Combining Statistics and Semantics for Word and Document Clustering

Alexandre Termier *, Marie-Christine Rousset *, Michèle Sebag †

Abstract

A new approach for constructing pseudo-keywords, referred to as Sense Units, is proposed. Sense Units are obtained by a word clustering process, where the underlying similarity reflects both statistical and semantic properties, respectively detected through Latent Semantic Analysis and WordNet.
Sense Units are used to recode documents and are evaluated from the performance increase they permit in classification tasks.
Experimental results show that accounting for semantic information in fact decreases the performances compared to LSI standalone.
The main weakenesses of the current hybrid scheme are discussed and several tracks for improvement are sketched.

1 Introduction

This paper focuses on document description and clustering. Learning and mining techniques meet particular difficulties when dealing with textual information [14]. These difficulties are related to the structured nature of texts (grammar), which requires advanced techniques to be accounted for; unfortunately, such techniques (e.g. syntactic analysers) are not yet as robust as desirable, and entail a non-negligible amount of noise. This is the reason why so many efficient approaches (see [3; 13] among many others) actually only rely on bag-of-words representation, even if this representation does not capture the whole semantics contents of a corpus (text set).

Canonical bag-of-word representations present several characteristics that adversely affect statistical approaches [17]. One is the huge number of attributes (number of words in the corpus, or dictionary size), and the fact that any text actually uses a small fraction of the dictionary. In other words, a text is a vector in a high-dimension space (each dictionary word corresponds to a dimension), and most of its components are equal to zero. Furthermore, a single dimension (word) might correspond to more than one semantic notion, due to polysemy; and conversely, distinct dimensions might correspond to same notion (synonymy).

An important research topic thus is to design new and better text descriptions (using word-windows [4] or syntactic analysis [5; 15]), such that semantically relevant patterns would correspond to statistically emergent ones, and *vice versa*. These approaches, which will be discussed in more detail in Section 5, proceed by specializing the texts, using adjacency relations [4] or syntactical taggers [15]. Such a specialization hopefully alleviates the polysemy effects. Still, it offers no remedy regarding the synonymy effects, and the resulting sparseness of the text distribution.

In this paper, we present a new approach for text description and clustering, termed *Semistics* for SEMantico–statISTICal System. *Semistics* involves the automatic construction of pseudo-keywords, which are bags of words referred to as *Sense Units*. Ideally, Sense Units allow for a synonymy and polysemy-free description of documents; furthermore, the number of SUs is controlled by the user in order to guarantee the scalability of the approach.

Sense Units are word clusters constructed by a preliminary clustering stage operating at the word level, using a standard distance-based clustering algorithm (Hierarchical Agglomerative Clustering). The novelty lies in the similarity employed, which combines statistical and semantic information. The statistical ingredient is borrowed from Latent Semantic Indexing [1] while the semantic one is provided by WordNet [7].
LSI achieves a statistical compression of the data based on a Singular Value Decomposition technique. This allows LSI to detect connections between words even though they never co-occur in a document, as opposed to window-based approaches [4] : words employed in a same context are found similar even though they are not employed together (Harrisian hypothesis).
WordNet is a publically available linguistic resource providing a thesaurus which organizes English words into sets of synonyms, termed *synsets*. A synset groups all words with same sense. Polysemy is accounted for by the fact that a word can appear in several synsets. In summary, WordNet can be viewed as a source of general domain knowledge about words.

Even though LSI is reasonably good at guessing synonyms or disambiguating words, there is no doubt it is outperformed

*LRI, CNRS UMR 8623, Bât. 490 Université de Paris-Sud 91405 Orsay Cedex, {termier, mcr}@lri.fr

†LMS, CNRS UMR 7649, Ecole Polytechnique 91128 Palaiseau Cedex, sebag@cmapx.polytechnique.fr

by WordNet in this respect. On the other hand LSI sees each document in the perspective of the corpus, limited to the application domain and vocabulary. In summary, WordNet provides a very general domain knowledge about words, while LSI constructs a specific, corpus-driven knowledge about words, expressed as a semi-distance.

The paper investigates how to combine both sources of knowledge in order to create an accurate and yet understandable description of the corpus, the Sense Units. The Sense Units are intended both to sustain an efficient distance-based clustering process, and to provide the user with many and simple opportunities to inspect the results and include extra knowledge.

Sense Units ideally correspond to the nodes in an ontology. The difference is that an ontology is structured according to logical relations (*is-a, part-of* relations), while Sense Units are constructed together with a similarity function, i.e. they are structured in a topological sense.

The paper is organized as follows. Section 2 provides an overview of *Semistics* ; it details the construction of Sense Units and how these are used to redescribe the texts. Section 3 gives the experiment goal and setting. Section 4 reports on the *Semistics* results obtained on the well-studied benchmark Reuters [22], and a real-world application concerned with the clustering of XML Document Type Definitions (DTD) [16]. Section 5 briefly reviews and discusses some related work, and the paper ends with some perspectives for further research.

2 Overview of the system

The input of the system is a collection D of documents, viewed as bags of words (subsets of the word set $\mathcal{W}$). It is worth keeping in mind that these words and documents are not necessarily "natural", in the sense that they might be produced by another text mining tool (this will be detailed in Section 3).

The data undergoes six stages:

1. The cleaning - downsizing of the data: deterministic and stochastic filters are used, in order to respectively remove poorly meaningful words, and keep the problem size under control,
2. The statistical reduction of the data through LSI, visible through a numerical word similarity $\ell_w(words, words)$.
3. The definition of synsets (set of synonyms) through WordNet, and the creation of a numerical synset similarity $\ell_s(synsets, synsets)$.
4. The creation of Sense Units, which are clusters of synsets. The synset clustering is a simple Hierarchical Agglomerative Clustering [21] based on similarity ℓ_s, and involving a specific stop criterion.
5. The redescription of all documents as vectors on the Sense Unit set. A cosine-based distance, noted $\ell_d(documents, documents)$ hence follows.
6. The evaluation of distance ℓ_d is then performed as detailed in Section 3.

2.1 Pre-processing

As in most real-world applications [6], the importance of document pre-processing cannot be underestimated.

Semistics first achieves a standard deterministic filter of poorly useful words. Such are stop-words (e.g. *the, is, for*). Words that are either too frequent[1] or too rare are difficult to exploit, too, from a statistical perspective. This deterministic filter is thus governed from a dictionary of stop-words, and two frequency thresholds f_{min} and f_{max} supplied by the expert.

Document pre-processing commonly reduces the vocabulary size to ensure the tractability of further computational issues. In a purely deterministic approach, this reduction is often based on using tight frequency thresholds, regrettably filtering out some highly relevant though rare words.

The alternative explored by *Semistics* is based on a stochastic filter. The expert supplies the total number of words to be considered further. The set $\mathcal{W}$ of words actually considered therafter is constructed by uniformly sampling the initial words whose frequency belongs to the desired interval ($f_w \in [f_{min}, f_{max}]$).

This hybrid deterministic-stochastic pre-process is intended to offer the expert a better chance to control the trade-off tractability *vs* representativity (or compression *vs* diversity) of the data representation. It is further expected that this reduction will better preserve the description of documents with mostly rare words.

Formally, pre-processing starts with a set of documents and constructs the documents $\times$ words matrix $\mathcal{M}$, where $\mathcal{M}_{ij}$ denotes the presence (1) or absence (0) of word w_j in document[2] d_i. Column i in $\mathcal{M}$ gives a vectorial description of word i, noted $\mathcal{M}_{i\star}$. Symetrically, row j in $\mathcal{M}$ gives a vectorial description of document j, noted $\mathcal{M}_{j\star}$. Note that this representation induces a natural similarity between respectively words and documents, by taking the vector cosine.

2.2 Latent Semantic Indexing

Latent Semantic Indexing (LSI) achieves a statistical compression of documents $\times$ words matrix $\mathcal{M}$, based on a singular value decomposition of $\mathcal{M}$ [3]:

$$\mathcal{M} = U \, \Lambda \, V$$

where Λ is a diagonal matrix ($\Lambda = diag(\lambda_1, ..\lambda_d)$, with $\lambda_i \geq \lambda_{i+1}$). The compression is achieved through cancelling out the smallest eigenvalues λ_i, $i > d'$ ($\Lambda' = diag(\lambda'_1, ..\lambda'_d)$, $\lambda'_i = \lambda_i \times \mathbb{1}_{i \leq d'}$), and considering the compressed matrix $\mathcal{M}'$:

$$\mathcal{M}' = U \, \Lambda' \, V$$

LSI differs from Principal Component Analysis [12] in two respects. First of all, LSI operates on the documents $\times$ words matrix $\mathcal{M}$, whereas PCA considers the word covariance matrix. Second, PCA cancels all but a few eigenvalues

[1] The frequency f_w of word w is computed as the fraction of documents d_w that contain w : $f_w = \frac{|\{d_w\}|}{|D|}$.

[2] A refinement is to set $\mathcal{M}_{ij}$ to the log-entropy of word w_i wrt document d_j [3]

$(d' = 2, 3)$, which allows for mapping the data in a 2- or 3-D space, enabling a visual detection of the word clusters.

On the contrary, LSI retains a significantly higher number of eigenvalues (set to $d = 100$ in the following). Any visual inspection is therefore forbidden. However, matrix $\mathcal{M}'$ gives an extended and saturated description of the documents; the contribution $\mathcal{M}'_{ij}$ of word w_j to document d_i is raised if w_j co-occurs frequently with the very words in d_i, even though w_j was not actually present in d_i. In this respect, $\mathcal{M}'$ can be viewed as a smooth "transitive closure" of the initial description $\mathcal{M}$.

This saturation effect might explain the fact that euclidean-based approaches are more robust when applied on $\mathcal{M}'$ instead of $\mathcal{M}$ [20]. Consider the descriptions of words w_i and w_j according to $\mathcal{M}'$, given as the i^{th} and j^{th} columns of $\mathcal{M}'$, further referred to as ${}^w\mathcal{M}'_i$ and ${}^w\mathcal{M}'_j$. LSI thus induces a similarity ℓ_w between words, defined as the cosine of ${}^w\mathcal{M}'_i$ and ${}^w\mathcal{M}'_i$:

$$\ell_w(w_i, w_j) = \frac{< {}^w\mathcal{M}'_i,\ {}^w\mathcal{M}'_j >}{||{}^w\mathcal{M}'_i||\ ||{}^w\mathcal{M}'_j||}$$

A dual similarity is likewise defined between documents, enabling the use of any distance-based clustering algorithm. Experimentally, it is observed than ℓ_w performs better (in a word disambiguation context) than the cosine similarity based on the initial matrix $\mathcal{M}$.

Notably, LSI is highly scalable with respect to the number of documents and words considered, due to sophisticated decomposition methods exploiting the sparsity of $\mathcal{M}$ (e.g. applications in the TREC context have considered up to several data gigabytes; our database is about 3 MB large).

2.3 Coupling LSI and WordNet

The success of LSI on several text mining tasks, e.g. word disambiguation or essay rating [8], confirms indeed that (restricted-scope) semantic information can be extracted from statistical estimates.

However, it is worth noting that sources of partial semantic information are commonly available. The resource we use in the following is WordNet, that is an electronic lexical database enriched with conceptual-semantic relations (linking concepts) and lexical relations (linking individual words) which is publicly available [10]. The use of WordNet for text mining has been investigated in several respects, e.g. supporting text retrieval through query expansion [19], or achieving sophisticated spell checking through word sense disambiguation [9].

It seems worth combining the complementary knowledge conveyed by statistical estimates and WordNet semantic relations. The question is how. A previous approach uses a tagged corpus to enrich WordNet relations with distributions [2].

Our approach does not require the preliminary and tedious labelling of the word senses in the corpus. Rather, *Semistics* looks for all senses, according to WordNet, associated to any word in the word set $\mathcal{W}$. To each such word sense $w.s$ it associates a synset $\mathcal{S}_{w.s}$, defined as the set of all words w' which are synonymous to $w.s$; it is further required that w' co-occurs with w in at least one document in the corpus (e.g. $\mathcal{S}_{work.4}$ = {*study, work, learning, acquisition* }). Note that $\mathcal{S}_{w.s}$ might be reduced to $\{w\}$; this typically happens when w is not recognized by WordNet (e.g. company names).

The similarity ℓ_s between synsets is defined as the average similarity of the words they contain.

$$\ell_s(\mathcal{S}_1, \mathcal{S}_2) = \frac{\sum_{w_1 \in \mathcal{S}_1,\ w_2 \in \mathcal{S}_2} \ell_w(w_1, w_2)}{|\ \mathcal{S}_1\ | \times |\ \mathcal{S}_2\ |}$$

Only synsets are considered thereafter. Interestingly, though individual words usually belong to many synsets due to polysemy, the number of synsets is close to the number of words due to construction requirements.

2.4 Constructing Sense Units

Similarity ℓ_s is exploited through a standard bottom-up clustering algorithm, namely Hierarchical Agglomerative Clustering (HAC). HAC starts with a set of singleton clusters, each one containing exactly one synset. At each step, the two most similar clusters are merged into a single one. The similarity of two clusters is the average similarity of the synsets they contain.

HAC produces a partition of the synsets into disjoint clusters, which strongly depends on the termination criterion. A first possibility is to set the desired number n of clusters, such that HAC stops after performing $s - n$ merging steps, with s denoting the number of synsets. Another possibility is to set a minimal similarity threshold, such that HAC stops when the current best similarity is lower than the threshold.

However, these stop criteria hardly cope with the varying granularity of natural language concepts; the similarity threshold should typically depend on the local density of the concepts. We therefore propose an adaptive criterion, based on controlling the cluster coverage and its growth. Let the coverage of cluster $\mathcal{C}$ be the number of documents in the corpus containing at least one word w belonging to some synset in $\mathcal{C}$. Merging clusters $\mathcal{C}$ and $\mathcal{C}'$ is said to be *admissible* iff their relative overlap is above a prescribed threshold τ, referred to as *growth limit*:

$$\frac{coverage(\mathcal{C}) \bigcap coverage(\mathcal{C}')}{min(coverage(\{\mathcal{C}\}), coverage(\{\mathcal{C}'\}))} > \tau$$

The idea is that, if the coverage of the cluster abruptly grows, the underlying concept is becoming exceedingly general.

Finally at each step HAC merges the most similar clusters such that their merge is admissible, until no more merge is admissible. The complexity is cubic in the number of synsets and linear in the number of documents.

Each cluster so constructed is a set of words labelled with their sense, referred to as *Sense Unit*. An example of sense unit constructed from the Reuters corpus (Section 4) is {*protests, protestings, leftist* }. Sense units containing a single word are filtered out.

Let $\mathcal{U}$ denote the set of sense units, the size of which is u. Note that u is indirectly controlled from threshold τ; experimentally, the number of sense units is lower than the number of synsets by two orders of magnitude.

2.5 Document Redescription and Clustering

Each document is redescribed with respect to the sense units, and mapped onto $\mathbb{R}^u$. The contribution of sense unit U_j to document d_i, noted $\mathcal{M}''_{ij}$ is computed as

$$\mathcal{M}''_{ij} = \begin{cases} \frac{1}{F(U_j)} & \text{if there exists } w \in d_i \text{ such as } w \in U_j \\ 0 & \text{otherwise} \end{cases}$$

where $F(U_j)$ stands for the frequency of sense unit U_j (number of documents containing at least one word in U_j).

¿From the mapping of documents onto the metric space $\mathbb{R}^u$ we derive a document similarity noted ℓ_d, given as the cosine of $\mathcal{M}''$ vectors:

$$\ell_d(w_i, w_j) = \frac{< \mathcal{M}''_{i\star}, \mathcal{M}''_{j\star} >}{||\mathcal{M}''_{i\star}|| \, ||\mathcal{M}''_{j\star}||}$$

3 Experiment goal and setting

This section details the questions experiments should enable to address, the performance criteria, and our experiment setting.

3.1 Experiment goal

We compare three (re-)descriptions of the corpus. The first description simply involves the initial words in the documents. The second description, built from the first one, is based on LSI: the "descriptors" of the documents are made of LSI eigenvectors (implicitly derived from the LSI eigenvalues). The third description, which is used by *Semistics* , relies on the Sense Units. These descriptions are materialized respectively by matrices $\mathcal{M}$, $\mathcal{M}'$ and $\mathcal{M}''$.

Each description is processed by the same similarity-based clustering algorithm (HAC): the document similarity is the standard cosine of row vectors in matrix $\mathcal{M}$, $\mathcal{M}'$, or $\mathcal{M}''$.

Evaluating a description ultimately amounts to evaluate the relevance of the provided clusters, and the flexibility of the re-description/clustering process (through diverse parameters such as word sampling rate, number of LSI eigenvalues retained, growth limit τ in *Semistics*).

3.2 Criteria

The difficulties of evaluating a clustering process have long been discussed [18; 22]. As many authors [22], we finally retain the classification predictive accuracy, derived from the 1-nearest and the 20-nearest neighbor classifier using the considered similarity.

So we will not evaluate the quality of the clusters produced, but rather the quality of the similarity measure leading to that clusters.

The considered data is a subset of the Reuters corpus, where the document class is given as the value of the attached field *Topics*. Following [22], documents with field *Topics* not informed are rejected; furthermore, we also reject documents attached to several *Topics*.

The number of documents is 8,842, partitioned in 135 disjoint classes, and involving about 40,000 words.

The quality of a description is finally estimated from the predictive accuracy of the 1-nearest neighbor (or 20-nearest neighbor) classifier. We have two different ways to produce a measure :

- A standard leave-one-out test process: each document is taken as correctly classified (legend $\%OK$) iff its nearest neighbor (or the majority of its 20 nearest neighbors) is labelled with the same *Topics* ;
- A less demanding evaluation of the description quality, obtained by considering that a document is reasonably classified (legend $\%OK$-rel) if its nearest neighbor (or the majority of its 20 nearest neighbors) falls in the same category for at least one of the six main fields qualifying the documents (*Location, People, Orgs,* ....).

3.3 Experiment setting

The baseline experiment considers all 40,000 words in the corpus.
The LSI-baseline involves 100 eigenvectors, built on the same 40,000 words.

In *Semistics* , a first sampling is effectuated on the corpus, retaining 4,000 among the 40,000 words. LSI is applied on the sampled description, and finally the Sense Units are built by combining the LSI and Wordnet techniques. The growth limit τ is set to 0.7.

Due to this sampling step, *Semistics* might be considered as a randomized algorithm, and experimental results should therefore be averaged over a number of independent runs. Unfortunately and due to the total computation time needed, results presented in the following will be based on a single run.

4 Results

It is observed that *Semistics* finally produces 634 Sense Units. As this might be insufficient to carry on all the corpus information, in other experiments (noted *Semistics* +) we consider an extended set of Sense Units, completed with the 200 most frequent synsets which were left apart by the HAC.

Table 1 displays the predictive accuracy of all compared approaches. First column is the word baseline, second column is the LSI baseline, third column corresponds to *Semistics* , and fourth column is *Semistics* +.

These results indeed show that much care must be exercised when combining statistical and semantic information. *Semistics* is clearly outperformed by both the initial description and LSI. The reason for such a failure remains to be explained.

4.1 Sense Units vs Words

A first and disappointing observation is that Sense Units appear to be less appropriate than the initial words in order to classify documents. This might be analysed under two directions.
A first point simply regards the amount of information conveyed by the description; the number of SUs appears too restricted to support a sufficiently detailed description. This is confirmed by the fact that adding 200 SUs improves all results by about 5%.

A second point regards the quality of the Sense Units. Some of them are impressively relevant (for example $\{$ *chevrolet, oldsmobile* $\}$). In other cases, the synsets inducemuch noise due to polysemy problems. For instance, *mark*

	Base	LSI	*Semistics*	*Semistics* +
1-NN $\%OK$	88,2	92,5	60,3	65
1-NN $\%OK$-rel	96	97,8	77,7	81,1
20-NN $\%OK$	88,6	95,7	68,5	72,1
20-NN $\%OK$-rel	95,7	97	83,1	86

Table 1: Results summary for Reuters

and *marker* naturally constitute a synset. Unfortunately, this will favor clustering documents concerned with *marks* (german currency) and documents about pencils, or some specific scientific documents.

One cause for the above difficulty is the fact that the word clustering process for building the Sense Units actually relies on the average similarity of the words contained in the clusters (section 2), even though some words are more central than others to a cluster. Further research will suggest a better cluster similarity.

On the other hand, the redescription step (deciding to which extent a document involves a Sense Unit) might be insufficiently elaborated; for instance, it does not take into account how many words in the SU appear in the document, nor the frequency of these words in the document. A worthwhile perspective would be to consider a Sense Unit as a surrogate document, and consider the LSI distance between the SU and the document to be redescribed.

4.2 Sense Units vs LSI

A second equally disappointing observation is that even if our Sense Units are built using a combination of techniques including LSI, they are not as efficient as LSI alone for classifying documents.

One can note that in [22], it is shown that the partition of Reuters that we consider is very favorable to pure statistical methods. Sense units are not purely statistical, and like rule-induction or decision-tree methods, fail to top those methods on this part of the corpus.

Of course, it is quite dissatisfactory to see that even if the projection made by LSI is made on much fewer "dimensions" than that of Sense Units (100 eigenvalues against 600 Sense Units), LSI performance is much better. This means that the eigenvectors of LSI are much richer for redescription of the documents than the Sense Units. Here we have an opportunity to improve our system : one idea could be to start from those information-rich eigenvectors, and use WordNet to dislocate them into Sense Units.

4.3 XML data

Apart from all those results obtained with Reuters, we also performed some testings on a small XML documents corpus containing about 2000 documents, provided by the Xyleme crawler ([11]). These documents did not came with labels, so it was impossible to make an evaluation of the different similarities as we did before. The only results we can give so far on that corpus are very subjective, and based on the Sense Units obtained and the final clusters of documents. 324 Sense Units were produced, and we obtained about 200 clusters of various sizes. A brief examination indicates that these clusters seem to make some sense. Many duplicates are present in the data, and have been detected. More difficult clusterings were performed appropriately, for example with some documents about biology.

5 Related work

Several approaches have been proposed in order to provide better document descriptors than simple words. Such better descriptions are sought for under diverse forms, ranging from ontologies to distributions.

These approaches can be characterized depending on the nature of the information used to rewrite the documents, which draws upon semantic or statistical methods, or both.

On the semantic side for instance, [19] maps the document words onto WordNet synsets, each synset accounting for a concept. Experiments in the domain of text retrieval show that the performance strongly depends on the word sense disambiguation method used, which still is a limitation.

More recently, a pure statistical approach has been proposed by [18], obtaining excellent experimental results on the 20Newsgroup corpus. This approach is quite similar to ours, in the sense that it involves a two-step process, clustering words first, then using word clusters to rewrite the documents, and clustering documents last. The difference lies in the criterion used to cluster words. [18] use a purely statistical criterion; words are clustered so as to minimize the information loss, i.e. the difference in the corpus quantity of information. In opposition, our criterion involves both statistical (through LSI) and semantic (through WordNet) information. As discussed in the previous section, the weaknesses of our approach are precisely blamed on the insufficient care exercised when clustering synsets.

The combination of semantic and statistic-based approaches has been investigated in several ways. Most works rely on a syntactic tagging of the sentences. In [5] for instance, a syntactic analyzer is used to spot relations in the sentences (*verb* + *preposition* + *complement*). Words are considered similar if they often occur together with same verb and preposition. Based on this similarity, the ASIUM system interactively constructs word clusters, which are modified and refined online by the expert. It is worth noting that ASIUM significantly reduces the time needed to manually build ontologies.
Another approach, also based on a syntactic parser [15], focuses on particular (binary) relations (*subject, verb*) in the sentences. These are used to construct a distribution over the pairs (words,verbs).

6 Conclusion

This paper has presented our first attempt in order to perform document clustering using a method for redescribing document that involves both statistical and semantic information.

The first results on Reuters have been quite disappointing. Part of this problem might be due to the choice of the corpus as discussed in Section 4. That is why ongoing experiments consider alternative corpus, like XML documents or 20-Newsgroups.

It appears that the major weakness of our approach is the weighting between statistical and semantic feature. This drawback can be blamed on two causes :

- insufficient care was exercised in building synset similarity. This similarity was defined as the average of similarities of words in the synset. So when dealing with small synsets, the polysemy effect might be amplified. One possibility to alleviate this limitation is to consider synsets as documents, and to use LSI to have directly similarity values between synsets.
- our document redescription method is not precise enough. Hence the algorithm has a "fuzzy" view of the documents after redescription. Taking more parameters into account during this step should take care of this problem.

Many people (see [15] for exemple) use statistical methods on top of semantic results. We tried to give the two techniques equal importance, but our system is very basic and will need some more tuning. Another way to explore is to use semantic techniques on top of statistical results, like the decomposition using WordNet of the bags-of-words constituted by LSI eigenvectors.

References

[1] M. Berry, S. Dumais, and G. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4):573–595, 1995.

[2] Leacok C. and Chodorov M. *WordNet: An Electronic Lexical Database and some of its Applications*, chapter 11: Combining local context and WordNet similarity for word sense identification. MIT Press, Christiane Fellbaum editor, 1998.

[3] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[4] R. Fagin, Y. Maarek, I. Ben-Shaul, and D. Pelleg. Ephemeral document clustering for web applications. Technical Report 10186, IBM, 2000.

[5] D. Faure and C. Nedellec. Knowledge acquisition of predicate argument structures from technical texts using Machine Learning: the system ASIUM. In Dieter Fensel Rudi Studer, editor, *11th European Workshop EKAW'99*, pages 329–334. Springer-Verlag, 1999.

[6] U.M. Fayyad and B.K. Irani. Multi-interval discretization of continuous valued attributes for classification learning. In *Proceedings of IJCAI-93*, pages 1022–1027. Morgan Kaufmann, 1993.

[7] C. Felbaum, editor. *WordNet: an electronic lexical database*. Boston: MIT Press, 1998.

[8] P. W. Foltz, D. Laham, and T. K. Landauer. Automated essay scoring: Applications to educational technology. In *Proceedings of EdMedia'99*, 1999.

[9] Hirst G. and St-Onge D. *WordNet: An Electronic Lexical Database and some of its Applications*, chapter 13: Lexical chains as representations of context for the detection and correction of malapropisms. MIT Press, Christiane Fellbaum editor, 1998.

[10] http://www.cogsci.princeton.edu/∼ wn.

[11] http://www.xyleme.com.

[12] A. K. Jain. *Fundamentals of Digital Image Processing*, chapter 5: Image transforms, pages 132–188. Prentice Hall, 1989.

[13] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Machine Learning: ECML-98, Tenth European Conference on Machine Learning*, pages 137–142, 1998.

[14] M. Grobelnik and D. Mladenic and N. Milic-Frayling. *Proceedings of the Workshop on Text Mining, held at KDD 2000*, 2000.

[15] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of english words. In *30th Annual Meeting of the ACL*, pages 183–190, 1993.

[16] Cluet S., Veltri P., and Vodislav D. Views in a large scale xml repository. Submitted, 2001.

[17] Gerard Salton. *Automatic text processing: the transformation, analysis, and retrieval of information by computer.* Addison Wesley, 1989.

[18] N. Slonim and N. Tishby. Document clustering using word clusters via the information bottleneck method. In *SIGIR 2000*, pages 208–215, 2000.

[19] EM. Voorhees. *WordNet: An Electronic Lexical Database and some of its Applications*, chapter 12: Using WordNet for Text Retrieval. MIT Press, Christiane FellBaum editor, 1998.

[20] P. Wiemer-Hastings, K. Wiemer-Hastings, and A Graesser. How latent is latent semantic analysis? In *Proceedings of the Sixteenth International Joint Congress on Artificial Intelligence*, pages 932–937, San Francisco, 1999. Morgan Kaufmann.

[21] P. Willett. Recent trends in hierarchic document clustering: A critical review. *Information Processing and Management*, 24(5):577–597, 1988.

[22] Yiming Yang. An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval*, 1(1/2):67–88, 1999.