Semi-supervised Graph-based Genre Classification for Web Pages

Noushin Rezapour Asheghi
School of Computing
University of Leeds
scs5nra@leeds.ac.uk

Katja Markert
L3S Research Center
Leibniz Universität Hannover
and School of Computing
University of Leeds
markert@l3s.de

Serge Sharoff
School of Modern
Languages and Cultures
University of Leeds
s.sharoff@leeds.ac.uk

Abstract

Until now, it is still unclear which set of features produces the best result in automatic genre classification on the web. Therefore, in the first set of experiments, we compared a wide range of content-based features which are extracted from the data appearing within the web pages. The results show that lexical features such as word unigrams and character n-grams have more discriminative power in genre classification compared to features such as part-of-speech n-grams and text statistics. In a second set of experiments, with the aim of learning from the neighbouring web pages, we investigated the performance of a semi-supervised graph-based model, which is a novel technique in genre classification. The results show that our semi-supervised min-cut algorithm improves the overall genre classification accuracy. However, it seems that some genre classes benefit more from this graph-based model than others.

1 Introduction

In Automatic Genre Identification (AGI), documents are classified based on their genres rather than their topics or subjects. Genre classes such as editorial, interview, news and blog which are recognizable by their distinct purposes, can be on any topic. The most important application of AGI could be in Information Retrieval. If a user could use the search engine to retrieve web pages from a specific genre such as news articles, reviews or blogs, search results could be more beneficial. With the aim of enhancing search engines, AGI has attracted a lot of attention (see Section 2).

In this paper, we investigate two important open questions in AGI. The first question is: what set of features produces the best result in genre classification on the web? The drawbacks of existing genre-annotated web corpora (low inter-coder agreement; false correlations between topic and genre classes) resulted in researchers' doubt on the outcomes of classification models based on these corpora (Sharoff et al., 2010). Therefore, in order to answer this question, we perform genre classification with a wide range of features on a reliable and source diverse genre-annotated web corpus. The second question that we investigate in this paper is: could we exploit the graph structure of the web to increase genre classification accuracy? With the aim of learning from the neighbouring web pages, we investigated the performance of a semi-supervised graph-based model, which is a novel technique in genre classification.

The remainder of this paper is structured as follows. After reviewing related work in Section 2, we compare different supervised genre classification models based on various lexical, POS-based and text statistics features in Section 3. Section 4 describes our semi-supervised graph-based classification experiment, where we use the multi-class min-cut algorithm as a novel technique in genre classification. Section 5 concludes the findings and discusses future work.

2 Related Work

There has been a considerable body of research in AGI. In previous studies on automatic genre classification of web pages, various types of features such as common words (Stamatatos et al., 2000), function words (Argamon et al., 1998), word unigrams (Freund et al., 2006), character n-grams (Kanaris and Stamatatos, 2007), part-of-speech tags (Karlgren and Cutting, 1994) , part-of-speech trigrams (Argamon et al., 1998; Santini, 2007), document statistics (e.g. average sentence length, average word length and type/token ratio) (Finn and Kushmerick, 2006; Kessler et

Proceedings of TextGraphs-9: the workshop on Graph-based Methods for Natural Language Processing, pages 39–47,
October 29, 2014, Doha, Qatar.

al., 1997), HTML tags (e.g. (Santini, 2007)) have been explored. However, researchers conducted genre classification experiments with different features on different corpora with different sets of genre labels. As a result, it is difficult to compare them. This motivated Sharoff et al. (2010) to examine a wide range of word-based, character-based and POS-based features on the existing genre-annotated corpora. They reported that word unigrams and character 4-grams outperform other features in genre classification. However, they concluded that the results cannot be trusted because of two main reasons. First, some of these collections exhibit low inter-coder agreement and any results based on unreliable data could be misleading. Second, the spurious correlation between topic and genre classes in some of these corpora was one of the reasons for some of the very impressive results reported by Sharoff et al. (2010). These good results were achieved by detecting topics rather than genres of individual texts. A similar point was made by Petrenz and Webber (2010) who examined the impact of topic change on the performance of AGI systems. They showed that a shift in topic can have a massive impact on genre classification models which are based on lexical features such as word unigrams or character n-grams. Therefore, the question which set of features produces the best result in automatic genre classification on the web is still an open question. In order to investigate this question, we perform genre classification with a wide range of features on a reliable and topically diverse dataset. Section 3.1 describes the dataset and the experimental setup.

Most of the current works in the field of AGI concentrated on extracting features from the content of the documents and classify them by employing a standard supervised algorithm. However, on the web there are other sources of information which can be utilized to improve genre classification of web pages. For instance, the web has a graph structure and web pages are connected via hyper-links. These connections could be exploited to improve genre classification. Various graph-based classification algorithms have been proposed to improve topic classification for web pages, such as the relaxation labelling algorithm (Chakrabarti et al., 1998), iterative classification algorithm (Lu and Getoor, 2003), Markov logic networks (Crane and McDowell, 2012), random graph walk (Lin and Cohen, 2010) and weighted-vote relational neighbour algorithm (Macskassy and Provost, 2007). These classification algorithms which utilize hyper-link connections between web pages to construct graphs, outperformed the classifiers which are solely based on textual content of the web pages for topic classification. Such connected data presents opportunities for boosting the performance of genre classification too.

Graph-based web page classification presented in studies such as (Crane and McDowell, 2012; Lu and Getoor, 2003; Macskassy and Provost, 2007) on the WebKB dataset (CRAVEN, 1998) could be considered as genre classification as opposed to topic classification. The WebKB dataset contains web pages from four computer science departments categorised into seven classes: student, faculty, staff, department, course, project and other. However, this dataset is very specific to the academic domain with low coverage for the web overall, whereas we examine graph-based learning for automatic genre classification of web pages on a much more general dataset with popular genre classes such as news, blog and editorial. Moreover, the graph-based algorithms used on the WebKB dataset are all supervised and were performed on a very clean and noise free dataset which was achieved by removing the class other. Class other contains all the web pages which do not belong to any other predefined classes. However, our experiment is in a semi-supervised manner which is a much more realistic scenario on the web, because it is highly unlikely that for each web page, we have genre labels for all its neighbouring web pages as well. Therefore, we perform our experiment on a very noisy dataset where neighbouring web pages could belong to none of our predefined genre classes. Section 4 describes our semi-supervised graph-based classification experiment, where we use a multi-class min-cut algorithm as a novel technique in genre classification.

3 Content-based Classification

3.1 Dataset and Experimental Setup

Petrenz and Webber (2010) and Sharoff et al. (2010) emphasize that the impact of topic on genre classification should be eliminated or controlled. In order to avoid the influence of topic on genre classification, some researchers (e.g. (Sta-

Genre	Number of		# of pages from the same website			Fleiss's κ
	web pages	websites	max	min	med	
Personal Homepage (php)	304	288	9	1	1	0.858
Company/ Business Homepage (com)	264	264	1	1	1	0.713
Educational Organization Homepage (edu)	299	299	1	1	1	0.953
Personal Blog /Diary (blog)	244	215	9	1	1	0.812
Online Shop (shop)	292	209	23	1	1	0.830
Instruction/ How to (instruction)	231	142	15	1	1	0.871
Recipe	332	116	8	1	1	0.971
News	330	127	12	1	1	0.801
Editorial	310	69	11	1	3	0.877
Conversational Forum (forum)	280	106	11	1	1	0.951
Biography (bio)	242	190	15	1	1	0.905
Frequently Asked Questions (faq)	201	140	8	1	1	0.915
Review	266	179	15	1	1	0.880
Story	184	24	38	1	7	0.953
Interview	185	154	11	1	1	0.905

Table 1: Statistics for each category illustrate source diversity and reliability of the corpus (Asheghi et al., 2014). To save space, in this paper we use the abbreviation of genre labels which are specified after the genre names.

matatos et al., 2000) and (Argamon et al., 1998)) use only topic independent features such as common words or function words in genre classification. However, neither of these features are exclusive to genre classification. Function words and common words are used in authorship classification (e.g. (Argamon et al., 2007)) because they can capture the style of the authors without being influenced by the topics of the texts. On the other hand, word unigrams are a popular document representation in topic classification. If we want these models to capture the genre of documents without being influenced by their topics or the style of their authors, we must eliminate the influence of these factors on genre classification by keeping them constant across the genre classes in the training data. That means all the documents in the training set should be about the same topic and written by the same person. However, constructing such a dataset is practically impossible for genre classes on the web. The other more practical solution to this problem would be to collect data from various topics and sources in order to minimize the impact of these factors on genre classification. For that reason, we (Asheghi et al., 2014) created a web genre annotated corpus which is reliable (with Fleiss's kappa (Fleiss, 1971) equal to 0.874) and source diverse. We tried to reduce the influence of topic, the writing style of the authors as well as the design of the websites on genre classification by collecting data from various sources and topics. The corpus consists of 3964 web pages from 2522 different websites, distributed across 15 genres (Table 1).

Moreover, we prepared two versions of the corpus: the original text and the main text corpora. First, we converted web pages to plain text by removing HTML markup using the KrdWrd tool (Steger and Stemle, 2009). This resulted in the original text corpus which contains individual web pages with all the textual elements present on them. Moreover, in order to investigate the influence of boilerplate parts (e.g. advertisements, headers, footers, template materials, navigation menus and lists of links) of the web pages on genre classification, we removed the boilerplate parts and extracted the main text of each web page using the justext tool [1]. This resulted in the creation of the main text corpus. This is the first time that the performance of genre classification models is compared on both the original and the main text of the web pages.

Since the outputs of the justext tool for 518 of the web pages were empty files, the main text corpus has fewer pages. However, the main text corpus still has a balanced distribution with a relatively large number of web pages per category. Table 2 compares the number of web pages in the two versions of the corpus. For all the experiments we use this corpus via 10-fold cross-validation on the web pages. Also, in order to minimize the effect of factors such as topic, the writing style of the authors and the design of the websites even further, we ensured that all the web pages from the same website are in the same fold. Many, if not all of the previous studies in automatic genre classification on the web ignored this essential step when dividing the data into folds. For machine learning, we

[1] http://code.google.com/p/justext/

Genre	Number of web pages in corpora	
	Original text	Main text
php	304	221
com	264	190
edu	299	191
blog	244	242
shop	292	221
instruction	231	229
recipe	332	243
news	330	320
editorial	310	307
forum	280	251
bio	242	242
faq	201	160
review	266	262
story	184	184
interview	185	183

Table 2: Number of web pages in individual genre classes in both original text and main text corpora.

chose Support Vector Machines (SVM) because it has been shown by other researchers in AGI (e.g. (Santini, 2007)) that SVM produces better or at least similar results compared to other machine learning algorithms. We used the one-versus-one multi-class SVM implemented in Weka [2] with the default setting. All the experiments are carried out on both the original text and the main text corpora.

3.2 Features

In order to compare the performance of different lexical and structural features used in previous work, we reimplemented the following published approaches to AGI: function words (Argamon et al., 1998), part-of-speech n-grams (Santini, 2007), word unigrams (Freund et al., 2006) and character 4-grams binary representation (Sharoff et al., 2010). We also explored the discriminative power of other features such as readability features (Pitler and Nenkova, 2008), HTML tags [3] and named entity tags in genre classification (Table 3). This is the first time that some of these features such as average depth of syntax trees and entity coherence features (Barzilay and Lapata, 2008) are used for genre classification. To set a base-line, we used a list of genre names (e.g. news, editorial, interview, review) as features. We used two different feature representations: binary and normalized frequency. In the binary representation of a document, the value for each feature is either one or zero which represents the presence or the absence of each feature respectively. In the normalized frequency representation of a document, the value for each feature is the frequency of that feature which is normalized by the length of the document.

For extracting lexical features, we tokenized each document using the Stanford tokenizer (included as part of the Stanford part of speech tagger (Toutanova et al., 2003)) and converted all the tokens to lower case. For extracting POS tags and named entity tags, we used the Stanford maximum entropy tagger [4] and the Stanford Named Entity Recognizer [5] respectively. For extracting some of the readability features such as average parse tree height and average number of noun and verb phrases per sentences, we used the Stanford Parser (Klein and Manning, 2003). However, web pages must be cleaned before they can be fed to a parser, because parsers cannot handle tables and list of links. Therefore, we only used the main text of each web page as an input to the parser. For web pages for which the justext tool produced empty files, we treated these features as missing values. Moreover, we used the Brown Coherence Toolkit [6] to construct the entity grid for each web page and computed the probability of each entity transition type. This tool needs the parsed version of the text as an input. Therefore, for web pages for which the justext tool produced empty files, we also treated these features as missing values.

3.3 Results and Discussion

Table 4 shows the result of the different feature sets listed in the previous section on both the original text and the main text corpora. At first glance, we see that the results of genre classification on the original text corpus are higher than the main text corpus. This shows that boiler plates contain valuable information which helps genre classification.

Moreover, the results show that binary representation of word unigrams is the best performing feature set when we use the whole text of the web pages. However, on the main text corpus, character 4-grams outperform other features. This confirms the results reported in (Sharoff et al., 2010). The results also highlight that the performance of POS-based features are much less accurate than that of textual features such as word unigrams and character n-grams. The results also show that the combination of word unigrams, text statistics and

[2] http://www.cs.waikato.ac.nz/ml/weka/

[3] http://www.w3schools.com/tags/ref_byfunc.asp

[4] http://nlp.stanford.edu/software/tagger.shtml

[5] http://nlp.stanford.edu/software/CRF-NER.shtml

[6] http://www.cs.brown.edu/ melsner/manual.html

Category	Features
Token features	number of tokens and number of types normalized frequency of punctuation marks and currency characters
Named entity tags	normalized frequency of tags: time, location, organization, person, money, date
Readability features	average parse tree height average sentence length and word length standard deviation of sentence length and of word length average number of syllables per word type/token ratio average number of noun phrases and verb phrases per sentence entity coherence features (Barzilay and Lapata, 2008)
HTML tags	normalized frequency of tags for: sections / style, formatting, programming, visual features such as forms, images, lists and tables

Table 3: List of text statistics features explored in this paper

part of speech features resulted in improving genre classification accuracy (compared to the accuracy achieved by word unigrams alone), for both original and main text corpora. However, while the improvement for the main text corpus is statistically significant [7], there is no significant difference between these two models for the original corpus. Surprisingly, adding part of speech 3-grams to the word unigrams features decreased the genre classification accuracy in both original and main text corpora. The reason could be that the model is over-fitted on the training data and as a result, it performs poorly on the test data. Therefore, combining various features will not always improve the performance of the classification task. Moreover, for extracting POS-based features and some of the text statistics features we rely on tools such as part-of-speech taggers and parsers whose performance varies for different genres. Even the best part-of-speech taggers and parsers are error prone and cannot be trusted on new unseen genres.

4 Graph-based Classification

Until now we extracted features only from the content of the web pages. However, other sources of information such as the connections and the link patterns between the web pages could be exploited to improve genre classification. The underlying assumption of this approach is that a page is more likely to be connected to pages with the same genre category. For example, if the neighbouring web pages of a particular web page are labelled as shop, it is more likely that this web page is a shop too, whereas, it is highly unlikely that it is a news or editorial. This property (i.e. entities with similar labels are more likely to be connected) is known as homophily (Sen et al., 2008). We hypothesis that homophily exists for genre classes and it can help us to improve genre classification on the web. In this paper, we use a semi-supervised graph-based algorithm namely, multi-class min-cut, which is a novel approach in genre classification. This algorithm, which is a collective classification method, considers the class labels of all the web pages within a graph.

4.1 Multi-class Min-cut: The Main Idea

The Min-cut classification algorithm originally proposed by Blum and Chawla (2001) is based on the idea that linked entities have a tendency to belong to the same class. In other words, it is based on the homophily assumption. Therefore, it should be able to improve genre classification on the web if our hypothesis holds. However, this technique is a binary classification algorithm, whereas, we have a multi-class problem. Unfortunately, multi-class min-cut is NP-hard and there is no exact solution for it. Nevertheless, Ganchev and Pereira (2007) proposed a multi-class extension to Blum and Chawla (2001)'s min-cut algorithm by encoding a multi-class min-cut problem as an instance of metric labelling. Kleinberg and Tardos (1999; 2002) introduced metric labelling for the first time. The main idea of metric labelling for web page classification can be described as follows:

Assume we have a weighted and undirected graph $G = (V, E)$ where each vertex $v \in V$ is a web page and the edges represent the hyper-links between the web pages. The task is to classify these web pages into a set of labels L. This task can be denoted as a function $f : V \rightarrow L$. In order to do this labelling task in an optimal way, we need to minimize two different types of costs. First, there is a non-negative cost $c(v, l)$ for assigning label l

[7] McNemar test at the significance level of 5%

Feature set	Original text	Main text
genre names bin	57.39	29.02
genre name nf	38.29	14.16
function words bin	65.71	55.57
function words nf	74.95	66.86
word unigrams bin	**89.32**	76.61
word unigrams nf	85.21	74.91
character 4-grams bin	87.96	**78.88**
POS-3grams bin	73.18	61.23
POS-3grams nf	70.28	57.83
POS-2grams bin	64.10	54.91
POS-2grams nf	68.94	60.76
POS nf	60.14	54.64
text statistics	55.47	59.17
word unigrams bin + text statistics	**89.48**	78.09
word uni-grams bin + text statistics + POS nf	**89.63**	78.24
word uni-grams bin + POS 3-grams bin	88.14	75.59

Table 4: Classification accuracy of different features in genre classification. *bin* and *nf* refer to the use of binary and normalized frequency representation of the features respectively.

to web page v. Second, if two web pages v_1 and v_2 are connected together with an edge e with weight w_e, we need to pay a cost of $w_e \cdot d(f(v_1), f(v_2))$ where $d(.,.)$ denotes distance between the two labels. A big distance value between labels indicates less similarity between them. Therefore, the total cost of labelling task f is:

$$E(f) = \sum_{v \in V} c(v, f(v)) + \sum_{e=(v_1,v_2) \in E} w_e \cdot d(f(v_1), f(v_2)) \quad (1)$$

Kleinberg and Tardos (1999; 2002) developed an algorithm for minimizing $E(f)$. However, their algorithm uses linear programming which is impractical for large data (Boykov et al., 2001). In a separate study for metric labelling problems, Boykov et al. (2001) have developed a multi-way min-cut algorithm to minimize $E(f)$. This algorithm is very fast and can be applied to large-scale problems with good performance (Boykov et al., 2001).

4.2 Selection of unlabelled data

A web page w has different kind of neighbours on the web such as parents, children, siblings, grand parents and grand children which are mainly differentiated based on the distance to the target web page as well as the direction of the links (Qi and Davison, 2009). Since the identification of children of a web page (i.e. web pages which have direct links from the target web page) is a straightforward task as their URLs can be extracted from the HTML version of the target web page, in this study, we explore the effect of the target web pages' children on genre classification. Therefore, in this experiment, by neighbouring web pages we mean the web pages' children. In order to collect the neighbouring web pages, for every web page in the data set, we extracted all its out-going URLs and downloaded them as unlabelled data. However, using all these neighbouring pages could hurt the genre classification accuracy because web pages are noisy (e.g. links to advertisements) and some neighbours could have different genres than the target page. In order to control the negative impact of such neighbours, we could preselect a subset of neighbours whose content are close enough to the target page. To implement this idea, we

Cosine similarity	# of unlabelled web pages	Average # of neighbours
≥ 0	103,372	40.65
≥ 0.1	98,824	39.08
≥ 0.2	87,834	34.23
≥ 0.3	70,602	26.46
≥ 0.4	50,232	17.52
≥ 0.5	28,437	8.62
≥ 0.6	13,919	3.77
≥ 0.7	7,241	1.86
$\geq$ **0.8**	**3,772**	**0.98**
≥ 0.9	1,732	0.44

Table 5: Number of unlabelled web pages with different cosine similarity thresholds. The last column shows the average number of neighbours per labelled page.

computed the cosine similarity between the web page w and its neighbouring web pages and used different threshold to select the neighbours. If u is a neighbour of w and $\overrightarrow{u}$ and $\overrightarrow{w}$ are the representative feature vectors of these two web pages respectively, we could compute the cosine similarity between these two web pages using the following formula:

$$\cos(\overrightarrow{w}, \overrightarrow{u}) = \frac{\overrightarrow{w} \cdot \overrightarrow{u}}{\| \overrightarrow{w} \| \| \overrightarrow{u} \|} = \frac{\sum_{i=1}^{n} w_i \times u_i}{\sqrt{\sum_{i=1}^{n}(w_i)^2} \times \sqrt{\sum_{i=1}^{n}(u_i)^2}} \quad (2)$$

where n is the number of the dimensions of the vectors and w_i is the value of the ith dimension of the vector $\overrightarrow{w}$. Since the word unigrams binary representation model yields the best result for content-based genre classification, we used this representation of web pages to construct their feature vectors. Table 5 shows the number of unlabelled data and the average number of neighbours per labelled web page for different cosine similarity thresholds.

4.3 Formulation of Semi-supervised Multi-class Min-cuts

The formulation of semi-supervised multi-class min-cut for genre classification involves the following steps:

1. We built the weighted and undirected graph $G = (V, E)$ where vertices are the web pages (labelled and unlabelled) and the edges represent the hyper-links between the web pages and set the weights to 1.

2. For training nodes, set the cost of the correct label to zero and all other labels to a large constant.

3. For test nodes and unlabelled nodes, we set the cost of each label using a supervised classifier (SVM) using the following formula:

$$c(w, l) = 1 - p_l(w) \quad (3)$$

   where $c(w, l)$ is the cost of label l for web page w and $p_l(w)$ is the probability of w belonging to the label l which is computed by a supervised SVM using word unigrams binary representation of the web pages.

4. Set $d(i, j)$, which denotes the distance between two labels i and j, to 1 if $i \neq j$ and zero otherwise.

5. Employ Boykov et al. (2001) algorithm to find the minimum total cost using multiway min-cut algorithm.

4.4 Results and Discussion

We divided the labelled data into 10 folds again ensuring that all the web pages from the same websites are in the same fold. We used 8 folds for training, one fold for validation and one fold for testing. We learnt the best cosine similarity threshold using validation data and then evaluated it on the test data. Tables 6 and 7 illustrate the results of the multi-class min-cut algorithm and the content-based algorithm (both using word unigrams as features) respectively. The results show that the multi-class min-cut algorithm significantly outperforms [8] the content-based classifier for the cosine similarity equal or greater than 0.8 which was chosen on the validation data. It must be noted that the result of the multi-class min-cut algorithm when we used all the neighbouring pages was much lower than the content-based algorithm due to noise. The results also shows that some genre classes such as news, editorial, blog, interview and instruction benefited more than other genre classes from the neighbouring web pages. Genre categories with improved results are shown in bold in Table 6. The homophily property of these genre categories was the reason behind this improvement. For example, the fact that a news article is more likely to be linked to other news articles, whereas, an editorial is more likely to be linked to other editorials, helped us to differentiate these two categories further. On the other hand, we observe no improvement or even decrease in F-measure for some genre categories such as frequently asked questions, forums and company home pages. Two reasons could have contributed to these results. First, the homophily property might not exist for these categories. Second, the homophily property holds for these categories, however, in order to benefit from this property, we need to examine other neighbours of the target web pages such as parents, siblings, grand parents, grand children or even more distant neigh-

[8] McNemar test at the significance level of 5%

class	Recall	Precision	F1-measure
php	**0.928**	**0.850**	**0.887**
forum	0.925	0.977	0.951
review	0.895	0.832	0.862
news	**0.897**	**0.798**	**0.845**
com	0.897	0.891	0.894
shop	**0.860**	**0.965**	**0.910**
instruction	**0.870**	**0.914**	**0.892**
recipe	**0.994**	**0.991**	**0.993**
blog	**0.889**	**0.879**	**0.884**
bio	**0.905**	**0.948**	**0.926**
editorial	**0.800**	**0.932**	**0.861**
faq	0.902	0.841	0.870
edu	**0.957**	**0.963**	**0.960**
story	**0.902**	**0.943**	**0.922**
interview	**0.870**	**0.809**	**0.839**
overall accuracy = **90.11%**			

Table 6: Recall, Precision and F-measure for multi-class min-cut genre classification.

class	Recall	Precision	F1-measure
php	0.938	0.798	0.862
forum	0.943	0.974	0.958
review	0.872	0.859	0.866
news	0.894	0.782	0.835
com	0.920	0.874	0.897
shop	0.849	0.950	0.897
instruction	0.866	0.889	0.877
recipe	0.988	0.988	0.988
blog	0.865	0.841	0.853
bio	0.884	0.926	0.905
editorial	0.765	0.926	0.837
faq	0.866	0.879	0.872
edu	0.950	0.969	0.959
story	0.864	0.941	0.901
interview	0.827	0.785	0.805
overall accuracy = 88.98% [9]			

Table 7: Recall, Precision and F-measure for content-based genre classification using word unigrams feature set

bours.

5 Conclusions and Future work

In the first set of experiments, we compared a diverse range of content-based features in genre classification using a reliable and source diverse genre-annotated corpus. The evaluation shows that lexical features outperformed all other features. Source diversity of the corpus minimized the influence of topic, authorship and web page design on genre classification. In the second experiment, we significantly improved the genre classification result using a semi-supervised min-cut algorithm by employing the children of the target web pages as unlabelled data. The results of this method which takes advantage of the graph structure of the web shows that some genre classes benefit more than others from the neighbouring web pages. The homophily property of genre categories such as news, blogs and editorial was the reason behind the improvement of genre classification in this experiment. In future work, we would like to examine the effect of other types of neighbours on genre classification of web pages and experiment with other graph-based algorithms.

References

Shlomo Argamon, Moshe Koppel, and Galit Avneri. 1998. Routing documents according to style. In *First international workshop on innovative information systems*, pages 85–92. Citeseer.

Shlomo Argamon, Casey Whitelaw, Paul Chase, Sobhan Raj Hota, Navendu Garg, and Shlomo Levitan. 2007. Stylistic text classification using functional lexical features. *Journal of the American Society for Information Science and Technology*, 58(6):802–822.

Noushin Rezapour Asheghi, Serge Sharoff, and Katja Markert. 2014. Designing and evaluating a reliable corpus of web genres via crowd-sourcing. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*.

Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Avrim Blum and Shuchi Chawla. 2001. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 19–26. Morgan Kaufmann Publishers Inc.

Yuri Boykov, Olga Veksler, and Ramin Zabih. 2001. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239.

Soumen Chakrabarti, Byron Dom, and Piotr Indyk. 1998. Enhanced hypertext categorization using hyperlinks. In *ACM SIGMOD Record*, volume 27, pages 307–318. ACM.

Robert Crane and Luke McDowell. 2012. Investigating markov logic networks for collective classification. In *ICAART (1)*, pages 5–15.

M CRAVEN. 1998. Learning to extract symbolic knowledge from the world wide web. In *Proc. of the 15th National Conference on Artificial Intelligence (AAAI-98)*.

[9] Please note that in this experiment we had less training data because we used 8 folds for training, one fold for validation and one fold for testing. As a result, the accuracy of word unigrams is slightly lower than the result reported in Table 4.

Aidan Finn and Nicholas Kushmerick. 2006. Learning to classify documents according to genre. *Journal of the American Society for Information Science and Technology*, 57(11):1506–1518.

J.L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378.

L. Freund, C.L.A. Clarke, and E.G. Toms. 2006. Towards genre classification for ir in the workplace. In *Proceedings of the 1st international conference on Information interaction in context*, pages 30–36. ACM.

Kuzman Ganchev and Fernando Pereira. 2007. Transductive structured classification through constrained min-cuts. *TextGraphs-2: Graph-Based Algorithms for Natural Language Processing*, page 37.

Ioannis Kanaris and Efstathios Stamatatos. 2007. Webpage genre identification using variable-length character n-grams. In *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, volume 2, pages 3–10. IEEE.

J. Karlgren and D. Cutting. 1994. Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of the 15th conference on Computational linguistics-Volume 2*, pages 1071–1075.

B. Kessler, G. Numberg, and H. Schutze. 1997. Automatic detection of text genre. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 32–38.

Dan Klein and Christopher D Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics.

Jon Kleinberg and Eva Tardos. 1999. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. In *focs*, page 14. Published by the IEEE Computer Society.

Jon Kleinberg and Eva Tardos. 2002. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. *Journal of the ACM (JACM)*, 49(5):616–639.

Frank Lin and William W Cohen. 2010. Semi-supervised classification of network data using very few labels. In *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*, pages 192–199. IEEE.

Q. Lu and L. Getoor. 2003. Link-based classification using labeled and unlabeled data. *The Continuum from Labeled to Unlabeled Data in Machine Learning & Data Mining*, page 88.

Sofus A Macskassy and Foster Provost. 2007. Classification in networked data: A toolkit and a univariate case study. *The Journal of Machine Learning Research*, 8:935–983.

P. Petrenz and B. Webber. 2010. Stable classification of text genres. *Computational Linguistics*, (Early Access):1–9.

Emily Pitler and Ani Nenkova. 2008. Revisiting readability: A unified framework for predicting text quality. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 186–195.

Xiaoguang Qi and Brian D Davison. 2009. Web page classification: Features and algorithms. *ACM Computing Surveys (CSUR)*, 41(2):12.

Marina Santini. 2007. *Automatic identification of genre in web pages*. Ph.D. thesis, University of Brighton.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. 2008. Collective classification in network data. *AI magazine*, 29(3):93.

S. Sharoff, Z. Wu, and K. Markert. 2010. The web library of babel: evaluating genre collections. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation*, pages 3063–3070.

Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. 2000. Text genre detection using common word frequencies. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*, pages 808–814.

Johannes M. Steger and Egon W. Stemle. 2009. KrdWrd – architecture for unified processing of web content.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180.