phrases in Automatic Text Analysis, *Computers and the Humanities,* 10:2, March-April 1976, 69-87.

7. M.E. Lesk, Word-word Associations in Document Retrieval Systems, *American Documentation,* 20:1, January 1969, 27-38.

8. P.S. Jacobs and L.F. Rau, *Natural Language Techniques for Intelligent Information Retrieval,* Proc. of the Eleventh International Conference on Research and Development in Information Retrieval, Y. Chiaramella, Editor, Grenoble, France, June 1988, 85-99.

9. M. Mauldin, J. Carbonell and R. Thomason, *Beyond the Keyword Barrier: Knowledge-Based Information Retrieval,* Proc. 29th Annual Conference of National Federation of Abstracting and Information Services, Elsevier Press, 1987.

10. U. Hahn and U. Reimer, *Informationslinguistische Konzepte der Volltextverarbeitung in TOPIC (Linguistic information concepts in the full text information processing system TOPIC)*, Report TOPIC 2/82, University of Konstanz, Germany, November 1982.

11. R.M. Tong, L.A. Appelbaum, U.N. Askman and J.F. Cunningham, *Conceptual Information Retrieval Using RUBRIC,* Proc. of the Tenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, C.T. Yu and C.J. van Rijsbergen, editors, New Orleans, LA, June 1987, 247-253.

12. D. DeJaco and G. Garbolino, *An Information Retrieval System Based on Artificial Intelligence Techniques,* Proc. of the Ninth International Conference on Research and Development in Information Retrieval, F. Rabitti, editor, Pisa, Italy, September 1986, 214-220.

13. M.F. Bruandet, *Outline of a Knowledge Base Model for an Intelligent Information Retrieval System,* Proc. of the Tenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, C.T. Yu and C.J. van Rijsbergen, editors, New Orleans, LA, June 1987, 33-43.

14. W.B. Croft and D.D. Lewis, *An Approach to Natural Language Processing for Document Retrieval,* Proc. of the Tenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, C.T. Yu and C.J. van Rijsbergen, editors, New Orleans, LA, June 1987, 26-32.

15. J. Fagan, The Effectiveness of a Nonsyntactic Approach to Automatic Phrase Indexing for Document Retrieval, *Journal of the ASIS,* 40:2, March 1989, 115-132.

16. A.H. Klingbiel, Machine Aided Indexing of Technical Literature, *Information Storage and Retrieval,* 9:2, 79-84, and 9:9, 477-494, 1973.

17. M. Dillon and A.S. Gray, Fully Automatic Syntactically Based Indexing System, *Journal of the ASIS,* 34:2, March 1983, 99-108.

18. F.J. Damerau, Automatic Parsing for Content Analysis, *Communications of the ACM,* 13:6, June 1970, 356-360.

19. D.J. Hillman and A.J. Kasarda, The Leader Retrieval Systems, *AFIPS Proceedings,* AFIPS Press, Montvale, NJ, 34, 1969, 447-455.

20. A.F. Smeaton and C.J. van Rijsbergen, Experiments on Incorporating Syntactic Processing of User Queries into a Document Retrieval Strategy, *Proceedings of the Eleventh International Conference on Research and*

Development in Information Retrieval, Y. Chiaramella, editor, Grenoble, France, June 1988, 31-52.

21. Y. Chiaramella, D. Defude, M.F. Bruandet and D. Kerkouba, IOTA: A Full Text Information Retrieval System, *Proc. of the Ninth International Conference on Research and Development in Information Retrieval,* F. Rabitti, editor, Pisa, Italy, September 1986, 207-213.

22. C. Berrut and P. Palmer, Solving Grammatical Ambiguities within a Surface Syntactical Parser for Automatic Indexing, *Proc. of the Ninth International Conference on Research and Development in Information Retrieval.* F. Rabitti, editor, Pisa, Italy, September 1986, 123-130.

23. G. Thurmair, A Common Architecture for Different Text Processing Techniques in an Information Retrieval Environment, *Proc. of the Ninth International Conference on Research and Development in Information Retrieval,* F. Rabitti, editor, Pisa, Itably, September 1986, 138-143.

24. G.E. Heidorn, K. Jensen, L.A. Miller, F.J. Byrd and M.S. Chodorow, The EPISTLE Text Critiquing System, *IBM Systems Journal,* 21:3, 1982, 305-326.

25. K. Jensen, G.E. Heidorn, L.A. Miller and Y. Ravin, Parse Fitting and Prose Fitting: Getting Hold of Ill Formedness, *American Journal of Computational Linguistics,* 9:3-4, July-December 1983, 147-160.

26. G. Salton, C. Buckley, and M. Smith, On the Application of Syntactic Methodologies in Automatic Text Analysis, *Information Processing and Management,* 26:1, 1990, 73-92.

27. J. Fagan, Experiments in Automatic Phrase Indexing for Document Retrieval: An Examination of Syntactic and Nonsyntactic Methods, *Proc. of the Tenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* C.T. Yu and C.J. van Rijsbergen, editors, Association for Computing Machinery, New York, 1987, 91-101.

28. S.J. DeRose, Grammatical Category Disambiguation by Statistical Optimization, *Computational Linguistics,* 14:1, Winter '88, 31-39.

29. R. Garside, G. Leech, and G. Sampson (editors), *The Computational Analysis of English - A Corpus Based Approach,* Longman,London, 1987

30. K. Church, A Stochastictic Parts Program and Noun Phrase Parser for Unrestricted Text, Second Conference for Applied Natural Language Processing, Austin, TX, 1988.

31. K. Church, W. Gale, P. Hanks, and D. Hindle, Parsing, Word Associations and Typical Predicate-Argument Relations, Technical Report, ATT Bell Laboratories, Murray Hill, NJ, 1989.

32. G. Salton, *Automatic Text Processing,* Addison-Wesley Publishing Co., Reading, MA 1989, Chapter 6.

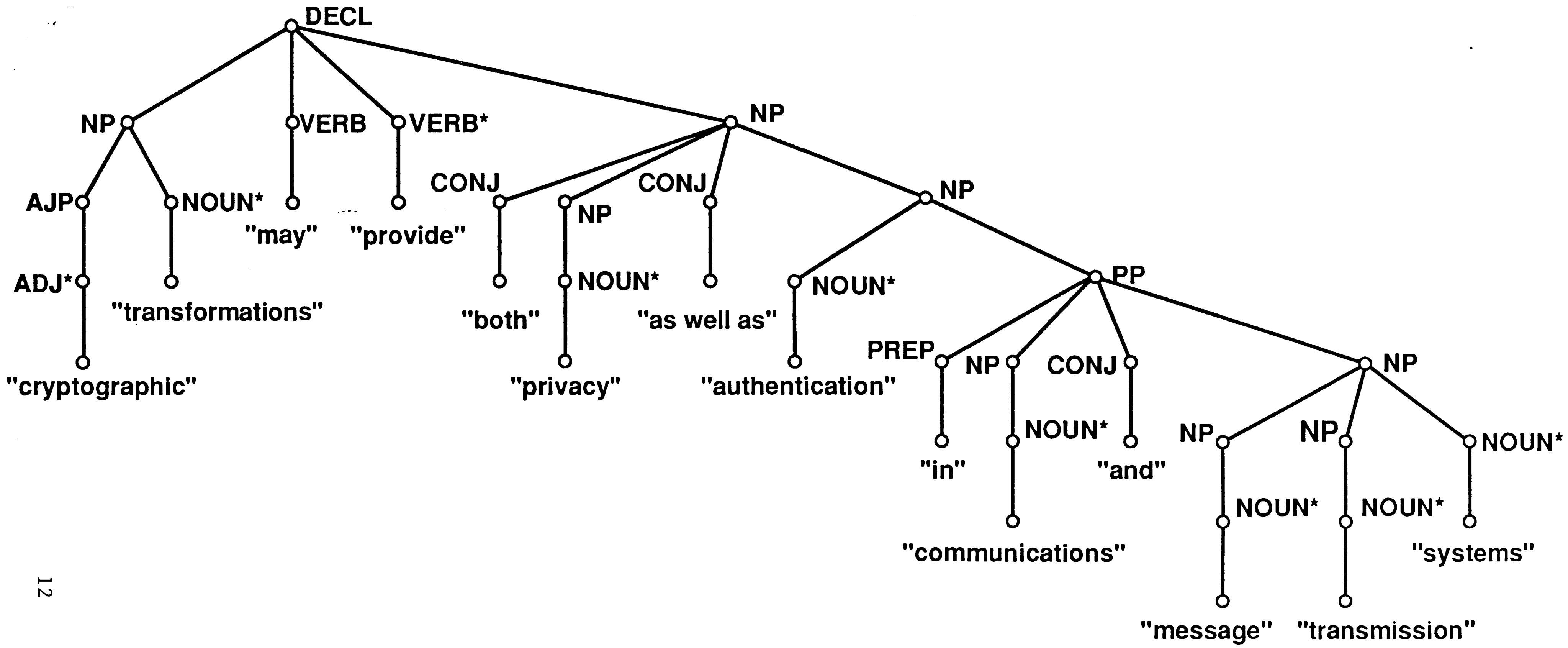

Fig. 1. Parse Tree Produced by PLNLP Grammar for Sample Sentence

("Cryptographic transformation may provide both privacy as well as authentication in communications and message transmission systems")

	Start of Sentence	Syntactic Tags
1	[Chapter 5 Text Compression]	NN/CD/NP/NP
2	[The usefulness]	AT/NN
	and	CC
3	[efficiency]	NN
	of	IN
4	[text-processing systems]	NN/NNS
	can - often - be	MD - RB - BE
	improved - greatly - by - converting	VBN - RB - IN - VBG
5	[normal natural-language text representations]	JJ/NN/NN/NNS
	into	IN
6	[a new form]	AT/JJ/NN
	better - adapted - to	RB - VBN - TOIN
7	[computer manipulation]	NN/NN
	End of Sentence	

Figure 2: Sample Output for Bell Laboratories Grammar

AT: article	NN: singular noun
BE: uninflected form of 'to be'	NNS: plural noun
CC: conjunction	NP: proper noun
CD: number	RB: adverb
IN: preposition	TOIN: 'to'
JJ: adjective	VBG: verb & 'ing'
MD: modal	VBN: verb & 'en'

Sample Error	Phrase		Explanation
1	[today	NN	"today" NN
	secrecy	NN	(wrong syntactic tag)
	transformations]	NNS	
2	on	IN	"the other hand"
	[the	AT	(failure to recognize
	other	JJ	idiom)
	hand]	NN	
3	he	PPS	"he or she"
	or	CC	(lack of conjunctive analysis)
	[she]	PPS	
	[deciphers]	NNS	"deciphers" NNS (wrong syntactic tag)
4	and	CC	"vice" NN
	[vice]	NN	(wrong syntactic tag; failure
	versa	RB	to recognize idiom)
5	[most	QL	"enciphering" VBG
	cryptographic]	JJ	(here interpreted as verb-gerund)
	enciphering	VBG	"most cryptographic" phrase
	and	CC	
	[deciphering	VBG	(phrase produced by VBG label and
	operations]	NNS	failure of conjunctive analysis)
6	[Fig	NP	"a" AT
	6.1]	CD	(here article is wrong syntactic
	[a]	AT	tag; produces phrase "a".)
7	[a	AT	"set" VBN
	message]	NN	(wrong syntatic tag; do not
	set	VBN	get phrase "a message set"
8	The need to transmit		"from place to place
	keys from place		(failure to recognize idiom)
	to	TOIN	
	[place	NN	
	limits	NNS	"limits" NNS
	conventional	JJ	(wrong syntactic tag;
	cryptographic	JJ	should be verb)
	systems]	NNS	(generates phrase "place limits
	severely	RB	conventional cryptographic systems)

Figure 3: Typical Errors in Phrase Formation for Bell Laboratories Grammar

[ ]	assigned phrase boundary	NP	proper noun
AT	article	PPS	subject pronoun
CC	conjunction	QL	qualifier
CD	number	RB	adverb
IN	preposition	TOIN	''to''
JJ	adjective	VBG	verb & ''ing''
NN	singular noun	VBN	verb & ''en''
NNS	plural noun		

saltonbook	Wed Aug 9 15:05:11 1989	Page 50, line 2941

I.254
Chapter 5 Text Compression

The usefulness and efficiency of text-processing systems can often be improved greatly by converting normal natural-language text representations into a new form better adapted to computer manipulation. For example, storage space and processing time are saved in many applications by using short document abstracts, or summaries, instead of full document texts. Alternatively, the texts can be stored and processed in encrypted form, rather than the usual format, to preserve the secrecy of the content.

.I 255
One obvious fact usable in text transformations is the redundancy built into normal natural-language representation. By eliminating redundancies – a method known as text compression – it is often possible to reduce text sizes considerably without any loss of text content. Compression was especially attractive in earlier years, when computers of restricted size and capability were used to manipulate text. Today large disk arrays are usually available, but using short texts and small dictionary sizes saves processing time in addition to storage space and still remains attractive.

Figure 4: Sample Document Texts (document 254, 255 of chapter 5)

Document 254

Phrases	Syntactic Tag	Frequency
computer manipulation	NN/NN/	1
document abstract	NN/NN/	1
full document texts	JJ/NN/NN/	1
natural-language text representation	NN/NN/NN/	1
processing time	NN/NN/	1
storage space	NN/NN/	1
Text Compression	NP/NP/	1
text-processing system	NN/NN/	1

a) Phrase list

Terms	Syntactic Tag	Frequency
application	NN/	1
Chapter	NN/	1
content	NN/	1
efficiency	NN/	1
example	NN/	1
form	NN/	2
format	NN/	1
secrecy	NN/	1
summary	NN/	1
text	NN/	1
usefulness	NN/	1

b) Single Terms

Document 255

Phrases	Syntactic Tag	Frequency
dictionary size	NN/NN/	1
disk array	NN/NN/	1
natural-language representation	NN/NN/	1
storage	NN/NN/	1
text compression	NN/NN/	1
text content	NN/NN/	1
text size	NN/NN/	1
text transformation	NN/NN/	1

a) Phrase list

Terms	Syntactic Tag	Frequency
addition	NN/	1
capability	NN/	1
Compression	NN/	1
computer	NN/	1
factor	NN/	1
los	NN/	1
redundancy	NN/	1
size	NN/	1
text	NN/	2
time	NN/	1
year	NN/	1

b) Single Terms

Figure 5: Phrases and Single Terms for Documents 254 and 255

	Document 254			Document 255			
Term Weight	Term	Term Weight	Term	Term Weight	Term	Term Weight	Term
0.216650	abstract	0.216650	adapt	0.215960	array	0.319460	attract
0.216650	chapt	0.362120	docu	0.215960	disk	0.266340	size
0.216650	format	0.222880	process	0.219060	redund	0.215960	usabl
0.216650	secrec	0.249310	stor	0.195300	text	0.215960	year
0.216650	use	0.181060	usual				

a) Top Term in Documents 254 and 255

Term	Syntactic Tag	Frequency	Term	Syntactic Tag	Frequency
Chapter	NN/	1	dictionary size	NN/NN/	1
document abstract	NN/NN/	1	disk array	NN/NN/	1
format	NN/	1	factor	NN/	1
full document texts	JJ/NN/NN/	1	redundancy	NN/	1
processing time	NN/NN/	1	size	NN/	1
secrecy	NN/	1	text	NN/	2
storage space	NN/NN/	1	text compression	NN/NN/	1
usefulness	NN/	1	text content	NN/NN/	1
			text size	NN/NN/	1
			text transformation	NN/NN/	1
			year	NN/NN/	1

b) Phrases and Single Terms with At Least One Component in Top Terms

Term	Syntactic Tag	Frequency	Term	Syntactic Tag	Frequency
Chapter	NN/	1	disk array	NN/NN/	1
document abstract	NN/NN/	1	redundancy	NN/	1
format	NN/	1	size	NN/	1
secrecy	NN/	1	text	NN/	2
usefulness	NN/	1	text size	NN/NN/	1
			year	NN/	1

c) Phrases and Single Terms with All Components in Top Terms

Figure 6: Top Phrases and Single Terms for Documents 254 and 255

Document	Number	Questionable Index Entries PLNLP Grammar	Questionable Index Entries Bell Laboratories Grammar
281	Chapter 5	produces substantial text text	local area
283	Chapter 5	system designer underestimates	net storage space system designer
296	Chapter 5	character-by-character	
299	Chapter 5	restricted code length code	
300	Chapter 5	half-byte bits standard eight bit	
308	Chapter 5		unchanging occurrence probability
321	Chapter 5	fragments corresponding	efficient method
325	Chapter 6	objects safe cryptography case	
328	Chapter 6	message transmission takes place	main method
330	Chapter 6	receiver enciphering key	
339	Chapter 6	language eliminating	

Figure 7: Sample Questionable Indexing Entries for Two Grammars

	Chapter 5 (5000 words occurrences)	Chapter 6 (7000 words occurrences)
Term Occurrences		
Total number of single terms and phrase occurrences	1481	1605
Proportion of acceptable single term and phrase occurrences	1043 (70%)	1350 (84%)
Total number of single terms occurrences	886	1077
Proportion of acceptable single term occurrences	465 (52%)	838 (78%)
Total number of phrase occurrences	595	527
Proportion of acceptable phrase occurrences	578 (97%)	511 (97%)
Distinct Terms		
Total number of distinct single terms and phrases	626	686
Total number of acceptable single terms and phrases	492 (79%)	575 (84%)
Total number of distinct single terms	250	1321
Proportion of acceptable distinct single terms	131 (52%)	226 (70%)
Total number of distinct phrases	376	365
Proportion of acceptable distinct phrases single terms	361 (96%)	349 (96%)

Table 1: Global Statistics for Single Terms and Phrases in Chapters 5 and 6 of [32]

	Chapter 5	Chapter 6
Term Occurrences		
Total number of single terms and phrase occurrences	876	935
Proportion of acceptable single term and phrase occurrences	702 (80%)	809 (87%)
Total number of single terms occurrences	463	546
Proportion of acceptable single term occurrences	298 (64%)	431 (79%)
Total number of phrase occurrences	413	398
Proportion of acceptable phrase occurrences	404 (98%)	378 (97%)
Distinct Terms		
Total number of distinct single terms and phrases	430	474
Proportion of acceptable distinct single terms and phrases	361 (84%)	407 (86%)
Total number of distinct single terms	153	204
Proportion of acceptable distinct single terms	94 (61%)	148 (73%)
Total number of distinct phrases	277	270
Proportion of acceptable distinct phrases	267 (96%)	259 (96%)

Table 2: Statistics for Single Terms and Phrases with at least One Highly-weighted Component in Chapter 5 and 6 [32]

	Chapter 5	Chapter 6
Term Occurrences		
Total number of single terms and phrase occurrences	606	674
Proportion of acceptable single term and phrase occurrences	454 (75%)	561 (83%)
Total number of single terms occurrences	471	566
Proportion of acceptable single term occurrences	321 (68%)	455 (80%)
Total number of phrase occurrences	135	108
Proportion of acceptable phrase occurrences	133 (99%)	106 (98%)
Distinct Terms		
Total number of distinct single terms and phrases	237	288
Proportion of acceptable distinct single terms and phrases	178 (75%)	232 (81%)
Total number of distinct single terms	153	206
Proportion of acceptable distinct single terms	96 (63%)	152 (74%)
Total number of distinct phrases	84	82
Proportion of acceptable distinct phrases	82 (98%)	80 (98%)

Table 3: Statistics for Single Terms and Phrases with all Highly-weighted Components in Chapter 5 and 6 [32]

Appendix: Syntactic Phrases for Chapter 5
(one term in top 10)

	Phrases	Tag(s)	Frequency
1	additive function	JJ/NN/	1
2	adjacent byte	JJ/NN/	1
3	adjacent character	JJ/NN/	2
4	adjacent entry	JJ/NN/	1
5	adjacent word entry	JJ/NN/NN/	1
6	alphabetic character	JJ/NN/	2
7	alphabetic dictionary files	JJ/NN/NN/	1
8	automatic text-processing application	JJ/NN/NN/	1
9	auxiliary case	JJ/NN/	5
10	auxiliary shift case	JJ/NN/NN/	1
11	average code length	JJ/NN/NN/	5
12	average entropy	JJ/NN/	1
13	average information content	JJ/NN/NN/	1
14	average length	JJ/NN/	1
15	average word length	JJ/NN/NN/	1
16	base case	NN/NN/	4
17	binary code	NN/NN/	1
18	binary digit	JJ/NN/	2
19	bit level	NN/NN/	2
20	bit scan	NN/NN/	1
21	bit string	NN/NN/	2
22	bits character	NN/NN/	1
23	buffer store	NN/NN/	1
24	byte length	NN/NN/	1
25	byte representation	NN/NN/	1
26	byte size	NN/NN/	1
27	byte-length code	NN/NN/	1
28	character code	NN/NN/	1
29	character dependency	NN/NN/	1
30	character dependency factor	NN/NN/NN/	1
31	character level	NN/NN/	1
32	character pair	NN/NN/	5
33	character string	NN/NN/	3
34	character-by-charactbasis basi	JJ/NN/	1
35	circuitous locution	JJ/NN/	1
36	code assignment	NN/NN/	1
37	code combination	NN/NN/	5
38	code combination	NN/NN/	1
39	Code efficiency	NN/NN/	1
40	code increment	NN/NN/	2

	Phrases	Tag(s)	Frequency
41	code length	NN/NN/	10
42	code length increase	NN/NN/NN/	1
43	code portion	NN/NN/	1
44	code table	NN/NN/	1
45	code transformation	NN/NN/	1
46	commercial file	JJ/NN/	1
47	communications theory	NN/NN/	2
48	complex turn	JJ/NN/	1
49	component unit	NN/NN/	1
50	compressed file	JJ/NN/	1
51	compressed form	JJ/NN/	2
52	compression effectivenes	NN/NN/	1
53	compression method	NN/NN/	3
54	compression problem	NN/NN/	1
55	compression ratio	NN/NN/	7
56	compression system	NN/NN/	2
57	compression technique	NN/NN/	2
58	current entry	JJ/NN/	1
59	data compression	NN/NN/	2
60	data record	NN/NN/	1
61	data size	NN/NN/	1
62	Data transmission costs	NN/NN/NN/	1
63	data-compaction	NN/	1
64	Data-compaction system	NN/NN/	1
65	data-compression method	NN/NN/	1
66	decimal digit	NN/NN/	1
67	decimal form	NN/NN/	1
68	decomposition graph	NN/NN/	1
69	dependent character	JJ/NN/	2
70	dictionary file	NN/NN/	1
72	dictionary size	NN/NN/	1
73	differential-coding technique	JJ/NN/	1
74	digit character string	NN/NN/NN/	1
75	disk array	NN/NN/	1
76	document abstract	NN/NN/	1
77	efficient method	JJ/NN/	1
78	eight-bit byte	JJ/NN/	5
79	eight-bit code	JJ/NN/	3
80	English text	NP/NN/	4

	Phrases	Tag(s)	Frequency
81	English word	NP/NN/	4
82	equiprobable character	JJ/NN/	1
83	equivalent compression ratio	JJ/NN/NN	1
84	essential information	JJ/NN/	1
85	even increment	NN/NN/	1
86	even number	JJ/NN/	1
87	file characteristic	NN/NN/	1
88	file merging	NN/NN/	1
89	file size	NN/NN/	1
90	five-bit chunck	JJ/NN/	1
91	five-bit code	JJ/NN/	1
92	Fixed Length Codes	NP/NP/NP/	1
93	fixed-length	NN/	1
94	fixed-length code	NN/NN/	8
95	fixed-length code string	NN/NN/NN/	1
96	fixed-length digram-encoding system	NN/NN/NN/	1
97	fixed length record	NN/NN/	1
98	fragment code	NN/NN/	1
99	fragment occurrence	NN/NN/	1
100	fragment representation	NN/NN/	1
101	fragment-encoding system	JJ/NN/	1
102	fragment-generation method	NN/NN/	1
103	fragment-selection proces	NN/NN/	1
104	frequency characteristic	NN/NN/	1
105	Frequency count	NN/NN/	1
106	frequency order	NN/NN/	2
107	full document texts	JJ/NN/NN/	1
108	full word	JJ/NN/	1
109	full-byte data	JJ/NN/	1
110	George Zipf	NP/NP/	1
111	half byte	NN/NN/	2
112	half-byte code	NN/NN/	1
113	half-byte information	NN/NN/	1
114	high-frequency character	JJ/NN/	1
115	high-frequency function words	JJ/NN/NN/	1
116	high-frequency symbol	JJ/NN/	1
117	high-frequency unit	JJ/NN/	1
118	high-frequency word	JJ/NN/	1
119	high-frequency word combination	JJ/NN/NN/	1
120	Highest-ranking term	NN/NN/	1

	Phrases	Tag(s)	Frequency
121	Huffman code	NP/NN/	4
122	information byte	NN/NN/	1
123	information content	NN/NN/	8
124	initial character	JJ/NN/	2
125	initial character pair	JJ/NN/NN/	1
126	initial clas	JJ/NN/	1
127	initial prefix	JJ/NN/	1
128	integral number	NN/NN/	2
129	irreversible data-compaction method	JJ/NN/NN/	1
130	least effort	JJ/NN/	1
131	left-to-right-scan	JJ/NN/	1
132	length increment	NN/NN/	1
133	length variation	NN/NN/	1
134	Letter combination	NN/NN/	1
135	letter occurrence	NN/NN/	2
136	linguistic redundancy	JJ/NN/	1
137	linguistic tool	JJ/NN/	1
138	logarighmic law	JJ/NN/	1
139	look-up	NN/	1
140	low-frequency word	JJ/NN/	2
141	master character	JJ/NN/	1
142	master character	NN/NN/	5
143	mean number	JJ/NN/	1
144	memory size	NN/NN/	1
145	message receiver	NN/NN/	1
146	message source	NN/NN/	1
147	message text	NN/NN/	1
148	minimal code length	JJ/NN/NN/	1
149	minimum redundancy	JJ/NN/	1
150	most-frequent word	JJ/NN/	1
151	multicase-coding method	NN/NN/	1
152	multicharacter combination	JJ/NN/	1
153	multicharacster string	JJ/NN/	1
154	multicharacter symbol	JJ/NN/	1
155	multicharacster system	JJ/NN/	1
156	multi-case approach	JJ/NN/	1
157	multiword fragment	NN/NN/	1
158	net storage space	JJ/NN/NN/	1
159	nonzero component	NN/NN/	2
160	nonzero digit	NN/NN/	5

	Phrases	Tag(s)	Frequency
161	nonzero element NN/NN/	1	
162	null element	JJ/NN/	1
163	numeric data	JJ/NN/	1
164	numeric digit	JJ/NN/	1
165	numeric information	JJ/NN/	1
166	occurrence probability	NN/NN/	5
167	one-bit prefix	JJ/NN/	1
168	ON-IT-IN unit	NP/NN/	1
169	optimal code length	JJ/NN/NN/	1
170	optimal compression ratio	JJ/NN/NN/	1
171	optimal length	JJ/NN/	1
172	original data length	JJ/NN/NN/	1
173	original text	JJ/NN/	1
174	otherwise-unused code combination	JJ/NN/NN/	1
175	output purpose	NN/NN/	1
176	partial message	NN/NN/	2
177	plain text	JJ/NN/	1
178	principal character dependency	JJ/NN/NN/	1
179	processing capability	NN/NN/	1
180	processing time	NN/NN/	2
181	psycholinguist	NN/	1
182	psycholinguistics	NN/	1
183	rank order	NN/NN/	1
184	rank-frequency formulation	NN/NN/	1
185	rarer word	JJ/NN/	1
186	redundant element	JJ/NN/	1
187	redundant fragment	JJ/NN/	1
188	Restricted Variable-length Codes	NP/NP/NP/	1
189	reverse shift symbol	JJ/NN/NN/	1
190	reverse transformation	JJ/NN/	1
191	run-length	NN/	1
192	sample string subdivision	NN/NN/NN/	1
193	sample word fragment	NN/NN/NN/	1
194	semantic consideration	JJ/NN/	1
195	semantic redundancy	JJ/NN/	2
196	seven-bit code	JJ/NN/	1
197	shift characster	NN/NN/	1
198	shift symbol	NN/NN/	1
199	single byte	JJ/NN/	1
200	single characster	JJ/NN/	4

	Phrases	Tag(s)	Frequency
201	single characster codes	JJ/NN/NN/	1
202	single combination	JJ/NN/	1
203	single-case byte structure	JJ/NN/NN/	1
204	single-character code	JJ/NN/	2
205	single-character fragment	JJ/NN/	1
206	single-character symbol	JJ/NN/	1
207	skewed occurrence probability	JJ/NN/NN/	1
208	sparse record	JJ/NN/	1
209	sparse vector	JJ/NN/	1
210	sparse-vector representation	JJ/NN/	2
211	Special-purpose Compression System	NP/NP/NP	1
212	speech sound	NN/NN/	1
213	standard computer environment	JJ/NN/NN/	1
214	standard utilization	JJ/NN/	1
215	statistical communications theory	JJ/NN/NN/	1
216	statistical component	JJ/NN/	1
217	Statistical Language Characteristic	NP/NP/NP/	1
218	Statistical methodology	JJ/NN/	1
219	statistical redundancy	JJ/NN/	1
220	Storage cost	NN/NN/	1
221	storage space	NN/NN/	1
222	straightforward mode	JJ/NN/	1
223	string character	NN/NN/	3
224	string-decoding method	NN/NN/	1
225	string-decomposition process	NN/NN/	1
226	subsequent characster	JJ/NN/	1
227	suppressed material	JJ/NN/	1
228	suppressed zero	JJ/NN/	1
229	target frequency	NN/NN/	3
230	telegraphic style	JJ/NN/	1
231	temporary shift	JJ/NN/	2
232	temporary shift character	JJ/NN/NN/	1
233	terminal space	JJ/NN/	1
234	Text Compression	NP/NP/	1
235	text compression	NN/NN/	3
236	text content	NN/NN/	1
237	text encryption	NN/NN/	1
238	text fragment	NN/NN/	1
239	text processing	NN/NN/	1
240	text size	NN/NN/	2

	Phrases	Tag(s)	Frequency
241	text string	NN/NN/	1
242	text transformation	NN/NN/	1
243	text word	NN/NN/	6
244	text-processing application	NN/NN/	1
245	text processing system	NN/NN/	1
246	text-transformation system	NN/NN/	1
247	three-bit code	JJ/NN/	1
248	total number	JJ/NN/	1
249	total probability	JJ/NN/	1
250	tree branche	NN/NN/	1
251	Typical value	JJ/NN/	1
252	unchanging occurrence probability	JJ/NN/NN/	1
253	unique code value	JJ/NN/NN/	1
254	usage characteristic	NN/NN/	1
255	variable length	JJ/NN/	4
256	Variable-length Code	NP/NP/	1
257	variable length code	JJ/NN/NN/	1
258	variable-length Huffman code	NN/NN/NN/	1
259	variable-length code	NN/NN/	3
260	variable-length record	NN/NN/	1
261	vector position	NN/NN/	1
262	Vocabulary growth	NN/NN/	1
263	vocabulary growth data	NN/NN/NN/	1
264	vocabulary size	JJ/NN/	1
265	Western Europe	NP/NP/	1
266	word boundary	NN/NN/	1
267	word ending NN/NN/	1	
268	word fragment	NN/NN/	5
269	Word-Fragment Encoding	NP/NP/	1
270	Word-frequency statistic	NN/NN/	1
271	word level	NN/NN/	1
272	word list	NN/NN/	1
273	word occurrence	NN/NN/	8
274	word prefix	NN/NN/	1
275	word-encoding method	JJ/NN/	1
276	word-frequency distribution	NN/NN/	1
277	word-probability distribution	NN/NN/	1
278	Zipf distribution characteristic	NN/NN/NN/	1
279	Zipf law	NN/NN/	1