Hypatia Digital Library:A text classification approach based on abstracts

FROSSO VORGIA[1,a], IOANNIS TRIANTAFYLLOU[1,b], ALEXANDROS KOULOURIS[1,c]

[1]Department of Library Science and Information Systems
Technological Educational Institute of Athens, Aegaleo, Athens, Greece

[a)]frossovorgia@gmail.com, [b)]triantafi@teiath.gr, [c)]akoul@teiath.gr

Abstract: The purpose of this paper is to investigate the application of text classification in Hypatia, the digital library of Technological Educational Institute of Athens, in order to provide an automated classification tool as an alternative to manual assignments. The crucial point in text classification is the selection of the most important term-words for document representation. Classic weighting method TF.IDF was investigated. Our document collection consists of 718 abstracts in Medicine, Tourism and Food Technology. Classification was conducted utilizing 14 classifiers available on WEKA. Classification process yielded an excellent ~97% precision score.

Keywords: **Digital libraries, Text classification, WEKA, Word stemming.**

Introduction

Digital libraries and repositories serve as valuable access points to information. Their continuous enrichment with digital objects indicates their significance and also raises a need for immediate classification (Triantafyllou I. et al. 2014). On the contrary, digital libraries still conduct manual subject classification based on classification systems, subject headings, thesauri, ontologies. Nevertheless, this process is time consuming, involving experienced human resources (Joorabchi A. and Mahdi A. 2014), and the results might differ from one library to the other.

The purpose of this paper is to examine a simple application of an alternative solution to the aforementioned problem. That is the application of text classification methods in digital libraries using the abstracts of digital objects. Abstracts are considered to be the best option to experiment with as they might be the only available texts which represent the content of resources, since full text is not always available due to copyrights constraints. The main source of abstracts is Hypatia, the digital library of Technological Educational Institute of Athens. We apply abstract representation by word weighting with TF.IDF. In the final phase, we use basic classification techniques in WEKA (Waikato Environment for Knowledge Analysis), an open source software which allows classification, clustering and association rule mining (Machine Learning Group at the University of Waikato n.d.; Bouckaert R.R. et al. 2010).

Methodology

Text classification/categorization (TC) is the task of classifying texts in classes which have been defined in advance (Sebastiani F. 2002). So far TC has been utilized in a machine learning approach, conducted with the use of classifiers (algorithms). The most extensively used ones for TC are NaïveBayes and NaïveBayesMultinomial (Witten I.H. et al. 2011) but there are more classifiers, such as Support Vector Machines (SVM), MultilayerPerceptron, IBk, DecisionTable, etc. which can be exploited (Triantafyllou I. et al. 2001). TC has achieved positive results from e-mail labeling (spam or no spam) to twitter trending toppings' classification (Irani D. et al. 2010; Awad W.A. and ELseuofi S.M. 2011).

Dataset collection

We collected the abstracts from 718 digital objects, considering that they are in Greek and already classified either in Medicine or Tourism or Food Technology, as these classes were the most populated. Although, Hypatia was the main source of abstracts, it was impossible to extract data from this source only, since it was still under enrichment process. Thus, we decided to derive abstracts from other DL aiming to create a balanced corpus for the three classes. Analytically, abstracts were assembled from 9 Greek academic digital libraries and repositories:

- Hypatia- Technological Institute of Athens (512),
- The digital repository of Agricultural University of Athens (AUA) (73),
- Eureka!- Technological Institute of Thessaloniki (47),
- Dioni- University of Piraeus (45),
- Psepheda- University of Macedonia (19),
- DSpace@NTUA- National Technical University of Athens (11),
- Nemertes- University of Patras (9),
- E-Locus- University of Crete (1),
- Anaktisis-Technological Educational Institute of Western Macedonia (1).

However, each digital library applies different subject classification tools, such as Library of Congress Subject Headings (LCSH) or Agrovoc thesaurus, to assign the subject categories. In order to ensure uniformity and accordance in our dataset, Dewey Decimal Classification was used as a guide to include or discard the abstracts. The only exception was a set of 22 abstracts from the digital repository of Agricultural University of Athens. These were theses from the department of Science and Food Technology, which also included relevant words, so they were considered to have a connection to Food Technology.

The final text corpus consisted of 373 abstracts in Medicine, 223 in Tourism and 122 in Food Technology.

Text handling and word stemming

Initially, a basic text pre-processing is necessary to minimize the noise. A system of natural language communication includes nouns, verbs, adverbs, conjunctions, etc. Not every part of speech has useful meaning. In addition, it is essential to stem the words of the texts. Greek is a highly inflected language, meaning that almost every word in a sentence has an affix. Stemming, or conflation, is the process of reducing the words to their stem by taking off the affixes (Croft W.B. et al. 2010). Word stemming or term conflation process is performed by using a score mechanism which is based on the similarity estimator (1), especially designed to assign higher scores to morphological variations of the same root form.

$$\mathrm{Similarity}(\mathrm{W1},\mathrm{W2}) = \mathrm{CommonPositionTrigrams}\big(\mathrm{Left}(\mathrm{W1},\mathrm{L})\,,\mathrm{Left}(\mathrm{W2},\mathrm{L})\big)/\mathrm{L}$$
$$\text{where } \mathrm{L} = \big(\mathrm{Length}(\mathrm{W1}) + \mathrm{Lenght}(\mathrm{W2})\big)/2\,, \mathrm{L}\epsilon\mathrm{N} \qquad (1)$$

Efficient grouping of words in terms has been achieved with a similarity score of 66,6%.

Abstract representation

The feature space is a crucial aspect in the performance of any text classification model. Any term-word within the abstracts corpus constitutes a candidate feature with the exception of functional words that are excluded. Feature selection consists of reducing the vocabulary size of the training corpus by selecting term-words with the highest indicative efficiency over the class variable. The TF.IDF metric (Jones K.S. 1972; Croft W.B. et al. 2010) is one classic approach to sort the candidates term-words in a list by scoring their correlation importance to the class variable. In our case TF is the frequency of feature f within the corpus, and IDF is the logarithm of N/Nf, where N is the total number of abstracts and Nf is the number of abstracts containing the feature f. The selected features are the most dominant ones based on that score.

An additional important issue to consider is the frequency of a term-word when determining the abstract vector. There are cases where a term-word is more indicative to the relevance of the abstract when it appears several times. However, this is not always true since long abstracts usually introduce a lot of noise. We experimented with two alternatives concerning the strength of the selected features: the binary (boolean) appearance (0 or 1), and the actual value of the term frequency in the abstract.

Text classification with WEKA

Following the extraction of the most important words in the corpus, the abstract representation sampling consisted of 10, 15, 20, 25, 50, 75, 100, 150, 200, 300, 500 and 750 term-words. In order to achieve accurate estimation (Kohavi R. 1995), a 10-fold cross-validation method was used. Precision, Recall and F-score were the evaluation metrics applied for comparing and evaluating the performance of classifiers.

The classifiers were chosen from version 3.7.12 of WEKA for developers. These were:

- Two Bayesian classifiers: NaïveBayes and NaïveBayesMultinomial,
- Three Function classifiers: MultilayerPerceptron, SimpleLogistic, and SMO(SVM),
- Two Lazy classifiers: IBk and Kstar,
- Two Metalearning classifiers: ClassificationViaRegression and LogitBoost,
- Three Rule classifiers: DecisionTable, JRip, and PART,
- Two Tree classifiers: LMT and RandomForest.

Results and Discussion

Table 1. F-score (%) with words from TF.IDF

	Vector Size	*10*	*15*	*20*	*25*	*50*	*75*	*100*	*150*	*200*	*300*	*500*	*750*
	Classifier	*W*	*W*	*W*	*W*	*W*	*W*	*W*	*W*	*W*	*W*	*W*	*W*
BIN	NaiveBayes(NB)	83	83	84	86	92	92	93	93	93	94	93	95
	NB Multinomial	77	82	85	88	93	94	94	94	93	95	95	96
	MLP	81	82	83	87	92	95	95	95	95	96	fail	fail
	SimpleLogistic	80	83	86	87	93	94	95	95	96	95	96	96
	SMO	84	83	86	87	92	93	93	94	95	95	95	96
	IBk	81	80	80	85	86	86	87	83	80	79	67	71
	Kstar	81	81	82	86	87	88	87	84	81	80	70	73
	ClassViaRegression	81	84	86	86	91	93	93	93	93	94	93	95
	LogitBoost	81	82	84	88	92	93	94	94	94	96	95	96
	DecisionTable	82	81	83	81	88	92	92	92	91	92	92	91
	JRip	79	81	83	83	90	91	93	92	92	90	91	93
	PART	82	81	84	86	90	91	92	92	92	92	93	94
	LMT	80	82	86	87	93	94	96	95	96	95	96	96
	RandomForest	82	82	86	89	93	95	96	96	96	97	96	97
TF	NB	74	75	77	80	85	87	89	90	90	92	93	92
	NB Multinomial	81	83	86	87	92	94	94	95	95	97	96	96
	MLP	80	81	84	87	91	94	92	93	91	84	fail	fail
	SimpleLogistic	82	84	86	87	93	94	95	94	94	95	95	95
	SMO	76	78	80	83	90	93	92	92	93	94	92	94
	IBk	75	75	76	80	79	82	79	78	75	75	71	66
	Kstar	79	77	79	80	80	80	77	73	72	70	60	57
	ClassViaRegression	81	84	86	87	90	92	92	91	92	92	92	92
	LogitBoost	80	83	85	87	92	94	93	94	94	96	95	95
	DecisionTable	82	82	81	81	89	92	91	91	91	91	91	92
	JRip	80	81	81	83	90	91	92	92	91	91	91	91
	PART	80	81	83	83	90	92	91	92	92	91	91	90
	LMT	82	84	86	87	93	94	95	94	94	95	95	95
	RandomForest	80	85	87	89	93	95	96	96	96	96	96	97

All of the 14 classifiers were tested (Table 1) and the results of the best classifiers are shown on Table 2.

Table 2. Results (%) of the Best Classifiers

Classifier	*Method*	*Vector*	*F-score*	*Precision*	*Recall*
RandomForest	TF.IDF-bin	300W	97,40	97,40	97,40
RandomForest	TF.IDF-tf	750W	97,40	97,40	97,40
NaïveBayesMultinomial	TF.IDF-tf	300W	97,25	97,30	97,20
SMO	TF.IDF-bin	750W	96,70	96,70	96,70

The best classifier was RandomForest which achieved the highest Precision, Recall and F-score rates in both methods: TF.IDF-bin (binary appearance) and TF.IDF-tf (frequency appearance).

Another critical observation is that binary representation of document vectors acts in a more beneficiary way than frequency representation in the performance of the examined classifiers. This is illustrated in Fig.1 where the dark line corresponds to binary representation while gray one indicates term frequency representation.

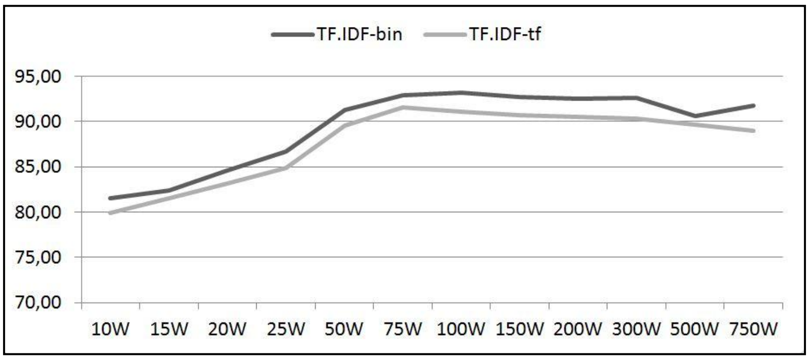

Fig 1. Average F-score (%) performance for all classifiers of Binary(bin) and Frequency(tf) representations

Conclusion

We assess the use of text classification in digital libraries. The classic weighting method TF.IDF with binary and term frequency appearance were used. The software used to apply classification algorithms was WEKA. Overall, this research indicated that digital libraries could substitute manual classification with our proposed approach. TF.IDF approach was proved to be effective, produced an F-score greater than 97% in some classifiers. However, this raises the question whether we could exploit the same approach using smaller texts and better term-word representation. Hence, in the future we would like to experiment with titles instead of abstracts. Another important future aspect is to apply clustering techniques to encourage and identify classes and topic fusion.

References

Awad, W.A., ELseuofi, S.M. (2011). Machine learning methods for spam e-mail classification. *International Journal of Computer Science & Information Technology*. 3(1), pp. 173-184.

Bouckaert, R.R., Frank, E., Hall, M.A., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. H. (2010). WEKA- experiences with a Java open-source project. *Journal of Machine Learning Research*. 11, pp. 2533-2541.

Croft, W.B., Metzler, D. and Strohman, T. (2010). *Search engines: information retrieval in practice.* Addison-Wesley.

Irani, D., Webb, S., Pu, C. and Li, K. (2010). Study of trend-stuffing on twitter through text classification. *Proceedings of Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS).*

Jones, K. S. (1972). A statistical interpretation of term frequency and its application in retrieval. *Journal of Documentation.* 28(1), pp. 11-21.

Joorabchi, A., Mahdi, A. (2011) An unsupervised approach to automatic classification of scientific literature utilizing bibliographic metadata. *Journal of Information Science.* 37(5), pp. 499-514.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI),* pp.1137-1145.

Machine Learning Group at the University of Waikato. (n.d.) *WEKA 3- data mining with open source machine learning software in Java.* Available at: http://www.cs.waikato.ac.nz/~ml/weka/index.html [Accessed: 30/06/2015]

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR).* 34(1), pp. 1-47.

Triantafyllou, I., Demiros, I. and Piperidis, S. (2001). Two Level Self-Organizing Approach to Text Classification. *Proceedings of RANLP-2001: Recent Advances in NLP*.

Triantafyllou, I., Koulouris, A., Zervos, S., Dendrinos, M., Kyriaki-Manessi, D. and Giannakopoulos, G. (2014). Significance of Clustering and Classification Applications in Digital and Physical Libraries. *Proceedings of 4th International Conference IC-ININFO 2014,* Madrid, Spain.

Witten, I.H., Frank, E. and Hall, M.A. (2011). *Data mining: practical machine learning tools and techniques*. Morgan Kaufmann.