COMPARISON OF THE PERFORMANCE OF GENRE CLASSIFIERS TRAINED BY DIFFERENT MACHINE LEARNING ALGORITHMS

Vedrana Vidulin, Mitja Luštrek, Matjaž Gams
Department of Intelligent Systems
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
Tel: +386 1 477 3147; fax: +386 1 425 1038
e-mail: vedrana.vidulin@ijs.si

ABSTRACT

Modern search engines aim at classifying web pages not only according to topics, but also according to genres. This paper presents the results of an attempt to train a genre classifier. We present features extracted from a 20-genre corpus used for training the genre classifiers and the results of using different machine learning (ML) algorithms in the process of learning. Success of the genre classifiers was measured by accuracy, precision, recall and F-measure. Accuracy did not turn out to be a good indicator of classifier success. In the case of other measures the results show that different algorithms should be used for training purposes depending on whether the user wishes to obtain high precision or high recall.

1. INTRODUCTION

A good question to start with is why we want to classify a web page according to genre. For example, if we are interested in elephants and search for the keyword “elephant”, we can get as a result links to pages that scientifically describe the life of elephants, but we can also get links to web pages that describe movie with the title “Elephant” or a newspaper article about saving the elephants in Africa. However, if we can define that we want to search only for journalistic materials about elephants, than we can get more specific results in accordance with our interest. Classification of web pages according to genres can make our life easier, but what exactly is a genre?

In general, a genre could be described as a style of a web page [4]. A web page is used to send a message to the user. Message has a topic, for example life of the elephants, but it also tries to communicate that topic in a specific way. To a zoologist it will give a high number of objective facts about elephants. To a user wishing to be entertained, it will communicate the message about elephants in other ways, e.g. by presenting pictures and video material about elephants. In the light of the previous explanation, we chose the definition of genres as “named socio-cultural communication artifacts, linked to a society or a community, bearing standardized traits, leaving space for the creativity of the text producer, and raising expectations in the text receiver” [12].

The field of our interest is finding the set of web page features that could be used for discriminating web pages according to genre and to train the classifier that could be used as a part of a search engine. Therefore, results of searching could be presented to a user not only according to topic, but also according to genre.

For training the classifier it is important to choose a suitable ML algorithm. The selection of algorithms tested was inspired by the literature about genres [1-6, 8, 11-13]. Weka [14] as an environment for conducting data mining and ML experiments was used for the experiments.

In Section 2 is described the corpus on which the experiments were conduced, in Section 3 the features extracted from web pages, in Section 4 ML problem, in Section 5 the results of the experiments and in Section 6 the conclusion.

2. 20-GENRE COLLECTION

20-Genre Collection was compiled at Jožef Stefan Institute and consists of 1539 web pages divided into 20 genres. The genre categories are: index, childrens’, journalistic, prose fiction, faq, scientific, entertainment, official, blog, error message, informative, poetry, personal, user input, gateway, shopping, commercial/promotional, adult, community, content delivery.

The web pages were collected from the Internet using the most popular keywords like “Britney Spears”. Keywords were chosen dependent of statistics provided by Google Zeitgeist with an intention to build the classifier that will not have a problem with recognizing the most popular web pages.

The corpus was manually annotated by two independent annotators. Their labels disagreed on about a third of the pages in the data set, so a reassessment was made for those documents. The intention was to collect an approximately equal number of pages for each genre, but this task proved to be very difficult.

Important characteristic of the corpus is that it is multilabeled, what means that one page can belong to the multiple categories.

3. FEATURES

Text is more that just a set of the words and can be described by a set of various features [4]. In this case, we do not have only texts that need to be described, because web pages also have formatting and multimedia elements and other special characteristics that can be used to discriminate between different genres. All the features we used are presented in Table 1 divided into three groups.

URL Features	• URL Depth • Document Type (html, script, doc, output, mix) • Appearance of "/~" in URL • Top-level domain (com, org, edu, net, gov, other) • Appearance of the 35 most commonly used words in URL address in the corpus (e.g. index, news, faq etc.)
HTML Features	• Number of hyperlinks to the same domain / Total number of tags used in a document • Number of hyperlinks to a different domain / Total number of tags used in a document • Total number of hyperlinks / Total number of characters in a document • Number of tags / Total number of tags used in a document for 73 different tags
Token/Lexical Fetures	• Number of characters • Number of words • Average number of characters per word • Number of content words / Total number of content words used in a document for 50 most commonly used content words in a corpus (e.g. new, dvd, post etc.) • Number of function words / Total number of function words used in a document for 50 most commonly used function words in a corpus (e.g. a, he, in etc.) • Number of punctuation symbols / Total number of punctuation symbols used in a document for 26 punctuation symbols (e.g. ., ;, -, etc.)

Table 1. Features used for the description of web pages

4. ML PROBLEM

The first problem that we encountered was how to handle a multilabeled data set in the process of ML. The problem arose because Weka does not support multilabeled learning. The chosen solution was to separate ML process into 20 sub-processes, one for each genre category. Hence, we prepared 20 data sets of the same data, i.e. values of the features extracted from the web pages. The only element that was changed in each data set was the class. The class is binary, which means that the examples that belong to the class are labeled with yes, and the examples that do not belong to the class are labeled with no.

Some initial experiments were conduced using the SVM algorithm [14], but we did not get satisfactory results. The next idea was to search the literature about genres for algorithms appropriate for training the classifier using our set of features. At the end we chose the following set of algorithms and variations of their parameters:

1. Bagging in combination with REPTree [14]
2. J48 algorithm – implementation of C4.5 [10] in Weka
3. J48 with reduced error pruning option
4. REPTree algorithm
5. IBk – the k-nearest neighbor algorithm [9] with k parameter equal 1
6. IBk – the k-nearest neighbor algorithm with k parameter equal 2
7. IBk – the k-nearest neighbor algorithm with k parameter equal 3
8. IBk – the k-nearest neighbor algorithm with k parameter equal 4
9. JRip rule learner [14]
10. AdaBoostM1 algorithm [14]

Except for the mentioned changes in the parameters of the algorithms, for all other parameters the default values set in Weka were used. Experiments were run in Weka Experimenter. The algorithms were compared to bagging algorithm used in the combination with REPTree that is the default Weka combination. Main reason for comparing performance of other algorithms to a bagging algorithm lies in our aspiration to explore the application of ensemble ML methods. 10-fold cross-validation [7] was used and four performance evaluation measures were observed, i.e. accuracy, precision, recall and F-measure [4].

5. RESULTS

Table 2 presents the level of performance of a bagging algorithm used in the combination with REPTree in terms of four performance measures, and Table 3 the results of the comparisons of algorithms performance.

	Accuracy	Precision	Recall	F-Measure
Adult	97.73	0.77	0.73	0.74
Blog	96.49	0.84	0.37	0.51
Childrens’	96.17	0.97	0.46	0.59
Commercial/promotional	92.14	0.00	0.00	0.00
Community	96.56	0.97	0.37	0.50
Content delivery	92.33	0.90	0.17	0.28
Entertainment	95.32	0.37	0.09	0.15
Error message	96.95	0.85	0.53	0.64
Faq	99.22	0.97	0.86	0.91
Gateway	94.87	0.00	0.00	0.00
Index	86.42	0.62	0.19	0.29
Informative	84.99	0.28	0.03	0.05
Journalistic	90.58	0.76	0.34	0.47
Official	96.56	0.10	0.03	0.05
Personal	93.63	0.64	0.18	0.27
Poetry	97.46	0.87	0.55	0.66
Prose fiction	96.88	0.80	0.40	0.53
Scientific	96.62	0.80	0.42	0.53
Shopping	95.78	0.30	0.05	0.08
User input	95.97	0.75	0.43	0.54

Table 2. Performance of a bagging algorithm used in the combination with REPTree.

	Accuracy	Precision	Recall	F-Measure
Bagging (RepTree)	-	-	-	-
J48	0/12/8	1/11/8	4/16/0	3/17/0
J48 (Reduced Error Pruning)	0/15/5	0/18/2	0/19/1	0/17/3
REPTree	0/17/3	0/18/2	0/20/0	0/20/0
IBk (k=1)	0/6/14	1/8/11	6/12/2	4/14/2
IBk (k=2)	0/2/18	1/6/13	6/13/1	3/12/5
IBk (k=3)	0/9/11	0/14/6	3/12/5	1/14/5
IBk (k=4)	0/9/11	1/11/8	5/11/4	3/12/5
JRip	0/16/4	0/15/5	4/16/0	0/20/0
AdaBoost	0/18/2	0/18/2	0/18/2	0/18/2

Table 3. Comparisons of ML algorithms performance relative to bagging algorithm – The numbers denote: on how many genres was the algorithm better than bagging algorithm / performed equally / performed worse.

Accuracy did not turn out to be a good indicator of the classifier performance. In the cases of all genres and all algorithms, accuracy was greater than 74%. This happened because transforming the 20-class ML problem into 20 binary problems resulted in 20 unbalanced data sets with many negative examples in comparison to the number of positive examples. In this case we could very easily train the classifier that will with high accuracy recognize negative examples but could not recognize positive examples. This was the motivation for searching other more appropriate measures.

In information retrieval measures like precision and recall are often used [4]. F-measure (see Eq. 1), which is combination of these measures, is also common indicator. We were very interested in precision measure that shows us how many positive examples classifier could recognize. In comparison with the bagging algorithm in the cases of 19 genre categories other algorithms were not significantly better. Four ML algorithms did outperform bagging algorithm, but all in the case of the same category, i.e. commercial/promotional. An interesting result is that for commercial/promotional category algorithms that use reduced error pruning had a precision of 0, and all other algorithms had a low precision.

$$\frac{2\times recall\times precision}{recall+precision} \tag{1}$$

In the case of recall measure the situation is highly dependent on the genre category. kNN algorithm was significantly better in the cases of index, journalistic, informative and shopping category. Its performance could be further improved by choosing appropriate value of k. Hence, we need to be cautious because changing the k value results in an increase of one measure and, at the same time, a decrease of other measures. The selection of an appropriate algorithm and parameters of the algorithm must therefore depend on the situation, e.g. if we want to have as a result a high precision classifier we choose one algorithm and parameters, and if we want a high recall classifier other.

An interesting result is that some of the genre categories are sensitive to reduced error pruning. This is manifested in recall equal or near zero when using ML algorithms with reduced error pruning option, whereas other algorithms showed higher level of performance (e.g. in the case of genre category entertainment).

In terms of F-Measure J48 algorithm performed significantly better than bagging algorithm in the case of three genre categories and in all other categories performed equally well. kNN algorithm also outperformed bagging algorithm in some categories but also had significantly lower results in other categories. Therefore, we can conclude that J48 algorithm without the option of reduced error pruning included could be better choice taking into account F-measure than bagging algorithm. Reduced error pruning also had a negative impact in this case.

It can be seen from the experiments that even by using different ML algorithms, performance of the classifiers in some genre categories could not be improved. Some categories like faq are well recognized and some like gateway are not. Apparently that problem lies in the set of chosen features and not in the chosen ML algorithms.

6. CONCLUSION

This paper could be described as lessons learned. From the experiments we learned that the set of features used is not adequate for describing genre categories and that it needs to be upgraded. Decision tree ML algorithms with reduced error pruning did not show as the best solution.

Accuracy did not showed as good measure because of the unbalanced data set with a high ration of negative to positive examples. In the cases of precision, recall and F-measure, which algorithm will be chosen is highly dependent on the need for the classifier that will show high precision, high recall or high values of F-measure.

In the case of a high precision classifier, the recommendation is to use bagging algorithm in combination with REPTree. A high recall classifier could be trained with kNN algorithm. If we take F-measure into account, J48 without the reduced error pruning option is the best solution. Therefore, we can finally conclude that the choice of the most suitable algorithm is highly dependant on the application.

References

[1] S. Argamon, M. Koppel, G. Avneri, *Routing Documents According to Style*, First International Workshop on Innovative Information Systems, 1998

[2] N. Dewdney, C. VanEss-Dykema, R. MacMillan, *The Form is the Substance - Classifications of Genres in Text*, 1998

[3] A. Finn, *Machine Learning for Genre Classification*, 2002

[4] J. Karlgren: *Stylistic Experiments for Information Retrieval*, 2000

[5] J. Karlgren, D. Cutting, *Recognizing Text Genres with Simple Metrics Using*, Proceedings of the 15th. International Conference on Computational Linguistics, 1994

[6] B. Kessler, G. Nunberg, H. Schütze, *Automatic Detection of Text Genre*, Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, 1997

[7] R. Kohavi, *A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection*, IJCAI, 1995

[8] C.S. Lim, K.L. Lee, G.C. Kim, *Multiple sets of features for automatic genre classification of web documents*, Information Processing & Management, 2005

[9] T.M. Mitchell, *Machine Learning*, McGraw-Hill, 1997

[10] J.R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1993

[11] M. Santini, *A Shallow Approach to Syntactic Feature Extraction for Genre Classification*, Proceedings of the 7th Annual Colloquium for the UK Special Interest Group for Computational Linguistics, 2003

[12] M. Santini, *Common Criteria for Genre Classification: Annotation and Granularity***,** Workshop on Text-based Information Retrieval (TIR-06), In Conjunction with ECAI 2006, Riva del Garda, 2006

[13] E. Stamatatos, G. Kokkinakis, N. Fakotakis, *Automatic Text Categorization in Terms of Genre and Author*, Computational Linguistics, 2000

[14] I.H. Witten, E. Frank, *Data Mining – Practical Machine Learning Tools and Techniques*, Elsevier Inc., 2005