Web Genre Classification via Hierarchical Multi-label Classification

Gjorgji Madjarov[1], Vedrana Vidulin[2], Ivica Dimitrovski[1], and Dragi Kocev[3,4](✉)

[1] Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Skopje, Macedonia
{gjorgji.madjarov,ivica.dimitrovski}@finki.ukim.mk
[2] Ruđer Bošković Institute, Zagreb, Croatia
vedrana.vidulin@irb.hr
[3] Department of Informatics, University of Bari Aldo Moro, Bari, Italy
[4] Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
dragi.kocev@ijs.si

Abstract. The increase of the number of web pages prompts for improvement of the search engines. One such improvement can be by specifying the desired web genre of the result web pages. This opens the need for web genre prediction based on the information on the web page. Typically, this task is addressed as multi-class classification, with some recent studies advocating the use of multi-label classification. In this paper, we propose to exploit the web genres labels by constructing a hierarchy of web genres and then use methods for hierarchical multi-label classification to boost the predictive performance. We use two methods for hierarchy construction: expert-based and data-driven. The evaluation on a benchmark dataset (`20-Genre collection corpus`) reveals that using a hierarchy of web genres significantly improves the predictive performance of the classifiers and that the data-driven hierarchy yields similar performance as the expert-driven with the added value that it was obtained automatically and fast.

Keywords: Web genre classification · Hierarchy construction · Hierarchical multi-label classification

1 Introduction

There is an increasing need for new ways of searching for desired web pages on the Internet (in April 2015 there were $9.4 \cdot 10^8$ websites – http://www.internetlivestats.com). Typically, searching is performed by typing keywords in a search engine that returns web pages of a topic defined by those keywords. The user can, however, obtain more precise results if web page genre is specified

The first two authors should be regarded as joint first authors.

K. Jackowski et al. (Eds.): IDEAL 2015, LNCS 9375, pp. 9–17, 2015.
DOI: 10.1007/978-3-319-24834-9_2

in addition to the keywords. *Web genre* represents form and function of a web page thus enabling a user to find a "Scientific" paper about the topic of text mining.

A web page is a complex document that can share conventions of several genres or contain parts from different genres. While this is recognized in the web genre classification community, state-of-the-art genre classifier implementations still attribute a single genre to a web page from a set of predefined genre labels (i.e., address the task as multi-class classification). However, a line of research [1–3] advocates that multi-label classification (MLC) scheme is more suitable for capturing the web page complexity. The rationale is that since several genres are easily combined in a single web page, such hybrid forms thus require attribution of multiple genre labels. For example, a story for children will belong to both "Childrens" and "Prose fiction" genres. Furthermore, web genres naturally form a hierarchy of genres. For example, "Prose fiction" is a type of "Fiction". Aforementioned properties of the web genre classification can be easily mapped to the machine learning task of hierarchical multi-label classification (HMC). HMC is a variant of classification, where a single example may belong to multiple classes at the same time and the classes are organized in the form of a hierarchy. An example that belongs to some class c automatically belongs to all super-classes of c. This is called the hierarchical constraint.

Although it can be easily conceived that the task of web genre classification can be mapped to HMC, the hierarchical and multi-label structure of web genres has not yet been explored. There are two major obstacles for this: lack of a comprehensive genre taxonomy with a controlled vocabulary and meaningful relations between genres and web-page-based corpora labelled with such a taxonomy [4]. In addition to these, from a machine learning point of view, methods that are able to fully exploit the complexity of such data started appearing only recently and have not yet gained much visibility (see [5,6]).

In this work, we aim to address these obstacles. First of all, we propose a hierarchy of web genres that is constructed by an expert and propose to use methods for generating hierarchies using the available data. The use of data-driven methods would bypass the complex process of hierarchy construction by experts: it is difficult (if at all possible) to construct a single hierarchy that would be acceptable for all of the experts. Second, we take a benchmark dataset for genre classification (from [1]) and convert it into a HMC dataset. Finally, we investigate the influence of the hierarchy of web genres on the predictive performance of the predictive models.

For accurately measuring the contribution of the hierarchy and reducing the model bias, we need to consider a predictive modelling method that is able to construct models for both MLC (predicting multiple web genres simultaneously without using a hierarchy of genres) and HMC(predicting multiple web genres simultaneously and expoliting a hierarchy of genres). Such methodology is offered with the predictive clustering trees (PCTs) [6]. PCTs can be seen as a generalization of decision trees towards the task of predicting structured outputs, including the tasks of MLC and HMC.

2 Hierarchical Web Genres Data

State-of-the-art web genre classification approaches mostly deal with feature construction and use benchmark 7-Web and KI-04 multi-class corpora to test the feature sets. The two corpora focus on a set of web genres that are at the same level of hierarchy [3] – experiments in [2] indicated that a mix of genres from different levels may significantly deteriorate multi-class classifier's predictive performance. In a MLC setting, typically used corpus is the `20-Genre Collection benchmark corpus` from our previous work [1]. A hierarchical (non multi-label) corpus is presented in [7]: An expert constructed a two-level tree-graph hierarchy composed of 7 top-level and 32 leaf nodes.

In this work, we use the dataset from [1]. It is constructed from 20-Genre Collection corpus and is composed of 2,491 features and 1,539 instances/web pages in English. The features are tailored to cover the different web genre aspects: content (e.g., function words), linguistic form (e.g., part-of-speech trigrams), visual form (e.g., HTML tags) and the context of a web page (e.g., hyperlinks to the same domain). All features, except those pertaining to URL (e.g., appearance of the word blog in a web page URL), are expressed as ratios to eliminate the influence of the page length. The average number of genre labels per page is 1.34. We then converted this dataset to a HMC dataset by expert- and data- driven hierarchy construction methods. We would like to note that the constructed hierarchies are tree-shaped.

Expert-driven hierarchy construction. Expert-based hierarchy was constructed (Fig. 1) by grouping web genres. To this end, we consulted the Web Genre Wiki (http://www.webgenrewiki.org) – it contains results of experts' efforts to construct an unified web genre hierarchy.

Data-driven hierarchy construction. When we build the hierarchy over the label space, there is only one constraint that we should take care of: the original MLC task should be defined by the leaves of the label hierarchy. In particular, the labels from the original MLC problem represent the leaves of the tree hierarchy, while the labels that represent the internal nodes of the tree hierarchy are so-called meta-labels (that model the correlation among the original labels).

In [8], we investigated the use of label hierarchies in multi-label classification, constructed in a data-driven manner. We consider flat label-sets and construct label hierarchies from the label sets that appear in the annotations of the training

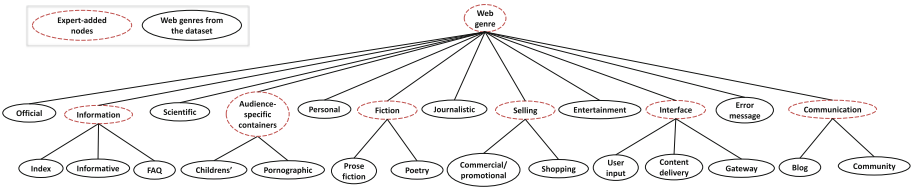

Fig. 1. Web genre hierarchy constructed by an expert.

data by using clustering approaches based on balanced k-means clustering [9], agglomerative clustering with single and complete linkage [10], and clustering performed with PCTs. Multi-branch hierarchy (defined by balanced k-means clustering) appears much more suitable for the global HMC approach (PCTs for HMC) as compared to the binary hierarchies defined by agglomerative clustering with single and complete linkage and PCTs. In this work, for deriving the hierarchy of the (original) MLC problem, we employ balanced k-means.

3 Predictive Modelling for Genre Classification

We present the methodology used to construct predictive models for the task of genre classification using PCTs. We first present general algorithm for constructing PCTs. Next, we otuline the specific PCTs able to predict all of the genres simultaneously but ignore the hierarchical information (i.e., address the task of genre prediction as a multi-label classification task). Furthermore, we give the PCTs able to predict all of the genres simultaneously and exploit the hierarchy information (i.e., address the task of genre prediction as a HMC task).

General algorithm for PCTs. The Predictive Clustering Trees (PCTs) framework views a decision tree as a hierarchy of clusters: the top-node corresponds to one cluster containing all data, which is recursively partitioned into smaller clusters while moving down the tree. The PCT framework is implemented in the CLUS system [6] – available for download at http://clus.sourceforge.net.

PCTs are induced with a standard *top-down induction of decision trees* (TDIDT) algorithm. It takes as input a set of examples and outputs a tree. The heuristic that is used for selecting the tests is the reduction in variance caused by the partitioning of the instances corresponding to the tests. By maximizing the variance reduction, the cluster homogeneity is maximized and the predictive performance is improved. The main difference between the algorithm for learning PCTs and a standard decision tree learner is that the former considers the variance function and the prototype function (that computes a label for each leaf) as *parameters* that can be instantiated for a given learning task. PCTs have been instantiated for both MLC [6,11] and HMC [12]. A detailed computational complexity analysis of PCTs is presented in [6].

PCTs for MLC. These can be considered as PCTs that are able to predict multiple binary (and thus discrete) targets simultaneously. Therefore, the variance function for the PCTs for MLC is computed as the sum of the Gini indices of the target variables, i.e., $Var(E) = \sum_{i=1}^{T} Gini(E, Y_i)$. The prototype function returns a vector of probabilities that an instance belongs to a given class for each target variable. The most probable (majority) class value for each target can then be calculated by applying a threshold on these probabilities.

PCTs for HMC. The variance and prototype for PCTs for the HMC are defined as follows. First, the set of labels of each example is represented as a vector with binary components; the i'th component of the vector is 1 if the example belongs to class c_i and 0 otherwise. The variance of a set of examples E

is defined as the average squared distance between each example's class vector (L_i) and the set's mean class vector ($\overline{L}$): $Var(E) = \frac{1}{|E|} \cdot \sum_{E_i \in E} d(L_i, \overline{L})^2$.

In the HMC context, the similarity at higher levels of the hierarchy is more important than the similarity at lower levels. This is reflected in the distance measure used in the above formula, which is a weighted Euclidean distance: $d(L_1, L_2) = \sqrt{\sum_{l=1}^{|L|} w(c_l) \cdot (L_{1,l} - L_{2,l})^2}$, where $L_{i,l}$ is the l^{th} component of the class vector L_i of an instance E_i, $|L|$ is the size of the class vector, and the class weights $w(c)$ decrease with the depth of the class in the hierarchy. More precisely, $w(c) = w_0 \cdot w(p(c))$, where $p(c)$ denotes the parent of class c and $0 < w_0 < 1$).

In the case of HMC, the mean $\bar{L}$ of the class vectors of the examples in the leaf is stored as a prediction. Note that the value for the i^{th} component of $\bar{L}$ can be interpreted as the probability that an example arriving at the given leaf belongs to class c_i. The prediction for an example that arrives at the leaf can be obtained by applying a user defined threshold τ to the probability. Moreover, when a PCT makes a prediction, it preserves the hierarchy constraint (the predictions comply with the parent-child relationships from the hierarchy).

4 Experimental Design

The comparison of the methods was performed using the CLUS system for predictive clustering implemented in Java. We constructed predictive models corresponding to the two types of modelling tasks, as described in the previous section: multi-label classification (MLC-one model for all of the leaf labels, without using the hierarchy) and hierarchical multi-label classification (HMC-one model for all of the labels by using the hierarchy). For each modeling task, we constructed single tree models.

We used F-test pruning to ensure that the produced models are not overfitted to the training data and have better predictive performance [12]. The exact Fisher test is used to check whether a given split/test in an internal node of the tree results in a statistically significant reduction in variance. If there is no such split/test, the node is converted to a leaf. A significance level is selected from the values 0.125, 0.1, 0.05, 0.01, 0.005 and 0.001 to optimize predictive performance by using internal 3-fold cross validation.

The balanced k-means clustering method that is used for deriving the label hierarchies, requires to be configured the number of clusters k. For this parameter, three different values (2, 3 and 4) were considered [8].

The performance of the predictive models was evaluated using 3-fold cross-validation (as in the study that published the data [1]). We evaluate the predictive performance of the models on the leaf labels in the target hierarchy. In this way, we measure more precisely the influence of the inclusion of the hierarchies in the learning process on the predictive performance of the models.

We used 16 evaluation measures described in detail in [11]. We used six *example-based* evaluation measures (*Hamming loss*, *accuracy*, *precision*, *recall*, F_1 *score* and *subset accuracy*) and six *label-based* evaluation measures (*micro precision*, *micro recall*, *micro* F_1, *macro precision*, *macro recall* and *macro*

F_1). These evaluation measures require predictions stating that a given label is present or not (binary 1/0 predictions). However, most predictive models predict a numerical value for each label and the label is predicted as present if that numerical value exceeds some pre-defined threshold τ. To this end, we applied a threshold calibration method by choosing the threshold that minimizes the difference in label cardinality between the training data and the predictions for the test data. In particular, values from 0 to 1 with step 0.05 for τ were considered.

5 Results and Discussion

In this section, we present the results from the experimental evaluation. The evaluation aims to answer three questions: (1) Which data-driven hierarchy construction method yields hierarchy of genres with best performance? (2) Does constructing a hierarchy improves the predictive performance? and (3) Does constructing a data-driven hierarchy yields satisfactory results when compared with expert-constructed hierarchy?

For answering question (1), we compare the performance of three different hierarchies obtained with varying the value of k in the balanced k-means algorithm. For question (2), we compare the performance of the models that exploit the hierarchy information (HMC) with the performance of the flat classification models (MLC). Finally, for addressing question (3), we compare the performance obtained with the expert hierarchy and the data-driven hierarchy.

Table 1 shows the predictive performance of the compared methods. To begin with, the results for the three hierarchies constructed data-driven methods show that the best hierarchy is the one obtained with k set to 4. This reveals that multi-branch hierarchy is more suitable for this domain. Hence, we select this hierarchy for further comparison. Next, we compare the performance of the hierarchical model with the one of the flat classification model. The results clearly show that using a hierarchy of genre labels significantly improve the performance over using the flat genre labels. Moreover, the improvement in performance is across all of the evaluation measures.

Furthermore, we compare the performance of the models obtained with the expert hierarchy and the data-driven hierarchy. We can see that these models

Table 1. The performance of the different approaches in terms of the label, example and ranking based evaluation measures.

	HammingLoss	*Accuracy*	*Precision*	*Recall*	*Fmeasure*	*SubsetAccuracy*	*MicroPrecision*	*MicroRecall*	*MicroF1*	*MacroPrecision*	*MacroRecall*	*MacroF1*	*OneError*	*Coverage*	*RankingLoss*	*AvgPrecision*
HMC - manual hiear.	0.094	**0.276**	**0.327**	**0.341**	**0.334**	0.172	0.31	**0.33**	0.32	**0.424**	**0.296**	**0.297**	0.643	5.561	0.238	0.47
HMC - BkM (k=4)	**0.081**	0.261	0.31	0.3	0.305	**0.177**	**0.368**	0.291	**0.325**	0.368	0.262	0.284	**0.635**	**5.435**	**0.232**	**0.475**
HMC - BkM (k=3)	0.09	0.223	0.273	0.272	0.273	0.131	0.301	0.263	0.281	0.328	0.212	0.211	0.677	5.878	0.254	0.44
HMC - BkM (k=2)	0.084	0.206	0.247	0.247	0.247	0.127	0.328	0.24	0.277	0.361	0.205	0.227	0.682	5.956	0.259	0.433
MLC	0.111	0.136	0.172	0.165	0.168	0.073	0.165	0.163	0.164	0.063	0.1	0.065	0.83	7.955	0.36	0.317

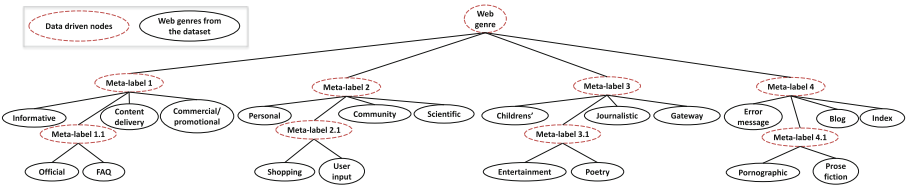

Fig. 2. Web genre hierarchy constructed by balanced k-means algorithm (for k = 4).

have relatively similar predictive performance – each of the models is better than the other according to 8 evaluation measures. It is worth mentioning that the data-driven hierarchy is better on the ranking-based evaluation measures (the last four columns in Table 1). This means that by improving the threshold selection procedure the other evaluation measures will further improve. Nevertheless, even with the results as they are, they convey an important message: The tedious, laborious and expensive method of hierarchy construction by experts can be replaced with a cheap, automatic, fast, data-driven hierarchy construction method without any loss in terms of predictive performance.

The data-driven hierarchy obtained with balanced k-means (and k set to 4) is depicted in Fig. 2. An inspection of the two hierarchies (the first constructed by an expert, Fig. 1, and the second constructed using only data) reveals that these two hierarchies differ to each other completely. Namely, there is no grouping of genres in the expert hierarchy that can be noted in the data-driven hierarchy. This means that there exist a semantic gap between the meaning of the genres and how these meaning are well represented in the data.

Considering that the PCTs are interpretable models, we briefly comment on the attributes selected on the top levels of the trees constructed with the different scenarios: MLC, HMC-manual and HMC-BkM. The MLC and HMC-BkM tree selected first information on the appearance of the word FAQ in the url of the web page and then focus on content related attributes. The HMC-BkM tree also uses the part-of-speech trigrams. Conversely, the HMC-manual tree used mainly content related features on the top levels of the tree-model accompanied with HTML tags information on the lower levels. All in all, the different scenarios exploit different attributes from the dataset.

6 Conclusions

In this paper, we advocated a new approach for resolving the task of web genres classification. Traditionally, this task is treated as a multi-class problem, while there are some recent studies that advise to treat it as a MLC problem. We propose to further exploit the information that is present in the web genres labels by constructing a hierarchy of web genres and then use methods for HMC to boost the predictive performance. Considering that hierarchical benchmark datasets for web genre classification do not exist, we propose to use data-driven methods for hierarchy construction based on balanced k-means. To investigate

whether there is a potential in this, we compare the obtained the data-driven hierarchy with a hierarchy based on expert knowledge.

In the evaluation, we consider a benchmark dataset with 1539 web pages with 20 web genres. The results reveal that using a hierarchy of web genres significantly improves the predictive performance of the classifiers. Furthermore, the data-driven hierarchy yields similar performance as the expert-driven with the difference that it was obtained automatically and fast. This means for even larger domains (both in terms of number of examples and number of web genre labels) it would be much simpler and cheaper to use data-driven hierarchies.

We plan to extend this work in two major directions. First, we plan to use more advanced predictive models such as ensembles for predicting structured outputs to see whether the improvement carries over in the ensemble setting. Second, we plan to develop hierarchies of web genres structured as directed acyclic graphs, which seems more natural in modelling relations between genres. It could also be useful to adapt the hierarchy construction algorithm to break down existing genres into sub-genres.

Acknowledgments. We acknowledge the financial support of the European Commission through the grant ICT-2013-612944 MAESTRA.

References

1. Vidulin, V., Luštrek, M., Gams, M.: Multi-label approaches to web genre identification. J. Lang. Tech. Comput. Linguist. **24**(1), 97–114 (2009)
2. Santini, M.: Automatic identification of genre in web pages. Ph.D. thesis, University of Brighton (2007)
3. Santini, M.: Cross-testing a genre classification model for the web. In: Mehler, A., Sharoff, S., Santini, M. (eds.) Genres on the Web, pp. 87–128. Springer, Heidelberg (2011)
4. Crowston, K., Kwaśnik, B., Rubleske, J.: Problems in the use-centered development of a taxonomy of web genres. In: Mehler, A., Sharoff, S., Santini, M. (eds.) Genres on the Web, pp. 69–84. Springer, Heidelberg (2011)
5. Silla Jr., C.N., Freitas, A.A.: A survey of hierarchical classification across different application domains. Data Min. Knowl. Discov. **22**(1–2), 31–72 (2011)
6. Kocev, D., Vens, C., Struyf, J., Džeroski, S.: Tree ensembles for predicting structured outputs. Pattern Recogn. **46**(3), 817–833 (2013)
7. Stubbe, A., Ringlstetter, C., Schulz, K.U.: Genre as noise: noise in genre. Int. J. Doc. Anal. Recogn. **10**(3–4), 199–209 (2007)
8. Madjarov, G., Dimitrovski, I., Gjorgjevikj, D., Džeroski, S.: Evaluation of different data-derived label hierarchies in multi-label classification. In: Appice, A., Ceci, M., Loglisci, C., Manco, G., Masciari, E., Ras, Z.W. (eds.) NFMCP 2014. LNCS, vol. 8983, pp. 19–37. Springer, Heidelberg (2015)
9. Tsoumakas, G., Katakis, I., Vlahavas, I.: Effective and efficient multilabel classification in domains with large number of labels. In: Proceedings of the ECML/PKDD Workshop on Mining Multidimensional Data, pp. 30–44 (2008)
10. Manning, C.D., Raghavan, P., Schütze, H.: An Introduction to Information Retrieval. Cambridge University Press, Cambridge (2009)

11. Madjarov, G., Kocev, D., Gjorgjevikj, D., Džeroski, S.: An extensive experimental comparison of methods for multi-label learning. Pattern Recogn. **45**(9), 3084–3104 (2012)
12. Vens, C., Struyf, J., Schietgat, L., Džeroski, S., Blockeel, H.: Decision trees for hierarchical multi-label classification. Mach. Learn. **73**(2), 185–214 (2008)