Temporal Language Models for the Disclosure of Historical Text

Franciska de Jong Henning Rode Djoerd Hiemstra

University of Twente, The Netherlands
{f.m.g.dejong, h.rode, d.hiemstra}@utwente.nl

1 Introduction

Historical and heritage collections consist for a considerable part of text and may incorporate diverse text types such as journals, archival documents, and catalogue descriptions. Because of the historical distance, access to this content is not straightforward. Historical variants of text are often more complex to identify and retrieve than modern variants. This is due to the less standardized spelling, the effect of on-going language change and different word (de)compounding principles. Moreover, more words are ambiguous because one or more meaning shifts may have occurred. Common full-text search tools can only be applied successfully by users who are able to formulate queries with (a) knowledge of historical language and (b) insight in the relevant time spam from which the words have evolved. This paper explores techniques which may compensate for these linguistic obstacles: linking of contemporary search terms to their historical equivalents and 'dating' of texts.

We envisage to restore the diachronic relationship between terms which may be obscured by language evolution and usage, by applying statistical language models. These models may support the automatic detection of semantic similarities between words and word ambiguities, and they also allow to classify a text according to the time span from which it originates. This approach involves building temporal profiles of words as longitudinal sections in a reference corpus and temporal language models as cross sections.

In section 2 some detailed examples will be presented of the added value of this approach both for the accessibility of historical content and the detection of language change in relatively recent corpora from the news domain. In section 3 an overview of related work will be given, plus some technical background on statistical language models. Section 4 describes the proposed methodology in more detail, and some experiments for it in the news domain will described in section 5.

2 What temporal models can do?

Two types of language change will be distinguished here: evolution that is exemplified by a series of etymologically cognate word forms, and evolution exemplified by a single word form with a series of different but distinguished meanings. An example of the first type is the series of Dutch forms for the concept *PILGRIM*: *pelegrime, pelgrime, peelgrime, peilgrime, pilgrime, ..., pilgrijm, pellegrijm, peregrijm.* An example of the latter would be the semantic evolution for Dutch words such as *wijf* (English: woman; shifted connotations) and the *kabinet* (English: cabinet). For the sake of simplicity we will ignore here combinations of these types of evolution.

Users of a search system will typically know one or more contemporary forms associated with the concept they want to search for. They would be helped if the search interface was enhanced with knowledge about diachronically related forms that can be considered synonyms. This knowledge

could be available via lexicographic resources, but could also be detected (semi-) automatically by using information on collocations (words that tend to occur in the same syntactic phrase; fixed meaning; e.g., *commit a crime; raise a question* and significant co-occurrence figures (for words that tend to occur in the same documents).

An early form for the concept PILGRIM is as likely to co-occur with (a form for) the concepts PILGRIMAGE or TRAVEL as a contemporary form. Language models can catch such dependencies per period, but via corpus-based normalization the temporal evolution of a concept profile could also be captured. Conversely, shifts in meaning will also bring shifts in co-occurrence figures. In principle statistical language model should therefore be able to help detecting diachronic synonymy.

A lot of research, including experimental work will be needed to deliver a proof-of-concept for this idea, and to get a better grip on the question which information sources can exploited and or combined to generate tools that are precise and refined enough to support the exploration of historical texts. In addition to plain text, all kinds of corpus annotation could be exploited, such as metadata on date of publication (or more precisely: conception), author, etc.

Keeping track of diachronic form evolution has features in common with paraphrase or synonymy detection and can thus be seen as a variant of translation. Therefore the approach proposed here shares some features with work on cross-language information retrieval (CLIR), and the therefore applicability of of CLIR-methods such as decribed in [7] should be investigated.

Moreover, as dictionaries have been demonstrated to be useful resources for automatic word sense disambiguation[1], we foresee a role for parsed entries from historical dictionaries in this context as well. Note that the approach oulined here could also be seen as a contribution to the construction of a diachronic WordNet.[2]

In this paper we will present some experimental results for the dating of contemporary news articles. The purpose is to illustrate the potential role that temporal profiles can play in the automatic annotation of texts. The choice for contemporary content is simply given in by practical considerations. The development of statistical language models requires the availability of a huge digitised reference corpus, and as digitisation of historical text corpora has only just begun.[3] The application of temporal language models to historical data is future work.

3 Related Research and Background

To assist the disclosure of historical documents, time needs to be modeled somehow by the system. There is extensive literature on automatic classification of texts: There are links with work on automatic thesaurus discovery [4] and other approaches to derive synonyms and concept hierarchies from text [9, 12, 16, 18]). Temporal modeling of text does not play a role in these approaches.

Time stamps of documents have been used in several studies for browsing document collections [1,8]. In these studies, temporal metadata such as publication date are used to present or visualise the temporal structure of media collections, or just those parts that are relevant to a query. Temporal metadata can help users to zoom in on a certain period, for instance because they expect it to be more relevant for a certain topic or event than other periods. Swan and Jensen [17] were among the first to investigate which kind of statistical models were appropriate for modeling of the temporal dimension of term usage. They used simple contingency tables, similar to the word-time matrix we introduce in the next section. In subsequent studies by Li and Croft [11] and Diaz and Jones [2], *statistical language models* are used for temporal models of term usage.

Statistical language models are simple models of language use, which were pioneered by researchers developing automatic speech recognition systems [15] and later taken up in the field of text retrieval [5,14]. A language model assigns a probability to a piece of text. Typically, we would expect a statistical language model for English to assign a much lower probability to the phrase

[1] Cf. First experiments were reported already in [10]

[2] The WordNet lexical database is a machine-readable thesaurus and semantic network developed and maintained by the Cognitive Science Laboratory at Princeton University. [3].

[3] Cf. for example projects at the Dutch National Royal Library(http://kranten.kb.nl) and the British Library (http://www.bl.uk/catalogues/newspapers/intro.asp).

"mice zoo meat queue" than to the phrase "nice to meet you". On the basis of such probabilities a speech recognition system can be supported in picking the most likely combination of words. In this paper, we will build a similar language model for the dating task. The model assigns a probability to pieces of text, given a particular time frame.

For reasons of simplicity, instead of sophisticated n-gram models, representing the probabilities of phrases or word sequences up to length n, we will only use *unigram* models here, i.e., n-gram models with $n = 1$. This type of language models is commonly used in information retrieval settings and captures word or term probabilities instead of sequence probabilities. Given a certain document D, the probability to encounter a word w is calculated by the frequency of w occurring in D divided by the total number of words in D. Hence, an unigram language model of any document can be represented by a table of word frequencies.

There are several standard measures for comparing two language models, such as cross-entropy or Kullback-Leibler divergence between the models. In this paper we use a normalised variant proposed by Kraaij [6], the so-called normalised log-likelihood ratio measure (NLLR) between a model Q and a model D, usually representing the query and a document to which the query is compared. C is a background model that is estimated on the entire collection.

$$NLLR(Q|D) = \sum_{w \in Q} P(w|Q) * \log \left(\frac{P(w|D)}{P(w|C)} \right)$$

It is easy to see that this measure is not useful if one of the terms from model Q is assigned zero probability by model D, i.e., if $P(w|D) = 0$, the logarithm is undefined.[4] This may occur in the case of 'unseen events', such as the absence of a word like *pilgrim* in a certain time span, e.g., 1920–1930. This would normally lead to probability $P(\text{pilgrim}|D = \text{1920–1930}) = 0$. To avoid this effect, a so-called smoothing method can be applied in estimating the probabilities of a model. In this example, smoothing would assign a very small (non-zero) probability to *pilgrim* in the time span 1920–1930.

Compared to typical document language models, temporal language models are rather large. Therefore, the issue of smoothing will probably be less important. It might even flatten out important characteristics of a specific time span. In this study we experimented with two smoothing approaches: so-called linear interpolation smoothing used by Kraaij [6] and Dirichlet smoothing [19]. In contrast to linear interpolation smoothing, the effect of Dirichlet-smoothing depends directly on the size of the smoothed model. Thus, for temporal language models built from large fractions of the reference corpus the smoothing is negligible, while it is effective for models built on a small fraction of the corpus.

4 Dating of text

An example application of temporal language models is the dating of texts. In this paper we would like to show how statistical language models can be used for this task. A more precise definition of the dating task is given below:

Task definition given a date-tagged reference corpus, consisting of documents from a certain time span, and a document X with unknown date within the same time span, the system should classify X according to time partitions of predefined granularity.

4.1 Reference Corpus

The task definition mentions a reference corpus. Such a collection of documents with known publication date is necessary as a base for comparison. The temporal language models derived from the corpus are supposed to capture the characteristics of the vocabulary used within a certain period. The reference corpus must meet several requirements. It needs to

[4] $P(w|D)$ is the probability that w shows up given D

- be sufficiently large,
- have a balanced distribution over the represented time span,
- cover the same domain as the documents to be dated,
- and cover at least the period from which the undated documents originate.

The first requirement is formulated vaguely, because the required corpus size depends on several other parameters, such as the level of granularity imposed by the actual dating task. The main concern, however, is that a sparse data set may cause the language models to be determined by specific document characteristics rather than by temporal patterns. For similar reasons, a balanced distribution of the corpus documents over the complete time span is needed. If one temporal language model is aggregated from a few documents, while another one is based on half of the corpus, the former may suffer from data sparsity, whereas the latter may be over-trained, resulting in a model hardly distinguishable from the background collection characteristics. But even these two requirements do not help if the corpus domain differs from the topic domain of the document to be dated. Obviously, we cannot date a sports article from a newspaper with a model trained on a corpus with personal correspondences from writers. Reliable dating also requires that the publication date of target articles is covered by the reference corpus.

4.2 Time Partitioning

In the Task Definition above there is mentioning of the granularity of the date classification. In fact, a document is not dated precisely, but our method just outputs the time span from which the document most probably originates. The reference corpus is therefore partitioned into smaller sets of documents, corresponding to time spans of the desired granularity, and the document is compared with all these time partitions. Only in case the granularity selected corresponds to temporal units of one day – in most cases not a reasonable choice – the classification will involve exact dating.

Formally the partitioning can be described as follows: if we divide n time marks t_i in ascending order over the full time span of the corpus, such that t_0 marks the start of the corpus period, then each pair of adjacent time marks ($t_i < t_{i+1}$) defines a time partition C_i of corpus documents. D_j denotes any document from the corpus and $\tau(D_j)$ it's date.

$$C_i = \{D_j | t_i \leq \tau(D_j) < t_{i+1}\}.$$

We also need to distinguish between output granularity and model granularity. Whereas the first will be determined by external factors – need for publication year, week, day, etc.,– a finer model granularity can be chosen as well. Suppose we like to classify newspaper articles for the *year* of publication. If we build temporal language models for news on a yearly scale, we won't get very characteristic time patterns. Specific topics usually are discussed during shorter time spans and within a year almost any topic can be mentioned. An alternative would be to build models on a smaller scale, e.g., weekly, while still letting the system produce labels for the year of publication. In general, as model granularity any non-overlapping partitioning can be chosen that is finer than or equal to the one wanted for the final output.

Within this section, time span partitioning is defined as a division in non-overlapping sections. However, in the newspaper case, the period in which a certain topic occurs in the news will not always coincide with a certain time partitioning. Normally the partition time marks t_i are equally divided over the corpus time span, resulting in an arbitrary crossing of such topic boundaries. A simple way to avoid this is to abandon the principle of overlap-free partitioning and to generate overlapping partitions with a time window moving in small steps (window size $w \leq s$ step length) over the corpus time span. For reasons of simplicity, though, we will stick here to the overlap-free partitioning.

4.3 Classification Approaches for the Dating Task

The basic idea is to compute a language model from the undated document and compare it to the language models built from the reference corpus. Comparison of language models is an often used technique, not only for the "classic" information retrieval task to find documents similar to a query model, but also for classification of documents, for instance in case of topic detection. In the following, we describe two such approaches for date classification. They differ in the way the language models for the reference corpus are built.

Method A: Comparison on Document Level The first approach is based on the idea of Diaz and Jones for temporal query profiles [2]. In a first step, all corpus documents are ranked according to their language model divergence to the undated document X, i.e., by the normalized log-likelihood ratio $NLLR(X|D_j),\ D_j \in C$. In a second step, a temporal profile of X is built from the set S of the top-n ranked documents by aggregating the sum of scores belonging to each time partition:

$$val(C_i) = \sum_{D_j} NLLR(X|D_j),\ D_j \in S \cap C_i.$$

The interpretation of the computed values is obvious: the higher the score of a time partition, the higher the probability that the document originates from its time span. Thus, the time partition with the highest value $val(C_i)$ is the best candidate to determine $\tau(X)$. Or in other words, the most likely publication period.

Method B: Comparison on Partition Level An alternative approach is to perform the aggregation beforehand by building temporal language models for each time partition, which requires to sum up the word frequencies from all documents belonging to a time partition:

$$|w \in C_i| = \sum_{D_j \in C_i} |w \in D_j|.$$

The next section discusses the building of temporal language models in more detail. Having language models for all time partitions, we are able to compare them directly with the language model for undated documents. The highest ranked time partition C_i then determines the systems output for $\tau(X)$. This is either the time span of C_i itself, or if model and output granularity differ from each other, the enclosing time span of the output partitioning.

4.4 Data structures for temporal language models

In the previous section it was explained how aggregated temporal language models can be used for the dating task. Here we will describe two simple data structures to maintain such language models.

Word	Partition	Freq.
gulden	2000	1498
gulden	2001	1615
gulden	2002	481
euro	2000	10339
euro	2001	13625
euro	2002	26905
toekomst	2000	7360
toekomst	2001	6962
toekomst	2002	5141

(a) Table

	2000	2001	2002
gulden	1498	1615	481
euro	10339	13625	26905
toekomst	7360	6962	5141

(b) Matrix

Figure 1: Data Structures: Table vs. Matrix Design

Instead of storing each temporal language model separately, which introduces organizational overhead, all frequency counts can be gathered in one data structure. We will discuss two options, which are visualized in Fig. 1:

- a 3-attribute database table of the form `[term-id, partition-id, frequency]`,
- or an 2-dimensional array with terms and partitions as its dimensions (word-time matrix).

The table design is more appropriate in case of a fine time granularity or sparse reference data, because it avoids to store zero frequencies, which occurs often in these settings. If the table is sorted on term identifiers, we can also very efficiently perform a ranking of all partitions (the procedure then equals the one of a complete collection ranking on an inverted index structure).

The matrix structure on the other hand enables fast and direct positional access to all fields, even without storing redundant term and partition identifiers.

4.5 Confidence in Dating

Unless we work on a very specific but time-characteristic document type, we cannot expect that dating based on pure statistical comparison of word frequencies delivers excellent results. It is easy to imagine that an historical source and a later secondary reference share a lot of common vocabulary, although they originate from different periods. For the dating approach described here this is reflected by the scores for the top ranked partitions. They tend to very close, even if the corresponding time spans are scattered over the entire corpus range. This observation suggests that the dating system might be able to decide itself how confident it is about the suggested classification. This is interesting because in general reliable confidence measures for statistical tasks highly increase the usability of systems applying them. A simple confidence measure for dating could be the relative distance between the score of the top-ranked time partition to the scores of the following ones. A more sophisticated measure could also take into account the level of timely scattering in the top-ranked partitions.

5 Empirical Tests on Newspaper Data

In order to illustrate the role of temporal language models and to assess their usefulness for dating techniques, we carried out some preliminary experiments. This section will describe the test corpus, the experiments, and the results.

5.1 The Reference Corpus

The reference corpus for our tests consisted of articles from two well-known Dutch newspapers, *De Volkskrant* and *Algemeen Dagblad*, from the time span ranging from January 1999 till February 2005, in total almost 2 GB of text material.[5] Newspaper articles represent a specific document genre, but such a corpus is heterogeneous in terms of topicality.

Indexing the corpus showed that our term vocabulary had a size of approximately 1.3 million different word forms, including unfortunately a large fraction of spelling mistakes. To reduce the vocabulary size, we simply neglected all words occurring less than 10 times in the whole corpus, a suitable threshold to cut out a large number of spelling mistakes. Furthermore we applied a Dutch stemming algorithm, a rule-based system to reduce words to their stems. Both techniques together reduced the total vocabulary size to 170.000 words.

5.2 Dating Experiments

As a set of test documents to be dated, we chose other Dutch newspapers, *Trouw*, *Het Parool* and *NRC Handelsblad*, originating from the same time span as the reference corpus, and let the system pick a random – thereafter fixed – sample of in total 500 articles.

[5] The data stem from the so-called Twente-Corpus [13].

Dating method A which compares models on document level, was tested on this sample, varying in the number (n) of top-ranked documents used to aggregate the temporal profile. Because pretests indicated that small values of n are in general beneficial, we only tested cases with $n = \{1, 10\}$. In fact, choosing $n = 1$ is equivalent to co-dating a text with the most similar corpus document.

To test dating method B, which compares on the model level, we built four word-time matrices differing in time granularity, ranging from a rough partitioning in quarter-years down to a granularity of two days. Only in the latter case, we tried the idea of overlapping time partitions by moving a 4-day window in 2-day steps over the corpus time span. In order to keep the results comparable, the output granularity was kept throughout all experiments to quarter-years.

The experimental results presented in the next section will show the dating performance of method B (comparison at model level) for two different smoothing techniques: linear interpolation smoothing and Dirichlet-smoothing. In both cases the smoothing parameters (λ, μ) are set in such a way that smoothing effects remain minimal. For the document-based dating techniques we used only linear interpolation smoothing, but here with a higher value for λ. Though in general the documents are short, their length is stable compared to the size of temporal language models, so Dirichlet-smoothing was irrelevant here.

Finally, we also tested the expressiveness of the suggested dating confidence measure. The computed value,

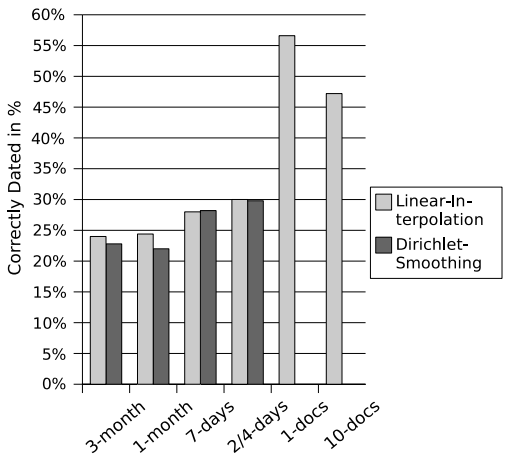
$$conf(\tau(X)) = -\left(\log \frac{score(C_j)}{score(C_i)}\right)$$

with C_i, C_j being the first and second ranked temporal language model, just reflects the idea to use the distance of the best scored time partition compared to the following as a confidence measure for the dating task. Within the tests we wanted to see whether the dating performance improves when a certain confidence threshold is required.

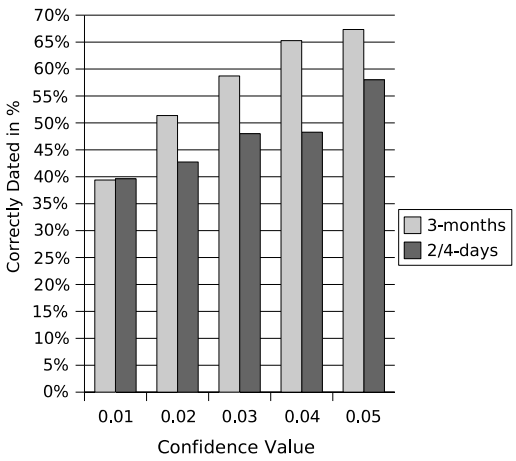
5.3 Results

Figure 2(a) shows the results of the dating experiments depending on model granularity. It displays the percentage of documents in the test set which were dated correctly. The last two bars represent the results using the direct document comparison of method A.

As said earlier, the proposed dating methods are based on word frequencies only and come with a certain error rate. However, we also see that temporal language models are far from being meaningless. Notice, that with 25 output partitions a random algorithm would only date 4% correctly.



(a) Performance of Model Granularity



(b) Performance at Confidence Level

Figure 2: Overview Dating Results

Furthermore Figure 2(a) makes clear that method A outperforms method B. An explanation for this might be that especially in the news domain articles reporting about the same event are likely to occur, and often in highly similar wordings. Therefore the similarity between an undated document and a reference document model tends to be much higher than between the undated document and a topically unbound temporal language model.

A second observation concerns the model granularity. Apparently it pays to work with the smallest possible time spans. This allows to interpret the superiority of method A in another way: the direct document comparison can also be regarded as working with the smallest possible time unit.

For the two smoothing techniques compared, only marginal differences were found. The result comes less as a surprise, since our newspaper corpus has a relatively balanced temporal distribution, so there is less room for positive effects from Dirichlet-smoothing.

Figure 2(b) shows that a higher confidence value correlates indeed with a better dating performance. Hence it can be concluded that *conf* is a useful additional metric. Still, even with a high confidence threshold, the dating reliability stays below 75%. Another important problem, not visible in the figures, is that the fraction of dating experiments gaining a confidence measure above the required threshold is decreasing rapidly. The correctly dated documents (almost 70%) at the left, are taken from the small fraction of dating experiments (approx. 10%) with a high confidence level. Here we also find an explanation for the observation that with courser model granularity more can be gained from using confidence measures. In fact, the filter effect just turned out to be higher in this case. So it is questionable whether the results on different granularities should be compared at all.

6 Conclusion and future work

We have shown that simple statistical methods can be used to model time in a large newspaper corpus. On the basis of the system for dating texts one of the next steps will be to develop techniques that can help to link modern Dutch queries to their historical equivalents and thereby support a wide group of users with an interest in historical textual collections and the objects linked to them.

There are several issue that could be explored further in future research. Time-word matrices such as Fig. 1(b) have a property that deserves some attention. Whereas the longitudinal sections are exactly the described temporal language models for time partitions, the cross sections can be interpreted as temporal *word profiles*, showing the usage pattern of a certain word over the corpus time span. Although we won't use temporal word profiles for the dating task, they can provide interesting information about language change. Computing the time frequency correlation on all word profiles, for instance, allows to search for the most trendy words. We could also extract highly time-specific words, by searching for word profiles with just one outstanding frequency peak. Finally, exploiting the full richness of n-gram models for more refined temporal models is part of our research agenda.

References

[1] C. Ahlber, B. Shneiderman. Visual information seeking: tight coupling of dynamic query filters with starfield displays. In *Proceedings of the ACM Conference on Human factors in computing systems (SIGCHI)*, pages 313 – 317. 1994.

[2] F. Diaz, R. Jones. Using Temporal Profiles of Queries for Precision Prediction. In M. Sanderson, et al. (eds.), *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 18–24. ACM, Sheffield, UK, 2004.

[3] C. Fellbaum. *WordNet: an electronic lexical database*. Speech, and Communication Series. MIT Press, 1998.

[4] G. Grefenstette. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, 1994.

[5] D. Hiemstra. A Linguistically Motivated Probabilistic Model of Information Retrieval. In *Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, pages 569–584. 1998.

[6] W. Kraaij. *Variations on language modeling for information retrieval*. Ph.D. thesis, University of Twente, Netherlands, 2004.

[7] W. Kraaij, F. de Jong. Transitive probabilistic CLIR models. In *Proceedings of RIAO 2004: Recherche d'Informations Assistee par Ordinateur*. 2004.

[8] V. Kumar, R. Furuta, R. Allen. Metadata Visualization for Digital Libraries: Interactive Timeline Editing and Review. In *Proceedings of the 3rd ACM conference on Digital libraries*, pages 126–133. 1998.

[9] D. Lawrie, W. Croft. Discovering and comparing topic hierarchies. In *Proceedings of RIAO*. 2000.

[10] Lesk M. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to tell a Pine Cone from an Ice Cream Cone. In *Proceedings of ACM SIGDOC*, pages. 24-26, 1986.

[11] X. Li, W. Croft. Time-Based Language Models. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM)*, pages 469–475. 2003.

[12] D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of COLING/ACL*. 1998.

[13] R. Ordelman. *Dutch Speech Recognition in Multimedia Information Retrieval*. PhD Thesis, University of Twente, Enschede, 2003.

[14] J. Ponte, W. Croft. A Language Modeling Approach to Information Retrieval. In *Proceedings of the 21st ACM Conference on Research and Development in Information Retrieval (SIGIR'98)*, pages 275–281. 1998.

[15] L. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In A. Waibel, K. Lee (eds.), *Readings in speech recognition*, pages 267–296. Morgan Kaufmann, 1990.

[16] M. Sanderson, W. Croft. Deriving concept hierarchies from text. In *Proceedings of ACM SIGIR Conference on Research and Advancements in Information Retrieval*. 1999.

[17] R. Swan, D. Jensen. Constructing Topic-Specific Timelines with Statistical Models of Word Usage. In *Proceedings of the 6th ACM Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 73–80. 2000.

[18] P. Turney. Mining the Web for synonyms. In *Proceedings of the Twelth European Conference on Machine Learning (ECML)*, pages 491–502. 2001.

[19] C. Zhai, J. D. Lafferty. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. In *ACM Trans. Inf. Syst.*, 22(2):179–214, 2004.