Gender Classification of Web Authors Using Feature Selection and Language Models

Christina Aravantinou(✉), Vasiliki Simaki, Iosif Mporas, and Vasileios Megalooikonomou

Multidimensional Data Analysis and Knowledge Management Laboratory,
Department of Computer Engineering and Informatics, University of Patras,
26500 Rion, Greece
{aravantino,simaki,vasilis}@ceid.upatras.gr,
imporas@upatras.gr

Abstract. In the present article, we address the problem of automatic gender classification of web blog authors. More specifically, we employ eight widely used machine learning algorithms, in order to study the effectiveness of feature selection on improving the accuracy of gender classification. The feature ranking is performed over a set of statistical, part-of-speech tagging and language model features. In the experiments, we employed classification models based on decision trees, support vector machines and lazy-learning algorithms. The experimental evaluation performed on blog author gender classification data demonstrated the importance of language model features for this task and that feature selection significantly improves the accuracy of gender classification, regardless of the type of the machine learning algorithm used.

Keywords: Text classification · Gender identification · Feature selection

1 Introduction

The growth of social media and its users is impressive and the exploration of the available information in terms of topic, author, genre, etc., is a basic task. Every user of social media leaves his digital traces, makes transactions, expresses his opinion about things and describes moments of his life. Many trends have sprung up over tweets, blog posts and Facebook statuses. These trends express not only an individual user, but often an entire social group. Therefore, it is an interesting task to detect demographic characteristics, such as gender, in users' text data. Information about the gender can be derived not only from the data the user provides about himself, but also implicitly, from the linguistic choices he/she makes. The automatic extraction of information from the everyday enormously growing volumes of data related to the gender, age and other demographic characteristics of the user is essential in the e-government, security and e-commerce market.

The online user's attitude can be observed and explained from a social perspective and his/her "digital traces" [10] may be very informative about current

A. Ronzhin et al. (Eds.): SPECOM 2015, LNAI 9319, pp. 226–233, 2015.
DOI: 10.1007/978-3-319-23132-7_28

trends in any domain. The user's online activity reveals several elements, not only about his/her preferences or transactions, but also about his/her identity. The user, consciously or not, provides information about his/her social status, gender, age and even his/her educational level and profession.

User's gender detection can be perceived as a text classification task, where machine learning techniques are used to identify the author's gender [4,5]. Koppel et al. [9] propose text classification methods to extract the author's gender from formal texts, using features such as n-grams and function words that are more frequent in authorship attribution. They combine stylometric and text classification techniques, in order to extract the author's gender. In a subsequent study [2], Argamon et al., by applying factor analysis and machine learning techniques, contribute with gender and age information, in texts mined from the blogosphere. Ansari et al. [1] use frequency counting of tokens, tf-idf and part-of-speech (POS) tags to find the gender of blog authors. For gender recognition in Twitter, Burger et al. [3] exploit the content of the tweet combined with the username and other pieces of information related to the user.

Recent studies in gender estimation [8,12,13,18] deal with social media and propose methods that detect the gender and, in some cases, the age of the users. The experiments were implemented with gender-polarized words, POS tags and sentence length among other features. In Sarawgi et al. [15], they perform a comparative study of gender attribution, without taking into account the topic or the genre of the selected text. In [17] they use the Naive Bayes classifier, combined with text features and features such as the web page color. Holmgren and Shyu [6] applied machine learning techniques using a feature vector containing word counts, in order to detect the author's gender of Facebook statuses. Rangel and Rosso [14] introduced a set of stylistic features to extract the gender and age of authors, using a large set of documents from the social web, written in Spanish. Finally, Marquardt et al. [11] focused on detecting the best feature set towards age and gender prediction in social media.

In the present article, we perform feature ranking and subset selection, aiming to improve the accuracy of author gender classification on web blogs. The feature selection is performed over a large set of features, using statistical, part-of-speech tagging and language model based feature extraction methodologies. These text features were examined by machine learning classification algorithms, in order to evaluate the gender classification performance for different numbers of features.

The rest of the paper is organized as follows. In Sect. 2 we describe the proposed methodology for gender identification of web users from blog posts. In Sect. 3 we demonstrate and analyze the experimental results and finally, in Sect. 4 we conclude this work.

2 Proposed Gender Identification Methodology

For the feature selection on the task of blog authors' gender identification, we adopted a standard approach followed in most of the previous related work, i.e. pre-processing, feature extraction and classification structure was utilized, as shown in Fig. 1.

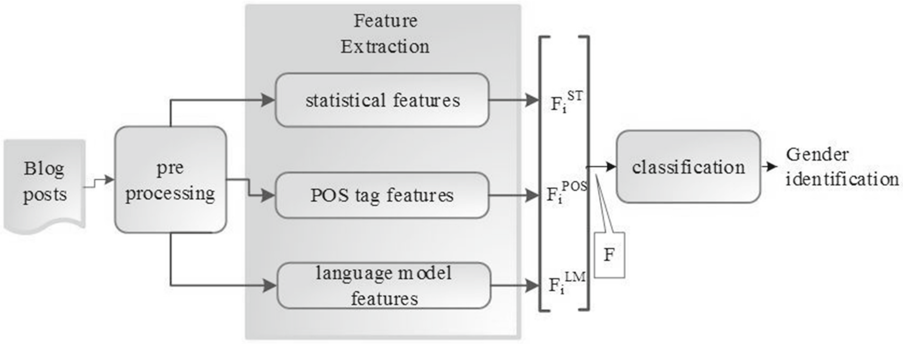

Fig. 1. Block diagram of the blog post users' gender identification scheme

Specifically, each blog post is initially preprocessed. During pre-processing, each post is split into sentences and each sentence is split into words. Afterwards, three feature extraction methodologies are applied in parallel and independently to each other to each post. In detail, statistical, part-of-speech (POS) tag and language model features are extracted constructing vectors F_i^{ST}, F_i^{POS} and F_i^{LM}, respectively. These features are consequently concatenated to a super vector $F = F_i^{ST}||F_i^{POS}||F_i^{LM}$. This results to one feature vector, F, per blog post, which is processed by a classification algorithm, in order to label each post with a gender class.

2.1 Feature Extraction

Three categories of features were computed for each blog post, namely the statistical features, the POS tag features and the language model features.

As far as the statistical features are considered, they consist of statistical values in character and word level. The statistical features that we employed are the following: the number of characters per web post; the normalized number of characters in capital; the normalized number of alphabetic characters; the normalized number of space characters; the normalized number of tab ("\t") characters; the number of occurrence of each alphabetic character; the normalized number of digit characters; the normalized number of occurrence of special characters (˜,@,#,$,%,∧,&,∗,−,_,=,+,>,<,[,],{,},/, \,|); the total number of words; the normalized number of words that consist of less than 4 characters (short words); the normalized number of characters per word; the average word length; the number of sentences; the number of paragraphs; the number of lines; the average number of characters per sentence; the average number of words per sentence; the normalized number of different words; the number of words that appear once in the document; the number of words that appear twice in the document; the number of punctuation symbols (",", ".", "?", "!", ":", ";", " ' "," " "); the number of function words; the average number of sentences per paragraph; the average number of characters per paragraph; the normalized number of words that start with a capital letter; the normalized number of emoticons; the normalized number of words whose letters are all capital; the standard deviation of the word length; the maximum word length; the minimum word length. All the above features compound the F_i^{ST} feature vector, which has dimensionality equal to 30.

POS tag features, which mainly represent a particular part of speech for every word in a given text, were then computed. These are: the number of nouns; the number of proper nouns; the number of adjectives; the number of prepositions; the number of verbs; the number of pronouns; the number of interjections; the number of adverbs; the number of articles. The F_i^{POS} feature vector comprises all these features and has dimensionality equal to 9.

Finally, for the language model features, we use 2 unigram, bigram and trigram language models, one for each gender class (female and male), in order to measure log likelihood and entropy as well as their normalized values for each model. These features are language independent. The F_i^{LM} feature vector contains all the above features and, therefore, is a 24-dimensional feature vector.

The concatenation of the three feature vectors results to F, as described above. Thus, for each blog post, one final feature vector, F, is constructed, which has dimensionality equal to $30 + 9 + 24 = 63$. For the estimation of the above text features we used the NLTK [19] open-source toolkit.

2.2 Feature Selection

For the evaluation of the importance of the features we relied on ReliefF algorithm [7]. ReliefF evaluates the worth of a feature by repeatedly sampling an instance and considering the value of the given feature for the nearest instance of the same and different class. ReliefF can operate on both discrete and continuous class data. In the present evaluation we used the Ranker [16] search method. Ranker method ranks features by their individual evaluations. For the feature selection step we used the WEKA machine learning toolkit software [20] implementation.

2.3 Classification

For the classification stage, we used a number of dissimilar machine learning algorithms, which are well studied and have been used extensively in several text classification tasks. In particular, we used a multilayer perceptron neural network (MLP), using the back-propagation algorithm for training and three layers, and support vector machines (SVMs) using the sequential minimal optimization algorithm, which were tested using radial basis kernel (rbf) and polynomial kernel (poly). In addition, we used four tree algorithms, namely the pruned $C4.5$ decision tree (J48), the random tree (RandTree), constructing a tree that considers K randomly chosen attributes at each node, the random forest (RandForest) constructing a forest of random trees and the fast decision tree learner (RepTree), that builds a decision tree using information gain or variance and prunes it using reduced-error pruning with back-fitting. Finally, we employed one lazy-learning algorithm, the k-nearest neighbor, IBk. All classifiers were implemented using the WEKA toolkit [20].

3 Experimental Setup and Evaluation

The feature selection methodology for blog post authors' gender classification described in the previous section was evaluated using the statistical, POS tag and language model features and the classifiers presented above. In order to avoid overlap between training and test subsets, a 10-fold cross validation evaluation protocol was followed.

For our evaluation we used the publically available "Blog author gender classification dataset", which was introduced in the work of Mukherjee and Liu [12]. It contains blog posts in English, 1390 written by female authors and 1546 written by male authors. The dataset includes 1319917 words and 6085202 characters. Each post was labeled as female or male, depending on the gender of its author, which was determined by the profile of each author.

At first, we tested the discriminative ability of the language model features, so we used only these features with the classifiers. The dimensionality of the feature vector is, therefore, 24. Table 1 shows the experimental results, in terms of percentages of correctly classified blog posts. SVM using polynomial kernel achieves the best accuracy, which is equal to 69.35 %.

As a second step, we evaluated the appropriateness of all the features on the task of web blog authors' gender identification, using the ReliefF criterion. As seen in Table 2, the language model based features were found to be the most relevant to gender classification, since sixteen out of the top-20 features derive from the language model based features we trained.

The absence of POS tag features from the top-20 list implies that these features are weakly correlated to the gender of an author. It is worth mentioning that the normalized number of characters per word, short words, different words and digits were found important for classifying a blog post as male or female, considering that they are ranked in the top-15, top-18, top-19 and top-20 features respectively.

In total, 62 out of the 63 text features were found to be to some degree relevant with the gender classification problem, i.e. they demonstrated positive attribute quality value. The remaining one, namely the normalized number of tab characters, obtained a negative value, which means that it is not a relevant attribute with respect to the gender classification problem.

The classification accuracy was evaluated in terms of percentages of correctly classified blog posts. The experimental results, in percentages, for the eight dif-

Table 1. Accuracy for gender classification using only the language model features.

J48	MLP	SVMrbf	SVMpoly	RandForest	RandTree	REPTree	IBk
68.19	68.02	67.91	**69.35**	67.92	60.42	67.40	68.66

Table 2. Ranking results for the top-20 features.

Rank	Score	Feature description	Feature type
1	0.00769	norm. entropy of female unigram lang. model	language model
2	0.00769	norm. entropy of male unigram lang. model	language model
3	0.00769	norm. log likel. for female unigram lang. model	language model
4	0.00769	norm. log likel. for male unigram lang. model	language model
5	0.00668	norm. entropy of female trigram lang. model	language model
6	0.00668	norm. entropy of male trigram lang. model	language model
7	0.00668	norm.log likel. for female trigram lang. model	language model
8	0.00668	norm. log likel. for male trigram lang. model	language model
9	0.00666	entropy for male trigram lang. model	language model
10	0.00489	norm. entropy for male bigram lang. model	language model
11	0.00489	norm. entropy for female bigram lang. model	language model
12	0.00489	norm. log likelihood for male bigram lang. model	language model
13	0.00489	norm. log likel. for female bigram lang. model	language model
14	0.00478	entropy for female trigram lang. model	language model
15	0.00354	norm. number of characters per word	statistical
16	0.00351	entropy for male bigram lang. model	language model
17	0.00309	entropy for female bigram lang. model	language model
18	0.00241	number of short words	statistical
19	0.00189	number of different words	statistical
20	0.00182	digits	statistical

ferent machine learning techniques and for each of the 6 different subsets of the top-n features with $n = \{10, 20, 30, 40, 50, 60\}$ are presented in Table 3. The last column presents the classification accuracy for each algorithm in the case that we do not perform feature selection ("All") and the dimensionality of the feature vector is 63. The best performing subset of features for each algorithm is indicated in bold. The top-10 features consist only of language model features. The top-20 features contain 16 language model features and so do the top-30 features. The top-40 features include 21 language model features; the top-50 features include 23 language model features and finally, the top-60 features contain all the language model features.

Comparing the results obtained with the use of feature selection to those using all 63 features, i.e. with no feature selection, we observe that for the all classification algorithms the former results are better compared to the latter. This fact verifies our initial hypothesis, namely that the feature selection is able to improve the accuracy of gender classification. As seen in Table 3, the improvement in accuracy varies from 0.51 % in SVM using rbf kernel to 7.12 %, in the case of Random Tree. The IBk shows a significant improvement in performance, which is approximately 6 %, followed by J48 and MLP, with improvements of 2.8 % and 1 % respectively.

Table 3. Accuracy for gender classification per feature subset and classification algorithm.

	top-10	top-20	top-30	top-40	top-50	top-60	All
J48	**68.32**	67.92	67.44	67.54	67.47	67.68	65.46
MLP	68.46	67.88	68.56	68.56	**68.66**	67.27	67.57
SVMrbf	68.02	68.26	68.32	**68.42**	68.36	68.32	67.91
SVMpoly	69.43	69.24	69.75	69.75	69.93	**70.27**	69.72
RandForest	69.52	68.46	69.41	**70.50**	70.03	69.96	69.69
RandTree	67.01	**68.43**	67.17	65.90	66.62	66.55	61.31
REPTree	67.23	**67.51**	66.49	66.79	66.38	67.34	66.62
IBk	68.12	68.29	**68.66**	68.56	68.56	68.49	63.01

The best classification accuracy, 70.50 %, was achieved by combining the top-40 features with Random Forest, followed by the 70.27 %, achieved using the top-60 features with SVM using polynomial kernel. It is worth mentioning that SVM using polynomial kernel performs better with a larger number of features. This is owed to the curse of dimensionality phenomenon, from which the SVMs do not suffer. One the other hand, J48 performs best when using only the top-10 features. In general, the top-10 and top-20 features obtain quite high results and specifically, Random Tree and REP Tree achieve their best performance using only the top-20 features. This shows the great impact of the language model features on the gender classification problem, since the top-20 features consist mainly of them. Finally, as can be seen in Table 1, the use of the language model features only, succeeded a quite high performance, having accuracy equal to 69.35 %.

4 Conclusion

In this article, we investigated the effectiveness of feature selection for improving the accuracy of web blog authors' gender identification. A combination of statistical, part of speech tagging and language model based features was used to represent each web post. We evaluated the gender identification performance using eight classification algorithms for different subsets of features according to ranking scores, produced by the ReliefF algorithm. Random Forest outperformed all the evaluated algorithms, when using the top-40 text-based features, with accuracy equal to 70.50 %. The experimental results showed that the use of subsets of features improved the overall gender identification accuracy for all the evaluated algorithms. Finally, the superiority of the language model features was proved, since the best classification results were obtained, mainly, due to their contribution.

References

1. Ansari, Y.Z., Azad, S.A., Akhtar, H.: Gender classification of blog authors. Int. J. Sustain. Dev. Green Econ. **2**(1) (2013). ISSN No: 2315–4721
2. Argamon, S., Koppel, M., Pennebaker, W., Schler, J.: Mining the Blogosphere: age, gender and the varieties of self-expression. First Monday **12**, 9 (2007)
3. Burger, J., Henderson, J., Kim, G., Zarrella, G.: Discriminating gender on Twitter. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1301–1309. Association for Computational Linguistics, Stroudsburg (2011)
4. Cheng, N., Chandramouli, R., Subbalakshmi, K.P.: Author gender identification from text. Int. J. Digit. Forensics Incident Response **8**(1), 78–88 (2011)
5. Company, J.S., Wanner, L.: How to use less features and reach better performance in author gender identification. In: Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC). Reykjavik, Iceland (2014)
6. Holmgren, J., Shyu, E.: Gender Classification of Facebook Posts (2013)
7. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: Proceedings of the 9th International Workshop on Machine Learning, pp. 249–256 (1992)
8. Kobayashi, D., Matsumura, N., Ishizuka, M.: Automatic estimation of Bloggers' gender. In: Proceedings of International Conference on Weblogs and Social Media (2007)
9. Koppel, M., Argamon, S., Shimoni, A.R.: Automatically categorizing written texts by author gender. Literary Linguist. Comput. **17**(4), 401–412 (2003)
10. Lazer, D., Pentland, A.S., Adamic, L., Aral, S., Barabasi, A.L., Brewer, D., Van Alstyne, M.: Life in the network: the coming age of computational social science. Science **323**(5915), 721 (2009). (New York, NY)
11. Marquardt, J., Farnadi, G., Vasudevan, G., Moens, M., Davalos, S., Teredesai, A., De Cock, M.: Age and Gender Identification in Social Media. Author Profiling Task at PAN (2014)
12. Mukherjee, A., Liu, B.: Improving gender classification of blog authors. In: Proceedings of EMNLP (2010)
13. Peersman, C., Daelemans, W., Van Vaerenbergh, L: Predicting age and gender in online social networks. In: Proceedings of the 3rd Workshop on Search and Mining User-Generated Contents, Glasgow, UK (2011)
14. Rangel, F., Rosso, P.: Use of language and author profiling: identification of gender and age. In: Proceedings of the 10th International Workshop on Natural Language Processing and Cognitive Science (2013)
15. Sarawgi, R., Gajulapalli, K., Choi, Y.: Gender attribution: tracing stylometric evidence beyond topic and genre. In: Proceedings of the 15th Conference on Computational Natural Language Learning, pp. 78–86. Association for Computational Linguistics, Stroudsburg (2011)
16. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan-Kaufman Series of Data Management Systems, 2nd edn. Elsevier, San Francisco (2005)
17. Yan, X., Yan, L.: Gender Classification of Weblog Authors. Computational Approaches to Analyzing Weblogs, AAAI (2006)
18. Zhang, C., Zhang, P.: Predicting gender from blog posts. Technical report. University of Massachusetts Amherst, USA (2010)
19. NLTK. http://www.nltk.org/
20. WEKA. http://www.cs.waikato.ac.nz/ml/weka/