Principal Components for Automatic Term Hierarchy Building

Georges Dupret and Benjamin Piwowarski

Yahoo! Research Latin America
gdupret@yahoo-inc.com
bpiwowar@yahoo-inc.com

Abstract. We show that the singular value decomposition of a term similarity matrix induces a term hierarchy. This decomposition, used in Latent Semantic Analysis and Principal Component Analysis for text, aims at identifying "concepts" that can be used in place of the terms appearing in the documents. Unlike terms, concepts are by construction uncorrelated and hence are less sensitive to the particular vocabulary used in documents. In this work, we explore the relation between terms and concepts and show that for each term there exists a latent subspace dimension for which the term coincides with a concept. By varying the number of dimensions, terms similar but more specific than the concept can be identified, leading to a term hierarchy.

Keywords: Term hierarchy, principal component analysis, latent semantic analysis, information retrieval.

1 Introduction

Automated management of digitalized text requires a computer representation of the information. A common method consists in representing documents by a bag-of-words or set of features, generally a subset of the terms present in the documents. This gives rise to the vector space model where documents are points in an hyperspace with features as dimensions: The more important a feature in a document, the larger the coordinate value in the corresponding dimension [12].

Clearly, much information is lost when discarding the term order but the more significant limitation is that only the presence and the co-occurrence of terms are taken into account, not their meaning. Consequently, synonyms appear erroneously as distinct features and polysemic terms as unique features. This serious limitation is an avatar of the feature independence assumption implicit in the vector representation.

In the more general *statistical models* [20] (OKAPI) representations of queries and documents are clearly separated. Relevance of a document to a query is estimated as the product of individual term contributions. The corresponding assumption is not much weaker than strict independence.

Term dependence is taken into account in Language Models like *n-grams* and their applications to Information Retrieval [18], but generally within windows of

F. Crestani, P. Ferragina, and M. Sanderson (Eds.): SPIRE 2006, LNCS 4209, pp. 37–48, 2006.

two or three terms. The Bayesian network formalism [19] also allows for term dependence, but its application to a large number of features is unpractical.

Principal Component Analysis (PCA) [5] for text (and the related Latent Semantic Analysis method) offers a different approach: Uncorrelated linear combinations of features –the latent "concepts"– are identified. The lack of correlation is taken to be equivalent to independence as a first approximation, and the latent "concepts" are used to describe the documents. This work shows that more than a list of latent concepts, Principal Component Analysis uncovers a hierarchy of terms that share a "*related and more specific than*" relation.

Together with [7], this work extends the results of [6] beyond the context of Latent Semantic Analysis and PCA to all type of symmetric similarity measures between terms and hence documents and proposes a theoretical justification of the results. The main contribution, a method to derive a term hierarchy, is presented in Sect. 3. Numerical experiments in Sect. 4 validate the method while a review of automatic hierarchy generation methods is proposed in Sect. 5.

2 Term Similarity Measure

The estimation of the similarity between terms in Information Retrieval is generally based on term co-occurrences. Essentially, if we make the assumption that each document of a collection covers a single topic, two terms that co-occur frequently in the documents necessarily refer to a common topic and are therefore somehow similar. If the documents are not believed to refer to a single topic, it is always possible to divide them into shorter units so that the hypothesis is reasonably verified.

The Pearson correlation matrix $\mathbf{S}$ associated to the term by document matrix $\mathbf{A}$ is a common measure of term similarity. Nanas et al. [15] count the number of term co-occurrence in sliding windows of fixed length, giving more weight to pairs of terms appearing close from each other. Park et al. [17] use a Bayesian network. The method we present here does not rely on a particular measure of similarity or distance. The only requirement is an estimate of the similarity between any two index terms, represented by a symmetric matrix $\mathbf{S}$.

In the vector space representation of documents, index terms correspond to the base vectors of an hyperspace where documents are represented by points. If to each term j corresponds a base vector $\mathbf{e}_j$, an arbitrary document d is represented by $\mathbf{d} = \sum_{j=1}^{N} \omega_j \mathbf{e}_j$ where ω_j is the weight of term j in the document d. Weights are usually computed using the well known *tf.idf* formula and then normalized. The inconvenient of this representation stems from the implicit assumption of independence between terms: Consider two documents d_a and d_b each composed of a different single term. Independently of whether the single terms are synonyms, unrelated or antonyms, the documents similarity in the hyperspace representation is null because their representations coincide with two different base vectors. A more desirable result would be a non null similarity between terms u and v. This can be achieved by redefining the similarity measure

between documents: We will use the dot product in base $\mathbf{S}$ between the normalized document vectors[1].

$$\frac{\mathbf{d}_a^T}{|\mathbf{d}_a|}\mathbf{S}\frac{\mathbf{d}_b}{|\mathbf{d}_b|} = S_{u,v}$$

Alternatively, we can define an *extended representation* of a document d as $(1/|\mathbf{d}|)\mathbf{d}^T\sqrt{\mathbf{S}}$ and use the traditional cosine similarity measure[2].

The idea of introducing the similarity between terms to compute document similarity is closely related to Latent Semantic Analysis and Principal Component Analysis for text [6]. In the latter, the similarity between a set of documents and a query is computed as $\mathbf{r}(k) = \mathbf{A}^T\mathbf{S}(k)\mathbf{q}$ where $\mathbf{A}$ is the matrix formed by the space vector representation of the documents and q is a query. The i^{th} component of $\mathbf{r}(k)$, noted $r_i(k)$, is the similarity of document i with the query. The analogy with the *extended document representation* is clear, but instead of using the original similarity matrix $\mathbf{S}$, we use the rank k approximation of its eigenvalue decomposition. The matrix $\mathbf{S}$ can be decomposed into a product including the orthonormal matrix $\mathbf{V}$ and the diagonal matrix $\mathbf{\Sigma}$ of its eigenvalues σ_ℓ in decreasing value order: $\mathbf{S} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$. The best approximation following the Frobenius norm of the matrix $\mathbf{S}$ in a subspace of dimensionality $k < N$ is obtained by setting to zero the eigenvalues σ_ℓ for $\ell > k$. Noting $\mathbf{V}(k)$ the matrix composed of the k first columns of $\mathbf{V}$ and $\mathbf{\Sigma}(k)$ the diagonal matrix of the first k eigenvalues, we have $\mathbf{S}(k) = \mathbf{V}(k)\mathbf{\Sigma}(k)\mathbf{V}(k)^T$.

We can now represent in extended form a document $\mathbf{t}_u$ formed of a unique index term u in the rank k approximation of the similarity matrix:

$$\mathbf{t}_u^T = \mathbf{e}_u^T\sqrt{\mathbf{S}} = \mathbf{e}_u^T\mathbf{V}(k)\mathbf{\Sigma}(k)^{1/2} = \mathbf{V}_{u,\cdot}(k)\mathbf{\Sigma}(k)^{1/2} \quad (1)$$

where $\mathbf{\Sigma}(k)^{1/2}$ is the diagonal matrix of the square root of the eigenvalues in decreasing order and $\mathbf{V}_{u,\cdot}(k)$ is the u^{th} row of $\mathbf{V}(k)$. By analogy with the terminology introduced by Latent Semantic Analysis, the columns of $\mathbf{V}(k)$ represent latent concepts. The documents in general as well as the single term documents are represented with minimal distortion[3] as points in the k dimensional space defined by the k first columns – i.e. the eigenvectors – of $\mathbf{V}$ instead of the N dimensional space of index terms. This is possible only if the selected eigenvectors summarize the important features of the term space, hence the idea that they represent latent concepts.

In the next sections, we analyze the properties of the rank k approximation of the similarity matrix for different ranks and show how a hierarchy can be deduced.

3 The Concepts of a Term

We explore in this section the relation between terms and concepts. Sending a similarity matrix onto a subspace of fewer dimensions implies a loss of

[1] $\mathbf{S}$ being symmetric, but not necessarily full rank, this dot product introduces a quasi-norm [10].

[2] $\sqrt{\mathbf{S}}$ always exists because the singular values of $\mathbf{S}$ are all positive or null.

[3] according to the Frobenius norm.

information. We will see that it can be interpreted as the merging of terms meanings into a more general concept that encompasses them. We first examine the conditions under which a term coincides with a concept. Then we use the results to deduce a hierarchy.

A similarity matrix row $\mathbf{S}_{j,.}$ and its successive approximations $\mathbf{S}(k)_{j,.}$ represent a single term document $\mathbf{t}_j$ in terms of its similarity with all index terms. We seek a representation that is sufficiently detailed or encompass enough information for the term to be correctly represented. A possible way is to require that a single term document is more similar to itself than to any other term document:

Definition 1 (Validity). *A term is correctly represented in the k-order approximation of the similarity matrix only if it is more similar to itself than to any other term. The term is then said to be valid at rank k.*

If we remember that the normalized single term documents correspond to the base vectors, $\mathbf{e}_u$, the definition of validity requires: $\mathbf{e}_u^T\mathbf{S}(k)\mathbf{e}_u > \mathbf{e}_u^T\mathbf{S}(k)\mathbf{e}_v \ \forall u \neq v$ or equivalently $\mathbf{t}_u^T\mathbf{t}_u > \mathbf{t}_u^T\mathbf{t}_v \ \forall u \neq v$. This is verified if the diagonal term of $\mathbf{S}$ corresponding to u is larger than any other element of the same column, i.e. if $\mathbf{S}(k)_{u,u} > \mathbf{S}(k)_{u,v} \quad \forall v \neq u$. In other words, even though the diagonal element corresponding to term i is not equal to unity –which denotes perfect similarity by convention, it should be greater than the non-diagonal elements of the same row[4] to be correctly represented.

It is useful to define the rank below which a term ceases to be valid:

Definition 2 (Validity Rank). *A term* $\mathtt{t}$ *is optimally represented in the k-order approximation of the similarity matrix if it is valid at rank k and if* $k-1$ *is the largest value for which it is not valid. Rank k is the validity rank of term* $\mathtt{t}$ *and is denoted* $\mathrm{rank}(\mathtt{t})$.

In practice it might happen for some terms that validity is achieved and lost successively for a short range of ranks. It is not clear whether this is due to a lack of precision in the numerically sensitive eigenvalue decomposition process or to more fundamental reasons. The definition of validity was experimentally illustrated in [6] and a theoretical justification can be found in [2].

At a given rank k, if a term $\mathtt{a}$ is more similar to a valid term $\mathtt{c}$ than to itself, the representation of term $\mathtt{c}$ represents a meaning more general than $\mathtt{a}$: We say that $\mathtt{a}$ is generalised by the concept $\mathtt{c}$.

Definition 3 (Concept of a Term). *A term* $\mathtt{c}$ *is a concept of term* $\mathtt{a}$ *if* $\mathrm{rank}(\mathtt{c}) < \mathrm{rank}(\mathtt{a})$ *and if for some rank k such that* $\mathrm{rank}(\mathtt{c}) \leq k < \mathrm{rank}(\mathtt{a})$, $\mathtt{a}$ *is more similar to* $\mathtt{c}$ *than to itself.*

The requirement that $\mathrm{rank}(\mathtt{c}) < \mathrm{rank}(\mathtt{a})$ ensures that $\mathtt{a}$ is never a concept of $\mathtt{c}$ if $\mathtt{c}$ is a concept of $\mathtt{a}$.

It is possible to determine at each rank k the concepts of a term. To derive a hierarchy, we incrementally reconstruct the similarity matrix based on its decomposition $\mathbf{S} = \sum_{k=1}^{N} \sigma_k \mathbf{V}_{k,.}\mathbf{V}_{k,.}^T$. We collect for each k the links between concepts – i.e. terms whose representation is valid at rank k – and invalid terms.

[4] $\mathbf{S}(k)$ is symmetric and the condition can be applied indifferently on rows or columns.

Table 1. The first 30 direct links in *Shopping* and *Science* databases, ordered by decreasing coverage and limited to the stable links. Links in bold are in the ODP database.

Shopping		Science	
alberta	→ **canada**	**humidor**	→ **cigar**
monorail	→ lighting	**cuban**	→ **cigar**
criminology	→ sociology	**alberta**	→ **canada**
prehistory	→ **archaeology**	**cuckoo**	→ **clock**
romanian	→ slovenian	**grandfather**	→ **clock**
gravitation	→ relativity	fudge	→ chocolate
forensics	→ forensic	**soy**	→ **candle**
aztec	→ maya	**putter**	→ **golf**
karelian	→ finnish	**quebec**	→ **canada**
oceania	→ asia	racquetball	→ racket
transpersonal	→ **psychology**	**tasmania**	→ **australia**
etruscan	→ greek	airbed	→ mattress
barley	→ wheat	**glycerin**	→ **soap**
papuan	→ eastern	**snooker**	→ **billiard**
quebec	→ **canada**	**housebreaking**	→ **dog**
cryobiology	→ cryonics	waterbed	→ mattress
soho	→ solar	oceania	→ asia
catalysis	→ **chemistry**	tincture	→ herbal
geotechnical	→ engineering	**gunsmithing**	→ **gun**
iguana	→ **lizard**	chrysler	→ chevrolet
sociologist	→ **sociology**	equestrian	→ horse
olmec	→ maya	flamenco	→ guitar
oceanographer	→ **oceanography**	**pistachio**	→ **nut**
canine	→ dog	condiment	→ sauce
neptunium	→ plutonium	appraiser	→ estate
lapidary	→ mineral	**salsa**	→ **sauce**
raptor	→ **bird**	**ontario**	→ **canada**
ogham	→ irish	volkswagen	→ volvo
governmental	→ **organization**	arthropod	→ insect
forestry	→ forest	bulldog	→ terrier

There is a typically a range of ranks between rank(c) and rank(a) where a term a points to its concept c. This motivates the following definition:

Definition 4 (Coverage of a Link). *Define* $k_{\min}$ *and* $k_{\max}$ *as the minimum and maximum* k *that verify* $\text{rank}(\mathtt{c}) \leq k < \text{rank}(\mathtt{a})$ *and for which* c *is a concept of term* a*. The coverage of the link between the two concepts is the ratio*

$$\textit{coverage} = \frac{k_{\max} - k_{\min} + 1}{\text{rank}(\mathtt{a}) - \text{rank}(\mathtt{c})}$$

The coverage has values in $]0, 1]$.

The coverage reflects "how long" with respect to the possible range defined by rank(c) and rank(a), the valid term was a concept for the other term. We will see when we illustrate the hierarchy building procedure in Section 4 that the coverage is a good predictor of interesting links.

4 Numerical Experiments

There are no standard procedures to evaluate hierarchies although some attempts have been made [13]. Beyond the fact that evaluation is difficult even

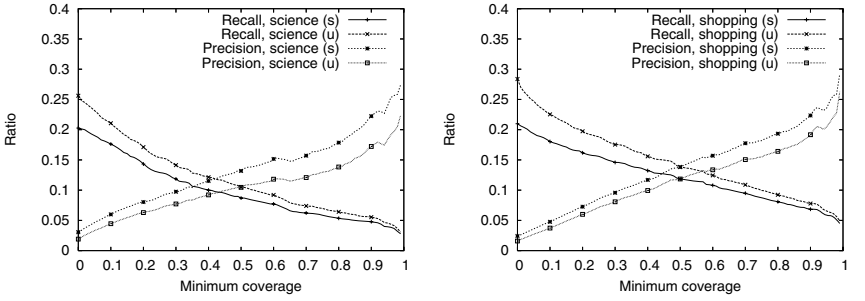

Fig. 1. Comparison of the PCA stable (s) and unstable (u) links with the ODP hierarchy on the Science and Shopping topics. *Recall* is the proportion of links in the original ODP hierarchy rediscovered by PCA. *Precision* is the proportion of ODP links among those retrieved by PCA. The x-axis is the coverage ratio: For a given value c, the PCA links we consider are those whose coverage is superior to c.

when a group of volunteers is willing to participate, it also depends on the task the hierarchy is designed for. For example, the measure used in [13] could not be applied here as the scoring is based on an estimate of the time it takes to find all relevant documents by calculating the total number of menus –this would be term nodes in this work– that must be traversed and the number of documents that must be read, which bears no analogy to this work.

We expect PCA to uncover two main types of relation between terms: The first one is semantic and can be found in dictionaries like WordNet[5]. These are relations that derives from the definition of the terms like "cat" and "animal" for example. The other kind of relation we expect to uncover is more circumstantial but equally interesting like, for example, "Rio de Janeiro" and "Carnival". These two words share no semantic relation, but associating them make sense. To evaluate the PCA hierarchy, we chose to compare the links it extracts from the document collection associated with the Open Directory Project[6] to the original, edited hierarchy. To identify the ability of PCA to extract "semantic" relations, we performed some experiments with WordNet which are not presented in this article due to a lack of space.

The *Open Directory Project* (ODP) is the most comprehensive human edited directory of the Web. We extracted two topics from this hierarchy, namely *Shopping* and *Science*. Out of the 104,276 and 118,584 documents referred by these categories, we managed to download 185,083 documents to form the database we use.

Documents were processed with a language independent part-of-speech tagger[7] and terms replaced by their lemmata. We extracted only adjectives and substantives to form the bag-of-word representations. Low and high frequency

[5] `http://wordnet.princeton.edu/`

[6] www.dmoz.org

[7] the TreeTagger home page can be found at `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/`

terms as well as stopwords were discarded unless they appeared in the ODP hierarchy. Original documents were divided in parts of 25 consecutive terms to form new, shorter documents. The objective is to reduce the confusion of topics inside a same document (Section 2).

A path in the ODP hierarchy is composed as a series of topics, from the most generic to the most specific. An example of such a path is "Health/Beauty/-Bath_and_Body/Soap". We discard concepts described as a sequence of terms. For example, the previous sequence is transformed into "Health/Beauty/Soap". The hierarchy is then decomposed into direct links – i.e. relations that exist between adjacent terms – and indirect links where relations between terms belong to the same path. The direct links in our example are Health ← Beauty and Beauty ← Soap and the set of transitive links is composed of the former links more Health ← Beauty.

In order to test the stability of the discovered links, we *bootstrapped* [8] the document database. The method consists in picking randomly with replacement 185,083 documents from the original database to form a new correlation matrix before deducing a new set of links. This process is repeated ten times. The number of replications where a particular link appears reflects its stability with respect to variations in the database. We say that a link is *stable* when the relationship between the two terms held the ten times, and in the opposite case it is said to be *unstable*. For the science and shopping topics, half of the links are stable.

From this set of links between two terms we can construct a hierarchy of terms. Although cycles can appear among unstable links, they are absent by construction from the stable links. It would also be interesting to consider links that always appear in each bootstrap replication but with a different direction: This could be a good indicator of a symmetric relationship between two concepts.

With respect to the complexity of the algorithm, the term by term matrix is not sparse and the computation of the singular value decomposition is of order $O(n^3)$. This becomes rapidly intractable on regular desktop PC unless the number of terms is restricted to a range of between 5.000 and 10.000 terms and less frequent terms are discarded. This need not be a problem, given that the similarity of infrequent terms will be poorly estimated anyway.

In the remaining of this section, we compute the proportion of direct and indirect links present in ODP that we retrieve automatically with our Principal Component Analysis method. We also study the impact of link stability and coverage (Definition 4). Note that a large intersection between human and automatically generated links increases the confidence on the validity of the automatic method, but it does not invalidate the automatic links absent from edited hierarchy because documents and topics can be organized in a variety of equally good ways. This is corroborated in Table 1 where links absent from ODP are in normal font.

4.1 Coverage and Stability of Direct Links

Coverage is perceived as a relevant indicator of link quality because it reflects the strength that unite the two terms linked by a hierarchical relation. In Table 2,

Table 2. Number of links discovered by PCA in *Science* documents as a function of the coverage and, in parenthesis, the size of the intersection with the 2,151 *Science* ODP links

coverage	stable	unstable
0%	14,266 (436)	28,859 (551)
20%	3,832 (308)	5,850 (368)
40%	1,867 (251)	2,831 (261)
60%	1,095 (166)	1,676 (198)
80%	644 (115)	998 (138)
99%	218 (59)	294 (65)

the number of links discovered from the *Science* documents are reported as a function of the minimum coverage in both the stable and unstable cases. We see that 70% and 80% of the links have a coverage lower than 20%. Discarding all the links below this level of coverage results in the lost of only 30% and 33% of ODP links.

The stability is also an important selection criteria. We observe that if we consider all the PCA links, stable or not, we retrieve 551 of the original 2,151 ODP links present in Science. If we select only the stable links, we retrieve 436 ODP links, but the total number of PCA links is divided by two from 28,859 to 14,266. Some of the links present in the ODP hierarchy are lost, but more than half of the PCA links are discarded. A similar conclusion holds when varying the coverage minimum threshold. This justifies stability as an important criteria for selecting a link.

By analogy with the Information Retrieval measures, we define *recall* as the proportion of links in the original hierarchy that the PCA method manages to retrieve automatically. The *precision* is defined as the proportion of ODP links present in the set of PCA links. If we denote by `H` the set of links in the ODP human edited hierarchy and by A the set in the PCA automatic hierarchy, these measures become $recall = |H \cap A|/|H|$ and $precision = |H \cap A|/A$. Recall answers the question "How many ODP link do I retrieve automatically?", while precision answers "What is the concentration of ODP links among all the PCA links?"

Fig. 1 offers a global view of the impacts of stability and coverage on recall and precision for topics *Science* on the left and *Shopping* on the right. The portion of common links is significantly larger when the coverage is closer to its maximum. On both graphs, if we select only links with a coverage superior to 0.8, one tenth of the links in A are present in ODP. These results are good since the number of links in ODP is quite high in comparison with the number of relevant documents in the ad-hoc task of Information Retrieval, thus penalizing the recall. Moreover, ODP is not a gold standard and links not present in this hierarchy might still be useful.

When varying the coverage threshold from 0 to 1, precision increases and recall decreases almost always. This means that coverage is a good predictor of the link "relevance". This was verified empirically as well by inspecting some part of the discovered links ordered by coverage.

Summarizing, stability and coverage are both important predictors of link quality and PCA is able to identify a significant number of ODP links.

4.2 Transitive Links

Some links present in ODP might appear as combination of links in PCA and vice-versa. We already explained how ODP was processed to obtain these links. For PCA, we create a link between two terms if there is a path from one term to another. A link is said to be direct if it appears in the original hierarchy, and indirect if it was discovered by transitivity. A set of links is transitive if it includes both direct and indirect links.

The coverage being a good indicator of the link quality, we tried to extend this notion to transitive links. We found experimentally that the minimum coverage of all the traversed links led to the best results: An indirect link is penalized if all the paths between the two terms traverse a link with a low coverage.

A study of the effect of coverage and stability on precision and recall is reported in Fig. 2 where we aggregated the results over the science and shopping topics, and compared the direct and transitive ODP and PCA links. The results being similar for both topics, there is no need to treat them separately. Precision and recall when both links set are either transitive or direct (PCA, ODP and PCA+, ODP+ curves on Fig. 2) are very similar: This shows that precision is not much affected by the new PCA indirect links (around 38% more links, from 31,611 to 43,632) while recall is not much affected by the new ODP links (around 126% more links, from 4,153 to 9391). It is interesting also to observe that among the 2,297 links common to the transitive PCA and ODP sets, 1,998 are present in the direct PCA set. This is reflected on Fig. 2 (ODP+, PCA plot) where the corresponding precision curve is significantly superior while recall is less affected. This suggests that the indirect links of PCA did not contribute much.

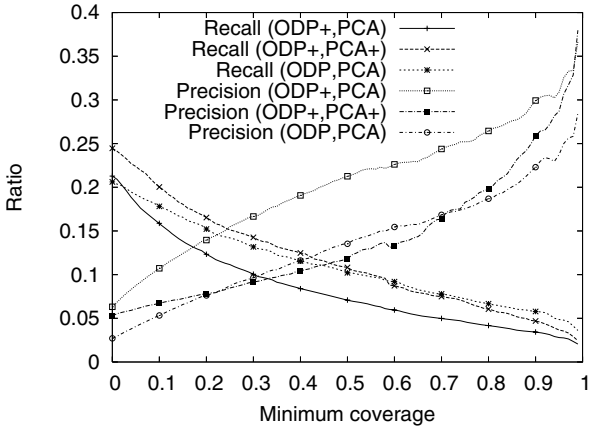

Fig. 2. Comparison of the PCA direct links (PCA) and transitive links (PCA+) with the ODP direct links (ODP) and transitive links (ODP+)

In conclusion, the new definition of coverage as the minimum on the path of traversed links proves a good selection indicator, as the precision increases with the coverage threshold. The manually derived and the PCA hierarchies share a significant amount of links and it seems that PCA is successful in discovering relations between terms. This is a specially good results given that the ODP directory is only one among numerous possible ways of organizing the documents in the database.

5 Related Work

Different fully automatic hierarchy discovery methods have already been proposed. The most popular one, from Sanderson and Croft [21], uses the co-occurrence information to identify a term that subsumes other terms. We tried various values of the unique parameter without succeeding in getting acceptable results. We suspect that a part of the problem stems from the heterogeneity of the corpus we used.

Njike-Fotzo and Gallinari [16] cluster documents prior to applying the Sanderson and Croft algorithm. This probably helps and will be used in future works. Nanas et al. [15] also proposed a method similar to Sanderson and Croft, but a subsumption relation is accepted if the terms involved are also correlated. The correlation is measured for terms appearing in windows of fixed length, and depends on the distance between them.

Hyponymy relations are derived from lexico-syntactic rules rather than plain co-occurrence in [11]. Another approach is to rely on frequently occurring words within phrases or lexical compounds. The creation of such *lexical hierarchies* has been explored and compared with subsumption hierarchies in [13]. In addition to the above two approaches, the same authors have investigated the generation of a concept hierarchy using a combination of a graph theoretic algorithm and a language model.

Glover et al. [9] base their hierarchy discovering algorithm on three categories: If a term is very common in a cluster of documents, but relatively rare in the collection, then it may be a good "self" term. A feature that is common in the cluster, but also somewhat common in the entire collection, is a description of the cluster, but is more general and hence may be a good "parent" feature. Features that are common in the cluster, but very rare in the general collection, may be good "child" features because they only describe a subset of the positive documents.

Application of traditional data mining and machine learning methods have also been tested. In [14], the learning mechanism is based on the Srikant and Agrawal [22] algorithm for discovering generalized association rules. A Bayesian network approach is proposed in [17]. Hierarchical clustering algorithm [1, 3] can be used to derive relations between terms, but cluster labelling is a challenging task. In [4] clustering is explicitly used to derive synonyms, hyperonyms and hyponyms relations.

6 Conclusion

We showed that the term similarity matrix induces a hierarchical relation among the terms. We computed this hierarchy based on the set of documents associated with two topics of the Open Directory Project hierarchy and observed significant similarities with the human edited original hierarchy.

We investigated different selection criteria and identified stability and coverage as good predictors of link quality. The coverage is especially interesting since it allows to order the links prior to selection. We also studied transitive links and showed that it is possible to extend to them the notion of coverage.

In conclusion, we observe that the hierarchy discovered by PCA is surprisingly good, especially if one considers only the stable links with a high coverage. The vast majority of links make sense and relations are uncovered than one would not expect to deduce from a simple co-occurence representation of documents.

References

1. L. D. Baker and A. K. McCallum. Distributional clustering of words for text classification. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *SIGIR-98, 21st ACM*, pages 96–103, Melbourne, AU, 1998. ACM Press, New York, US.
2. H. Bast and D. Majumdar. Understanding spectral retrieval via the synonymy graph. In *SIGIR-05, 28th ACM*, 2005.
3. C. Y. Chung and B. Chen. Cvs: a correlation-verification based smoothing technique on information retrieval and term clustering. In *KDD '02: Eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 469–474, New York, NY, USA, 2002. ACM Press.
4. C. Y. Chung, R. Lieu, J. Liu, A. Luk, J. Mao, and P. Raghavan. Thematic mapping - from unstructured documents to taxonomies. In *CIKM '02*, pages 608–610, New York, NY, USA, 2002. ACM Press.
5. S. Deerwester, S. Dumais, G. Furnas, and T. Landauer. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41:391–407, 1990.
6. G. Dupret. Latent concepts and the number orthogonal factors in latent semantic analysis. In *SIGIR-03, 26th ACM*, pages 221–226. ACM Press, 2003.
7. G. Dupret and B. Piwowarski. Deducing a term taxonomy from term similarities. In *Second International Workshop on Knowledge Discovery and Ontologies, Porto, Portugal*, 2005.
8. B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, May, 15 1993.
9. E. Glover, D. M. Pennock, S. Lawrence, and R. Krovetz. Inferring hierarchical descriptions. In *CIKM '02*, pages 507–514, New York, NY, USA, 2002. ACM Press.
10. D. Harville. *Matrix Algebra from a Statistician's Perspective*. Springer-Verlag, New York, 1997. 14.
11. M. A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *14th conference on Computational linguistics*, pages 539–545, Morristown, NJ, USA, 1992. Association for Computational Linguistics.

12. W. P. Jones and G. W. Furnas. Pictures of relevance: a geometric analysis of similarity measures. volume 38, pages 420–442, New York, NY, USA, 1987. John Wiley & Sons, Inc.
13. D. Lawrie and W. Croft. Discovering and comparing topic hierarchies. In *Proceedings of RIAO 2000*, 2000.
14. A. Maedche and S. Staab. Discovering conceptual relations from text. pages 321–325, 2000.
15. N. Nanas, V. Uren, and A. D. Roeck. Building and applying a concept hierarchy representation of a user profile. In *SIGIR-03, 26th ACM*, pages 198–204, New York, NY, USA, 2003. ACM Press.
16. H. Njike-Fotzo and P. Gallinari. Learning generalization/specialization relations between concepts - application for automatically building thematic document hierarchies. In *RIAO 2004*, Apr. 2004.
17. Y. C. Park, Y. S. Han, and K.-S. Choi. Automatic thesaurus construction using bayesian networks. In *CIKM '95*, pages 212–217, New York, NY, USA, 1995. ACM Press.
18. J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR-98, 21st ACM*, pages 275–281, New York, NY, USA, 1998. ACM Press.
19. B. A. N. Ribeiro and R. Muntz. A belief network model for ir. In *SIGIR-96: 19th ACM*, pages 253–260, New York, NY, USA, 1996. ACM Press.
20. S. Robertson and K. S. Jones. Simple proven approaches to text retrieval. Technical report tr356, Cambridge University Computer Laboratory, 1997.
21. M. Sanderson and B. Croft. Deriving concept hierarchies from text. In *SIGIR-99, 22th ACM*, pages 206–213, New York, NY, USA, 1999. ACM Press.
22. R. Srikant and R. Agrawal. Mining generalized association rules. *Future Generation Computer Systems*, 13(2–3):161–180, 1997.