Applying Collocation Segmentation to the ACL Anthology Reference Corpus

Vidas Daudaravičius
Vytautas Magnus University / Vileikos 8, Lithuania
v.daudaravicius@if.vdu.lt

Abstract

Collocation is a well-known linguistic phenomenon which has a long history of research and use. In this study I employ collocation segmentation to extract terms from the large and complex ACL Anthology Reference Corpus, and also briefly research and describe the history of the ACL. The results of the study show that until 1986, the most significant terms were related to formal/rule based methods. Starting in 1987, terms related to statistical methods became more important. For instance, *language model*, *similarity measure*, *text classification*. In 1990, the terms *Penn Treebank*, *Mutual Information* , *statistical parsing*, *bilingual corpus*, and *dependency tree* became the most important, showing that newly released language resources appeared together with many new research areas in computational linguistics. Although *Penn Treebank* was a significant term only temporarily in the early nineties, the corpus is still used by researchers today. The most recent significant terms are *Bleu score* and *semantic role labeling*. While *machine translation* as a term is significant throughout the ACL ARC corpus, it is not significant for any particular time period. This shows that some terms can be significant globally while remaining insignificant at a local level.

1 Introduction

Collocation is a well-known linguistic phenomenon which has a long history of research and use. The importance of the collocation paradigm shift is raised in the most recent study on collocations (Seretan, 2011). Collocations are a key issue for tasks like natural language parsing and generation, as well as real-life applications such as machine translation, information extraction and retrieval. Collocation phenomena are simple, but hard to employ in real tasks. In this study I introduce collocation segmentation as a language processing method, maintaining simplicity and clarity of use as per the n-gram approach. In the beginning, I study the usage of the terms *collocation* and *segmentation* in the ACL Anthology Reference Corpus (ARC), as well as other related terms such as *word*, *multi-word*, and *n-gram*. To evaluate the ability of collocation segmentation to handle different aspects of collocations, I extract the most significant collocation segments in the ACL ARC. In addition, based on a ranking like that of $TF\text{-}IDF$, I extract terms that are related to different phenomena of natural language analysis and processing. The distribution of these terms in ACL ARC helps to understand the main breakpoints of different research areas across the years. On the other hand, there was no goal to make a thorough study of the methods used by the ACL ARC, as such a task is complex and prohibitively extensive.

2 ACL Anthology Reference Corpus

This study uses the ACL ARC version 20090501. The first step was to clean and preprocess the corpus. First of all, files that were unsuitable for the analysis were removed. These were texts containing characters with no clear word boundaries, i.e., each character was separated from the next by whitespace. This problem is related to the extraction of text from .pdf

Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries, pages 66–75,
Jeju, Republic of Korea, 10 July 2012.

format files and is hard to solve. Each file in the ACL ARC represents a single printed page. The file name encodes the document ID and page number, e.g., the file name *C04-1001_0007.txt* is made up of four parts: *C* is the publication type, *(20)04* is the year, *1001* is the document ID, and *0007* is the page number. The next step was to compile files of the same paper into a single document. Also, headers and footers that appear on each document page were removed, though they were not always easily recognized and, therefore, some of them remained. A few simple rules were then applied to remove line breaks, thus keeping each paragraph on a single line. Finally, documents that were smaller than 1 kB were also removed. The final corpus comprised 8,581 files with a total of 51,881,537 tokens.

3 Terms in the ACL ARC related to collocations

The list of terms related to the term *collocation* could be prohibitively lengthy and could include many aspects of *what it is* and *how it is used*. For simplicitys sake, a short list of related terms, including *word*, *collocation*, *multiword*, *token*, *unigram*, *bigram*, *trigram*, *collocation extraction* and *segmentation*, was compiled. Table 2 shows when these terms were introduced in the ACL ARC: some terms were introduced early on, others more recently. The term collocation was introduced nearly 50 years ago and has been in use ever since. This is not unexpected, as collocation phenomena were already being studied by the ancient Greeks (Seretan, 2011). Table 2 presents the first use of terms, showing that the terms *segmentation*, *collocation* and *multiword* are related to a similar concept of gathering consecutive words together into one unit.

Term	Count	Documents	Introduced in
word	218813	7725	1965
segmentation	11458	1413	1965
collocation	6046	786	1965
multiword	1944	650	1969
token	3841	760	1973
trigram	3841	760	1973/87
bigram	5812	995	1988
unigram	2223	507	1989
collocation extraction	214	57	1992

Table 1: Term usage in ACL ARC

While the term *collocation* has been used for many years, the first attempt to define what *a collocation is* could be related to the time period when statistics first began to be used in linguistics heavily. Until that time, *collocation* was used mostly in the sense of an expression produced by a particular syntactic rule. The first definition of collocation in ACL ARC is found in (Cumming, 1986).

(Cumming, 1986): *By "collocation" I mean lexical restrictions (restrictions which are not predictable from the syntactic or semantic properties of the items) on the modifiers of an item; for example, you can say* **answer the door** *but not* **answer the window**. *The phenomenon which I've called* **collocation is** *of particular interest in the context of a paper on the lexicon in text generation because this particular type of idiom is something which a generator needs to know about, while a parser may not.*

It is not the purpose of this paper to provide a definition of the term collocation, because at the moment there is no definition that everybody would agree upon. The introduction of *unigrams*, *bigrams* and *trigrams* in the eighties had a big influence on the use of collocations in practice. N-grams, as a substitute to collocations, started being used intensively and in many applications. On the other hand, n-grams are lacking in generalization capabilities and recent research tends to combine n-grams, syntax and semantics (Pecina, 2005) .

The following sections introduce collocation segmentation and apply it to extracting the most significant collocation segments to study the main breakpoints of different research areas in the ACL ARC.

4 Collocation Segmentation

The ACL ARC contains many different segmentation types: discourse segmentation (Levow, 2004), topic segmentation (Arguello and Rose, 2006), text segmentation (Li and Yamanishi, 2000), Chinese text segmentation (Feng et al., 2004), word segmentation (Andrew, 2006). Segmentation is performed by detecting boundaries, which may also be of several different types: syllable boundaries (Müller, 2006), sentence boundaries (Liu et al., 2004), clause boundaries (Sang and Dejean, 2001), phrase boundaries (Bachenko and Fitzpatrick, 1990), prosodic boundaries (Collier et al., 1993), morpheme bound-

Term	Source and Citation
word	(Culik, 1965) : 3. Translation "**word** by **word**" . "Of the same simplicity and uniqueness is the decomposition of the sentence S in its single **words** w_1 , w_2 , ..., w_k separated by interspaces, so that it is possible to write s = (w_1 w_2 ... w_k) like at the text." A **word** is the result of a sentence decomposition.
segmentation	(Sakai, 1965): The statement "x is transformed to y" is a generalization of the original fact, and this generalization is not always true. The text should be checked before a transformational rule is applied to it. Some separate steps for this purpose will save the machine time. (1) A text to be parsed must consist of segments specified by the rule. The correct **segmentation** can be done by finding the tree structure of the text. Therefore, the concatenation rules must be prepared so as to account for the structure of any acceptable string.
Collocation	(Tosh, 1965): We shall include features such as lexical **collocation** (agent-action agreement) and transformations of semantic equivalence in a systematic description of a higher order which presupposes a morpho-syntactic description for each language [8, pp. 66-71]. The following analogy might be drawn: just as strings of alphabetic and other characters are taken as a body of data to be parsed and classified by a phrase structure grammar, we may regard the string of rule numbers generated from a phrase structure analysis as a string of symbols to be parsed and classified in a still higher order grammar [11; 13, pp. 67-83], for which there is as yet no universally accepted nomenclature.
multi-word	(Yang, 1969): When title indices and catalogs, subject indices and catalogs, business telephone directories, scientific and technical dictionaries, lexicons and idiom-and-phrase dictionaries, and other descriptive **multi-word** information are desired, the first character of each non-trivial word may be selected in the original word sequence to form a keyword. For example, the rather lengthy title of this paper may have a keyword as SADSIRS. Several known information systems are named exactly in this manner such as SIR (Raphael's Semantic Information Retrieval), SADSAM (Lindsay's Sentence Appraiser and Diagrammer and Semantic Analyzing Machine), BIRS (Vinsonhaler's Basic Indexing and Retrieval System), and CGC (Klein and Simmons' Computational Grammar Coder).
token	(Beebe, 1973): The type/**token** ratio is calculated by dividing the number of discrete entries by the total number of syntagms in the row.
trigram	(Knowles, 1973): sort of phoneme triples (**trigrams**), giving list of clusters and third-order information-theoretic values. (D'Orta et al., 1987): Such a model it called **trigram** language model. It is based on a very simple idea and, for this reason, its statistics can be built very easily only counting all the sequences of three consecutive words present in the corpus. On the other hand, its predictive power is very high.
bigram	(van Berkelt and Smedt, 1988): **Bigrams** are in general too short to contain any useful identifying information while tetragrams and larger n-gram are already close to average word length. (Church and Gale, 1989): Our goal is to develop a methodology for extending an n-gram model to an (n+l)-gram model. We regard the model for **unigrams** as completely fixed before beginning to study **bigrams**.
unigram	the same as bigram for (Church and Gale, 1989)
collocation extraction	(McKeown et al., 1992): Added syntactic parser to Xtract, a **collocation extraction** system, to further filter collocations produced, eliminating those that are not consistently used in the same syntactic relation.

Table 2: Terms introductions in ACL ARC.

aries (Monson et al., 2004), paragraph boundaries (Filippova and Strube, 2006), word boundaries (Rytting, 2004), constituent boundaries (Kinyon, 2001), topic boundaries (Tur et al., 2001).

Collocation segmentation is a new type of segmentation whose goal is to detect *fixed word sequences* and to segment a text into word sequences called collocation segments. I use the definition of a sequence in the notion of one or more. Thus, a collocation segment is a sequence of one or more consecutive words that collocates and have collocability relations. A collocation segment can be of any

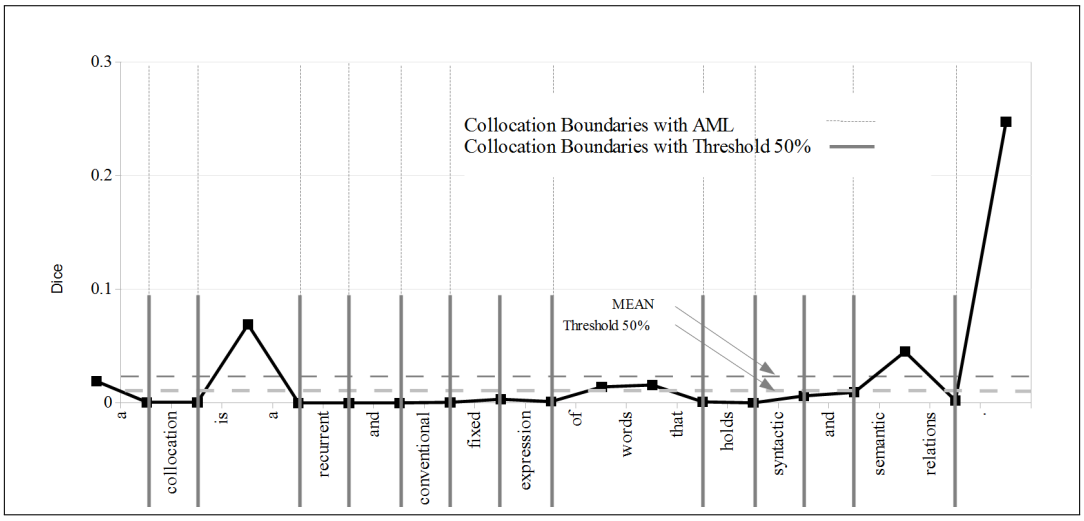

Figure 1: The collocation segmentation of the sentence *a collocation is a recurrent and conventional fixed expression of words that holds syntactic and semantic relations . (Xue et al., 2006).*

length (even a single word) and the length is not defined in advance. This definition differs from other collocation definitions that are usually based on n-gram lists (Tjong-Kim-Sang and S., 2000; Choueka, 1988; Smadja, 1993). Collocation segmentation is related to collocation extraction using syntactic rules (Lin, 1998). The syntax-based approach allows the extraction of collocations that are easier to describe, and the process of collocation extraction is well-controlled. On the other hand, the syntax-based approach is not easily applied to languages with fewer resources. Collocation segmentation is based on a discrete signal of associativity values between two consecutive words, and boundaries that are used to chunk a sequence of words.

The main differences of collocation segmentation from other methods are: (1) collocation segmentation does not analyze nested collocations it takes the longest one possible in a given context, while the n-gram list-based approach cannot detect if a collocation is nested in another one, e.g., *machine translation system*; (2) collocation segmentation is able to process long collocations quickly with the complexity of a bigram list size, while the n-gram list-based approach is usually limited to 3-word collocations and has high processing complexity.

There are many word associativity measures, such as Mutual Information (MI), T-score, Log-Likelihood, etc. A detailed overview of associativity measures can be found in (Pecina, 2010), and any of these measures can be applied to collocation segmentation. MI and Dice scores are almost similar in the sense of distribution of values (Daudaravicius and Marcinkeviciene, 2004), but the Dice score is always in the range between 0 and 1, while the range of the MI score depends on the corpus size. Thus, the Dice score is preferable. This score is used, for instance, in the collocation compiler XTract (Smadja, 1993) and in the lexicon extraction system Champollion (Smadja et al., 1996). Dice is defined as follows:

$$D(x_{i-1}; x_i) = \frac{2 \cdot f(x_{i-1}; x_i)}{f(x_{i-1}) + f(x_i)}$$

where $f(x_{i-1}; x_i)$ is the number of co-occurrence of x_{i-1} and x_i, and $f(x_{i-1})$ and $f(x_i)$ are the numbers of occurrence of x_{i-1} and x_i in the training corpus. If x_{i-1} and x_i tend to occur in conjunction, their Dice score will be high. The Dice score is sensitive to low-frequency word pairs. If two consecutive words are used only once and appear together, there is a good chance that these two words are highly related and form some new concept, e.g., a proper name. A text is seen as a changing curve of Dice values between two adjacent words (see Figure 1). This curve of associativity values is used to detect the boundaries of collocation segments, which can be done using a threshold or by following certain rules, as described in the following sections.

length	unique segments	segment count	word count	corpus coverage
1	289,277	31,427,570	31,427,570	60.58%
2	222,252	8,594,745	17,189,490	33.13%
3	72,699	994,393	2,983,179	5.75%
4	12,669	66,552	266,208	0.51%
5	1075	2,839	14,195	0.03%
6	57	141	846	0.00%
7	3	7	49	0.00%
Total	598,032	41,086,247	51,881,537	100%

Table 3: The distribution of collocation segments

2 word segments	CTFIDF
machine translation	10777
speech recognition	10524
training data	10401
language model	10188
named entity	9006
error rate	8280
test set	8083
maximum entropy	7570
sense disambiguation	7546
training set	7515
noun phrase	7509
our system	7352
question answering	7346
information retrieval	7338
the user	7198
word segmentation	7194
machine learning	7128
parse tree	6987
knowledge base	6792
information extraction	6675

3 word segments	CTFIDF
in terms of	4099
total number of	3926
th international conference	3649
is used to	3614
one or more	3449
a set of	3439
note that the	3346
it is not	3320
is that the	3287
associated with the	3211
large number of	3189
there is a	3189
support vector machines	3111
are used to	3109
extracted from the	3054
with the same	3030
so that the	3008
for a given	2915
it is a	2909
fact that the	2876

4 word segments	CTFIDF
if there is a	1690
human language technology conference	1174
is defined as the	1064
is used as the	836
human language technology workshop	681
could be used to	654
has not yet been	514
may be used to	508
so that it can	480
our results show that	476
would you like to	469
as well as an	420
these results show that	388
might be able to	379
it can also be	346
have not yet been	327
not be able to	323
are shown in table	320
is that it can	311
if there is an	305

5 word segments	CTFIDF
will not be able to	255
only if there is a	212
would not be able to	207
may not be able to	169
a list of all the	94
will also be able to	43
lexical information from a large	30
should not be able to	23
so that it can also	23
so that it would not	23
was used for this task	23
indicate that a sentence is	17
a list of words or	16
because it can also be	16
before or after the predicate	16
but it can also be	16
has not yet been performed	16
if the system has a	16
is defined as an object	16
is given by an expression	16

Table 4: Top 20 segments for the segment length of two to five words.

4.1 Setting segment boundaries with a Threshold

A boundary can be set between two adjacent words in a text when the Dice value is lower than a certain threshold. We use a dynamic threshold which defines the range between the minimum and the average associativity values of a sentence. Zero equals the minimum associativity value and 100 equals the average value of the sentence. Thus, the threshold value is expressed as a percentage between the minimum and the average associativity values. If the threshold is set to 0, then no threshold filtering is used and no collocation segment boundaries are set using the threshold. The main purpose of using a threshold is to keep only strongly connected tokens. On the other hand, it is possible to set the threshold to the maximum value of associativity values. This would make no words combine into more than single word segments, i.e., collocation segmentation would be equal to simple tokenization. In general, the threshold makes it possible to move from only single-word segments to whole-sentence segments by changing the threshold from the minimum to the maximum value of the sentence. There is no reason to use the maximum value threshold, but this helps to understand how the threshold can be used. (Daudaravicius and Marcinkeviciene, 2004) uses a global constant threshold which produces very long collocation segments that are like the clichs used in legal documents and hardly related to collocations. A dynamic threshold allows the problem of very long segments to be reduced. In this study I used a threshold level of 50 percent. An example of threshold is shown in Figure 1. In the example, if the threshold is 50 percent then segmentation is as follows: *a* | *collocation* | *is a* | *recurrent* | *and* | *conventional* | *fixed* | *expression* | *of words that* | *holds* | *syntactic* | *and* | *semantic relations* | . To reduce the problem of long segments even more, the Average Minimum Law can also be used, as described in the following section.

4.2 Setting segment boundaries with Average Minimum Law

(Daudaravicius, 2010) introduces the Average Minimum Law (AML) for setting collocation segmentation boundaries. AML is a simple rule which is applied to three adjacent associativity values and is expressed as follows:

$$boundary(x_{i-2}, x_{i-1}) = \\ = \begin{cases} True & \frac{D(x_{i-3}; x_{i-2}) + D(x_{i-1}; x_i)}{2} < D(x_{i-2}; x_{i-1}) \\ False & otherwise \end{cases}$$

The boundary between two adjacent words in the text is set where the Dice value is lower than the average of the preceding and following Dice values. In order to apply AML to the first two or last two words, I use sequence beginning and sequence ending as tokens and calculate the associativity between the beginning of the sequence and the first word, and the last word and the end of the sequence as shown in Figure 1. AML can be used together with Threshold or alone. The recent study of (Daudaravicius, 2012) shows that AML is able to produce segmentation that gives the best text categorization results, while the threshold degrades them. On the other hand, AML can produce collocation segments where the associativity values between two adjacent words are very low (see Figure 1). Thus, for lexicon extraction tasks, it is a good idea to use AML and a threshold together.

5 Collocation segments from the ACL ARC

Before the collocation segmentation, the ACL ARC was preprocessed with lowercasing and tokenization. No stop-word lists, taggers or parsers were used, and all punctuation was kept. Collocation segmentation is done on a separate line basis, i.e., for each text line, which is usually a paragraph, the average and the minimum combinability values are determined and the threshold is set at 50 percent, midway between the average and the minimum. The Average Minimum Law is applied in tandem. The tool *CoSegment* for collocation segmentation is available at (`http://textmining.lt/`).

Table 3 presents the distribution of segments by length, i.e., by the number of words. The length of collocation segments varies from 1 to 7 words. In the ACL ARC there are 345,455 distinct tokens. After segmentation, the size of the segment list was 598,032 segments, almost double the length of the single word list. The length of the bigram list is

4,484,358, which is more than 10 times the size of the word list and 7 times that of the collocation segment list. About 40 percent of the corpus comprises collocation segments of two or more words, showing the amount of *fixed language* present therein. The longest collocation segment is *described in section 2 . 2 ,* which contains seven words (when punctuation is included as words). This shows that collocation segmentation with a threshold of 50 percent and AML diverges to one-, two- or three-word segments. Despite that, the list size of collocation segments is much shorter than the list size of bigrams, and shorter still than that of trigrams.

After segmentation, it was of interest to find the most significant segments used in the ACL ARC. For this purpose I used a modified TF-IDF which is defined as follows:

$$CTFIDF(x) = TF(x) * ln\left(\frac{N - D(x) + 1}{D(x) + 1}\right)$$

where $TF(x)$ is the raw frequency of segment x in the corpus, N is the total number of documents in the corpus, and $D(x)$ is the number of documents in which the segment x occurs. Table 4 presents the top 20 collocation segments for two-, three-, four- and five-word segments of items that contain alphabetic characters only. The term *machine translation* is the most significant in CTFIDF terms. This short list contains many of the main methods and datasets used in daily computational linguistics research, such as: *error rate*, *test set*, *maximum entropy*, *training set*, *parse tree*, *unknown words*, *word alignment*, *Penn Treebank*, *language models*, *mutual information*, *translation model*, etc. These terms show that computational linguistics has its own terminology, methods and tools to research many topics.

Finally, 76 terms of two or more words in length with the highest CTFIDF values were selected. The goal was to try to find how significant terms were used yearly in the ACL ARC. The main part of the ACL ARC was compiled using papers published after 1995. Therefore, for each selected term, the average CTFIDF value of each document for each year was calculated. This approach allows term usage throughout the history of the ACL to be analysed, and reduces the influence of the unbalanced amount of published papers. Only those terms whose average CTFIDF in any year was higher than 20 were kept. For instance, the term *machine translation* had to be removed, as it was not significant throughout all the years. Each term was ranked by the year in which its average CTFIDF value peaked. The ranked terms are shown in Table 5. For instance, the peak of the CTFIDF average of the term *statistical parsing* occurred in 1990, of the term *language model* in 1987, and of the term *bleu score* in 2006. The results (see Table 5) show the main research trends and time periods of the ACL community. Most of the terms with CTFIDF peaks prior to 1986 are related to formal/rule-based methods. Beginning in 1987, terms related to statistical methods become more important. For instance, *language model*, *similarity measure*, and *text classification*. The year 1990 stands out as a kind of breakthrough. In this year, the terms *Penn Treebank*, *Mutual Information*, *statistical parsing*, *bilingual corpus*, and *dependency tree* became the most important terms, showing that newly released language resources were supporting many new research areas in computational linguistics. Despite the fact that *Penn Treebank* was only significant temporarily, the corpus is still used by researchers today. The most recent important terms are *Bleu score* and *semantic role labeling*.

This study shows that collocation segmentation can help in term extraction from large and complex corpora, which helps to speed up research and simplify the study of ACL history.

6 Conclusions

This study has shown that collocation segmentation can help in term extraction from large and complex corpora, which helps to speed up research and simplify the study of ACL history. The results show that the most significant terms prior to 1986 are related to formal/rule based research methods. Beginning in 1987, terms related to statistical methods (e.g., *language model*, *similarity measure*, *text classification*) become more important. In 1990, a major turning point appears, when the terms *Penn Treebank*, *Mutual Information*, *statistical parsing*, *bilingual corpus*, and *dependency tree* become the most important, showing that research into new areas of compu-

tational linguistics is supported by the publication of new language resources. The *Penn Treebank*, which was only significant temporarily, it still used today. The most recent terms are *Bleu score* and *semantic role labeling*. While *machine translation* as a term is significant throughout the ACL ARC, it is not significant in any particular time period. This shows that some terms can be significant globally, but insignificant at a local level.

References

Galen Andrew. 2006. A hybrid markov/semi-markov conditional random field for sequence segmentation. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 465–472, Sydney, Australia, July. Association for Computational Linguistics.

Jaime Arguello and Carolyn Rose. 2006. Topic-segmentation of dialogue. In *Proceedings of the Analyzing Conversations in Text and Speech*, pages 42–49, New York City, New York, June. Association for Computational Linguistics.

J. Bachenko and E. Fitzpatrick. 1990. A computational grammar of discourse-neutral prosodic phrasing in english. *Computational Linguistics*, 16:155–170.

Ralph D. Beebe. 1973. The frequency distribution of english syntagms. In *Proceedings of the International Conference on Computational Linguistics, COLING*.

Y. Choueka. 1988. Looking for needles in a haystack, or locating interesting collocational expressions in large textual databases. In *Proceedings of the RIAO Conference on User-Oriented Content-Based Text and Image Handling*, pages 21–24, Cambridge, MA.

Kenneth W. Church and William A. Gale. 1989. Enhanced good-turing and cat.cal: Two new methods for estimating probabilities of english bigrams (abbreviated version). In *Speech and Natural Language: Proceedings of a Workshop Held at Cape Cod*.

René Collier, Jan Roelof de Pijper, and Angelien Sanderman. 1993. Perceived prosodic boundaries and their phonetic correlates. In *Proceedings of the workshop on Human Language Technology*, HLT '93, pages 341–345, Stroudsburg, PA, USA. Association for Computational Linguistics.

Karel Culik. 1965. Machine translation and connectedness between phrases. In *International Conference on Computational Linguistics, COLING*.

Susanna Cumming. 1986. The lexicon in text generation. In *Strategic Computing - Natural Language Workshop: Proceedings of a Workshop Held at Marina del Rey*.

V. Daudaravicius and R Marcinkeviciene. 2004. Gravity counts for the boundaries of collocations. *International Journal of Corpus Linguistics*, 9(2):321–348.

Vidas Daudaravicius. 2010. The influence of collocation segmentation and top 10 items to keyword assignment performance. In Alexander F. Gelbukh, editor, *CICLing*, volume 6008 of *Lecture Notes in Computer Science*, pages 648–660. Springer.

Vidas Daudaravicius. 2012. Automatic multilingual annotation of eu legislation with eurovoc descriptors. In *In Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'2012)*.

Paolo D'Orta, Marco Ferretti, Alessandro Martelli, and Stefano Scarci. 1987. An automatic speech recognition system for the italian language. In *Third Conference of the European Chapter of the Association for Computational Linguistics*.

Haodi Feng, Kang Chen, Xiaotie Deng, and Weimin Zheng. 2004. Accessor variety criteria for chinese word extraction. *Computational Linguistics*, 30:75–93.

Katja Filippova and Michael Strube. 2006. Using linguistically motivated features for paragraph boundary identification. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 267–274, Sydney, Australia, July. Association for Computational Linguistics.

Alexandra Kinyon. 2001. A language independent shallow-parser compiler. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 330–337, Toulouse, France, July. Association for Computational Linguistics.

F. Knowles. 1973. The quantitative syntagmatic analysis of the russian and polish phonological systems. In *Computational And Mathematical Linguistics: Proceedings of the International Conference on Computational Linguistics, COLING*.

Gina-Anne Levow. 2004. Prosodic cues to discourse segment boundaries in human-computer dialogue. In Michael Strube and Candy Sidner, editors, *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue*, pages 93–96, Cambridge, Massachusetts, USA, April 30 - May 1. Association for Computational Linguistics.

Hang Li and Kenji Yamanishi. 2000. Topic analysis using a finite mixture model. In *2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 35–44, Hong Kong, China, October. Association for Computational Linguistics.

D. Lin. 1998. Extracting collocations from text corpora. In *First Workshop on Computational Terminology*, Montreal.

Yang Liu, Andreas Stolcke, Elizabeth Shriberg, and Mary Harper. 2004. Comparing and combining generative and posterior probability models: Some advances in sentence boundary detection in speech. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 64–71, Barcelona, Spain, July. Association for Computational Linguistics.

Kathleen McKeown, Diane Litman, and Rebecca Passonneau. 1992. Extracting constraints on word usage from large text corpora. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman*.

Christian Monson, Alon Lavie, Jaime Carbonell, and Lori Levin. 2004. Unsupervised induction of natural language morphology inflection classes. In *Proceedings of the Seventh Meeting of the ACL Special Interest Group in Computational Phonology*, pages 52–61, Barcelona, Spain, July. Association for Computational Linguistics.

Karin Müller. 2006. Improving syllabification models with phonotactic knowledge. In *Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology and Morphology at HLT-NAACL 2006*, pages 11–20, New York City, USA, June. Association for Computational Linguistics.

Pavel Pecina. 2005. An extensive empirical study of collocation extraction methods. In *Proceedings of the ACL Student Research Workshop*, pages 13–18, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Pavel Pecina. 2010. Lexical association measures and collocation extraction. *Language Resources and Evaluation*, 44(1-2):137–158.

C. Anton Rytting. 2004. Segment predictability as a cue in word segmentation: Application to modern greek. In *Proceedings of the Seventh Meeting of the ACL Special Interest Group in Computational Phonology*, pages 78–85, Barcelona, Spain, July. Association for Computational Linguistics.

Itiroo Sakai. 1965. Some mathematical aspects on syntactic discription. In *International Conference on Computational Linguistics, COLING*.

Erik F. Tjong Kim Sang and Herve Dejean. 2001. Introduction to the conll-2001 shared task: clause identification. In *Proceedings of the ACL 2001 Workshop on Computational Natural Language Learning*, Toulouse, France, July. Association for Computational Linguistics.

Violeta Seretan. 2011. *Syntax-Based Collocation Extraction*, volume 44 of *Text, Speech and Language Technology*. Springer.

Frank Smadja, Vasileios Hatzivassiloglou, and Kathleen R. McKeown. 1996. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22:1–38.

Frank Smadja. 1993. Retrieving collocations from text: Xtract. *Computational Linguistics*, 19:143–177.

E. Tjong-Kim-Sang and Buchholz S. 2000. Introduction to the conll-2000 shared task: Chunking. In *Proc. of CoNLL-2000 and LLL-2000*, pages 127–132, Lisbon, Portugal.

L. W. Tosh. 1965. Data preparation for syntactic translation. In *International Conference on Computational Linguistics, COLING*.

Gokhan Tur, Andreas Stolcke, Dilek Hakkani-Tur, and Elizabeth Shriberg. 2001. Integrating prosodic and lexical cues for automatic topic segmentation. *Computational Linguistics*, 27:31–57.

Brigitte van Berkelt and Koenraad De Smedt. 1988. Triphone analysis: A combined method for the correction of orthographical and typographical errors. In *Proceedings of the Second Conference on Applied Natural Language Processing*, pages 77–83, Austin, Texas, USA, February. Association for Computational Linguistics.

Nianwen Xue, Jinying Chen, and Martha Palmer. 2006. Aligning features with sense distinction dimensions. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 921–928, Sydney, Australia, July. Association for Computational Linguistics.

Shou-Chuan Yang. 1969. A search algorithm and data structure for an efficient information system. In *International Conference on Computational Linguistics, COLING*.

	65	67	69	73	75	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	00	01	02	03	04	05	06
parsing algorithm	25						2	9	5	13					13	7											10			10				
lexical entry			36		21		4						14												9						13			11
source language		11	21				4			15						9					12													
word senses		10		31					22	12			6	10	7	7	19	17		13						6			9				9	
target language		11	15		24		2			18			5			11				10	13				10		17							
brown corpus							4	36		16			6			30			21	18	21	20	6		14			9	29					
logical form							8	21	11	17	13	2	6	9	18	12	19	15	16	16	17	14	8		13	8			11	11		10	10	
semantic representation				9		4	3			21	9						11																	
multi - word											22																21	9						
reference resolution							4	7	8					41	9		30	17	13	16	18		9			9	23			14		12		
language model											9				34		11	19	14	13	12	18				7		13	11			12	10	9
text generation			24					17	9	25	25		13	9	29			19		12			7											
spoken language		6														37	23	20	19	21	13		14											
speech recognition													12			11	33	19	21	19	16	16												
similarity measure			13													13	33	17				15									10			
text classification																		55		23	17									11	16			
statistical parsing																		30																
tree adjoining grammars												3		14		22	19	29	19	15		13					12		11					
mutual information																12	17	27	12	15			9	11				15						
penn treebank																		26					6											
bilingual corpus																		22		11		19			9			11	9	16	17	12	11	
dependency tree	10	8	9							11	8			9	7	10		21		15			12		11			13	16		11	16	15	11
pos tagging																			23		14			10	11			10		10	10	11	13	9
spontaneous speech				16												8	34	22	20	42	16	17		18				9			13	10		
text categorization																			21	25								16	9	15	14	12		10
feature selection																			20	11	51				10	7	10	11	9	12	14	15	10	10
translation model																		28	27			49						17	11	15	18	15	14	10
spelling correction									10	14	9		7			17		17	16	19		42	9	27	26			15	10	17	10	13	19	16
edit distance																						37		21			13	16	10	14	11	12	15	12
target word																							42		13	9		12	9	16	14	20	13	11
speech synthesis							3	8			11			9		8	14	18	16		11		29		9				11	11		10		10
search engine																					27			31	16			12	20		11		9	10
maximum entropy																			22	12	22	25		27	12	10	10	9		12	10			
lexical rules				12		6	2	10					5	18	9		11		36			18	8	24	46		21	18	11		11			
annotation scheme																						15		14	31		21	14	15	11	18		10	
coreference resolution																							11	21	34	37		10	16	27	20	22	16	27
text summarization														15												36	17	14		13	13	13	12	
naive bayes																									32	23	43	14	12	13	23	20	12	17
trigram model																		20		20	16	13		13		14	38	10	13	11	13			10
named entity																						20	12	14	11	14	31	12	10	19	16		10	11
anaphora resolution									5						10	9						15	17	10			29	13	21	11		12	12	11
word segmentation																				20	19	26		16	16	13		10	11	20	31	26	24	30
word alignment																								10	13		19	20		11	24	24	22	17
semantic role labeling																																25	25	28
bleu score																														12	18	16	14	21
	65	67	69	73	75	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	00	01	02	03	04	05	06

Table 5: The list of selected terms and the yearly importance in terms of CTFIDF.