Filtering Contents with Bigrams and Named Entities to Improve Text Classification

François Paradis and Jian-Yun Nie

Université de Montréal, Canada
paradifr,nie@iro.umontreal.ca

Abstract. We present a new method for the classification of "noisy" documents, based on filtering contents with bigrams and named entities. The method is applied to *call for tender* documents, but we claim it would be useful for many other Web collections, which also contain non-topical contents. Different variations of the method are discussed. We obtain the best results by filtering out a window around the least relevant bigrams. We find a significant increase of the micro-F1 measure on our collection of call for tenders, as well as on the "4-Universities" collection. Another approach, to reject sentences based on the presence of some named entities, also shows a moderate increase. Finally, we try combining the two approaches, but do not get conclusive results so far.

1 Introduction

Text classification techniques rely heavily on the presence of a good *feature set*, or indexing terms, and the selection of discriminant features with regards to the classes. This task to the "cleanliness" of the documents: the presence of non-relevant or repetitive contents, as is often found on the Web, will degrade their performance. In our work, we are especially interested in a particular kind of Web documents, *call for tenders*, in which a contracting authority invites contractors to submit a tender for their products and/or services. These documents can be found on the contracted organisation Web site, or on dedicated tendering sites. In earlier work [1] we hypothesized that the noise in such documents was caused by the use of a *sublanguage* [2, 3] that describes the procedural aspects of the tenders submission, rather than their topic.

While feature selection undoubtedly brings a significant improvement to some classification methods [4], it is not clear whether it is adequate to filter such "procedural" noise. Indeed in our experiments with call for tenders we have found it difficult to extract either the procedural language (i.e. non-relevant features), or the tenders topic language (i.e. relevant features). There seems to be a significant overlap between the two vocabularies. However certain patterns or constructs of the procedural language can immediately be seen in the documents, and their presence should be an indication of the relevance of the surrounding context.

In this paper we propose to combine feature selection by *vocabulary*, with a *contextual* approach to filter out words or passages in the documents. That

is, we first select some n-grams features or named entities, and accept or reject passages based on their presence or absence. Our aim is to improve classification removing the "noise" from documents.

First, we briefly review related work and the context of our study. We present our first approach to content filtering based on filtering out a passage around the least relevant n-grams. We obtain a significant increase of the micro-F1 measure by using bigrams and a window passage: +11% on our collection of call for tenders, and +3.6% on the 4 Universities dataset. Our second approach, filtering around named entities, gives a moderate increase (+2.6%). We have also tried combining the two approaches, but with little success to report so far.

2 Related Work

The search for a better feature set for classification is hardly new. It has been demonstrated that feature selection is central to some algorithms such as Naive Bayes [4], and therefore several techniques have been proposed, the most popular being *InfoGain*. In early work by Lewis [5] the use of *phrases*, i.e. terms syntactically connected, was considered as a replacement for single-term features. The results were discouraging, which could be partly explained by the fact that there were too many index terms, with low frequency and high redundancy. Still, the idea was revisited by many. More recently Tan et al. [6] find an improvement on the classification of Web pages, by using a combination of bigrams and unigrams selected on the merits of their InfoGain score. However the same technique applied to the Reuters collection did not yield the same gain, mostly because of its over-emphasis on "common concepts". Since their method favours recall, the authors conclude it was harder to improve Reuters because it already had high recall.

The traditional use of the term *filtering* in classification refers the selection of documents relevant to a user profile. There has been much interest lately with spam filtering [7]. Content filtering, such as discussed in this paper, is also not a new idea, although it has not often been linked with classification. Early work with filtering based on character n-grams met with surprising success [8]. In [9] the notion of non-relevant passages in a document is exploited: a document is classified based on the relevance of its passages and their sequence as modeled with Hidden Markov Models. The area of automatic summarisation [10] is also related, since one of its subgoals is also to identify the most meaningful sentences. For example, the relevancy of a sentence can be defined based on its position, length, the frequency of the terms and its similarity with the title [11].

Text classification is often used in the process of named entity extraction [12] but rarely the other way around. Its use in classification is mostly restricted to replacing common strings such as dates or money amounts with tokens, to increase the ability of the classifier to generalise.

3 Classification of call for tenders

3.1 The MBOI project

This study is part of the MBOI project (Matching Business Opportunities on the Internet), which deals with the discovery of business opportunities on the Internet [13]. The project aims to develop tools for business watch, including spidering, information extraction, classification, and search. The aspect of interest here, classification, consists of classifying call for tenders by industry type, according to one of the existing norms: SIC (Standard Industrial Classification), NAICS (North American Industry Classification System), FCS (Federal Supply Codes), CPV (Common Procurement Vocabulary), etc.

A difficulty in the classification of call for tenders is to identify the relevant information amongst submission instructions, rules, requirements, etc. Sometimes the notice posted on the Web will have all relevant information to determine the subject, sometimes very little, and in some extreme cases none. Often the contracting authority will have the applicant pay to get a full description of the call for tender.

Furthermore, since we are spidering very different sites, the style and format of the documents vary a lot. Although a given organisation will tend to reuse the same patterns, it would not be feasible to manually define filters based on these patterns, without falling into a maintenance nightmare.

3.2 The test collection

For our experiments, we created a collection of call for tenders documents by downloading the XML daily synopsis from the FedBizOpps Web site (tenders solicited by American government agencies, available at `http://www.fedbizopps.gov/`). The XML documents have the same contents as the HTML documents found on the same site. The period downloaded ranged from September 2000 to October 2003. We kept only one document per tender, i.e. chose a document amongst pre-solicitations and amendments. Our collection (thereafter called FBO) is available at `http://iro.umontreal.ca/~paradifr/fbo/`.

An example of call for tender is shown in figure 1. It includes some meta-data such as the date of publication ("21 May 2001"), classification codes (NAICS "424120" and FCS "75"), the contracting authority ("Office of Environmental Studies"), etc. The body of the document is composed of the subject line and the description; only these fields will be used for classification. Only a portion of the body is indicative of the tender subject (shown in bold). The rest concerns dates and modalities for submission.

We considered only documents with two classification codes, FCS and NAICS (although FCS will not be used here). Since the NAICS codes were not tagged in XML at the time (as they now are), they were extracted from the free text description. This resulted in 21945 documents (72Megs), which were splitted 60% for training, and 40% for testing.

The NAICS codes are hierarchical: every digit of a six-digit code corresponds to a level of the hierarchy. For example, for industry code 424120 (Stationery and Office Supplies Merchant Wholesalers) the sector code is 424 (Merchant Wholesalers, Nondurable Goods). Each of the three participating countries, the U.S., Canada and Mexico, have their own version of the standard, which mostly differ at the level of industry codes (5th or 6th digit). We reduced the category space by considering only the first three digits, i.e. the corresponding "sector". This resulted in 92 categories (vs. 101 for FCS). We did not normalise for the uneven distribution of categories: for NAICS, 34% of documents are in the top two categories, and for FCS, 33% are in the top five.

Our baseline for this collection is a Naive Bayes [14] classifier trained and tested on the unfiltered documents. Naive Bayes is a common choice in the literature for baseline, and it is known to be sensitive to feature selection, which makes it appropriate to our study. Furthermore, some of the better performing but costlier techniques, such as SVM, do not scale up to our project requirement of handling more than 100K documents.

The 8,000 top terms were selected according to their InfoGain score. The following thresholds were applied: a rank cut of 1 (*rcut*), a fixed weight cut of 0.001 (*wcut*), and a category cut learnt after cross-sampling 50% of the test set over 10 iterations (*scut*). More details about these thresholding techniques can be found in [15, 16].

The rainbow software [17] was used to perform our experiments. The results for our baseline classifier are shown in table 1, under the label "unfiltered". The next line, "unfiltered bigram", is provided as an indication of the effect of bigrams alone on classification. It is again unfiltered contents, but using only bigrams as features (see our definition of bigram below). Since the feature space is much larger, we have selected 64,000 features this time. Surprisingly, there is a great drop in the micro-F1 measure. A quick look at the bigrams shows that many were actually part of the procedural language. It is difficult to filter them out, because when we select less features, we also remove "good" features and decrease the micro-F1 measure further.

4 Passage filtering

Two levels of passage filtering will be considered, depending on the unit being filtered: sentences or *windows* (i.e. sequence of words). Window filtering is appealing on our collection, because sentences can be long, and relevant and non-relevant information is often mixed in a sentence. Also, segmenting into sentences is not trivial in this collection, because it is not well formatted: for example the end-of-sentence period could be missing, or a space could appear inside an acronym (e.g. "U. S.").

4.1 Supervised filtering of sentences

In a first experiment we manually labeled 1000 sentences from 41 documents of FBO. The label was "positive" if the sentence was indicative of the tender's

<PRESOL>
<DATE> 0521
<YEAR> 01
<CLASSCOD> 75
<NAICS> 424120
<OFFADD> Office of Environmental Studies; 1323 Y Street, Washington, DC 22030
<SUBJECT> **Office supplies and devices**
<SOLNBR> N00140-04-Q-4555
<ARCHDATE> 07131999
<CONTACT> Mary Ann Deal, Contract Specialist
<DESC> The office of Environmental Studies intends **to procure printer toner cartridges and supplies** for the Naval Inventory Control Point in Mechanicsburg, PA. Request for Quotation (RFQ) N00140-04-Q-4555 contemplates an indefinite delivery type firm fixed price order. This is a combined synopsis/solicitation for commercial items prepared in accordance with the format in FAR Subpart 13.5, Test Program for Certain Commercial Items, as supplemented with additional information included in this notice. This announcement constitutes the only solicitation; proposals are being requested, and a written solicitation will not be issued. This is a 100% Total Small Business Set-Aside. etc.
<URL> http://www.oes.gov
<EMAIL>
<ADDRESS> johndoe@usa.gov
<SETASIDE> Total Small Disadvantage Business
<POPZIP> 22030
<POPCOUNTRY> US
</PRESOL>

Fig. 1. A Call for Tender

subject, or "negative" if not. Sentences with descriptive contents were labeled positive, while sentences about submission procedure, rules to follow, delivery dates, etc. were labeled negative. In the example of figure 1, only the first sentence would be labeled positive. Overall, almost a quarter of the sentences (243) were judged positive.

Intuitively, one would think that the first sentence(s) would often be positive, i.e. the author would start by introducing the subject of the tender, and then explain the rules and requirements. This is not always the case. In *combined* tenders, the text often starts with background information, and then define each item. In some cases, the subject is scattered amongst negative sentences.

We trained a Naive Bayes classifier on the 1000-sentence collection, for the positive and negative classes. The task seems to be relatively simple, since when we tested the classifier on a 40/60 split we obtained a micro-F1 measure of 85%. We thus filtered the whole collection with this classifier, keeping only the positive sentences. The collection size went from around 600,000 sentences to 96,811. The new, filtered documents were then classified with another Naive Bayes classifier.

Table 1 shows that this classification ("trained") gives an increase of the micro-F1 measure, 7.6% over the baseline ("unfiltered"). Although this result in itself is interesting, our real aim is to achieve unsupervised filtering, i.e. not requiring a training collection and labeled sentences. We propose in the next section a technique to select sentences based on the presence of vocabulary.

Table 1. FBO classification with passage filtering

method	macro-F1	micro-F1
unfiltered	.3297	.5498
unfiltered-bigrams	.2622 (-20%)	.4863 (-12%)
trained	.3223 (-2.2%)	.5918 (+7.6%)
sent-unigram	.3323 (≈)	.5701 (+3.7%)
sent-bigram	.3585 (+8.7%)	.5891 (+7.1%)
sent-trigram	.3394 (+2.9%)	.5866 (+6.6%)
window-bigram	.3787 (+14.9%)	.6101 (+11%)
window-trigram	.3497 (+6%)	.5927 (+7.8%)

4.2 Unsupervised filtering of sentences

Our approach to unsupervised filtering of sentences is to build a list of n-grams from the collection, and then filter out either a sentence or a window of terms around each of their occurences in the documents. We define an n-gram as a consecutive sequence of n words, after removal of stop words. For example, we have found the following top 5 n-grams in FBO:

– unigrams: "commercial", "items", "acquisition", "government" and "information"

- bigrams: “items-commercial”, “business-small”, “conditions-terms”, “fedbizopps-link” and “document-fedbizzopps”
- trigrams: “link-fedbizopps-document”, “supplemented-additional-information”, “additional-information-included”, “information-included-notice”, “prepared-accordance-format”.

We have tried two metrics for the selection of n-grams: the InfoGain measure and the frequency of the n-gram. Although one would expect the InfoGain measure to be a better discriminant, it turns out that the high-frequency terms in FBO are uniformely distributed in the classes, so the simpler frequency works as well if not slightly better. Also, since we are trying to select non-relevant features, i.e. features with low InfoGain, we will also capture the unfrequent features, whether they are distributed evenly or not.

Table 1 shows results of sentence filtering with unigram, bigrams and trigrams (i.e. “sent-unigram”, “sent-bigram” and “sent-trigram”). Only the most frequent 1,500 n-grams in the collection were kept (this parameter was determined manually). The criterion for a sentence to be filtered out was the following: a sentence was rejected if 1/8 of its n-grams were in the reject list (again this parameter was determined empirically).

The best result is obtained with bigrams (0.5891, or an increase of 7.1% over our baseline), which is quite similar to the results obtained with the trained classifier in the preceding section.

4.3 Window filtering

As mentioned before, although the sentence seems like a good logical unit to perform filtering, it is a bit problematic in our collection because it is not so well delimited, and it is not guaranteed to have the right granularity (i.e. sentences can contain both relevant and non-relevant information). Another approach is to ignore punctuation and sentence markers, and to filter a window around a term.

We select n-grams as above, and filter out a region of m words preceding, up to m words after the n-gram. Additionaly, two regions to be filtered out are connected if “close” enough.

Table 1 shows the results for bigrams and trigrams (“window-bigram” and “window-trigram”, respectively). of window filtering for bigrams and trigrams, using term frequency. A window of size 2 was used, i.e. the region filtered out started with the two terms preceding the n-gram, up to the two succeeding terms. Two regions to be filtered out were “connected” if less than 6 terms apart. The window-bigrams filter gives our best results: a micro-F1 of 0.6101 micro-F1 (+11%) and a macro-F1 of .3787 (+14.9%).

Note that to avoid any bias, the term distribution was computed over the training set only. Slightly higher figures can be reached by calculating term frequency over both the training and the test set (the default in rainbow). This is possible in a real scenario, if we constantly update the term distribution with new documents.

Also, we have presented results of filtering *out* sentences or windows, based on non-relevant features. We have also tried the opposite, i.e. selecting relevant features and keeping only those sentences or windows where they appeared. The results are similar.

5 Named entities

5.1 Entities as indicators of relevance

Table 2. Named entities in 1000 sentences

type	accuracy	type	accuracy
-location	72% (252/348)	-person	82% (351/429)
-organisation	75% (357/479)	-date	95% (59/62)
-time	98% (42/43)	-money	100% (18/18)
-URL & email	100% (38/38)	-phone number	98% (39/40)
-FAR	100% (56/56)		
+CLIN	80% (4/5)	+dimensions	100% (8/8)

Named entities are expressions containing names of people, organisations, locations, time, etc. These often appear in call for tenders, but are rarely indicative of the subject of the tender. Therefore, we hope that by identifying these expressions, we can either filter out passages that contain them, or reduce their impact on the classifier.

We take a somewhat broad definition of named entities, to include the following:

- *geographical location*. In a call for tender, this can be an execution or delivery location. A location can also be part of an address for a point of contact or the contracting authority (although these are often tagged as meta-data in FBO, they often appear in the text body).
- *organisation*. Most often the organisation will be the contracting authority or one its affiliates. For pre-determined contracts it can be the contractor.
- *date*. This can be a delivery date or execution date (opening and closing dates are often explicitely tagged as meta-data, and therefore do not need to be extracted).
- *time*. A time limit on the delivery date, or business hours for a point of contact.
- *money*. The minimum/maximum contract value, or the business size of the contractor.
- *URL*. The Web site of the contracting authority or a regulatory site (e.g. a link to CCR - Central Contract Registry).
- *email*, *phone number*. The details of a point of contact.

Although these entities have a particular use in our collection, they are generic in the sense that they also apply to many other domains. We have also considered the following entities, specific to our collection:

- *FAR* (Federal Acquisition Rules). These are tendering rules for U.S. government agencies. A call for tender may refer to an applicable paragraph in the FAR (e.g. "FAR Subpart 13.5").
- *CLIN* (Contract Line Item Number). The line item define a part or subcontract of the tender. Line items usually appear as a list (e.g. "CLIN 0001: ...").
- *dimensions*. In the context of a tender, a dimension almost always refers to the physical characteristics of a product to deliver (e.g. "240MM x 120MM").

All entities except CLIN and dimensions are *negative* indicators: their presence is an indication of a negative passage or sentence, i.e. not relevant to the subject of the tender. CLIN and dimensions on the other hand are *positive* indicators, since they introduce details about the contract or product.

The entities were identified in the collection using a mix of regular expressions and *Nstein NFinder*, a tool for the extraction of named entities. Table 2 shows the accuracy of the entities as positive/negative indicators on the 1000 training sentences. For example, dates (a negative indicator) appeared in 62 sentences, 59 of which were labeled negative. Dimensions (a positive indicator) appeared in 8 sentences, all of which were labeled positive.

Locations, persons and organisations are the most problematic entities, with an accuracy around or lower that of an "always-negative" classifier (which would be correct 75.7% of the time on our 1000 sentences). That is partly because they often appear along with the subject in an introductory sentence. For example in figure 1 the first sentence contains an organisation, "Office of Environmental Studies", a location, "Mechanicsburg, PA", as well as the subject, "toner cartridges and supplies". Furthermore, these entities are inherently more difficult to recognise than date and time, which only require a few simple patterns, and can achieve near-perfect recognition accuracy. To make matters more difficult, some documents are all in capital letters, which make the task more difficult because there are no clues to distinguish proper and common nouns. Some examples of errors were: "Space Flight" as a person, "FOB" as an organisation, or "184BW Contracting Office" as a location.

5.2 Classification with entities

As noted above, a common use of named entities in text categorisation is to replace each instance in the text with a generic token. That strategy does not seem to pay off on FBO, as shown in table 3, under the label "tokens". The micro-F1 measure does not change and the macro-F1 decreases by 1.5%. We have tried different combinations of entities, especially leaving out locations and organisations, all with similar results.

Organisation names sometimes provide valuable clues to the tender's subject. For example, knowing that the contracting authority is the USDA (U.S.

Department of Agriculture) increases the likelihood of a tender to be relevant to agriculture. This information is already taken into account by the classifier if the full name appears in the text. However if the acronym alone appears, only limited inference is possible (unless the acronym systematically appeared in all tenders of its kind).

We have tried to expand acronyms based on information collected from the training collection. Firstly we have built an acronym list from all organisation entities of the form: "full name (acronym)". We thus collected 1068 acronyms, excluding two-letters acronyms, which were deemed too ambiguous, especially since our collection includes many two-letter state abbreviations. We then expanded acronyms in the documents (except when they appeared inside brackets), and used the window-bigrams selection. Unfortunately, as shown in the last line of table 3, "acronym", this approach yielded a micro-F1 of .5265, a decrease of 4.2% over the baseline. One possible explanation for this poor performance is the high degree of ambiguity in the acronyms. For example, ISS refers to "Integrated Security System" or "International Space Station". In this case we put both expansions in the document.

Another possible use of named entities is to exploit the accuracy information from table 2. We have built a sentence filter that rejects a sentence if enough negative indicators are found. For indicators with a 100% accuracy, one instance is enough to reject a sentence. For others, we give a weight to each entity equals to its accuracy minus 75.7% (i.e. the accuracy of the always-negative classifier). We sum up the weights, and reject the sentence if it is above a threshold (which we have set to .40 in this experiment). The results of this filtering, under label "indicators", yields a micro-F1 of 0.5640, or +2.6% of the baseline.

We have tried combining bigrams and named entities in the following way. Each instance was replaced with a generic token, as above, and the bigrams were computed with those tokens. The aim was to find more generic patterns inside the bigrams. For example, the bigrams now included patterns such as 'exceed-[money]' (as in "business size should not exceed $10.4M"). Such a pattern could not be picked up before because money amounts, as other numbers, would be rejected by the tokeniser. Furthermore, using an entity tag increases the frequency of the bigram and therefore its chance to be included in the filter list. Unfortunately, as shown in table 3 under labels "trained" and "window-bigram", this combination does not significantly increase over the sentence and window filtering of the preceding section: the trained method goes from .5918 to .5935, and the window-bigram from .6101 to .6077.

6 Results on 4-Universities

We have tried these techniques on the 4-Universities collection, which can be obtained from CMU World Wide Knowledge Base project at URL `http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/`. It contains 8282 Web pages collected mainly from four American universities, and manually classi-

Table 3. FBO classification with named entities

method	macro-F1	micro-F1
tokens	.3246 (-1.5%)	.5514 (≈)
acronym	.3222 (-2.3%)	.5265 (-4.2%)
indicators	.3325 (≈)	.5640 (+2.6%)
trained	.3507 (+6.4%)	.5935 (+8%)
window-bigram	.3703 (+12.3%)	.6077 (+10.5%)

fied in seven categories: student pages, faculty, staff, department, course, project or other.

The baseline is again a Naive Bayes classifier, this time using InfoGain with 2000 features, as suggested by the authors. We have also used their script to replace some numbers with generic tokens. We have not however implemented their cross-tests, i.e. train on three, and test on one university.

Table 4. Classification results on 4 Universities

method	macro-F1	micro-F1
unfiltered	.5918	.6492
unfiltered-bigram	.6205 (+4.8%)	.7055 (+8.6%)
window-bigram	.5346 (-9.7%)	.6723 (+3.6%)
tokens	.5864 (≈)	.6664 (+2.6%)

Table 4 shows some results. One can see some differences with our FBO collection. Contrary to FBO, the use of bigrams and named entity tokens (tokens) had a positive impact on the unfiltered collection, with an increase of 8.6 and 2.6%, respectively. We did not test the "indicators" filtering, since we did not have information about the accuracy of the entities.

We tested the window-bigram method with the 1500 top bigrams according to the InfoGain measure, and obtained a micro-F1 value of 0.6723, an increase of 3.6% over the baseline. However, the macro-F1 has suffered a hefty drop (-9.7%). We have not yet studied the reasons for this.

7 Conclusion

We have investigated the use of bigrams and named entities to perform content filtering. Our domain of application was the classification of call for tenders. Our findings are that filtering a windows of terms around most frequent bigrams works well for this kind of collection: we could obtain an increase of 11% of micro-F1. We also get a moderate improvement of 2.6% by filtering sentences based on named entities; however this method relies on an accuracy estimate,

which is not always practical to get in reality. We have tried combining the two approaches but so far our results are rather inconclusive.

We have tried this approach on the 4-Universities dataset and also found an increase of the micro-F1 using the window-bigram method. This is, we hope, an indication that this method is well-suited for Web collections. We plan to do more tests in the future to verify that claim.

When filtering with bigrams, we did not include simple terms in the list; similarly for trigrams we did not include bigrams. This might explain the lower results for trigrams, because they are too restrictive, and fail to capture common two-terms relationships. An obvious extension, as in [6], is to combine them.

Another idea worth pursuing is taking advantage of the sequence of relevant and non-relevant sentences in the document. This idea is similar to the HMM proposed in [9].

Acknowledgments

This project was financed jointly by Nstein Technologies and NSERC.

References

1. Paradis, F., Nie, J.Y.: Étude sur l'impact du sous-langage dans la classification automatique d'appels d'offres. In: CORIA, Grenoble, France. (2005)
2. Lehrberger, J. In: Automatic translation and the concept of sublanguage, in R. Kittredge et J. Lehrberger (eds.), Sublanguage: Studies of Language in Restricted Semantic Domains. (1982)
3. Biber, D.: Using register-diversified corpora for general language studies. Computational linguistics **19** (1993)
4. Yiming Yang, J.O.P.: A comparative study on feature selection in text categorization. In: Proceedings of ICML-97, 14th International Conference on Machine Learning. (1997)
5. Lewis, D.: An evaluation of phrasal and clustered representations on a text categorization task. In: 15th ACM International Conference on Research and Development in Information Retrieval (SIGIR). (1992) 37–50
6. Tan, C.M., Wang, Y.F., Lee, C.D.: The use of bigrams to enhance text categorizatio. Information Processing and Management: an International Journal **38** (2002) 529 – 546
7. Zhang, L., Yao, T.: Filtering junk mail with a maximum entropy model. In: Proceeding of 20th International Conference on Computer Processing of Oriental Languages (ICCPOL03). (2003) 446–453
8. Cavnar, W.: N-gram-based text filtering for trec-2. In: Second Text REtrieval Conference (TREC). (1993)
9. Denoyer, L., Zaragoza, H., Gallinari, P.: Hmm-based passage models for document classification and ranking. (2001)
10. Orasan, C., Pekar, V., L., L.H.: A comparison of summarisation methods based on term specificity estimation. In: Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC-04). (2004) 1037–1041

11. Nobata, C., Sekine, S., Murata, M., Uchimoto, K., Utiyama, M., Isahara, H.: Sentence extraction system assembling multiple evidence. (2001)
12. M., J.: Named entity extraction with conditional markov models and classifiers. In: The 6th Conference on Natural Language Learning. (2002)
13. Paradis, F., Ma, Q., Nie, J.Y., Vaucher, S., Garneau, J.F., Gérin-Lajoie, R., Tajarobi, A.: Mboi: Un outil pour la veille d'opportunités sur l'internet. In: Colloque sur la Veille Strategique Scientifique et Technologique, Toulouse, France. (2004)
14. Jason D. M. Rennie, Lawrence Shih, J.T., Karger, D.R.: Tackling the poor assumptions of naive bayes text classifiers. In: Proceedings of the Twentieth International Conference on Machine Learning. (2003)
15. Yang, Y.: An evaluation of statistical approaches to text categorization. Journal of Information Retrieval **1** (1999) 67–88 An excellent reference paper for comparisons of classification algorithms on the Reuters collection.
16. Yang, Y.: A study on thresholding strategies for text categorization. In: Proceedings of SIGIR-01, 24th ACM International Conference on Research and Development in Information Retrieval. (2001)
17. McCallum, A.K.: Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. `http://www.cs.cmu.edu/~mccallum/bow` (1996)