International Journal on Artificial Intelligence Tools
Vol. XX, No. X (2006) 1–16
© World Scientific Publishing Company

AUTHORSHIP ATTRIBUTION BASED ON FEATURE SET SUBSPACING ENSEMBLES

EFSTATHIOS STAMATATOS

Dept. of Information and Communication Systems Eng., University of the Aegean,
Karlovassi, Samos - 83200, Greece
stamatatos@aegean.gr

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

Authorship attribution can assist the criminal investigation procedure as well as cybercrime analysis. This task can be viewed as a single-label multi-class text categorization problem. Given that the style of a text can be represented as mere word frequencies selected in a language-independent method, suitable machine learning techniques able to deal with high dimensional feature spaces and sparse data can be directly applied to solve this problem. This paper focuses on classifier ensembles based on feature set subspacing. It is shown that an effective ensemble can be constructed using, exhaustive disjoint subspacing, a simple method producing many poor but diverse base classifiers. The simple model can be enhanced by a variation of the technique of cross-validated committees applied to the feature set. Experiments on two benchmark text corpora demonstrate the effectiveness of the presented method improving previously reported results and compare it to support vector machines, an alternative suitable machine learning approach to authorship attribution.

Keywords: text categorization, authorship attribution, classifier ensembles.

1. Introduction

Authorship Attribution (AA) is the task of identifying the author of a text given a predefined set of candidate authors. Until recently, AA had only limited application mainly to literary works of unknown or disputed authorship[1] providing, in many cases, controversial results[2]. However, by taking advantage of machine learning techniques to exploit high dimensional low-level and language-independent information, the field is now mature to handle real-world cases where only short texts by many candidate authors are available. Therefore, effective identification of text authorship can now significantly assist intelligence and security by providing evidence about the identity of the authors of given texts. For instance, existing authorship analysis approaches can be applied to the verification of authorship of emails and electronic messages[3,4,5], plagiarism detection in student essays[6], and forensic cases[7].

Research in AA focuses mainly on the extraction of the most appropriate features for quantifying the style of an author (the so-called *stylometry*). Several measures have been proposed, including attempts to quantify the diversity of the vocabulary used by the author, function word frequencies (like ‘and’, ‘to’, etc.) and syntactic annotation measures. A good review of stylometry techniques is given by Holmes[8] while Rudman[9] estimates that nearly 1,000 features have been proposed. The style features used in early studies included word and sentence length[10], syllable distribution per word and punctuation mark counts. Although useful for specific cases, they certainly lack generality. However, they can be used in combination with other, stylistically richer features. Another family of measures attempts to model the richness of the vocabulary used by the author[11] (e.g., hapax legomena, type-token ratio etc). The main problem of these features is their strong dependency on text length while they are quite unstable for short texts.

More complicated features are based on syntactic information[12] (e.g., part-of-speech frequencies, use of rewrite rules, etc). In the framework of automatic AA, such features require robust Natural Language Processing (NLP) tools able to provide accurate measures. The use of NLP tools can also provide useful measures related to the specific procedure followed to analyze the text[13]. Another source of useful information is provided by idiosyncratic usage in formatting and spelling[14] given that they can be detected effectively. However, syntactic annotation features are relatively computationally expensive. In addition, so far only a few natural languages are supported by reliable NLP tools. Therefore, such measures make the extraction of stylometric feature a language-dependent procedure.

The most straightforward approach to represent a text is by using word frequencies, a method widely applied to topic-related text categorization as well. To this end, the most appropriate words for AA may be selected in an arbitrary way according to their discriminatory potential on a given set of candidate authors. In a famous case of disputed authorship, Mosteller and Wallace[1] make use of one such hand-crafted word list in order to discriminate between Alexander Hamilton and James Madison, both claimed to be the true authors of 12 *Federalist Papers*. Burrows[15] first indicated that the most frequent words of the texts (like ‘and’, ‘to’, etc.) have the highest discriminative power for stylistic purposes. Interestingly, these words are usually excluded from topic-related text categorization systems. Additionally, the latter approach for selecting words as features for AA is language-independent.

From the machine learning point of view, AA can be viewed as a single-label multi-class text categorization problem where the candidate authors play the role of the classes[16]. Provided a set of texts of undisputed authorship is available for each candidate author, a learning algorithm can be trained using these texts as training set. In this study, we use the most frequent words of the training corpus (the collection of all texts of known authorship by the candidate authors) as style markers. Hence, each text of either the training or test set can be represented as a vector of word frequencies. The obvious question is how many frequent words to use? If the feature set is small (50 to 100 words), the style of the authors is not adequately represented. On the other hand, if the size of the feature set is arbitrarily long (let’s say 1,000 words), the problem of overfitting arises, that is, the learning algorithm loses its generalization ability trying to perfectly fit on the training data. To handle the high dimensional feature space, the following solutions can be applied:

- Feature subset selection: The feature set is reduced to a smaller subset of the most valuable features for the given problem[17]. Alternatively, a reduced set of new synthetic features can be generated from the initial feature set (feature extraction). This approach presumes unavoidable loss of information since many of the available features should be discarded from the classification procedure. Previous studies have shown that every word can contribute to the classification model[18].
- Using a learning algorithm robust to overfitting: Some kernel learning algorithms, especially *Support Vector Machines*[18,19] (SVM), can scale up to considerable dimensionalities. This method depends on the credibility of the learning algorithm to deal with a large number of features given certain limitations in the training data. Moreover, SVM is conceived only for binary classification, hence this method has to be adapted somehow to multi-class classification problems.
- Constructing an ensemble of classifiers based on feature set subspacing: The feature set is divided into smaller parts and multiple classifiers are constructed based on these parts using the same learning algorithm and the same training set[20]. The resulting base classifiers are, then, combined to guess the most likely author.

In this paper, we discuss the latter two approaches that better suit the AA task. Since SVMs is a well-studied learning algorithm[19] the focus will be on the ensemble approach. Constructing classifier ensembles is one of the most active areas in machine learning. However, in order to construct multiple base classifiers, emphasis is given mainly to the manipulation of the training set (e.g., bagging, boosting) or the combination of training and feature set manipulation[21]. This is due to the fact that in most problems a limited number of valuable features is available. A classifier ensemble entirely based on feature set subspacing fits perfectly to AA (and text categorization in general) since there are few irrelevant features. In this study, a series of approaches for constructing an effective ensemble for AA will be explored.

The rest of this paper is organized as follows: Section 2 describes the ensemble-based approach for AA. Section 3 includes the AA experiments for evaluating the proposed approach and Section 4 summarizes the main conclusions drawn by this study and proposes future work directions.

2. Feature Set Subspacing

In order to construct an ensemble based on the feature set subspacing approach it is necessary for the following issues to be addressed:

- The number of feature subsets (and consequently base classifiers) extracted from the available feature set.
- The number of features included in each feature subset.
- The selection procedure for choosing the features that group together in a subset.
- Will the feature subsets be disjoint or not?

- The most appropriate learning algorithm for the base classifiers of this kind of ensemble.
- The most appropriate combination method of the base classifiers.

For the sake of simplicity, we assume equally-sized feature subsets drawn at random. The base learning algorithm will be *linear discriminant analysis*, a standard technique from multivariate statistics. This is a stable classification algorithm proven to be a good compromise between classification accuracy and training time cost[22]. Moreover, this algorithm is able to provide posterior probabilities for each class, which is essential for the presented approach.

2.1. *Building the Model*

Each text is represented as a vector of word frequencies. Let $\mathbf{W}_n=\{w_1, w_2, \ldots, w_\mathrm{n}\}$ be the ordered set of the most frequent word-tokens of the training set (in decreasing frequency of occurrence). A conversion to lower case is assumed and no stemming procedure is performed. Consider f_{ij} as the normalized (by the text size) frequency of the j-th word of $\mathbf{W}_n$ in the i-th text. Then, a text x_i is represented as the ordered vector $<f_{i1}, f_{i2}, \ldots, f_{in}>$. Recall that no stemming or lemmatization is involved in preprocessing the word tokens. Therefore, this text representation is language-independent.

Let $W_{m:n}$ be a subset of m words drawn (without replacement) at random from the set $\mathbf{W}_n$ of the most frequent words of the training corpus ($m \leq n$). Consider $C(W_{m:n})$ as a single linear discriminant classifier trained on the frequencies of these m words in the training set texts. Then, $E(C(W_{m:n}), \mathit{combination})$ is an ensemble of such base classifiers according to the *combination* method.

Consider $\mathbf{L}$ as the set of all possible classes (authors), then the i-th classifier assigns a posterior probability $P_i(C_i(W_{m:n}), x, c)$ to an input text x for each $c \in \mathbf{L}$, so that

$$\sum_{j=1}^{|L|} P_i(C_i(W_{m:n}), x, c_j) = 1 \tag{1}$$

where $|\mathbf{L}|$ is the size of $\mathbf{L}$. In case of learning algorithms that provide only crisp predictions, the posterior probabilities can take only binary values (0 or 1). Provided the posterior probabilities of the constituent classifiers, an ensemble assigns a posterior probability to an input text for each class according to the combination method. For example, the posterior probabilities of the base classifiers can be combined by summation (a combination scheme we call *mean*) or by multiplication (a combination scheme we call *product*), that is:

$$P(E(C(W_{m:n}), mean), x, c) = \frac{1}{k}\sum_{i=1}^{k} P_i(C_i(W_{m:n}), x, c) \tag{2}$$

$$P(E(C(W_{m:n}), product), x, c) = \sqrt[k]{\prod_{i=1}^{k} P_i(C_i(W_{m:n}), x, c)} \tag{3}$$

where *k* is the number of the base classifiers. Comparison of these two combination rules has shown that under the assumption of independence the *product* rule should be used. However, in case of poor posterior probability estimates, the *mean* rule is proved to be more fault tolerant[23].

Considering these two combination methods, a slightly more complicated combination scheme, we call *mp*, which is just the average of *mean* and *product* will be used. Note that *mean* is affected by high values of posterior probabilities, therefore it is favorable for cases where a few base classifiers have assigned a high posterior probability to a class. On the other hand, *product* is affected by low values of posterior probabilities, therefore it is favorable for cases where none of the base classifiers have assigned very low posterior probability to a class. Hence, *mp* is a good compromise of these two. In other words, *mp* selects the author to whom the base classifiers have assigned the best combination of many high scores and few low scores.

To complete the classification model, let *label*(*classifier*, *text*) be the class assigned by a *classifier* to an input *text*. Then, a classifier ensemble chooses the class that maximizes the posterior probability for an input text *x*, that is:

$$label(ensemble, x) = \arg\max_{c \in L}(P(ensemble, x, c)) \tag{4}$$

2.2. *Effectiveness Measures*

The performance of a classifier ensemble is directly measured by the classification accuracy on the test set. Moreover, the effectiveness of an ensemble is indirectly indicated by the diversity among the predictions of the base classifiers as well as the accuracy of the individual base classifiers. In particular, many measures have been proposed to represent the diversity of an ensemble in machine learning literature[24]. In this study, we will use a common measure, the *entropy*, that is:

$$entropy = \frac{1}{|T|}\sum_{i=1}^{|T|}\sum_{c=1}^{|L|} -\frac{N_c^i}{k}\log_{|L|}(\frac{N_c^i}{k}) \tag{5}$$

where *k* is the number of base classifiers, |**T**| is the total number of test texts and N^i_c is the number of base classifiers that assign instance (text) *i* to class (author) *c*. Notice that log is taken in base |**L**| to keep the entropy within the range [0,1]. The higher the entropy of an ensemble, the more diverse the answers of the individual constituent classifiers. In addition, the higher the entropy, the more likely the ensemble to be accurate.

The accuracy of a classifier is the percentage of the correctly classified instances of the test set. However, a more detailed insight into the credibility of a learner is provided by considering the posterior probabilities assigned to each class for a particular test case. More specifically, *Mean Reciprocal Rank* (MRR) is based on the ordered list of classes (from most likely to least likely) predicted by a classifier. The MRR of a classifier is:

$$MRR(classifier) = \frac{1}{|T|}\sum_{i=1}^{|T|}\frac{1}{R(classifier, x_i)} \tag{6}$$

where $R(classifier, x_i)$ is the rank of the true class of test text x_i in the prediction produced by the *classifier*. The higher the MRR, the better the ranking of the true author in the ordered list of the classifier answers (for 10 authors, MRR would range from 0.1 to 1).

3. Authorship Attribution Experiments

3.1. *Text Corpora*

The text corpora used in the following experiments are two benchmark corpora for AA. In particular, the texts were published within 1998 in the Modern Greek weekly newspaper *TO BHMA*, (the tribune) and were downloaded from the WWW site of the newspaper. The texts are divided into two groups of authors:

- Group A (hereafter GA): It consists of ten randomly selected authors whose writings are frequently found in the section A of the newspaper. This section comprises texts written mainly by journalists on a variety of current affairs. Moreover, for a certain author there may be texts from different text genres (e.g., editorial, reportage, etc.). Note that in many cases such texts are highly edited in order to conform to a predefined style, thus washing out specific characteristics of the authors which complicate the task of attributing authorship.
- Group B (hereafter GB): It consists of ten randomly selected authors whose writings are frequently found in section B of the newspaper. This supplement comprises essays on science, culture, history, etc. in other words, texts in which the idiosyncratic style of the author is not overshadowed by functional objectives. In general, the texts included in the supplement B are written by scholars, writers, etc., rather than journalists.

Table 1. The text corpora used in the experiments.

	GA	GB
Average words per text	866.8	1148.2
Authors	10	10
Texts per author	20	20
Texts per author in training set	10	10
Texts per author in test set	10	10

Each corpus is divided into disjoint training and test parts of equal size in terms of texts per author (i.e., ten texts per author in the training set and ten texts per author in the test set for each group). Some brief information about these data sets is summarized in Table 1. More detailed information is given by Stamatatos *et al.*[13] Intuitively, for the GB it is easier to discriminate between the authors, since the texts are more stylistically homogenous. In addition, GB texts are significantly longer than GA texts.

3.2. *Setting the Baseline*

These text corpora were used to test several AA approaches and the best reported results are shown in Table 2. As can be seen, the above considerations are reflected in the reported results since the classification accuracy for GB is much higher in comparison to GA. Although these studies were tested on the same text corpora, they were based on different features to represent the style of the authors: Stamatatos *et al*.[13] exclude any lexical measure and use a natural language processing tool in order to extract 22 structural and syntax-related style markers. Peng *et al*.[25] apply language modeling technology in character-level *n*-grams. Keselj *et al*.[26] represent an author's style by profiles as long as 5,000 character-level *n*-grams. Finally, Peng, *et al*.[27] uses *n*-gram language models (best results for GB are reported for *n*-grams on the word level). Therefore, the exact data sets previous studies extracted from the corpora cannot be directly compared to the ones used in this paper.

Table 2. Best reported accuracy results so far for the GA and GB text corpora.

Approach	Features	GA	GB
Stamatatos *et al*. 2000	NLP-based	72%	70%
Peng *et al*. 2003	Character *n*-grams	74%	90%
Keselj *et al*. 2003	Character *n*-grams	85%	97%
Peng *et al*. 2004	Word *n*-grams	-	96%

To set a fairer baseline for evaluating the ensemble model, a SVM classifier was trained based on the frequencies of occurrence of the 1,000 most frequent words of the training corpus. To that end, the popular Weka toolbox of machine learning algorithms[28] was used with default parameter values. A similar approach based on word-form frequencies and a SVM classifier was followed by Diederich *et al*.[29] and tested to a German corpus with remarkable results. Actually, the performance of the SVM model is competitive in comparison with the best reported results for these text corpora, since it achieved accuracy of 86% and 96% for GA and GB, respectively. First, this indicates that the word frequency vector is quite effective for representing an author's style. Second, this constitutes another supporting case that SVMs are indeed a suitable algorithm for dealing with a high dimensional feature space and sparse data.

3.3. *Two Simple Ensembles*

Let's first examine two simple models for constructing ensembles by feature set subspacing:

- *k-Random Classifiers* (*k*RC): One subset $W_{m:n}$ of $\mathbf{W}_n$ is randomly selected without replacement and a new base learner is constructed. This process is repeated *k* times. The *k* resulting base classifiers are combined according to a predefined combination method. In this model, each individual feature is used at random (it may be included either in many subsets or none).

- *Exhaustive Disjoint Subspacing* (EDS): The feature set $\mathbf{W}_n$ is randomly divided into equally-sized disjoint subsets of size *m*. Each subset is used to build a base learner. The base classifiers are combined according to a predefined combination method. In this model, each individual feature is used exactly once and *n*/*m* (integer division) distinct base classifiers are built.

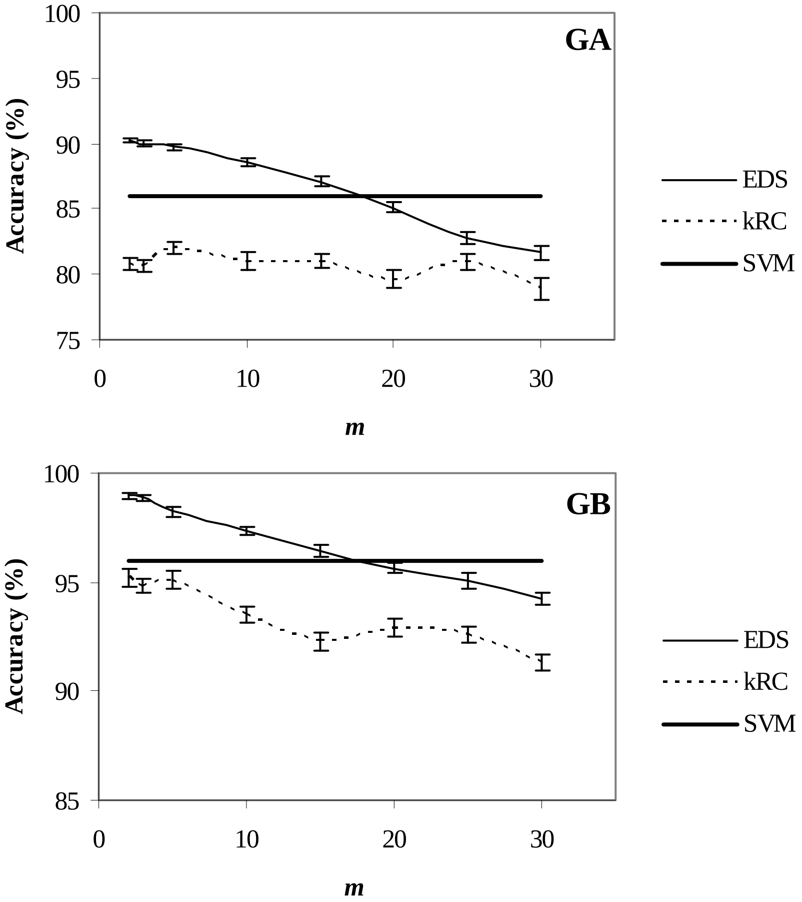

Figure 1. Classification accuracy results on GA (up) and GB (down) for EDS and *k*RC ensembles (*n*=1000) with different values of feature subset size (*m*). All measures averaged over 50 tries with standard error bars. SVM performance is also indicated for comparison purposes.

These two models were tested on both GA and GB for *n*=1,000 (i.e., using 1,000 most frequent words of the training corpus to represent the texts) and different feature subset sizes (*m*=2,3,5,10,15,20,25, and 30). In each case, equal number of base classifiers was used for the two methods (i.e., *k*=*n*/*m*). Figure 1 shows the average ensemble

Table 3. Classification effectiveness measures for GB (*n*=1,000). Accuracy (%) and Mean Reciprocal Ratio of the base classifiers, diversity (entropy) and accuracy (%) of the *k*RC and EDS ensembles (all measures averaged over 50 tries).

m	Base classifiers		*k*RC Ensemble		EDS Ensemble	
	Acc.	MRR	Div.	Acc.	Div.	Acc.
2	16	0.38	0.97	94	0.97	99
3	18	0.38	0.98	95	0.98	99
5	22	0.39	0.96	94	0.97	98
10	29	0.45	0.93	94	0.93	97
15	34	0.49	0.89	94	0.89	96
20	38	0.52	0.85	94	0.86	96
25	41	0.54	0.82	93	0.83	95
30	43	0.56	0.78	93	0.80	94

classification accuracy (% of correctly classified test texts) of *k*RC and EDS ensembles over 50 tries on GA and GB. Standard error bars are also depicted. Surprisingly, ensembles based on small feature subsets (i.e., low *m*) achieve the best performance for both data sets. Indeed, the performance of the ensembles with *m*=2 (GA: 90 and GB: 99%, on average) is better from the best reported results for these text corpora (see Table 2) and the SVM baseline. Moreover, the lower the *m* values, the lower the standard error for EDS. The difference in performance between EDS and *k*RC is statistical significant (*p*=0.01) in all cases.

A more detailed look at the produced ensembles is given in Table 3, where the average accuracy and MRR of the base classifiers and the diversity among their predictions for the EDS and *k*RC ensembles (on GB) are shown. As can been seen, there is a trade-off between base classifier accuracy and ensemble diversity. Ensembles based on low *m* have base classifiers of very low accuracy (recall that random guessing provides accuracy of 10%). However, their diversity is quite high (actually, for *m*=2 or 3 entropy reaches 1, which means almost randomized error), hence the combination of these poor individual classifiers leads to an accurate ensemble. Notice that the MRR is also reduced with *m* but to a lower extent in comparison with accuracy. In other words, the true class is ranked in good positions by the base classifiers when they fail to guess it. A significant remark, therefore, is that best ensemble accuracy results are achieved when a large feature set is divided into many small disjoint subsets rather than a few bigger subsets that correspond to better individual base classifiers.

Clearly, EDS outperforms SVM for small feature subset size (*m*<20) and *k*RC in all cases. The latter is not adequately explained by the results of Table 3. The base classifiers used to construct both EDS and *k*RC seem to have the same properties. There is no statistically significant difference in the diversity of the two methods (although EDS diversity tends to be slightly higher). Therefore, the key factor is that EDS ensembles are based on all available features and *k*RC on a subset of the entire feature set (with random selection). Actually, *k*RC ensembles are based on 64% (on average) of the entire feature set. In other words, 1 out of 3 features is not used at all in a *k*RC ensemble (recall that the number of base learners was set equal for both methods). Therefore, the richer the feature set, the more accurate the AA ensemble. This confirms the conclusion drawn by other researchers that all words are important for text categorization tasks[18].

To further illustrate the difference between *k*RC and EDS, let's increase the number of base classifiers that constitute the *k*RC ensemble for m=2 while maintaining the same for the EDS ensemble. In particular, if we use double base classifiers for *k*RC (that is, k = 2*n/m)* and check the performance over 50 tries, the classification accuracy is 86% for GA and 98% for GB, that is, still lower than that of the corresponding EDS model (90% and 99%, respectively). Despite doubling k in the *k*RC ensemble, 87% (in average) of the features were used at least once in the base classifiers. That is, a significant (but smaller than before) part of the feature set was not taken into account.

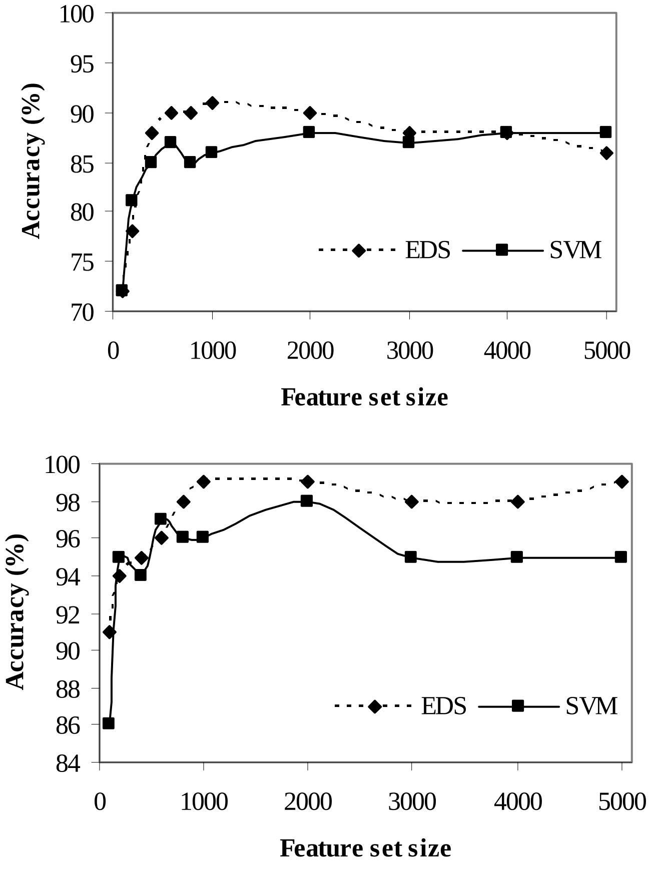

Figure 2. Classification accuracy for the EDS ensemble (m=2) and the SVM classifier for various feature set sizes on GA (up) and GB (down).

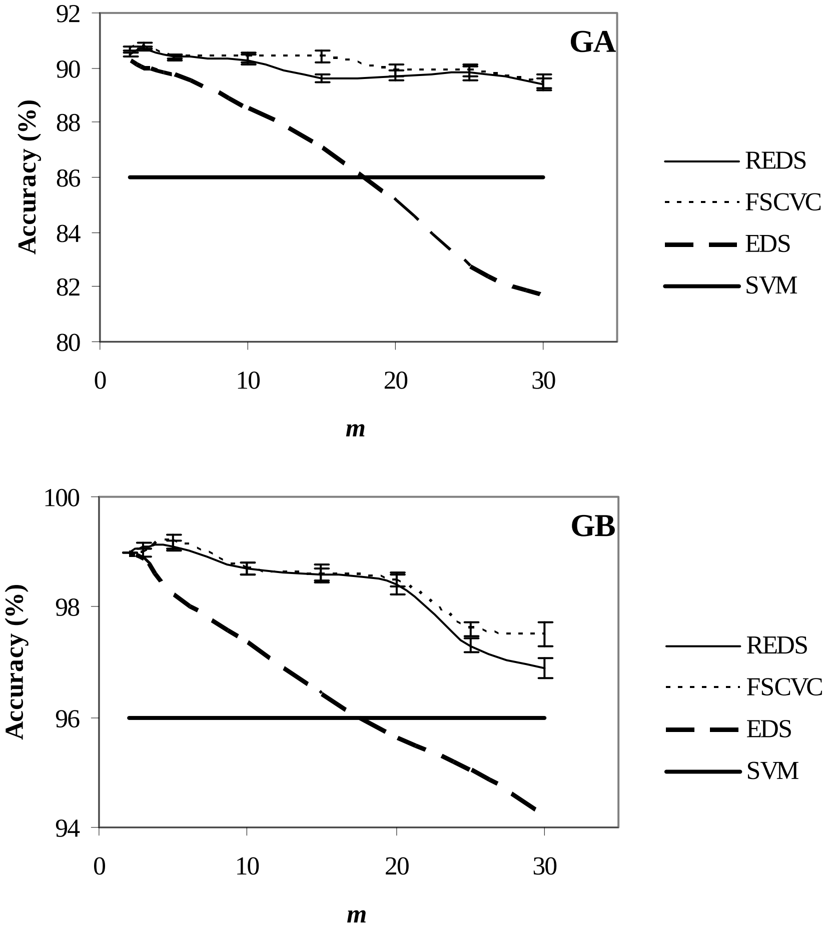

Figure 3. Classification accuracy (averaged over 50 tries plus standard error bars) of the enhanced models REDS and FSCVC on GA (up) and GB (down) for different feature subset size (*m*) and *n*=1,000. For comparison purposes, the performance of the simple EDS model as well as the SVM classifier are also depicted.

So far, the size of the feature set was selected arbitrarily at 1,000 words. What happens in case we use more (or less) words to represent the style of the authors? This is depicted in Figure 2. Here, the performance of the SVM and the EDS model is shown for different values of *n* (*m*=2). As can be seen, for both models, the performance is not notably improved when more words are added to the feature set. It seems that 1,000 words are adequate to capture the differences between the authors.

3.4. *Enhancing the Model*

The simple EDS ensemble can be easily enhanced by building a *stacked* model. In this section the following methods will be examined:

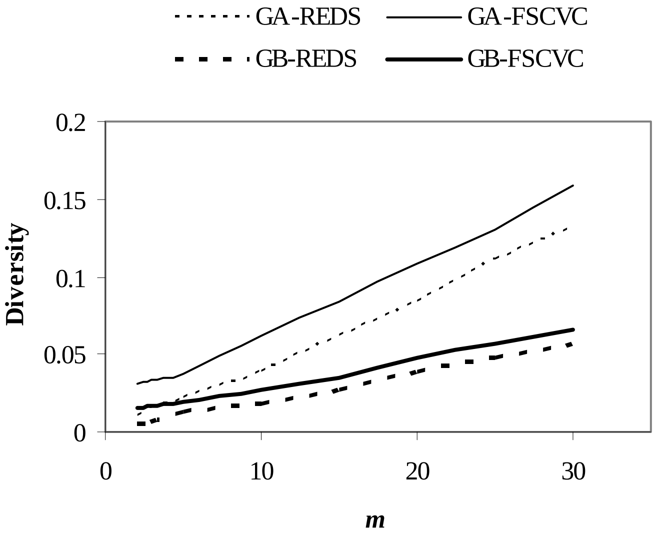

Figure 4. Diversity (in terms of entropy) of the enhanced models REDS and FSCVC on GA and GB.

- *Repeated Exhaustive Disjoint Subspacing* (REDS): A random subspacing of the entire feature set is conducted and a new EDS ensemble is constructed. This process is repeated *k* times. At the end of the iterations, the resulting EDS ensembles are combined (by *mp*) to provide the most likely class. All the features are used exactly *k* times and *k*(*n*/*m*) different base classifiers are constructed in total.
- *Feature Set Cross-Validated Committees* (FSCVC): Cross-validated committees is an ensemble construction method based on manipulating the training set[30]. However, here we apply the main idea of this method to the manipulation of the feature set. In particular, the feature set (the *n* most frequent words of the training set) is randomly divided into *k* disjoint subsets and *k* overlapping feature subsets are constructed by dropping out a different one of these *k* subsets. Then, each feature subset, containing *n*(*k*-1)/*k* features, is used to construct a simple EDS ensemble. Finally, the *k* resulting EDS ensembles are combined (by *mp*) to provide the most likely class. Using this method, all the features are used exactly *k*-1 times.

Essentially, the enhanced models are stacked classifiers, where the first level consists of individual EDS ensembles. Comparative results for REDS and FSCVC (for *n*=1,000) on GA and GB are given in Figure 3. In more detail, in both cases *k* was set to 10 (i.e., 10 EDS ensembles constitute each enhanced model) and 50 different tries were conducted. To illustrate the differences, the performance of the simple EDS model and the SVM approach are also depicted. Clearly, the enhanced models outperform simple EDS and SVM in all cases. The improvement is more remarkable for high values of feature subset size (*m*). However, as with the simple models, the best results are given by using low values of *m* (i.e, yielding the highest possible number of base classifiers).

Although FSCVC tends to outperform REDS in most cases, the difference between these models is not statistically significant (*p*=0.01). However, a more detailed look can confirm the superiority of FSCVC. Figure 4 depicts the diversity of the enhanced models (in terms of entropy between the predictions of the constituent EDS ensembles) for both GA and GB. At first, notice that the diversity is much more lower than that of Table 3, since now we deal with much more accurate constituent base learners (i.e., EDS ensembles). In addition, the difference in diversity between GA and GB can also be explained since more accurate classifiers are constructed for GB, therefore less diversity will exist among their predictions. The most important point is that there is a considerable difference in diversity between REDS and FSCVC for each data set. Recall that FSCVC uses a different subset of the initial feature set to construct each constituent EDS ensemble ensuring the diversity among their predictions. Since FSCVC is based on a more diverse set of classifiers is more likely to provide a better classification model in comparison with REDS.

4. Conclusions

Machine learning techniques can be directly applied to AA, producing effective systems that can assist criminal identification and cybercrime analysis. To that end, both feature set subspacing ensembles and SVMs provide reliable solutions able to deal with high-dimensional feature spaces and sparse data. Low-level information, such as word frequencies can adequately represent an author's style. Recall that the method of using the most frequent words of the training corpus as style markers is based on a language-independent procedure. Comparative results with previous work based on NLP-based features[13] indicate that a high dimensional word-based feature set in combination with an appropriate classification scheme is a more effective approach to AA. As indicated by the performance of the simple word-based SVM model, the contribution of the lexical measures plays the most important role while the ensemble-based approach optimizes the results. However, by definition, the NLP-based features capture different type of stylistic information and could be used as additional features to a word-based feature set[31].

An alternative solution, already examined by some researchers[25, 26] make use of sub-word information, such as character-level *n*-gram frequencies. Although *n*-grams are able to capture subtle information on the lexical, syntactical, and structural level, they considerably increase the dimensionality of the problem in comparison with the word-based approach. However, recent remarkable results of a character *n*-gram-based method in a competition of AA approaches[32] indicate that this representation is suitable for the AA task. Note also that a preliminary version of the presented approach was entered in this competition (a REDS ensemble with feature set size=200, combination= *product*, and *m*=2).

Simple but effective ensembles based on feature set subspacing can be constructed using the exhaustive disjoint subspacing technique. Although such ensembles are based on poor constituent classifiers, their combination provides a diverse set of answers in which the true class is more likely to outperform the others. Previous studies have also shown that diversity alone can be used as a guide for constructing good ensembles[33]. The approach followed in this study ensures an extremely high level of diversity. To that end, the combination method of the base classifiers plays an important role. Moreover, a stacked model enhances significantly the simple EDS ensemble.

A basic assumption considered in the construction of the tested ensemble models was that feature subsets should be of equal size. However, this consideration has no significant impact in the generality of the presented approach. Recall that minimal sized feature subsets proved to be the most effective solution. Additionally, this method can be applied to any number of candidate classes. Another basic assumption is that features are grouped at random. An alternative approach would require a search procedure through the space of all the possible feature groupings based on an evaluation function, e.g., taking account of base classifier accuracy and diversity of the resulting ensemble. However, such measures indicate indirectly the effectiveness of an ensemble (i.e., the ensemble maximizing the evaluation function is not the best solution necessarily). To this end, random subspacing could serve as the initial state, an approach followed by Opitz and Shavlik[20]. More significantly, for large values of feature set size and feature subset size the set of possible feature groupings grow exponentially. Training time cost would increase significantly as well.

As concerns the task of attributing authorship, there are still open questions. In particular, limited text length can affect the performance of the model. Many researchers indicate that 1,000 words is a good lower limit for text length in order to obtain reliable results. However, this restriction does not hold for many practical applications (e.g., email messages). In fact, many of the texts used in the presented experiments were pretty shorter than that, especially for the GA corpus. Another problem may arise in case of an imbalanced training set, that is, when the training set is unequally distributed over the authors. Moreover, open-class problems, that is, the true author is not included in the candidate authors, should be thoroughly tested as well. Last but not least, it would be useful to transform the quantified differences between the authors into human readable terms. From a literary point of view, this could assist the procedure of objectively defining the style of an author.

References

1. F. Mosteller and D. Wallace. *Applied Bayesian and Classical Inference: The Case of the Federalist Papers,* (Addison-Wesley: Reading, MA, 1984).
2. C. Labbé and D. Labbé, Inter-textual distance and authorship attribution: Corneille and Molière, *Journal of Quantitative Linguistics*, 8 (2001) 213-31.
3. O. de Vel, A. Anderson, M. Corney, and G.M. Mohay, Mining e-mail content for author identification forensics, *SIGMOD Record*, 30(4) (2001) 55-64.
4. S. Argamon, M. Saric, and S. Stein, Style mining of electronic messages for multiple authorship discrimination: First results, in *Proc. of the 9th ACM SIGKDD* (2003) 475-480.
5. A. Abbasi and H. Chen, Applying authorship analysis to extremistgGroup web forum messages, *IEEE Intelligent Systems*, 20(5) (2005) 67-75.

6. H. van Halteren, Linguistic profiling for author recognition and verification, in *Proc. of the 42nd Annual Meeting of the Association for Computational Linguistics* (2004) 199-206.
7. C. Chaski, Empirical evaluations of language-based author identification techniques, *Forensic Linguistics*, 8(1) (2001) 1-65.
8. D. Holmes, The evolution of stylometry in humanities scholarship, *Literary and Linguistic Computing*, 13(3) (1998) 111-117.
9. J. Rudman, The state of authorship attribution studies: Some problems and solutions, *Computers and the Humanities*, 31 (1998), 351-365.
10. A.Q. Morton, The authorship of Greek prose, *Journal of the Royal Statistical Society*, Series A, 128 (1965) 169–233.
11. H. Sichel, On a distribution law for word frequencies, *Journal of the American Statistical Association*, 70 (1975) 542–547.
12. H. Baayen, H. Van Halteren, and F. Tweedie, Outside the cave of shadows: Using syntactic annotation to enhance authorship attribution*.* *Literary and Linguistic Computing*, 11(3) (1996) 121–131.
13. E. Stamatatos, N. Fakotakis, and G. Kokkinakis, Automatic text categorization in terms of genre and author, *Computational Linguistics*, 26(4) (2000) 471-495.
14. M. Koppel, and J. Schler, Exploiting stylistic idiosyncrasies for authorship attribution, in *Proc. of the IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis* (2003).
15. J.F. Burrows, Word patterns and story shapes: The statistical analysis of narrative style, *Literary and Linguistic Computing*, 2(1987), 61–70.
16. F. Sebastiani, Machine learning in automated text categorization, *ACM Computing Surveys*, 34(1) (2002) 1-47.
17. R. Kohavi and G. John, Wrappers for feature subset selection, *Artificial Intelligence*, 97 (1997) 273-324.
18. T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in *Proc. of the European Conference on Machine Learning*, (1998).
19. V. Vapnik, *The nature of statistical learning theory* (Springer, New York, 1995).
20. D. Opitz and J. Shavlik, A genetic algorithm approach for creating neural network ensembles, in *Combining Artificial Neural Nets*, A. Sharkley (ed.) (1999) pp. 79-99.
21. L. Breiman, Random forests, *Machine Learning* 45(1) (2001) 5-32.
22. T. Lim, W. Loh, and Y. Shih, A comparison of prediction accuracy, complexity and training time of thirty-three old and new classification accuracy, *Machine Learning*, 40(3) (2000) 203-228.
23. D. Tax, M. van Breukelen, R. Duin, and J. Kittler, Combining multiple classifiers by averaging or by multiplying?, *Pattern Recognition*, 33 (2000) 1475-1485.
24. L. Kuncheva and C. Whitaker, Measures of diversity in classifier ensembles, *Machine Learning*, 51 (2003) 181-207.
25. F. Peng, D. Shuurmans, V. Keselj, and S. Wang, Language independent authorship attribution using character level language models, in *Proc. of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, (2003).
26. V. Keselj, F. Peng, N. Cercone, and C. Thomas, N-gram-based author profiles for authorship attribution, in *Proc. of the Conference of the Pacific Association for Computational Linguistics*, (2003).
27. F. Peng, D. Shuurmans, and S. Wang. Augmenting naive bayes classifiers with statistical language models, *Information Retrieval Journal*, 7(1) (2004) 317-345.
28. I.H. Witten and E. Frank, *Data mining: Practical machine learning tools with Java implementations* (Morgan Kaufmann, San Francisco, 2000).

29. J. Diederich, J. Kindermann, E. Leopold, and G. Paass, Authorship attribution with support vector machines, *Applied intelligence*, 19(1/2) (2003) 109-123.
30. B. Parmanto, P.W. Munro, and H.R. Doyle, Improving committee diagnosis by resampling techniques, in *Advances in Neural Information Processing Systems*, Touretzky, D.S, Mozer M.C., Heselmo, M.E. (eds), 8 (1996) 882-888.
31. E. Stamatatos, N. Fakotakis, and G. Kokkinakis, Computer-based authorship attribution without lexical measures, *Computers and the Humanities*, 35(2) (2001) 193-214.
32. P. Juola, Ad-hoc authorship attribution competition, in *Proc. of the Joint Int. Conference ALLC/ACH* (2004) 175-176.
33. G. Zenobi, and P. Cunningham, Using diversity in preparing ensembles of classifiers based on different feature subsets to minimize generalization error, in *Proc. of 12th European Conference on Machine Learning* (2001) 576-587.