

HYBRED: An OCR document representation for classification tasks

Sami Laroum¹, Nicolas Béchet², Hatem Hamza³, and Mathieu Roche²

¹ Biopolymères Interactions Assemblages, INRA
BP 71627, 44316 Nantes - France

² LIRMM, Université Montpellier, II CNRS
161 Rue Ada, 34392 Montpellier - France

³ ITESOFT: Parc d'Andron
Le Séquoia 30470 Aimargues - France

Abstract. The classification of digital documents is a complex task in a document analysis flow. The amount of documents resulting from the OCR retro-conversion (optical character recognition) makes the classification task harder. In the literature, different features are used to improve the classification quality. In this paper, we evaluate various features on OCREd and non OCREd documents. Thanks to this evaluation, we propose the HYBRED (**HY**Brid **RE**presentation of **D**ocuments) approach which combines different features in a single relevant representation. The experiments conducted on real data show the interest of this approach.

1 Introduction

In this paper we present an approach of text classification for OCREd documents. The OCR (optical character recognition) technology allows to transfer digital images into an editable textual document. The task of OCR documents classification is very difficult because of the process of transformation (spelling errors, missing letters, etc.). As a result, the identification and categorization of documents based on their contents are becoming crucial.

Text classification attempts to associate a text with a given category based on its content. This paper focuses on OCREd document classification improvement. [9] addresses the same problems by evaluating OCR documents using various representations. In our work, we propose an evaluation of more significant amount of features. The aim of this paper is to combine the best features on a single representation in order to enhance the classification results. For this purpose, we first propose a hybrid approach. Then, we apply basic classification algorithms (KNN, Naive Bayes, SVM) to evaluate our approach.

To test the relevance of our approach, we have used datasets given by the ITESOFT⁴ company. The ITESOFT datasets result from an OCR retro-conversion of digital documents. This kind of noisy textual data makes the classification

⁴ <http://www.itesoft.fr/>

tasks much harder than using a clean text. ITESOFT uses classes like *attestation salaire* (salary slip), *facture optique* (lenses bill), *frais médecin* (doctor fee), etc.

The paper is organized as follows: section 2 describes different document representation approaches. Section 3 introduces our HYBRED approach. Section 4 discusses the experiments and results. Finally, section 5 concludes the paper giving a summary and the future work.

2 Related work

Text categorization [16] is the task of automatically sorting documents into categories from a predefined set. An important task in text categorization is to prepare and represent text in a way usable by a classifier. Many methods can be used in document representation. In [9] the authors investigated seven text representations to evaluate the classification of OCR texts. Among the different methods of text representation, we present some of the relevant works in related areas as following (1) document features and (2) document representation.

Document features: Different features can be used in order to represent data for a classification task:

- **N-grams of words:** N words sequence [4], for instance:
 - N=1 (unigrams): "biology", "medical", "fee", "disease", etc.
 - N=2 (bigrams): "medical-biology", "disease-fee", etc.
 - N=3 (trigrams): "sick-n°-bill", "prescription-acts-information", etc.
- **Lemmatization:** lemmatization consists in using the canonical form of each word (e.g. the verbs by their infinitive form and the nouns by their singular form). [3] reveals that it is necessary to be careful with the use of the lemmatization in the extraction of relevant words. Indeed, by using the lemmatization technique, we are likely to lose crucial information because the contexts of the words in singular and plural can be associated with distinct concepts.
- **N-grams of characters:** The N-gram of characters is a sequence of N characters from any given sequence ([15], [17], [18]). For example:

"the nurse feeds infant".

if $N = 3$, we have: [the, he_, e_n, _nu, nur, urs, rse, se_, e_f, _fe, fee, eed, eds, ds_, s_i, _in, inf, nfa, fan, ant].

- **Statistical filtering:** We can apply two main methods to assign a statistical weight to each feature: on the one hand, the "document frequency" approach calculates the frequency of a word in the document. On the other hand, "TF.IDF method" [8] measures the importance of a word based on its frequency in the document weighted by the frequency of the word in the complete corpus.

- **Grammatical filtering:** This approach selects the words according to a Part-of-Speech (POS) tag (grammatical label: Noun, verb, adjective, etc.) or combination of Part-of-Speech tags (noun-verb, noun-adjective, etc.). To label a word, the "*Tree Tagger*⁵" [6] tool can be used.

Document representation: The Salton [1] representation consists in building a vector for each text of the corpus. In general, the only used information is the presence and/or the frequency of words. This representation transforms each text into a vector of n dimensions. The n features may be the n different words appearing in the documents. A segmentation step [2] is necessary for segmenting the sentences of the text into distinct words in order to identify lexical units which constitute the vector. There are also several ways to assign a weight to a feature. Two main methods can generally be used. The first one is based on a *boolean* model (i.e. "1" if the word is present in the text, "0" otherwise) and the second one is based on a *frequency* model (i.e. number of occurrences of the word in the text).

After a relevant document representation, we apply classical algorithms which will be described in section 4

3 Our Approach

In this section, we present our approach called HYBRED (*HYB*rid *RE*presentation of *D*ocuments). This approach is used to represent data for a classification process.

3.1 Motivation

In our context, it is important to propose an approach that improves the performances of complex data classification. Various experiments are carried out on different features: N-grams of words, n-grams of characters, and selection of grammatical categories. These features are associated with statistical approaches (TF, TF-Idf). This work enabled us to identify and combine the most relevant features for classification tasks.

3.2 Choice of the features

The experiments presented in section 4 and the results of the state-of-the-art lead us to select three methods:

- Grammatical filtering.
- N-grams of characters.
- Statistical filtering.

⁵ <http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/>

This choice is motivated by the following reasons. The choice of grammatical filtering is to retain only data respecting a grammatical category (noun, verb, adjective, noun-verb, noun-adjective, etc.). The main objective of this process is to keep relevant semantic knowledge.

Many studies in the literature [22] show that nouns are relevant for the determination of the documents meaning and topics.

The N-gram of characters based data representation is motivated by the data complexity (noisy data from the OCR retro-conversion). Indeed, the use of N-grams of characters is adapted to the context of OCRed data [9], which our experiments have also confirmed. The last selected method is the application of statistical measures in order to give a weight to the selected features. By assigning a weight, we favor the most discriminating features in a specific class.

3.3 Order of features

The selection of methods depends on the order in which they are used. The application of grammatical filtering can only be applied on a "full" word. In other words, it is not applicable on words truncated by N-grams of characters. The established order is: application of POS tagging (selection of words according to their tag) followed by a representation of N-grams of characters. After the features selection and their N-gram of characters representation, we apply a statistical filtering. This one allows to give an adapted weight to the different features.

3.4 How to combine the approaches?

In the previous section, we showed that the different processing techniques must be applied in a certain order. Now we describe the combination of these approaches.

As a first step, the selection of words with Part-of-Speech tagging is done. The selection depends on the tags. For instance, with the "noun" and "verb" selection, the sentence "or le bijoux plaqué or a du charme (the gold-plated jewellery has charm)", returns the following results: "bijoux plaqué or a charme". We can note that the grammatical filtering can distinguish the word "or" (*French conjunction*) from the *noun* "or" (gold).

After this initial processing, we represent the words from the N-grams of characters. The application of N-grams of characters process gives three possibilities of representation:

- The first representation is based on a bag of selected words (grammatical selection). The application of N-grams where $N = 5$ gives the following result:

"_bijo, bijou, ijoux, joux_, oux_p, ux_pl, x_pla, _plaq, plaqu, laqué, aqué_, qué_o, ué_or, é_or_, _or_a, or_a_, **r_a_c**, **_a_ch**, **a_cha**, _cham, chamr, harme, arme."

This application is flawed because it adds noise and unnecessary N-grams. For example: "a_cha" is a N-gram which represents noise (N-grams from the

fragment "a du charme (has charm)" where the word "du" was deleted). Indeed, the elimination of the words of the initial sentence returns irrelevant results.

- A second kind of representation is based on the application of N-grams of characters for each extracted word separately. As result, we have:

"_bijo, bijou, ijoux, joux_, _plaq, plaqu, laqué, aqué_, _cham, chamr, harme, arme_"

This representation corrects the defects caused by the previous method but provides fewer data (in particular with short words). For example, by using the N-grams of characters with $N \geq 5$, the noun "or" cannot be identified. This deletion causes a loss of information.

- These representations have major defects with the introduction of **noise** (first method) and **silence** (second method). Thus, we have introduced a principle of border. It corrects the noise added during the first proposed processing. This method takes into account groups of words (e.g. "plaqué or"). The result according to the principle of border is shown below:

"X bijoux plaqué or a X charme", "X" represents the border.

Then we can extract the 5-grams of characters in the two text fragments(i.e. " bijoux plaqué or a " and " charme ") :

"_bijo, bijou, ijoux, joux_, oux_p, ux_pl, x_pla, _plaq, plaqu, laqué, aqué_, qué_o, ué_or, é_or_, _or_a, or_a_, _cham, chamr, harme, arme_"

In the next subsection, we introduce our HYBRED approach.

3.5 The HYBRED Approach

The principles that we have chosen for data representation in our system are the following.

Step (a): Selection according to a Part-of-Speech tag.

A selection of data according to a Part-of-Speech tag can keep discriminating data. The experiments carried out in section 4.2 show that a selection according to the tags NV (**NounVerb**), NA (**NounAdjective**), or NVA (**NounVerbAdjective**) is relevant and provides improved quality while reducing the size of the vector representation.

Step (b): Application of the principle of border.

The LEXTER system [5], proposed by D. Bourigault is a terminology extraction tool which uses the border technique. This system extracts maximum nominal phrases by identifying border marker. These boundaries are linguistically determined (examples of border: "preposition + possessive adjective", "preposition

+ determiner”, etc.). The candidate terms (maximum nominal phrases) are extracted thanks to the use of the border marker information.

In our study, the words giving less information (i.e. with less relevant Part-of-Speech tags) are replaced by a border. The objective is the same as the LEXTER system. The difference of our work is that our borders are the words/tags that are less relevant to the classification task (adverb, preposition, etc.).

Step (c): Representation with the N-grams of characters.

After retaining the data belonging to a Part-of-Speech tag and applying the principle of border, the next step is to represent the characters with the N-grams. It is a merger of N-grams of different fragments separated by the border.

$$\text{Nbr-N-grams of characters} = \sum_{i \in \{\text{all fragments}\}} \text{N-grams}(\text{fragment}_i)$$

After this N-grams of characters representation, we realize a filtering step of irrelevant N-grams. Our goal is to remove the uncommon N-grams which may constitute noise (number of N-grams \leq threshold).

Step (d): Statistical filtering.

Finally, following many works in the literature, we apply a filter based on statistical TF.IDF to assign weights to features.

3.6 Example of the application of the HYBRED approach

This section develops a complete example of the HYBRED approach. We consider the sentence "It requires infinite patience to wait always what never happen".

(a) The selection of data according to the combination NVA (Noun Verb Adjective) returns: "requires infinite patience wait happen".

(b) The application of the principle of border, gives us: "X requires infinite patience X wait X happen".

(c) The N-grams of characters representation, where $N = 3$ returns:

word	N-grams of characters
[_requires infinite patience.]	[_re, req, equ, qui, uir, ire, res, es_,s_i, _in, inf, nfi, fn, ini, nit, ite, te_, e_p, _pa, pat, ati, tie, ien, enc, nce, ce_]
[_wait_]	[_wa, wai, ait, it_]
[_happen_]	[_ha, hap, app, ppe, pen, en_]

Thus, we can calculate the sum of all 3-grams:
N-grams(" _requires infinite patience_") + N-grams(" _wait_") + N-grams(" _happen_").

Finally, digital filtering (pruning according to a threshold and TF.IDF) will be applied. The threshold (manually given) is used to reduce the number of features in step (c) (N-grams representation).

After applying these different stages we obtain a representation for each text of the corpus.

4 Experiments

In this section, we present the various tests that we carried out to determine the relevant features. Then we test the relevance of our HYBRED approach. First, we present our experimental protocol.

4.1 Experimental protocol

To assess the relevance of the various features, we choose to use supervised learning algorithms as following: K Nearest Neighbours (KNN) [14], Naive Bayes ([11], [19]), and Support Vector Machines (SVM) algorithm [12]. These algorithms are often efficient for automatic classification of documents ([10], [13]).

Note that we have not implemented these algorithms, but we used the Weka⁶ software [7]. Weka contains a set of data-mining algorithms implemented in Java language. It contains tools for data pre-processing, classification, regression, clustering, and visualization. The parameters used with the classification algorithms are the default Weka ones.

In order to determine which features are most relevant, we compare the results with the three algorithms. The results of the classification are based on the accuracy of these algorithms (correct classification of documents in the appropriate categories).

$$\text{Accuracy} = \frac{\text{number of documents correctly classified}}{\text{all documents classified}}$$

This accuracy measure has been calculated after applying a 10 fold cross validation.

To assess the usage performance of our various features for a classification task, we used two OCRed text collections:

Dataset A is a corpus obtained after applying an OCR system, it consists of 250 documents divided into 18 categories. This one is given by ITESOFT company. The texts contain only few sentences that are well formulated in natural language. Note that some classes contain more documents than others (unbalanced distribution between classes). The second collection (dataset B) is given by the same company (ITESOFT): 2021 OCRed texts divided into 24 categories. The main characteristic of the dataset B is the diversity of documents and their lack of content. These collections represent newsletters, certificates, expenses, invoices, etc.

Table 1 summarizes the main features of the used datasets .

4.2 Experimental Results

We present the experiments under different features and HYBRED approach. We give results on the dataset B. For dataset A, the experiments are given in

⁶ www.cs.waikato.ac.nz/ml/weka

	Dataset A	Dataset B
Number of texts	250	2000
Number of categories	18	24
Size (MB)	52	29.7
Type of texts	OCR documents	OCR documents

Table 1. Summary of main characteristics of the used datasets.

annex. Actually, corpus B is the most representative one according to its size and its difficulty of processing (heterogeneous and noisy data). In addition, the documents of dataset B has fewer words: 130 words per document(average) for the dataset B and 295 words(average) per document for the dataset A.

The table 2 presents the results obtained with the various features according to a 10 fold cross-validation.

algorithm	K-NN		SVM		NB	
	frequency	TF.IDF	frequency	TF.IDF	frequency	TF.IDF
word	91.1	91.1	95.8	95.8	94.1	93.8
2-words	92.2	90.9	93.7	93.7	91.9	92.2
3-words	90.5	90.5	90.1	89.9	82.8	86.1
2-character	73.7	72.6	89.6	88.2	74.3	58.5
3-character	85.7	86.0	96.5	96.8	93.4	91.9
4-character	95.0	96.1	96.0	96.3	93.1	90.7
5-character	91.4	92.5	96.2	95.6	92.0	90.8
Lemme	92.3	93.8	95.4	95.5	93.7	94.4
N	91.1	93.0	95.6	95.1	93.6	94.6
V	88.2	87.5	88.4	87.8	85.2	84.9
NV	92.4	92.7	95.5	95.5	94.1	94.3
NVA	93.3	92.6	95.6	95.8	94.1	94.5
NA	92.8	92.4	95.6	95.4	93.9	94.8
VA	92.0	91.4	93.7	93.7	91.7	91.4

Table 2. Results of dataset B with various features (accuracy).

We observe that the best results are obtained with the SVM algorithm compared to the K-NN and Naive Bayes approaches. The representation with the N-grams of characters gives good results with both datasets. In general, we note that the results are significantly poor when $N = 2$. The application of the selection according to a Part-of-Speech tag may, in some cases, prove the ability of our method to select only the discriminating data (with a significant reduction of the representation space, as we will show in subsection 4.3). Our results show that a combination as NV (**N**oun **V**erb), NVA (**N**oun **V**erbe **A**djective), and NA (**N**oun **A**djective) is the most relevant among the possible combinations.

The results with the dataset A (presented in annex) are usually better than with the corpus B. We can explain the results by the complexity of the corpus B (corpus with noise and with fewer words per document).

The table 3 presents the results obtained by applying the HYBRED approach on the dataset B. The results of the dataset A are given in annex. In all cases, we obtain the best quality with the **SVM** algorithm compared to the K-NN and Naive Bayes approaches. We note that the selection NV (Noun Verb) associated with the 4-grams gives very satisfying results on the dataset B. We noted generally the same behavior on the dataset A.

K-NN Algorithm (TF.IDF)						
Features	N	V	NV	NVA	NA	VA
2-character	74.8	85.5	74.6	72.7	74.6	77.3
3-character	85.0	85.5	95.8	87.0	86.3	86.6
4-character	85.0	86.5	96.7	92.6	92.1	90.2
5-character	91.8	88.4	93.0	93.4	92.4	90.0
SVM Algorithm (TF.IDF)						
Feature	N	V	NV	NVA	NA	VA
2-character	89.5	89.9	87.4	86.43	89.0	88.1
3-character	96.4	94.2	96.6	96.9	96.8	96.3
4-character	96.5	93.8	98.0	96.8	96.7	95.8
5-character	96.4	93.2	96.8	96.8	96.7	95.2
NB Algorithm (TF.IDF)						
features	N	V	NV	NVA	NA	VA
2-character	61.4	92.6	60.7	59.4	73.5	63.6
3-character	61.4	88.3	60.7	92.2	73.5	91.4
4-character	92.6	88.3	96.9	92.2	92.7	91.9
5-character	92.6	86.9	93.1	92.1	92.4	91.5

Table 3. Accuracy obtained with the HYBRED approach for the corpus B

4.3 Summary

As presented in the experiments, it is very difficult to determine the best methods (a lot of results seem very close). However, we find that some methods behave differently.

Compared to the usage of each method separately, we observe that HYBRED method based on the combination of features tends to improve the quality of classification. We can see this improvement on both datasets. The table 4 presents a comparison between the proposed approach with a combination of NV (NounVerb) combined with a 4-grams of characters representation and the various features.

Algorithm	Dataset A			Dataset B		
	KNN	SVM	NB	KNN	SVM	NB
word	96.5	97.5	96.7	91.1	95.8	93.8
3-character	94.7	97.9	93.5	86.0	96.8	91.9
4-character	97.5	98.3	94.3	96.1	96.3	90.7
5-character	96.7	98.3	95.1	92.5	95.6	90.8
NV	95.9	98.0	96.7	92.7	95.5	94.3
NVA	95.5	98.0	96.7	92.6	95.8	94.5
NA	95.1	98.0	96.7	92.4	95.4	94.8
HYBRED	96.8	98.4	93.6	96.7	98.0	96.9

Table 4. Comparison of different features used for the HYBRED approach

In general, improvements were obtained by applying the HYBRED approach. Thus, table 4 shows that the HYBRED approach always improves the results (more or less significant depending on the dataset) with the SVM algorithm. This is particularly interesting because this algorithm shows the best behavior. Moreover, this improvement is particularly important with the most complex corpus (corpus B).

In table 5, we present a comparison of representation space (size) with and without applying the HYBRED approach. We note that the application of HYBRED approach reduces significantly the representation space. The results obtained using the HYBRED approach are given with the NV (Noun Verb) combination followed by a 4-grams of characters representation.

Research Area	Without applying HYBRED		After applying HYBRED
	Word	N-grams (N=4)	NV + N-grams (N=4)
Corpus A	12307	2087	1603
Corpus B	37837	4485	3294

Table 5. Comparison of representation space with and without the HYBRED approach

5 Conclusion and Future Work

In this paper, we proposed a new document representation approach for textual documents classification. The experiments carried out on OCRed data produced good classification results. One perspective of this work is based on the application of semantic knowledge (for instance, the use of specialized dictionaries to enrich the selected feature) which might improve classification quality. However, such dictionaries are not available for all domains. The second perspective is related to the classification techniques. We are currently experimenting other

algorithms to study the different results according to features and algorithms. We also wish to apply the approach on other types of noisy data in order to reinforce the relevance of our approach in other contexts. We can finally apply our approach for the classification of opinion data (with feature selections as adjective and/or adverb often relevant for opinion documents). Indeed, such opinion data as Blogs has a lot of noise. Then, our approach can be adapted for this type of task (e.g. Blog classification).

References

1. G. Salton: The SMART Retrieval System—Experiments in Automatic Document Processing, Prentice-Hall, Inc, Upper Saddle River, NJ, USA.(1971).
2. C. Hori: Advances in Automatic Speech Summarization, Proc. EUROSPEECH2001, vol. III, 1771–1774, Aalborg. (2001)
3. B. Lemaire: Limites de la lemmatisation pour l'extraction de significations, S. Heiden and B. Peiden (eds.): 9th International Conference on the Statistical Analysis of Textual Data, JADT'2008, 2, 725–732, Lyon, France. (2008)
4. C.M. Tan and Y.F. Wang and C.D. Lee: The use of bigrams to enhance text categorization, Inf. Process. Manage, 38, 4, 0306-4573, 529–546, Pergamon Press, Inc. Tarrytown, NY, USA. (2002)
5. D. Bourigault: LEXTER un Logiciel d'EXtraction de TERminologie. Application à l'extraction des connaissances à partir de textes, Paris, Thèse en mathématiques, informatique appliquée aux sciences de l'Homme, École des Hautes Études en Sciences Sociales. (1994)
6. H. Schmid: Improvements in Part-of-Speech Tagging with an Application to German. In Proceedings of the ACL SIGDAT-Workshop, Dublin. (1995)
7. I. Witten and E. Frank and L. Trigg and M. Hall and G. Holmes and S. Cunningham: Weka: Practical machine learning tools and techniques with java implementations. In: Proc ICONIP/ANZIIS/ANNES'99 Int. Workshop: Emerging Knowledge Engineering and Connectionist-Based Info. Systems. 192-196. (1999)
8. G. Salton and C. Buckley: Term-weighting approaches in automatic text retrieval. Inf. Process. Manage, 24, 5, 513–523, Pergamon Press, Inc. Tarrytown, NY, USA. (1988)
9. M. Junker and R. Hoch: Evaluating OCR and Non-OCR Text Representations for Document Classification. In Proceedings ICDAR-97, Fourth International Conference on Document Analysis and Recognition, Ulm, Germany, August. (1997)
10. F. Sebastiani: A Tutorial on Automated Text Categorisation, Analia Amandi and Ricardo Zunino, Proceedings of the 1st Argentinian Symposium on Artificial Intelligence (ASAI'99), Buenos Aires, AR. (1999)
11. T. Joachims: A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization, ICML, 143-151. (1997)
12. T. Joachims: Text categorization with support vector machines: learning with many relevant features, Proceedings of ECML-98, 10th European Conference on Machine Learning, 1398, Springer Verlag, Heidelberg, Chemnitz, DE, Claire Nédellec and Céline Rouveïrol, 137–142. (1998)
13. Y. Yang and X. Liu: A Re-Examination of Text Categorization Methods, SIGIR, 42-49. (1999)
14. Y. Yang: An Evaluation of Statistical Approaches to Text Categorization, Information Retrieval, 1, 1/2, Kluwer Academic Publishers, 69–90. (1999)

15. W.B. Cavnar and J.M. Trenkle: Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, 1430, categorization, classification, information retrieval, information-retrieval, ir, 161–175, 2, N-Gram-Based Text Categorization, Las Vegas, US. (1994).
16. F. Sebastiani: Machine learning in automated text categorization, *ACM Comput. Surv.*, 34, 1, 0360-0300, 1–47, ACM, New York, NY, USA. (2002).
17. M. Mansur and N. UzZaman and M. Khan: Analysis of N-gram based text categorization for Bangla in a newspaper corpus, *Proc. of 9th International Conference on Computer and Information Technology (ICCIT 2006)*, Dhaka, Bangladesh. (2006).
18. B. Vishnu Vardhan and L. Pratap Reddy and A. VinayBabu: Text categorization using trigram technique for Telugu script, *Journal of Theoretical and Applied Information Technology*, 3, 1-2, 9-14. (2007).
19. D. D. Lewis and M. Ringuette: A comparison of two learning algorithms for text categorization, In *Third Annual Symposium on Document Analysis and Information Retrieval*, 81–93. (1994).
20. S. S. Sterling and S. Argamo and O. Frieder: The Effect of OCR Errors on Text Classification, August, In *Poster Proc. SIGIR*. (2006).
21. U. S. Kohomban and W. S. Lee: Optimizing Classifier Performance in Word Sense Disambiguation by Redefining Sense Classes, *IJCAI*, 1635-1640, *IJCAI 2007*, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12. (2007).
22. I. Bayoudh and N. Béchet and M. Roche: Blog Classification: Adding Linguistic Knowledge to Improve the K-NN Algorithm, *Intelligent Information Processing*, 68-77, 5th IFIP International Conference on Intelligent Information Processing, October 19-22, 2008, Beijing, China, (2008).

Annex

Algorithm	K-NN		SVM		NB	
	frequency	TF.IDF	frequency	TF.IDF	frequency	TF.IDF
Word	95.7	96.5	97.9	97.5	96.7	96.7
2-words	88.7	85.5	90.3	86.7	91.9	89.5
3-words	77.1	76.3	74.2	74.6	77.1	73.0
2-character	94.3	77.9	95.9	85.9	87.9	77.9
3-character	96.3	94.7	97.9	97.9	96.3	93.5
4-character	94.3	97.5	97.9	98.3	96.3	94.3
5-character	95.5	96.7	97.5	98.3	96.7	95.1
Lemme	95.3	95.1	95.8	96.7	95.1	95.9
N	95.9	95.9	97.1	97.1	96.7	97.1
V	84.7	83.5	86.3	83.9	92.7	84.3
NV	96.3	95.9	97.1	98.0	96.7	96.7
NVA	95.9	95.9	97.5	98.0	97.1	96.7
NA	95.5	95.1	97.5	98.0	97.1	96.7
VA	93.1	92	95.5	95.0	95.1	91.0

Table 6. Results obtained with dataset A with various features (accuracy).

K-NN Algorithm						
Feature	N	V	NV	NVA	NA	VA
2-character	75.6	78.4	72.8	71.2	71.2	77.6
3-character	92.0	92.0	94.0	95.2	94.0	90.4
4-character	94.8	86.0	96.8	98.0	96.8	90.8
5-character	94.4	85.2	95.2	96.4	94.0	90.4
SVM Algorithm						
Feature	N	V	NV	NVA	NA	VA
2-character	90.0	81.6	88.8	84.4	86.4	87.2
3-character	97.6	91.2	98.0	97.6	98.4	96.4
4-character	97.6	95.2	98.4	98.4	98.0	98.0
5-character	96.8	94.0	98.0	98.4	98.0	96.4
NB Algorithm						
Feature	N	V	NV	NVA	NA	VA
2-character	77.6	76.4	78.4	76.8	76.8	77.6
3-character	93.6	88.8	90.4	92.0	91.6	91.6
4-character	94.8	91.2	93.6	94.8	96.0	92.4
5-character	92.8	90.8	92.0	94.0	96.4	91.6

Table 7. Accuracy obtained with the approach HYBRED for dataset A