Age and Gender Classification of English Tweets using Support Vector Machines

Roy Khristopher Bayot
Teresa Gonçalves, PhD

Universidade de Évora, Department of Informatics,
Rua Romão Ramalho nº59, 7000-671 Évora, Portugal
d11668@alunos.uevora.pt, tcg@uevora.pt

Abstract. Author profiling from twitter data was done in this paper. It focuses on English language and specifically age and gender classification. The individual effects of bag of words, text ngrams, and POS ngrams on the classification were explored and different SVM kernels and parameters were also tested. Finally, the study was able to achieve 80.92% and 80.79% for age and gender respectively using only *tfidf* and different SVM kernels.

Keywords: author profiling, twitter, tfidf, normalized term frequency, SVM, POS, ngrams

1 Introduction

Author profiling has been of importance in the recent years. From a forensic standpoint for example, it could be used to determine potential suspects by getting linguistic profiles and identifying characteristics. From a business intelligence perspective, companies could target specific people through online advertising. By knowing the profile of the authors, companies would easily find what a specific group of people talk about online and device strategies to advertise to these people. They could also analyze product reviews and know what types of products are liked or disliked by certain people.

One specific problem, which is the aim of this work, is to investigate age and gender classification of twitter users through Support Vector Machines [3] using only tweets from a dataset given in PAN 2015 [13]. The dataset contains twitter data from 4 different languages - English, Dutch, Italian, and Spanish. These tweets are used to profile an author based on age, gender, and 5 personality traits - agreeability, conscientiousness, extrovertedness, openness, and stability. This paper limits the investigation to age and gender classification for English tweets.

2 Related Literature

One of the first few works on author profiling is that of Argamon et al. in [1] where texts are categorized base on gender, age, native language, and personality. For personality, only neuroticism was checked. The corpus comes from different sources. The age and gender have the same corpus taken from blog postings. The native language corpus was taken from International Corpus of Learner English. Personality was taken from essays of pyschology students from University of Texas in Austin. Two types of features were obtained: content-based features and style-based features and Bayesian Multinomial Regression was used as a classifier. From the gender task, they were able to achieve 76.1% accuracy using style and content features. For age task with 3 classes, the accuracy was at 77.7% also using style and content features. For the native language task, the classifiers were able to achieve 82.3% using only content features. And finally, in checking for neuroticism, the highest obtained was 65.7% using only style features.

There has also been some research that uses datasets collected from social media. A particular example is that of Schler et al. in [15] where writing styles in blogs are related to age and gender. Stylistic and content features were extracted from 71,000 different blogs and a Multi-Class Real Winnow was used to learn the models to classify the blogs. Stylistic features included parts-of-speech tags, function words, hyperlinks, and non-dictionary words. Content features included word unigrams with high information gain. The accuracy achieved was around 80% for gender classification and 75% for age identification.

The work or Argamon et al. [1] became the basis for the work in PAN. It is an ongoing project from CLEF with author profiling as one of its tasks. It currently has three editions. In the first edition of PAN [12] in 2013, the task was age and gender profiling for English and Spanish blogs. There were a variety of methods used. One set includes content-based features such as bag of words, named entities, dictionary words, slang words, contractions, sentiment words, and emotion words. Another would be stylistic features such as frequencies, punctuations, POS, HTML use, readability measures, and other various statistics. There are also features that are n-grams based, IR-based, and collocations-based. Named entities, sentiment words, emotion words, and slang, contractions and words with character flooding were also considered. After extracting the features, the classifiers that were used were the following - decision trees, Support Vector Machines, logistic regression, Naïve Bayes, Maximum Entropy, Stochastic Gradient Descent, and random forests. The work of Lopez-Monroy in [6] was considered the winner for the task although they placed second for both English and Spanish with an accuracy of 38.13% and 41.58% respectively. They used second order representation based on relationships between documents and profiles. The work of Meina et al. [9] used collocations and placed first for English with a total accuracy of 38.94%. On the other hand, the work of Santosh et al. in [14] gave a total accuracy of 42.08% after using POS features for Spanish.

In PAN 2014 [11], the task was profiling authors with text from four different sources - social media, twitter, blogs, and hotel reviews. Most of the approaches used in this edition are similar to the previous year. In [5], the method used to represent terms in a space of profiles and then represent the documents in the space of profiles and subprofiles were built using expectation maximization clustering. This is the same method as in the previous year in [6]. In [7], ngrams were used with stopwords, punctuations, and emoticons retained, and then idf count was also used before placed into a classifier. Liblinear logistic regression returned with the best result. In [18], different features were used that were related to length (number of characters, words, sentences), information retrieval (cosine similarity, okapi BM25), and readability (Flesch-Kincaid readability, correctness, style). Another approach is to use term vector model representation as in [17]. For the work of Marquardt et al. in [8], they used a combination of content-based features (MRC, LIWC, sentiments) and stylistic features (readability, html tags, spelling and grammatical error, emoticons, total number of posts, number of capitalized letters number of capitalized words). Classifiers also varied for this edition. There was the use of logistic regression, multinomial Naïve Bayes, liblinear, random forests, Support Vector Machines, and decision tables. The method of Lopez-Monroy in [5] gave the best result with an average accuracy of 28.95% on all corpus-types and languages.

3 Dataset and Tools

The dataset for the problem at hand is composed of a set of tweets for 4 different languages - English, Dutch, Italian, and Spanish. We limited the investigation to just English and decided to build one model for age classification and another model for gender classification. There were 4 categories for the age classification - 18-24, 25-34, 35-49, and 50 and above. There were 152 users for English. Each user has different number of tweets but the dataset is balanced based on gender. The table 1 gives the minimum, maximum, average, standard deviation, as well as the median of the number of tweets of each user, total length of tweets for each user, and the average tweet length of each user.

	min	max	average	std	median
number of tweets of each user	32	100	93.20	16.82	100
total length of tweets of each user	1979	12485	7445.30	2389.61	7438.5
average tweet length of each user	29.46	124.85	79.61	20.28	80.35

Table 1. Various statistics on twitter data.

Processing the data was done through Python using the scikits-learn [10] library. For POS tags, TreeTagger as described in [16] was used with a python wrapper.

4 Experiments

The approach used was straightforward. First, there is preprocessing. Then for the first set of experiments, feature extraction. We investigate three different sets of features. The first set is the bag of words features in the form of term frequency and term frequency inverse document frequency. The second set are text ngrams. And finally, we also study part of speech ngrams to see if information extracted from these features are useful for characterizing twitter users. After determining which features are useful in classification, we then proceed to reduce the number of features that could be used to evaluate. In the second set of experiments, we used the features from the previous experiment to find the optimal Support Vector Machine parameters and kernels for classification. Finally, in the last set of experiments, we also check different for three other methods. We want to know if additional preprocessing would affect the results. This was done by substituting strings such as hyperlinks and hashtags into " hashtag_here " and " link_here ". We also want to know if information on the other class affects the classification. Lastly, we want to know if ensemble methods affect the result. Experiments were made to check for these objectives. More details are described in the succeeding sections.

4.1 Preprocessing

Xml files from each user are read. Then the tweets taken from each user are extracted and concatenated into one line to form one training example. The examples are then transformed by putting them all in lower case. No stop words are removed. Hashtags, numbers, mentions, shares, and retweets were not processed or transformed to anything else. They were retained as is and therefore will correspond to another item in the dictionary of features. The resulting file is used for feature extraction. The resulting file is used for feature extraction.

4.2 Determining A Viable Feature Set

The following set of features were extracted after preprocessing. Normalized term frequency and *tfidf* both yielded 26264 different features. Text bigrams gave 104435, text trigrams with 147577, while POS unigrams gave 35 and POS bigrams gave 895.

Bag-of-Words Two different bag-of-words features were extracted. The first is normalized term frequency. Terms are normalized by the number of terms in a training example. No terms were discarded. The second one is *tfidf* where terms were normalized by the inverse document frequency. No terms were also discarded.

Text Ngrams Text ngrams were the final set of features that was examined. Counts of bigrams and trigrams were taken after preprocessing. No normalization was done and no terms were discarded.

POS Ngrams Part of speech tags were also another set of features that was examined in this study. It was extracted by using a python wrapper to Schmid's TreeTagger program as detailed in [16]. Counts for unigrams, bigrams, and the combination of the two were used as features. The English parameter file for TreeTagger has 36 different tags. No terms were discarded.

Evaluating the Effect of Feature Types on Classification Table 2 compares the performance of the different feature types. We can see that *tfidf* features give better results than normalized term frequency results. We also see that text bigrams gives a better accuracy than text trigrams and POS bigrams give a better accuracy than the two others. Of course *tfidf* has the highest accuracy among all. However, looking at p-values after performing Wilcoxon signed rank test, we can see that only the results for bag-of-words features are statistically different from each other. This is indicated in boldface in the table.

	BoW		Text Ngrams		POS Ngrams		
	tf	tfidf	bigrams	trigrams	unigrams	bigrams	uni+bi
gender	0.511	**0.683**	0.623	0.652	0.659	0.659	0.660
age	0.395	**0.691**	0.638	0.553	0.579	0.611	0.593

Table 2. Accuracy results for age and gender classification using different types of features. Highest accuracy is given in bold face.

We then chose to just use *tfidf* as the set of features for classification. However, the set is still big. And we reduced this even further by ranking the features by information gain and gain ratio and try to reduce the feature set from there.

The results for feature reduction are given in table 3. The highest accuracy could be attained when using all the different features. However, Wilcoxon signed rank test [19] was done between the experiment with the highest accuracy and other subsequent experiments within each task and each ranking function. For example, in age classification, information gain features were used and the experiment that used 26263 features was compared to the other experiment that used 10000, 9000, and so on. The numbers in boldface are those which are not statistically different from each other. Therefore, for further experiments, 9000 features were used for classification while 7000 features were used for gender classfication. Information gain ranking was used instead of gain ratio because the words given by information gain were subjectively more descriptive than those given by gain ratio.

4.3 Determining Optimal SVM Kernel Parameters

Using Support Vector Machines [3] entails the use of kernels that maps the features into a different space such that separation would be done in the new space. In the earlier section, only the linear kernel was used but we further used

	Age		Gender	
Num Features	info gain	gain ratio	info gain	gain ratio
26263	**0.6913**	**0.6913**	**0.6825**	**0.6825**
10000	**0.6321**	**0.6321**	**0.6167**	**0.6167**
9000	**0.6321**	**0.6321**	**0.6163**	**0.6163**
8000	0.6188	0.6188	**0.6233**	**0.6233**
7000	0.6188	0.6188	**0.6300**	**0.6300**
5000	0.6188	0.6188	0.4863	0.4863
2000	0.6188	0.6188	0.4983	0.4983
1000	0.6188	0.6188	0.4983	0.4983
700	0.6188	0.6188	0.4921	0.4921
500	0.6188	0.6188	0.4921	0.4921
300	0.6188	0.6188	0.4921	0.4921
200	0.6188	0.6125	0.4921	0.4921
100	0.6188	0.3950	0.4921	0.4921

Table 3. Accuracies that result from using different number of important features ranked by information gain and gain ratio. SVM linear kernel with C=1 was used.

polynomial kernels and radial basis function kernels. Polynomial kernels maps the input features to another feature space that uses the polynomial function over the similarity of the input features. Mathematically, the kernel is given by equation 1 but scikits-learn implementation has a gamma to scale the dot product as in equation 2.

$$K(x, y) = (x^T y + c)^d \tag{1}$$

$$K(x) = (\gamma(x^T x) + c)^d \tag{2}$$

For our experiments, we set the gamma to be equal to 1 but the degrees d to vary between 1, 2, and 3. For both age and gender classification, c varies between 0.0001, 0.001, 0.1, 1, 10, 1000, 10000.

Radial basis function kernel is another kernel explored. Mathematically, it is given by equation 3. It is similar the scikits implementation but with a regularization factor c.

$$K(x, x') = exp\left(-\frac{\parallel x - x' \parallel^2}{2\sigma^2}\right) \tag{3}$$

For both age and gender classification, σ and c were chosen to be one among 0.0001, 0.001, 0.1, 1, 10, 1000, and 10000.

The results for age classification using support vector machines with polynomial kernel is given by table 4. The highest accuracy obtained was 80.92%. This was obtained from three different settings, degree 3 with C to be either 10, 1000, 10000. However, checking on Wilcoxon signed rank test [19], these results are not statistically significant to 80.25% given by settings with degree 2 and C to be either 10, 1000, 10000. These results are also not statistically significant against 75% with degree 3 and C to be 1. The results shown in boldface are the highest accuracies that are not statistically different from each other.

C	polynomial			radial basis function						
	degree			gamma						
	1	2	3	0.0001	0.001	0.1	1	10	1000	10000
0.0001	0.6321	0.6192	0.5121	0.3950	0.3950	0.3950	0.3950	0.3950	0.3950	0.3950
0.001	0.6321	0.6192	0.5121	0.3950	0.3950	0.3950	0.3950	0.3950	0.3950	0.3950
0.1	0.6321	0.6254	0.5992	0.3950	0.3950	0.3950	0.4746	0.3950	0.3950	0.3950
1	0.6321	0.7104	**0.7500**	0.3950	0.3950	0.6054	**0.6517**	0.5933	0.3950	0.3950
10	0.6321	**0.8025**	**0.8092**	0.3950	0.3950	**0.6846**	**0.8025**	0.6196	0.3950	0.3950
1000	0.6321	**0.8025**	**0.8092**	0.6188	**0.6971**	**0.8025**	**0.8025**	0.6196	0.3950	0.3950
10000	0.6321	**0.8025**	**0.8092**	0.6971	**0.8092**	**0.8025**	**0.8025**	0.6196	0.3950	0.3950

Table 4. SVM with polynomial and radial basis function kernel on age classication.

The results for age classification using support vector machines with radial basis function kernel is also given by table 4. Results show that the highest achieved accuracy was 80.92%. It was obtained from the kernel with gamma to be 0.001 and with a C to be 10000. This wasnt statistically different from the settings that gave 80.25%, 69.71%, 68.46%, and 65.17% which are all shown in boldface.

Comparing the two different kernels for age classification including different parameters, the settings settled for would be with the polynomial kernel of degree 3 with C equal to 10.

The results for gender classification using support vector machines with polynomial function kernel is given by table 5. Results show that the highest achieved accuracy was 79.54%. This comes from six different settings. The first three were when the degree is 2 and C was chosen to be either 10, 1000, or 10000. The last three settings were when degree is 3 with C also either 10, 1000, or 10000. These results are not statistically different from that which gave 70.25% which are all written in boldface.

The results for gender classification using support vector machines with radial basis function kernel is also given by table 5. Results show that the highest achieved accuracy was 80.79%. This comes from the settings with gamma to be 1 and C to be 10. However, these arent statistically different from settings which gave 80.21%, 80.13%, and 79.54% accuracy, all of which are written in boldface in the table.

Comparing the two different kernels for gender classification including different parameters, the settings settled for would be with the radial basis function kernel with gamma equal to 1 and C equal to 10.

4.4 Determining Effects of Other Priors and Ensemble Methods

Classification with string substitution One of the interesting experiments is to check if the addition of some twitter specific features could affect the clas-

C	polynomial			radial basis function						
	degree			gamma						
	1	2	3	0.0001	0.001	0.1	1	10	1000	10000
0.0001	0.6300	0.4983	0.4733	0.5233	0.5233	0.5233	0.5171	0.4921	0.5171	0.4733
0.001	0.6300	0.4983	0.4733	0.5233	0.5233	0.5233	0.5171	0.4921	0.5171	0.4733
0.1	0.6300	0.5233	0.4733	0.5233	0.5233	0.5233	0.5171	0.4921	0.5171	0.4733
1	0.6300	0.6367	**0.7025**	0.5233	0.5233	0.5233	0.6167	0.6629	0.4796	0.4733
10	0.6300	**0.7954**	**0.7954**	0.5233	0.5233	0.6238	**0.8079**	0.7025	0.4796	0.4733
1000	0.6300	**0.7954**	**0.7954**	0.5233	0.6367	**0.8021**	**0.8013**	0.7025	0.4796	0.4733
10000	0.6300	**0.7954**	**0.7954**	0.6367	**0.7954**	**0.8021**	**0.8013**	0.7025	0.4796	0.4733

Table 5. SVM with polynomial and radial basis function kernel on gender classication.

sification. For this experiment we only look into hashtags and links because user mentions are already handled previously since users have been anonymized. The treatment is simple and done in the preprocessing side. After taking the text from the html and converting them into lowercase, string substitution was performed on links and hashtags. All links were transformed into the text " LINK_HERE " while hashtags were turned into " HASHTAG_HERE ". The idea is that each link and hashtag wont be considered as a unique and that it would just add to the number of links or hashtags. After string substitution, *tfidf* was used and then features were ranked based on information gain. Top 9000 and 7000 words were used age and gender classification respectively. Finally, classification was done using the optimal settings given by the previous experiments and that given in the succeeding chapter.

Comparisons were also performed between the the best from previous parameter tuning experiments and that where links and hashtags were substituted with a different string tag. The performance drops when there's string substitution even while using the same optimal parameters for the classifier obtained from previous experiments. Optimal parameters means using polynomial kernel with degree 3 and C to be 10 in age classification, while using radial basis function with gamma to be 1 and C to be 10 for gender classification. For age classification using string substitution, the accuracy is 78.92% as compared to 80.92% from the best from previous experiments. For gender classification using string substitution, the accuracy is 74.42% as compared to 80.70% from the best from previous experiments. However, the results are not statistically different.

Using Information of the Other Class as a Feature Another interesting experiment is that of classification when using the information from the other class as a feature. For example, the results of gender classification will be used as a feature for age classification. And this will be done vice-versa with gender classification.

Comparisons were also performed between the the best from previous parameter tuning experiments and another method where the training set features is augmented by the information of the other class. For age classification, the accuracy result using features augmented with the gender information gives 82.92%

while the best from previous experiments gives 80.79%. It's a difference of about 2%, however the difference is not statistically different. For gender classification, the results are almost the same with the augmented set giving 80.21% while the best from previous experiments give 80.79%, with the difference also not statistically significant. Optimal parameters for age classification uses a polynomial kernel with degree 3 and C to be 10, while gender classification uses radial basis function with gamma to be 1 and C to be 10.

Exploring Ensemble Methods Finally, a comparison with ensemble methods was also done. Two different ensemble methods were considered - Random Forests and AdaBoost. Random Forests is one of the ensemble methods wherein results are averaged in the end. The full details of the algorithm are given by the paper of Breiman in [2]. The idea is to build a multitude of decision trees and averaging their prediction results. The trees built will vary since they are built by taking a random sample with replacement from the training set. Furthermore, the features will also vary since it will be a random subset of features that are selected when the splitting is done in the training phase.

Adaptive Boosting or AdaBoost is another ensemble method formulated by Freund and Schapire in [4]. The idea is to combine different weak models to produce a powerful ensemble. It is considered adaptive since the succeeding weak learners give more weight in classifying correctly instances in the training set which was misclassified by previous weak learners.

Accuracy results for random forests are given in table **??**. The highest accuracy obtained for age and gender classification using random forests are 61.12% and 70.29% respectively. However, after performing Wilcoxon signed rank test [19] towards the other results within each task, the p-values show that the values are not statistically different from each other.

Random Forest			AdaBoost		
n-estimators	Age	Gender	n-estimators	Age	Gender
10	0.6058	0.5983	50	**0.5479**	0.6787
100	0.6046	0.6500	100	0.5217	0.7187
1000	0.6112	**0.7029**	150	0.5475	0.7183
2000	0.6046	0.6762	200	0.5146	**0.7500**
5000	0.6108	0.6963	250	0.5275	0.7308
10000	**0.6175**	0.6967			

Table 6. Accuracy results for age and gender classification using forests of randomized trees and AdaBoost.

5 Conclusion and Recommendations

There are numerous conclusions that could be found from the results. First, we observed that *tfidf* performed best among the different types of features. Second, we can conclude that using minimal preprocessing, using the top 9000 features as ranked by information gain for age classification would yield the same results as using all the different features, from a statistical standpoint. This also goes true for gender classification but with the top 7000 features. Third, looking into the differences in kernel functions, the results show that the highest accuracy that could be achieved were 80.92% and 80.79% for age and gender respectively. These accuracies could be attained could from either polynomial kernels or radial basis function kernels. For instance, for age classification, the highest accuracy could either be obtained with a polynomial kernel with degree 3 and C to be 10 or it could also be a through a radial basis function with gamma to be 0.001 and C to be 10000. For gender classification, the similarity exists as well. A polynomial kernel could be used with the second degree and C to be 10000 and it will yield the same accuracy as the one with a radial basis function with gamma to be 0.01 and and C to be 10000. For the purposes of our next experiments, we fixed the parameters such that for age classification, we an SVM with polynomial kernel with degree to be 3 and C to be 10, while for gender classification, we use a radial basis function kernel with gamma to be 1 and C to be 10. Fourth, looking into features that are specific to twitter data, we see that substituting a hyperlinks and hashtags lower the accuracy as compared to the best among the previous experiments, although they arent statistically different. Fifth, we can also see that using information given by another class does not improve the classification results. Finally, results from ensemble methods are not statistically different from each other and accuracy results are lower than the others. However, on this front, there are still a lot of things to consider such as number of estimators or even the estimator type and this is definitely something to consider for future work.

For more future work, it should be noted that the current features are very minimal. The preprocessing is minimal. The features are *tfidf* ranked by information gain. For twitter specific features, only the hashtags, user mentions, and hyperlinks were identified and checked. It would be better if other features specific to twitter could be incorporated. Examples of such would be emoticons and character flooding. Experiments with character ngrams is also another direction to check.

References

1. Shlomo Argamon, Moshe Koppel, James W Pennebaker, and Jonathan Schler. Automatically profiling the author of an anonymous text. *Communications of the ACM*, 52(2):119–123, 2009.
2. Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
3. Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
4. Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.
5. A Pastor López-Monroy, Manuel Montes-y Gómez, Hugo Jair Escalante, and Luis Villaseñor-Pineda. Using intra-profile information for author profiling.

6. Adrian Pastor Lopez-Monroy, Manuel Montes-y Gomez, Hugo Jair Escalante, Luis Villasenor-Pineda, and Esaú Villatoro-Tello. Inaoe's participation at pan'13: Author profiling task. In *CLEF 2013 Evaluation Labs and Workshop*, 2013.
7. Suraj Maharjan, Prasha Shrestha, and Thamar Solorio. A simple approach to author profiling in mapreduce.
8. James Marquardt, Golnoosh Farnadi, Gayathri Vasudevan, Marie-Francine Moens, Sergio Davalos, Ankur Teredesai, and Martine De Cock. Age and gender identification in social media. *Proceedings of CLEF 2014 Evaluation Labs*, 2014.
9. Michał Meina, Karolina Brodzinska, Bartosz Celmer, Maja Czoków, Martyna Patera, Jakub Pezacki, and Mateusz Wilk. Ensemble-based classification for author profiling using various features. *Notebook Papers of CLEF*, 2013.
10. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
11. Francisco Rangel, Paolo Rosso, Irina Chugur, Martin Potthast, Martin Trenkmann, Benno Stein, Ben Verhoeven, and Walter Daelemans. Overview of the 2nd author profiling task at pan 2014. In *Proceedings of the Conference and Labs of the Evaluation Forum (Working Notes)*, 2014.
12. Francisco Rangel, Paolo Rosso, Moshe Moshe Koppel, Efstathios Stamatatos, and Giacomo Inches. Overview of the author profiling task at pan 2013. In *CLEF Conference on Multilingual and Multimodal Information Access Evaluation*, pages 352–365. CELCT, 2013.
13. Francisco Rangel, Paolo Rosso, Martin Potthast, Benno Stein, and Walter Daelemans. Overview of the 3rd Author Profiling Task at PAN 2015. In L Cappellato, N Ferro, J Gareth, and E San Juan, editors, *CLEF 2015 Labs and Workshops, Notebook Papers*. CEUR-WS.org, 2015.
14. K Santosh, Romil Bansal, Mihir Shekhar, and Vasudeva Varma. Author profiling: Predicting age and gender from blogs. *Notebook Papers of CLEF*, 2013.
15. Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. Effects of age and gender on blogging. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, volume 6, pages 199–205, 2006.
16. Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the international conference on new methods in language processing*, volume 12, pages 44–49. Citeseer, 1994.
17. Julio Villena-Román and José Carlos González-Cristóbal. Daedalus at pan 2014: Guessing tweet author's gender and age.
18. Edson RD Weren, Viviane P Moreira, and José PM de Oliveira. Exploring information retrieval features for author profiling—notebook for pan at clef 2014. *Cappellato et al.[6]*.
19. Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics bulletin*, pages 80–83, 1945.