litsift: Automated Text Categorization in Bibliographic Search

Lukas C. Faulstich
Peter F. Stadler
Bioinformatik, Institut für Informatik
Universität Leipzig, Germany
{lukas.faulstich, peter.stadler}
@bioinf.uni-leipzig.de

Caroline Thurner
Christina Witwer
Institut für Theoretische Chemie und
Strukturbiologie, Universität Wien, Austria
{caro,xtina}@tbi.univie.ac.at

ABSTRACT

In bioinformatics there exist research topics that cannot be uniquely characterized by a set of key words because relevant key words are (i) also heavily used in other contexts and (ii) often omitted in relevant documents because the context is clear to the target audience. Information retrieval interfaces such as entrez/Pubmed produce either low precision or low recall in this case. To yield a high recall at a reasonable precision, the results of a broad information retrieval search have to be filtered to remove irrelevant documents. We use automated text categorization for this purpose.

In this study we use the topic of conserved secondary RNA structures in viral genomes as running example. Pubmed result sets for two virus groups, *Picornaviridae* and *Flaviviridae*, have been manually labeled by human experts. We evaluated various classifiers from the Weka toolkit together with different feature selection methods to assess whether classifiers trained on documents dedicated to one virus group can be successfully applied to filter literature on other virus groups. Our results indicate that in this domain a bibliographic search tool trained on a reference corpus may significantly reduce the amount of time needed for extensive literature recherches.

Keywords

Automated Text Categorization, Document Filtering

1. INTRODUCTION

An important part of bioinformatics research is the comparison of computational results with experimental results. These are, unfortunately, often hidden in the vast body of molecular biology literature. More often than not, the data that are of interest for a particular computational study are mentioned only in passing and in a different context in the experimental literature. As a concrete example we consider here the survey of conserved RNA secondary structures in viral genomes[1] that has been initiated a few years ago by the Vienna group [9, 14, 11]. To our surprise, the bibliographic search for experimental evidence of and further information on *RNA secondary structures* in a given group of virus — a seemingly rather straightforward task — turned out to be more tedious than the work on the actual sequence and structure data.

There are several reasons for this difficulty: (i) RNA secondary structure is usually referred to only as *secondary structure* or simply as *structure* since the context *RNA* is clear. The term *secondary structure*, however, appears much more frequently in the context of protein structures for the same virus group because proteins are usually discussed more frequently and in much more detail. (ii) RNA secondary structures are rarely the main topic of research papers on viruses. Rather, only one or a few paragraphs are devoted to them. (iii) With few exceptions there is no well-established nomenclature of RNA features in viruses so that keyword searches for specific structural motifs are not very effective. (iv) Relevant articles are written by authors from rather diverse scientific communities, from clinical virologists to structural biologists.

Our target topic of *"conserved RNA secondary structure in viral genomes"* consists of several subtopics, each dedicated to a specific group of RNA viruses (e.g., *Picornaviridae*, *Flaviviridae*, *Coronaviridae*, or *Hepadnaviridae*). For some of these subtopics, manually labeled document corpora exist. The question addressed in this exploratory study is whether classifiers trained for one subtopic can be applied successfully to other subtopics. This would be in particular attractive for subtopics with a large amount of available literature, e.g., on the HIV virus in the case of *Retroviridae.* In our context, successful means a high recall (e.g., 80%) with a not too low precision (e.g., 30%) because the emphasis is on finding most of the relevant literature with a tolerable overhead caused by false positives.

Our goal is to make bibliographic search more effective by using classifiers trained on sample corpora in a system that filters and ranks search results from bibliographic databases such as Pubmed. This kind of application is known as *document filtering.* The filtering part is essentially a binary text categorization problem. Ranking comes for free in conjunction with distribution classifiers because they return probabilities that can be used as document scores. The vast body of literature on automated text categorization is surveyed in [8]. In the Information Retrieval community, much work (from [6] to [1, 3, 4, 10, 15]) has been done on *adaptive* document filtering, where relevance feedback from users is

[1] http://rna.tbi.univie.ac.at

Data Mining and Text Mining for Bioinformatics
Workshop at the ECML/PKDD 2003 in Dubrovnik-Cavtat, Croatia;
22. September, 2003.

Table 1: The training corpora.

Corpus	Source	Size	Positive
picorna	Pubmed query: picornavirus RNA secondary structure	40	68%
picorna2	picorna + 24 extra documents	64	58%
flavi	Pubmed query: RNA AND (IRES OR "secondary structure" OR "conserved structure" OR "5'utr" OR "3'utr" OR "coding region") AND ("hepatitis C virus" OR "hepatitis G virus" OR pestivirus OR dengue OR "japanese encephalitis virus" OR "yellow fever virus"OR "tick-borne encephalitis virus")	153	8%
flavi2	flavi + 34 extra documents	187	12%
hepadna	Pubmed query: (Hepadnaviridae OR "Hepatitis B" OR "HBV") AND (RNA secondary structure) NOT delta	16	69%

employed to adjust document filters.

The preliminary results presented here indicate that a classifier trained on one virus group can be applied successfully to search the literature on other virus groups. Therefore, a system for supporting bibliographic search based on automated text categorization seems feasible for our target topic.

The remainder of this article is organized as follows: in Sec. 2 we present the data sets used for this work. The methods and tools used are described in Sec. 3. Our experiments and their results are presented in Sec. 4. Finally we give in Sec. 5 a conclusion and outline our future research.

2. DATA SETS

Training data has been obtained from searching the Pubmed collection via the entrez interface[2] and then downloading the referenced articles as PDF documents (as far as available). The search queries (see Table 1) have been specified by our domain experts (PFS, CT, CW). The resulting corpora are referred to as picorna, flavi, and hepadna. They are dedicated to the virusgroups *Picornaviridae*, *Flaviviridae*, and *Hepadnaviridae*, respectively.

Since corpus picorna is quite small and corpus flavi contains only few positive examples, we decided to add more documents. These documents were provided by our domain experts from their private bibliographical collections. The resulting corpora are referred to as picorna2 and flavi2. The small corpus hepadna is only used for testing classifiers trained on the latter two corpora.

A document is considered a positive example within its corpus if it contains information on the secondary structure of the RNA of viruses belonging to the virus group the corpus is dedicated to.

3. METHODS

3.1 Data Preparation

The PDF documents where converted into text using the Unix tools pdftotext and ps2ascii. The ConceptComposer text analysis suite [2] was used to build a full text index of the resulting text documents in a relational database (mysql).

Based on this index, the documents were transformed into vector representation using a SQL script. We computed term weights according to the standard tfidf method (see e.g. [7]). Each corpus is stored in a separate mysql database.

For feature selection we implemented the term relevance measures *Odds Ratio* and *Mutual Information* (see [8]). In addition we implemented derived term relevance measures where the original relevance value for a term is weighted with its frequency in the test database that is used for evaluation.

3.2 Text Categorization

We built the Java application litsift on top of the Weka 3 machine learning software [13] to classify the document corpora. This enabled us to experiment with the variety of classifiers provided by Weka. Further parameters that can be varied are

- the term relevance measure to use for feature selection
- the number of features to be taken into account
- the target recall when evaluating a classifier on the test corpus
- classifier specific parameters

The application reads class labels for documents and their term weights for the selected features from the training database and creates a set of Weka instances from it. This instance set is either used for cross evaluation on the training corpus or it is used to train a classifier that is evaluated on a separate test corpus. In the latter case, only those documents are classified as positive whose predicted class-membership probability exceeds a certain threshold. This threshold is adjusted automatically to achieve at least the chosen target recall (if possible at all) in a trade-off with the achieved precision. The threshold is found by computing histograms on the number of positives and true positives over the predicted probabilities.

4. RESULTS

Before we assess the applicability of classifiers trained on one corpus to another corpus, we present cross-evaluation results on each corpus as a base line for comparison.

4.1 Feature Selection

To assess the performance of different term relevance measures, we varied the number N of features. From the corpus we filtered those documents that contained at least one of the N best terms of the chosen measure. Then we computed precision and recall of this filter by counting the selected documents as positives and the rest of the corpus as negatives. The results are shown in Table 2. It shows that 10–30 features are always sufficient to retrieve all positive examples. Moreover it shows that the corpora picorna and picorna2 are quite trivial since they can be classified completely and correctly by using just the first 20 (picorna) or 30 (picorna2) features selected by Mutual Information.

[2] http://www.ncbi.nlm.nih.gov/Entrez/

Table 2: Filtering results for different corpora, and relevance measures (column "msr"), with target recall 100%. The relevance measures Mutual Information and Odds Ratio are abbreviated as "MI" and "OR", respectively. Column "p_{avg}" shows the average precision over all feature counts where the target recall is exceeded. Column "p_{max}" shows the maximum precision. The minimum feature count at which this maximum precision is reached is labeled"d". The recall achieved with this number of features is shown in column "r".

corpus	msr	p_{avg}	p_{max}	r	d
flavi	MI	11.2%	23.1%	100.0%	20
flavi	OR	7.8%	7.9%	100.0%	10
flavi2	MI	20.2%	40.7%	100.0%	20
flavi2	OR	11.8%	11.8%	100.0%	10
picorna	MI	76.7%	100.0%	100.0%	20
picorna	OR	67.6%	69.2%	100.0%	10
picorna2	MI	69.3%	100.0%	100.0%	30
picorna2	OR	58.0%	59.7%	100.0%	10

4.2 Cross Evaluation on Each Corpus

As a base line for comparison we cross-evaluated several classifiers from the Weka toolkit, namely C4.5 ("J48"), Support Vector Machine ("SMO"), and Naive Bayes ("N.B."), in combination with the available term relevance measures on each corpus. The results are shown in Table 3. It shows that

1. on flavi and flavi2 the target recall of 80% can be reached only by the NaiveBayes classifier
2. with few exceptions, less than 50 features are needed to achieve maximum recall
3. corpora picorna and picorna2 can be almost perfectly classified in most cases
4. J48 seems sensitive with respect to the relevance measure: on flavi2, Odds Ratio performs much better, on picorna2, Mutual Information performs much better.

4.3 Validation on a Separate Test Corpus

We first present some exemplary experiments with SMO and then give in an overview of all experiments in form of a table.

4.3.1 *Training on* flavi*, Validation on* picorna

A SMO classifier trained on flavi with Odds Ratio measure evaluated on picorna2 reaches the target recall of 80% beginning with 30 features. The precision reaches a maximum of 80% at about 150 features (see Fig. 1a). Using Mutual Information yields similar results.

4.3.2 *Training on* flavi2*, Validation on* picorna2

Compared to Sec. 4.3.1, the average precision of SMO drops slightly from 74% to 66% (see Fig. 1b) which is still quite acceptable for bibliographic search.

Table 3: Cross evaluation results for different corpora, classifiers, and relevance measures, with target recall 80%. The classifiers shown in column "class" are C4.5 ("J48"), Support Vector Machine ("SMO"), and Naive Bayes ("N.B."). For the meaning of the remaining columns see Table 2. The average precision is ommitted in cases where the target recall was not reached.

corpus	class	msr	p_{avg}	p_{max}	r	d
flavi	J48	MI	–	85.7%	60.0%	10
flavi	J48	OR	–	72.7%	66.7%	60
flavi	N.B.	MI	21.3%	38.5%	83.3%	30
flavi	N.B.	OR	25.2%	44.0%	91.7%	40
flavi	SMO	MI	–	75.0%	30.0%	10
flavi	SMO	OR	–	80.0%	33.3%	30
flavi2	J48	MI	–	73.7%	63.6%	160
flavi2	J48	OR	–	77.3%	77.3%	30
flavi2	N.B.	MI	28.3%	41.9%	81.8%	80
flavi2	N.B.	OR	37.8%	58.1%	81.8%	30
flavi2	SMO	MI	–	66.7%	54.5%	30
flavi2	SMO	OR	–	73.3%	50.0%	40
picorna	J48	MI	99.3%	100.0%	100.0%	10
picorna	J48	OR	89.2%	92.6%	92.6%	50
picorna	N.B.	MI	79.3%	100.0%	95.2%	10
picorna	N.B.	OR	80.5%	100.0%	100.0%	20
picorna	SMO	MI	90.6%	100.0%	100.0%	10
picorna	SMO	OR	91.7%	100.0%	85.2%	30
picorna2	J48	MI	95.3%	100.0%	100.0%	10
picorna2	J48	OR	85.2%	88.2%	81.1%	110
picorna2	N.B.	MI	77.6%	100.0%	96.7%	10
picorna2	N.B.	OR	79.7%	100.0%	94.6%	20
picorna2	SMO	MI	93.5%	100.0%	100.0%	10
picorna2	SMO	OR	93.1%	100.0%	81.1%	40

4.3.3 *Training on* picorna*, Validation on* flavi

While SMO trained on corpus flavi can be successfully applied to the corpora picorna and picorna2, the inverse setting is not as successful. At 80% recall, SMO achieves a maximum precision of 23% precision at 60 features (see Fig. 2a).

With Mutual Information, the precision is even lower (about 10%).

Using a derived term relevance measure (Odds Ratio, weighted with term frequencies from flavi) did not yield any improvement, either.

23% precision may not seem high, but in our application to bibliographic search it is still more tolerable than in other fields of text classification.

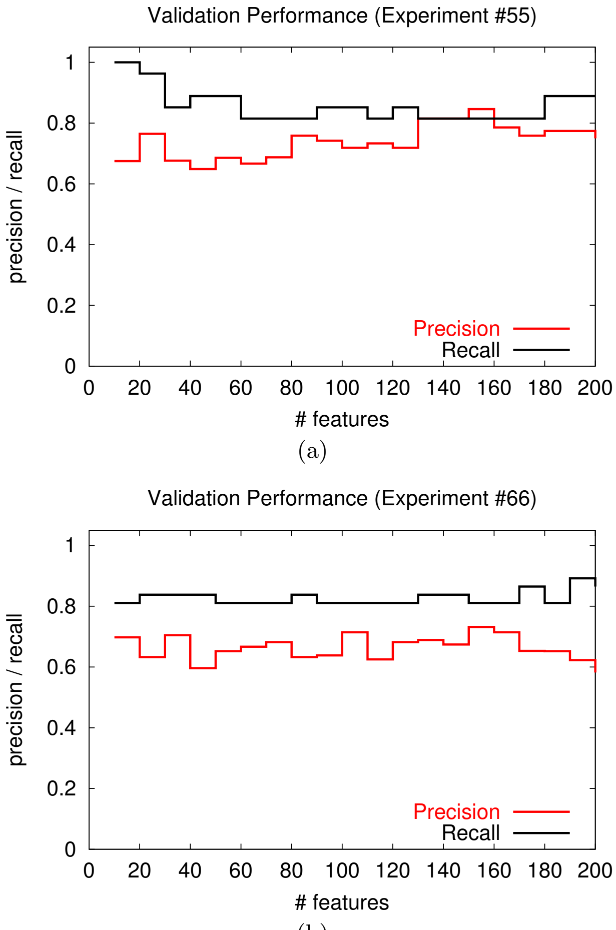
4.3.4 *Training on* picorna2*, Validation on* flavi2

Compared to Sec. 4.3.3, precision of SMO increases to 30% starting from 40 features (see Fig. 2b).

4.3.5 *Discussion*

In Table 4, all experiments with evaluation on a separate corpus are listed. We may summarize these results as follows:

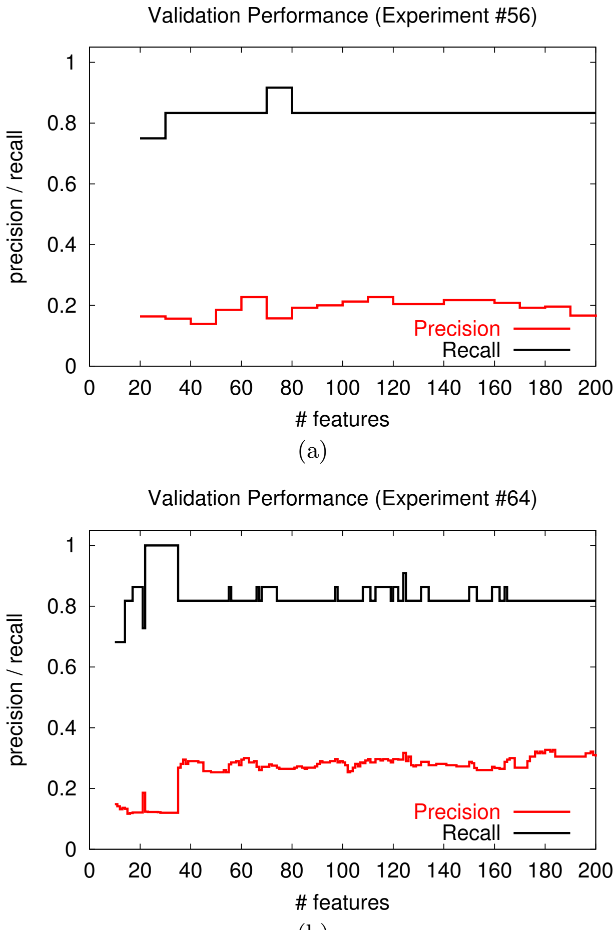
1. Corpora picorna and picorna2 can quite successfully be classified after training on flavi and flavi2, respectively.



(a)

(b)

Figure 1: Performance for Weka SMO with Odds Ratio, target recall 80%: (a) on corpus picorna after training on corpus flavi, (b) on corpus picorna2 after training on corpus flavi2.



(a)

(b)

Figure 2: Performance for Weka SMO with Odds Ratio, target recall 80%: (a) on corpus flavi after training on corpus picorna, (b) on corpus flavi2 after training on corpus picorna2.

   (a) With J48 or NaiveBayes, 100% recall can be achieved with maximum precisions above 70%, using only few features (10–30).

   (b) Mutual Information seems to perform better than Odds Ratio.

2. Corpora flavi and flavi2 can not as easily classified after training on picorna and picorna2, respectively.

   (a) The best maximum precision is achieved by SMO with Odds Ratio

   (b) Corpus flavi2 is easier to classify than flavi

3. Corpus hepadna can quite successfully be classified after training on picorna2 or flavi2.

   (a) In both cases, SMO performs best, reaching a maximum precision of 90%.

   (b) In most cases Odds Ratio performs much better than Mutual Information, i.e., it needs much fewer features to achieve a better maximum precision.

4. In most cases the difference between average and maximum precision is quite small. This supports the observation from Figs. 1 and 2 that precision does not depend too much on the number of features.

The asymmetry between the picorna* and flavi* corpora can to some extent be explained by the fact that the *Flaviviridae* virus group is more heterogenous than the *Picornaviridae* group. For instance, while all *Picornaviridae* genomes have so-called IRES (Internal Ribosomal Entry Site) regions, this does not hold for all *Flaviviridae*. This means that a classifier trained on a picorna* corpus only finds those positive examples in flavi* that are similar to those in the training corpus. In the other direction this partition within a flavi* corpus seems to be sufficient to learn the characteristics of the positive examples in the picorna* corpora. The additional positives in corpus flavi2 might be more “picorna”-like which would explain the better performance when testing on flavi2 instead of flavi.

Table 4: Transfer results for different training and test corpora, classifiers, and relevance measures, with target recall 80%. The classifiers shown in column "class" are C4.5 ("J48"), Support Vector Machine ("SMO"), and Naive Bayes ("N.B."). For the meaning of the remaining columns see Table 2.

training	test	class	msr	p_{avg}	p_{max}	r	d
flavi	picorna	J48	MI	69.1%	76.7%	100.0%	30
flavi	picorna	J48	OR	67.3%	67.5%	100.0%	10
flavi	picorna	N.B.	MI	69.1%	76.7%	100.0%	30
flavi	picorna	N.B.	OR	67.5%	67.5%	100.0%	10
flavi	picorna	SMO	MI	74.4%	80.6%	92.6%	160
flavi	picorna	SMO	OR	74.0%	84.6%	81.5%	150
flavi2	picorna2	J48	MI	65.8%	100.0%	80.0%	10
flavi2	picorna2	J48	OR	57.8%	57.8%	100.0%	10
flavi2	picorna2	N.B.	MI	65.9%	83.3%	100.0%	10
flavi2	picorna2	N.B.	OR	57.8%	57.8%	100.0%	10
flavi2	picorna2	SMO	MI	68.3%	83.3%	100.0%	10
flavi2	picorna2	SMO	OR	66.2%	73.2%	81.1%	150
picorna	flavi	J48	MI	12.6%	16.4%	91.7%	90
picorna	flavi	J48	OR	13.7%	15.7%	91.7%	60
picorna	flavi	N.B.	MI	9.5%	14.7%	83.3%	10
picorna	flavi	N.B.	OR	9.4%	12.2%	100.0%	30
picorna	flavi	SMO	MI	12.3%	20.0%	83.3%	110
picorna	flavi	SMO	OR	18.6%	22.7%	83.3%	60
picorna2	flavi2	J48	MI	15.1%	22.6%	86.4%	130
picorna2	flavi2	J48	OR	16.3%	19.3%	100.0%	40
picorna2	flavi2	N.B.	MI	16.1%	20.0%	100.0%	20
picorna2	flavi2	N.B.	OR	14.0%	15.4%	95.5%	180
picorna2	flavi2	SMO	MI	18.7%	23.8%	86.4%	130
picorna2	flavi2	SMO	OR	26.2%	32.7%	81.8%	180
flavi2	hepadna	J48	MI	78.4%	81.8%	81.8%	180
flavi2	hepadna	J48	OR	68.8%	71.4%	90.9%	20
flavi2	hepadna	N.B.	MI	78.4%	81.8%	81.8%	180
flavi2	hepadna	N.B.	OR	68.8%	68.8%	100.0%	10
flavi2	hepadna	SMO	MI	75.0%	75.0%	81.8%	200
flavi2	hepadna	SMO	OR	73.6%	90.9%	90.9%	50
picorna2	hepadna	J48	MI	76.6%	83.3%	90.9%	90
picorna2	hepadna	J48	OR	70.1%	71.4%	90.9%	40
picorna2	hepadna	N.B.	MI	76.6%	83.3%	90.9%	90
picorna2	hepadna	N.B.	OR	69.6%	75.0%	81.8%	170
picorna2	hepadna	SMO	MI	76.7%	81.8%	81.8%	90
picorna2	hepadna	SMO	OR	77.9%	90.0%	81.8%	70

4.3.6 Usefulness

How useful could a bibliographic search tool based on automated classification be for a scientist who wants to perform a literature recherche? To assess this, we consider the following scenario: the scientist wants to identify relevant literature with minimal effort without loosing to many relevant articles. For a fixed recall r, the amount of work is determined by the number of articles that the scientist has to inspect.

We assume a fixed corpus of articles that have been returned by the bibliographic database (i.e., Pubmed). By randomly selecting documents with a probability r, we achieve also recall r since the probability for a relevant document to be selected is r. In this case, the scientist has to inspect a fraction $P_{\text{rand}} = r$ of all documents. This is the baseline for a comparison with an automated classifier.

The fraction P_{auto} of documents selected by an automated classifier with precision p and recall r on a corpus with a fraction c of relevant documents is $P_{\text{auto}} = cr/p$. Hence the work reduction is $s = (P_{\text{rand}} - P_{\text{auto}})/P_{\text{rand}} = 1 - c/p$. In Table 5 the work reduction s is shown for some of the cases from Table 4. Most work can be saved on corpora such as flavi2 with few relevant documents (assuming that the precision does not deteriorate too much). The percentage of relevant documents in the small corpora picorna2 and hepadna is too high to reach large work reductions.

Table 5: Work reduction s for selected sample configurations.

training	test	class	msr	p_{max}	r	s
flavi2	picorna2	SMO	MI	83.3%	100.0%	30%
picorna2	flavi2	SMO	OR	32.7%	81.8%	63%
flavi2	hepadna	SMO	OR	90.9%	90.9%	25%
picorna2	hepadna	SMO	OR	90.0%	81.8%	25%

5. CONCLUSION AND OUTLOOK

The results presented in this study are rather heterogeneous. Nevertheless, they indicate that classifiers trained on one subtopic can be applied to another subtopic and achieve precisions (here 20% – 100%) that will result in cost savings when searching for relevant literature while not too many (here 20%) relevant documents are lost.

The complications of bibliographic search that plague the case of RNA secondary structure features in viral RNAs are not a restricted to this particular topic. Whenever the available literature has to be searched for information that is rarely the main focus of the publication keyword-based searches tend to have either low recall or low precision. Regulatory sequences associated with certain classes of genes may serve as another example.

We thus plan to extend the litsift application into a bibliographic search tool that sends a user query to a bibliographic database such as Pubmed, retrieves the search results and the articles cited therein, and ranks the results according to the predictions of a classifier previously trained using the same tool. The user may choose to re-label some of the results manually and retrain the classifier in order to enhance its performance. An interesting option for further improving this tool would be to include classification techniques that take unlabeled data into account, e.g. [5, 12].

Open questions that require further research are: (i) what are good heuristics for choosing a number of features and (ii) are there indicators for the transferability of a classifier to another corpus?

6. ACKNOWLEDGEMENTS

This work is supported by the Austrian *Fonds zur Förderung der Wissenschaftlichen Forschung*, Project Nos. P-13545-MAT and P-15893 and the German *DFG* Bioinformatics Initiative.

7. REFERENCES

[1] G. Amati and F. Crestani. Probabilistic learning for selective dissemination of information. *Information Processing and Management*, 35(5):633–654, 1999.

[2] G. Heyer, U. Quasthoff, and C. Wolff. Automatic analysis of large text corpora — A contribution to structuring WEB communities. In H. Unger, T. Böhme, and A. Mikler, editors, *Innovative Internet Computing Systems*, volume 2346 of *Lecture Notes in Computer Science*, pages 15–26. Springer-Verlag, Heidelberg, 2002.

[3] R. D. Iyer, D. D. Lewis, R. E. Schapire, Y. Singer, and A. Singhal. Boosting for document routing. In A. Agah, J. Callan, and E. Rundensteiner, editors, *Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management*, pages 70–77, McLean, US, 2000. ACM Press, New York, US.

[4] Y.-H. Kim, S.-Y. Hahn, and B.-T. Zhang. Text filtering by boosting naive Bayes classifiers. In N. J. Belkin, P. Ingwersen, and M.-K. Leong, editors, *Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pages 168–175, Athens, GR, 2000. ACM Press, New York, US.

[5] K. Nigam. *Using Unlabeled Data to Improve Text Classification*. PhD thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, US, 2001.

[6] J. J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The SMART retrieval system: experiments in automatic document processing*, pages 313–323. Prentice-Hall, Englewood Cliffs, USA, 1971.

[7] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[8] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[9] R. Stocsits, I. L. Hofacker, and P. F. Stadler. Conserved secondary structures in hepatitis B virus RNA. In *Computer Science in Biology*, pages 73–79, Bielefeld, D, 1999. Univ. Bielefeld. Proceedings of the GCB'99, Hannover, D.

[10] D. R. Tauritz, J. N. Kok, and I. G. Sprinkhuizen-Kuyper. Adaptive information filtering using evolutionary computation. *Information Sciences*, 122(2/4):121–140, 2000.

[11] C. Thurner, C. Witwer, I. Hofacker, and P. F. Stadler. Conserved RNA secondary structures in Flaviviridae genomes. 2003. submitted.

[12] J.-N. Vittaut, M.-R. Amini, and P. Gallinari. Learning classification with both labeled and unlabeled data. *Lecture Notes in Computer Science*, 2430:468–??, 2002.

[13] I. H. Witten and E. Frank. Nuts and bolts: Machine learning algorithms in java,. In *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations.*, pages 265–320. Morgan Kaufmann, 1999.

[14] C. Witwer, S. Rauscher, I. L. Hofacker, and P. F. Stadler. Conserved RNA secondary structures in Picornaviridae genomes. *Nucl. Acids Res.*, 29:5079–5089, 2001.

[15] K. L. Yu and W. Lam. A new on-line learning algorithm for adaptive text filtering. In G. Gardarin, J. C. French, N. Pissinou, K. Makki, and L. Bouganim, editors, *Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management*, pages 156–160, Bethesda, US, 1998. ACM Press, New York, US.