International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems
© World Scientific Publishing Company

Competitive author profiling using compression-based strategies*

FRANCISCO CLAUDE

Universidad Diego Portales, Escuela de Informática y Telecomunicaciones
Santiago, Chile
Sudo Technologies Inc., Menlo Park, California, USA
fclaude@recoded.cl

DANIIL GALAKTIONOV

Universidade da Coruña, Database Laboratory
Elviña, 15071, A Coruña, Spain
d.galaktionov@udc.es

ROBERTO KONOW

Universidad Diego Portales, Escuela de Informática y Telecomunicaciones
Santiago, Chile
eBay Inc., San Jose, California, USA
roberto.konow@mail.udp.cl, rkonow@ebay.com

SUSANA LADRA†

Universidade da Coruña, Database Laboratory
Elviña, 15071, A Coruña, Spain
sladra@udc.es

ÓSCAR PEDREIRA

Universidade da Coruña, Database Laboratory
Elviña, 15071, A Coruña, Spain
opedreira@udc.es

Received (received date)
Revised (revised date)

Author profiling consists in determining some demographic attributes -such as gender, age, nationality, language, religion, and others- of an author for a given document. This task, which has applications in fields such as forensics, security, or marketing, has been approached from different areas, especially from linguistics and natural language processing, by extracting different types of features from training documents, usually content- and style-based features.

*A preliminary version of this work appeared in *Proc. 4th Spanish Conference on Information Retrieval (CERI)*, ACM, 2016, Article No.4.
†Corresponding author.

In this paper we address the problem by using several compression-inspired strategies that generate different models without analyzing or extracting specific features from the textual content, making them style-oblivious approaches. We analyze the behavior of these techniques, combine them and compare them with other state-of-the-art methods. We show that they can be competitive in terms of accuracy, giving the best predictions for some domains, and they are efficient in time performance.

Keywords: author profiling; compression-based classification; age prediction; gender prediction.

1. Introduction

The huge amount of digitally published information and the multiple industrial applications of text mining have attracted a significant research effort to this area. One of the problems considered in text mining is that of processing documents to obtain information about the authors. In this paper we focus on the particular text mining problem known as *author profiling*, that is, automatically determining different attributes of the author of a document, such as gender, age, nationality, language, religion, among others.

Author profiling is relevant for both research purposes and for industrial applications. Although different author attributes can be of interest depending on the application domain, determining the gender and age of an author, for example, can be useful in applications as targeted advertising, prevention of child grooming, or as a relevant information source for psychological studies.

This task has been the subject of the *PAN International Competition on Author Profiling*[a], which has already completed four editions, starting in 2013. The four editions considered gender and age prediction, and the 2015 edition considered personality prediction too. Besides having attracted the interest of tens of research groups that participated, these competitions have also attracted the interest of the industry, being sponsored by companies working on the domains of forensic linguistics, social network analysis, semantic analysis, and text analytics.

The problem of author profiling has been addressed by researchers of different areas, but specially from the community of linguistics and natural language processing. The task is usually treated as a classification problem, since each attribute of interest can take a reduced number of values. For example, we are not supposed to guess the exact age of the author, but to frame it within a set of possible age ranges (10-20, 20-30, etc.). The main approach would consist in building a classification model from sets of features extracted from the content and/or style of the documents. These methods rely on the fact that the author's features, such as gender or age, can be determined by the use of different function words, part-of-speech, the existence of grammatical and orthographic errors, the morphological, syntactic and structural attributes, and other stylistic characteristics. For instance, male/female individuals of specific age may tend to use prepositions and determiners differently,

[a] `http://pan.webis.de`

or young people include emoticons and other common non-dictionary terms more frequently.[1] While these approaches have been popular and achieve good accuracy results, they are usually complex to implement and require huge amounts of computing resources.

In this paper, we present several compression-inspired techniques for the author profiling task. That is, instead of basing our classification of the documents on elements identified with natural language processing, we use similarity features derived from the compression of the documents, such as term frequency, entropy, etc. We show that using a simple idea, easy to implement and deploy, it is in practice faster than most natural language processing techniques using complex machine learning classification models. Regarding the accuracy results, our approach is competitive when compared with the state-of-the-art for all the scenarios described in the PAN competition, achieving in some cases results among the best ones.

The compression-inspired author profiling strategies we propose build models M_i for each class C_i, and assign these classes to new documents using frequency-, entropy-, compression-inspired similarity measures or word-based statistical models. This last approach is very common in compression-based text classification. In order to build the models M_i, we use a training document collection. We then compress each document in the test collection using different models, and then determine the corresponding M_i that achieves the best compression. We then assign the class C_i to the corresponding best model. The motivation behind the compression-inspired classification is that, if a document belongs to a class C_i, then the model M_i is the best one describing its structure, and therefore obtains the best compression ratio.

One of the most important advantages of this approach is that compression-inspired methods do not require, in general, any previous knowledge of the linguistic properties of the corpus of documents over which they are applied. These techniques are of special interest for those domains where we do not have an *a priori* intuition of their properties, such as for DNA and protein sequences, stock market data, or medical monitoring.[2] However, they may not be as powerful as those techniques that exploit stylistic features when the task involves natural language text. Thus, the main interest of this work is to evaluate the overall performance of some compression-inspired classifiers, which are style-oblivious approaches, over social media text of different nature. We evaluate both the efficiency and accuracy for identifying the age, gender or personality traits of the authors that have written documents in different scenarios.

The rest of the paper is structured as follows. We start with a revision of the related work. Next, we propose some simple approaches based on compression features to tackle the author profiling task. We also propose an approach to combine these methods to obtain better results using machine learning techniques. We then include an experimental evaluation where we compare the results of the approaches described for age, gender, and personality prediction over texts of different nature, and analyze their behavior. We conclude with a summary and future lines of work.

2. Related Work

Compression methods and their related concepts have been used for tasks different than the compression itself. Compression-based language models and entropy-based measures have been already used for NLP tasks such as text classification,[3,4,5] spam filtering,[6] automatic language identification,[2,7] or authorship attribution.[2,8,9,10]

Teahan proposed several methods for text classification and text segmentation using the cross-entropy computed as a fixed order character-based Markov model adapted from the PPM[b] text compression algorithm.[3] While for general text classification the compression-inspired techniques have been usually outperformed by other methods, they have been proved to be more effective for spam filtering than traditional machine learning systems.[6] The compression-oriented technique that obtained the best results also combined the usage of Dynamic Markov Coding and PPM.

Benedetto et al. reviewed several applications where this kind of compression-inspired techniques could be useful.[2] In particular, they analyzed the potential use of these methods for tasks such as language recognition, sequence classification or authorship attribution. After this work, many other researchers have addressed those same tasks by using compression-inspired approaches. For instance, Ferragina et al. studied compression-inspired classification of biological sequence,[12] and Bergsma et al. analyzed the performance of PPM for automatic language identification.[7]

Compression-inspired techniques have been also used for the task of authorship attribution, which is a related task to the one addressed in this work. Authorship attribution consists in identifying the author of a given text. Thus, the task can be formulated as a typical classification problem, which depends on discriminant features to represent the style of an author. Pavelec et al. demonstrated that compression algorithms (more specifically, PPM) can be successfully used for author identification, obtaining similar results to those obtained using a classical pattern recognition framework, where linguistic features proposed by forensic experts are used to train a support vector machine classifier.[8] In the case of the task of author verification, which consists in, given a set of documents by a single author and a questioned document, determining whether the questioned document was written by that particular author or not, some compression-inspired techniques, also based in PPM, have been already evaluated, but obtaining no relevant results.[13]

To the best of the authors' knowledge, the applicability of compression-inspired techniques has not been studied for the specific task of author profiling. On the other hand, this problem has been approached using traditional machine learning techniques, such as exponential gradient,[14] support vector machines,[15] or linear regression;[1] and over different dataset, such as blogs[16] or tweets.[15,1] For instance,

[b] PPM (*Prediction by Partial Matching*) is a k-th order statistical compressor that achieves high compression by computing the frequency of the contexts of the last seen symbols and predicting the next symbol of the sequence.[11]

the best four participants in the Author Profiling Task at PAN 2014 used machine learning approaches, most of them considering stylistic features.[26] `lopezmonroy14` used libLINEAR with a second order representation based on relationships among terms, documents, profiles and subprofiles. `liau14` used logistic regression with a bag-of-words approach, including stylistic features such as the occurrence of emoticons or specific phrases such as *my husband* or *my wife*. `shrestha14` also used logistic regression and features such as the frequency of emoticons or different punctuation signs, but with an n-gram approach. `weren14` used different approaches depending on the corpus (logic boost, rotation forest, multi-class classifier, multilayer perceptron and simple logistic) with IR features, and also considering the correctness, cleanliness and diversity of the texts, readability features, and information included in HTML tags. We will compare our approaches with these methods in Section 5.5.

3. Proposed compression-inspired classifiers

In this section we present our compression-inspired author profiling classifiers, focusing on their intuitive idea. We call $\mathcal{T}$ the *training* set, and $\mathcal{V}$ the *test* set. Each element in these sets is a triple $p = (g, a, T = \{t_1, t_2, ...\})$, where T corresponds to a set of texts associated with one user with age range a and gender g. We visualize each text t_i as a sequence of tokens, or words, $(w_1, w_2, ...)$, and we call T_w the concatenation of all the tokens in T, in the order they appear in $T = \{t_1, t_2, ...\}$.

We describe all methods considering only age and gender features. We will also use these techniques for personality traits, by addressing it as a classification problem using classes associated with each possible value of personality trait.

3.1. *Word Counting*

Word counting is our simplest classifier. It works by remembering the average number of words a document has for each possible pair of gender and age, (g, a). Given a document, the classifier counts the number of different words and returns the pair (g, a) for which the average is closest. We also perform a normalization mechanism, in which we divide the amount of different words by the total number of words in all the documents given a training $\mathcal{T}$ input.

The idea behind this method, although simple, allows us to capture an interesting feature of younger writers, which is the use of a more limited vocabulary and repetition of words. In case of gender prediction, women usually show a more elaborated speech, using more words than men. This classifier is somehow similar to the one proposed later, called Entropy, but simplified as an average of different elements, ignoring their probabilities.

3.2. *Move-To-Front*

Given a sequence of symbols to compress, Move-To-Front is a transformation algorithm that tries to reduce its redundancy.[17] It maintains a list with the different elements of the sequence and encodes each symbol of the sequence by its position in the list. After it encodes the symbol, the list is rearranged by moving this element to the front of the list. With this rearrangement of the vocabulary list, the most recent elements have lower position indexes, thus, in case of redundancy, the algorithm outputs codes corresponding to small indexes.

Hence, we propose a classifier using this idea. We create a set associated to each gender and age (g, a) and for each $w_i \in T_w$ obtained from $\mathcal{T}$ we insert w_i to the corresponding list in a move-the-front fashion. The classification step is as follows: Given a document we compute the information required to encode the position of each word using the lists for all pairs (g, a), encoding the position p of each word in $\log_2(p)$ bits, and penalizing with $1 + \log_2 k$ words that do not appear in the list, where k is the number of words in the list. The pair (g, a) whose list encodes the input document with less amount of bits becomes the resulting pair.

The intuition behind this mechanism is that the (g, a) list will tend to an ordering of the most used words of documents that belong to that class. Therefore, the document that most likely follows the tendency should be considered as a member of the class (g, a).

3.3. *Entropy*

The zero-order empirical entropy of a text is a lower bound for a compressor that encodes each symbol independently of any of its preceding or following symbols. With this idea, we propose the following classifier. For each $p \in \mathcal{T}$ we compute the empirical entropy of T_w for this sample, this is, $H_p = \frac{1}{|T_w|} \sum_{w \in T_w} \#(w) \log_2 \left(\frac{|T_w|}{\#(w)} \right)$, where $\#(w)$ corresponds to the number of times w appears in T_w. Finally, we average the entropies obtained for each (g, a) pair. In order to classify an element $p \in \mathcal{T}$, we compute H_p, and then produce the class whose entropy is closer to H_p.

The main intuition behind this classifier is that more descriptive texts will tend to have a higher entropy, since they need more vocabulary, and also, repetitive texts will tend to have lower entropy. This can help detecting the age of a user because of their use of the language, and the gender because in general women tend to describe things better than men.

3.4. *Compressor*

For each possible gender, and for each possible age range, we generate a simple compression model based on the elements of $\mathcal{T}$. Given a gender g, we generate a sequence S that concatenates all lists T_w for each $p \in \mathcal{T}$ such that $p = (g, X, T_w)$. Then we count how many times each word w appears in S and we create an array A of unique words sorted by the number of occurrences in S. This is our model for g, and we repeat this for each possible gender and age.

Given an element $p \in \mathcal{T}$, we compute the size of compressing T_w using each array A generated. The size of compressing with an array A is $C = \sum_{w \in T_w} \log_2(pos_A(w))$, where $pos_A(w)$ is the position in A where w appears, or $|A| + 1$ if w does not appear in A. We pick the class for which we obtain the smaller C. Notice that this classifier corresponds to using the same model as a zero-order word-based semi-static statistical compressor, such as in Huffman coding[18], but simplifying the encoding scheme. We encode a source symbol just by using the exact number of bits required by the binary representation of its position in the frequency-sorted vocabulary.

The intuition behind this classifier is that texts written by a similar group should be more compressible (or predictable) for someone that knows how this group writes, and the compression model serves this role.

4. Combining techniques using SVM

All the previously described classifiers may be combined in an attempt to achieve a better accuracy. In this work we explored the possibility of using a Support Vector Machine (SVM),[19] a machine learning technique that was devised as a binary classifier that is trained with a set of examples from two different classes. For multiple classes, the common approach is to train multiple binary classifiers and combining them in either a one-vs-one or a one-vs-all strategy.[20] The classification itself is achieved by looking for a hyperplane that best separates the samples in both classes. When the samples are not linearly separable, they must be transformed to linear space. The most common way to achieve it is known as the kernel method.[21]

We define a sample for each $p \in \mathcal{T}$ with $n_a + 5$ features, being n_a the number of age ranges. The features are extracted from the following classifiers:

- **Word Counting**: We use both the average number of different words per document in $t_i \in p$ and the normalized version (where we divide the number of different words by the total number of words).
- **Entropy**: We use the average entropy for every document in $t_i \in p$.
- **Compressor**: After generating the A array for every gender and age for all elements in $\mathcal{T}$, we calculate the compressed size C from every class model over T_w and then normalize each size as $C/|T_w|$, emulating a compression ratio per word. This produces n_a features for every possible age range plus two for both genders.

Two SVM classifiers are trained, one for genders and one for ages. Both use the Word Counting and Entropy features. However, the gender classifier uses the two genders compression ratios, while the age classifier uses the n_a age ratios.

5. Experimental Results

5.1. *Experimental Framework*

We use the TIRA experimentation platform, which provides a service to handle software submissions for PAN competitions.[22,23] This allows the access to the test datasets used in previous PAN competitions. We deploy our software on a virtual machine with the same characteristics as the ones used in the competition, thus, we can also compare execution times directly with those obtained by other participants. It is accessed using a web interface that allows the participants to execute and test their algorithms in a remote fashion.[24]

The virtual machine used dedicated server with one processor Intel® Xeon® E5-2620 v2 @ 2.10GHz, 4 GB of RAM. The operating system was Ubuntu 16.04 with kernel 4.4.0-62 (64 bits). We implemented the proposed classifiers methods using python 2.7.12. The source code for reproducibility is available at `https://github.com/fclaude/pan`.

We will refer to the word counting classifier as `WC`, the normalized word count version as `NWC`, move-to-front as `MTF`, entropy as `Entr`, the compressor classifier as `Comp` and the one using support vector machines as `SVM`. We used the scikit-learn SVM implementation[c], with an RBF kernel and the C parameter at its default value of 1.0.[25] The one-vs-one strategy was used for the age classification. It is not needed for gender classification, as there are only two genders. All the features extracted from $\mathcal{T}$ were standardized to have zero mean and unit variance, while the features from $\mathcal{V}$ were scaled with the same mean and variance from $\mathcal{T}$.

5.2. *Datasets*

For the experimental evaluation we use the dataset from the 2nd Author Profiling Task at PAN 2014,[26] the 3rd Author Profiling Task at PAN 2015,[27] and the 4th Author Profiling Task at PAN 2016.[28] These datasets are composed of several corpora from different scenarios and languages:

- **PAN 2014 dataset** is composed of four different domains (Blogs, Social Media, Hotel Reviews, and Twitter), each of them in two different languages (English and Spanish), except for the corpus of reviews (only English). The documents from the dataset contain the gender (female or male) and the age ranges (18–24, 25–34, 35–49, 50–64, and $\geq$65) of its author. A complete description of this dataset is included in the PAN 2014 overview.[26]
- **PAN 2015 dataset** contains corpora only from one domain (Twitter), but in four different languages (English, Spanish, Italian and Dutch). The documents from the dataset contain the gender (female or male), the age ranges (18–24, 25–34, 35–49, and $\geq$50), and five personality traits (extroversion, emotional stability/neuroticism, agreeableness, conscientiousness, and openness) of its author. These personality traits were self-assessed and included in the dataset as scores normalized between -0.5 and +0.5. A complete description of this dataset is included in the PAN 2015 overview.[27]
- **PAN 2016 dataset** was designed from cross-genre perspective. The training dataset consists of Twitter corpora in three different languages (English, Spanish, and Dutch), and the test dataset contains documents from other

[c] `http://scikit-learn.org/stable/modules/svm.html`

Table 1. Age accuracy for the compression-inspired approaches over the training dataset of the Author Profiling Task at PAN 2014. We mark in bold font the best result.

	Blogs		Twitter		Social Media		Reviews
	EN	ES	EN	ES	EN	ES	EN
WC	33.50	36.83	31.02	23.68	18.93	24.51	16.17
NWC	15.00	26.83	20.66	23.40	13.30	26.58	22.04
Entr	24.67	30.32	37.95	27.01	23.26	19.71	21.68
MTF	23.33	34.29	37.40	27.50	23.60	19.96	21.44
Comp	**40.17**	47.62	39.73	44.38	35.46	40.11	30.94
SVM	40.00	**50.16**	**41.40**	**46.77**	**35.86**	**40.36**	**31.78**

domains: Social Media and Blogs in English and Spanish languages, and Reviews in Dutch language. The documents contain the gender (female or male) and the age ranges (18–24, 25–34, 35–49, 50–64, and $\geq$65) of its author (except for the Dutch corpora, which only contain gender information). A complete description is included in the PAN 2016 overview.[28]

5.3. *Performance measures*

We use accuracy for evaluating the performance when predicting age and gender. More specifically, the accuracy is calculated as the ratio between the number of authors correctly predicted by the total number of authors. The accuracy is calculated separately for each subcorpus, language, gender, and age class. In addition, the combined accuracy for the joint identification of age and gender is also computed.

We use the Root Mean Square Error (RMSE) for evaluating the personality recognition. Note that in this case the measure indicates the distance of the predicted value with the true value, so the lower the scorer, the better is the method.

We also compute the total time needed to process the test data. TIRA platform shows the elapsed time for the testing process for each method, that is, the time taken from start of the program to the end.

5.4. *Comparison of the proposed methods*

We first compare our methods in terms of accuracy when predicting age and gender over the training dataset of the Author Profiling Task at PAN 2014 competition. For this purpose, we use a k-fold cross-validation method with $k = 10$ and report the average of the results.

Tables 1 and 2 show the accuracy obtained when predicting age and gender, respectively, using the different approaches presented in this paper. We can observe that, in average, SVM and Comp obtain in general the best results. They are also the most complex strategies from the ones presented in this paper, as they create a more elaborated model for each class. It is worth noting the remarkable results obtained by WC for predicting the gender of the authors of the Spanish Blogs and Twitter, as it beats the rest of the strategies by only counting the number of different words for

Table 2. Gender accuracy for the compression-inspired approaches over the training dataset of the Author Profiling Task at PAN 2014.

	Blogs		Twitter		Social Media		Reviews
	EN	ES	EN	ES	EN	ES	EN
`WC`	57.17	**68.57**	62.05	**65.63**	51.13	53.81	50.55
`NWC`	61.33	64.13	52.00	57.70	51.09	52.61	49.32
`Entr`	51.00	67.14	49.88	53.33	50.70	54.29	48.39
`MTF`	50.33	68.25	50.90	54.38	50.78	53.82	49.21
`Comp`	71.83	62.38	65.42	65.07	53.89	**61.74**	**64.52**
`SVM`	**72.50**	59.84	**68.84**	64.58	**54.29**	60.94	63.08

each gender class. It also obtains good results for the rest of the datasets, achieving generally the third best result. This classifier is taking advantage of the intuition already mentioned in Section 3.1, as in these datasets female authors use almost two times the number of different words than male authors. This is not happening for the rest of the corpora. For example, the number of different words in the Reviews corpora is almost identical for both genders. For some datasets `NWC` outperforms `WC`. They perform poorly when two or more classes have the same number of different words, as it happens for some age classes over English social media. `MTF` and `Entr` obtain their best results for predicting age and gender of authors of Spanish blogs.

5.5. *Performance in the Author Profiling Task at PAN 2014*

To compare our approaches with the techniques of the state of the art, we evaluate our best compression-based strategies over the test dataset of the Author Profiling Task at PAN 2014. Tables 3 and 4 show the results obtained for predicting age and gender, respectively, using `WC`, `Comp`, and `SVM`. In addition, we included a variant of `Comp`, denoted as `Comp n-grams`, where instead of words, we use n-grams, more concretely, 3-grams for tweets and reviews, 5-grams for English blogs and social media in both languages; and 7-grams for Spanish blogs. These configurations obtained the best result among all possible n-grams for each subcorpus. We also include the results of the best four participants of PAN 2014 competition.[26]

We can observe than `Comp` is the preferred choice to predict gender, as it obtains the best result for 3 out of the 7 subcorpora. In addition, it also beats the rest of the methods for predicting the age of Spanish blogs. For the rest of the subcorpora, it also obtains remarkable results. `SVM` obtains the best result for predicting the gender of English tweets, and the second best result for English blogs. It generally predicts the age better than `Comp`, obtaining three times the second best result. Using n-grams instead of words works well for predicting the age in English tweets and predicting gender in English tweets, Spanish blogs and reviews. For this test dataset, `WC` does not obtain good results.

Table 5 shows the result of the joint accuracy for predicting both age and gender for each subcorpus, and the average accuracy for each method, which is the score used to rank the participants at the competition. We can observe that `Comp`

Table 3. Age accuracy over the test dataset of the Author Profiling Task at PAN 2014. We include the results of the best four participants of the competition.

	Blogs		Twitter		Social Media		Reviews
	EN	ES	EN	ES	EN	ES	EN
`WC`	29.49	14.29	22.08	15.56	27.90	18.55	16.02
`Comp`	44.87	**48.21**	49.35	53.33	35.84	42.93	30.51
`Comp n-grams`	43.59	44.64	**50.65**	53.33	35.60	41.52	28.38
`SVM`	39.74	46.43	50.00	56.67	35.55	43.29	33.56
`lopezmonroy14`	39.74	**48.21**	49.35	53.33	35.52	45.23	33.37
`liau14`	34.62	44.64	**50.65**	50.00	36.05	**48.94**	**35.02**
`shrestha14`	38.46	46.43	44.16	**61.11**	**36.52**	42.76	33.31
`weren14`	**46.15**	25.00	33.12	52.22	34.89	43.82	33.43

Table 4. Gender accuracy over the test dataset of the Author Profiling Task at PAN 2014. We include the results of the best four participants of the competition.

	Blogs		Twitter		Social Media		Reviews
	EN	ES	EN	ES	EN	ES	EN
`WC`	50.00	46.43	48.70	42.22	47.51	51.24	49.63
`Comp`	**71.79**	51.79	70.78	**71.11**	**54.41**	59.72	66.81
`Comp n-grams`	70.51	55.36	72.08	65.56	54.03	58.83	67.05
`SVM`	67.95	48.21	**74.03**	58.89	53.02	61.31	64.49
`lopezmonroy14`	67.95	**58.93**	72.08	60.00	52.37	64.84	68.09
`liau14`	65.38	50.00	73.38	63.33	53.85	**68.37**	**72.59**
`shrestha14`	57.69	42.86	66.88	65.56	53.82	64.49	66.87
`weren14`	64.10	53.57	57.14	53.33	53.61	63.07	67.78

obtains the best joint accuracy for English blogs, and also the second best average joint accuracy thanks to its good results for all the subcorpora. Thus, `Comp` would rank second in the Author Profiling Task at PAN 2014 competition. Moreover, if we select the best compression-based strategy for each subcorpus, using `SVM` for Spanish social media content and English reviews, and `Comp n-grams` for Spanish blogs and English tweets, we would obtain an average joint accuracy higher than the one obtained by the winner of the competition.

Compression-inspired text classification methods have generally shown slower performances than techniques based on other approaches.[29] Time results in Table 6 show that the approaches proposed in this paper have been proved to be competitive in terms of time performance, being comparable to those obtained by the winner (`lopezmonroy14`), and only clearly beaten by the second classified in the competition (`liau14`). Moreover, this paper has been focused as a proof of concept on the efficacy of compression-based approaches for author profiling, and not on efficiency. For example, we use a generic XML parser in the evaluation phase that consumes most of the reported runtime. Thus, we believe that it is possible to obtain much better results with a more specific implementation of the methods, using a more efficient language, such as C or C++, and a specific parser for each subcorpus.

Table 5. Joint accuracy over the test dataset of the Author Profiling Task at PAN 2014. We include the results of the best four participants of the competition.

	Blogs		Twitter		Social Media		Reviews	
	EN	ES	EN	ES	EN	ES	EN	avg
`WC`	14.10	5.36	11.04	6.67	13.66	8.30	8.10	9.60
`Comp`	**32.05**	26.79	33.12	41.11	19.99	25.27	20.46	28.40
`Comp n-grams`	26.92	28.57	34.42	40.00	19.91	23.85	19.98	27.66
`SVM`	26.92	19.64	33.12	37.78	18.90	27.92	21.25	27.93
`lopezmonroy14`	30.77	**32.14**	**35.71**	34.44	19.02	28.09	22.47	28.95
`liau14`	26.92	23.21	35.06	32.22	19.52	**33.57**	**25.64**	28.02
`shrestha14`	23.08	25.00	30.52	**43.33**	**20.62**	28.45	22.23	27.60
`weren14`	29.49	17.86	20.13	27.78	19.14	27.92	22.11	23.49
`our best`	32.05	28.57	34.42	41.11	19.99	27.92	21.25	**29.33**

Table 6. Time results for the test dataset of the Author Profiling Task at PAN 2014. We include the results of the best four participants of the competition.

	Blogs		Twitter		Social Media		Reviews
	EN	ES	EN	ES	EN	ES	EN
`WC`	00:00:42	00:00:32	00:16:25	00:11:33	00:38:14	00:05:44	00:00:42
`Comp`	00:00:49	00:00:37	00:18:04	00:12:36	00:49:13	00:06:54	00:00:51
`Comp n-grams`	00:02:47	00:02:21	00:31:44	00:22:50	08:00:18	00:39:53	00:02:33
`SVM`	00:00:49	00:00:38	00:18:10	00:13:01	00:49:41	00:06:59	00:00:53
`lopezmonroy14`	00:03:47	00:03:22	00:07:02	00:05:36	00:34:06	00:06:25	00:04:01
`liau14`	00:00:06	00:00:04	00:00:55	00:00:27	00:12:53	00:00:27	00:00:12
`shrestha14`	00:01:56	00:00:39	00:02:31	00:01:10	00:26:31	00:03:26	00:02:13
`weren14`	00:04:46	00:04:06	00:41:32	01:33:48	30:18:02	02:34:33	01:17:29

5.6. *Performance in the Author Profiling Task at PAN 2015*

Table 7 shows the performance of our two best classifiers for the Author Profiling Task at PAN 2015. It shows the accuracy for age, gender, and joint prediction. It also shows the RSME obtained for each of the five personality traits: extroversion (E), emotional stability / neuroticism (S), agreeableness (A), conscientiousness (C), and openness to experience (O), in addition to the overall RSME, computed as the arithmetic mean of each trait RSME. Finally, it shows the global score that combines both the joint accuracy and the average RSME.

We can observe that `Comp` obtains better global results for the Italian and Dutch corpora, whereas `SVM` obtains better results for the English and Spanish corpora. Even when age, gender, and personality traits are better predicted by `Comp`, the joint accuracy of age and gender prediction is higher for `SVM`.

Compared with the results obtained by the participants of the competition,[d] `Comp` obtains the best accuracy for age prediction in English tweets, and the second

[d] Due to space restrictions, we cannot include here the results of the best participants at the competition. These results can be seen at the TIRA website (`http://www.tira.io/task/author-profiling/`), or at the PAN 2015 overview.[27]

Table 7. Results over the test dataset of the Author Profiling Task at PAN 2015.

	EN		ES		IT		NL	
	`Comp`	`SVM`	`Comp`	`SVM`	`Comp`	`SVM`	`Comp`	`SVM`
Age	0.7817	0.7676	0.9545	0.9432	-	-	-	-
Gender	0.7958	0.7324	0.7045	0.7386	0.6944	0.6944	0.9063	0.8438
Joint	0.6127	0.6408	0.6705	0.7045	-	-	-	-
E	0.1439	0.1463	0.1745	0.1856	0.1155	0.1202	0.1287	0.1237
S	0.2253	0.2239	0.2153	0.2624	0.2186	0.2321	0.1346	0.1611
A	0.1501	0.1583	0.1716	0.1658	0.0972	0.1202	0.1159	0.1500
C	0.1242	0.1482	0.1153	0.1348	0.1404	0.1607	0.1479	0.1458
O	0.1487	0.1178	0.1617	0.1373	0.2068	0.2273	0.0771	0.1199
RSME	0.1584	0.1589	0.1677	0.1772	0.1557	0.1721	0.1208	0.1401
Global	0.7271	0.7410	0.7514	0.7637	0.7694	0.7612	0.8927	0.8518
Time	0:06:22	0:05:49	0:04:24	0:04:07	0:01:37	0:01:13	0:01:21	0:01:27

Table 8. Results of `Comp` over the test dataset of the Author Profiling Task at PAN 2016.

	EN		ES		NL	
	Social Media	Blogs	Social Media	Blogs	Reviews 1	Reviews 2
Age	0.3075	0.5385	0.3281	0.4821	-	-
Gender	0.5230	0.5897	0.6094	0.7679	0.6400	0.5800
Joint	0.1695	0.3205	0.1563	0.4107	-	-
Time	0:05:19	0:01:14	0:01:12	0:00:53	0:00:17	0:00:22

best accuracy for gender and conscientiousness prediction in Spanish tweets. `SVM` obtains also good results for age and openness prediction for English tweets, and gender and joint accuracy for Spanish tweets. Overall, our approaches rank among the best 5 participants of this task.

5.7. *Performance in the Author Profiling Task at PAN 2016*

Table 8 shows the performance of `Comp` for the Author Profiling Task at PAN 2016. We omit `SVM` here, as it is not suitable for this cross-genre task. Notice that the entropy and number of different words extracted from one domain is not applicable for other domains; thus combining these techniques for this task is not convenient.

Compared with the results obtained by the participants of the competition,[28] `Comp` obtains the best result for the first Dutch corpus (Reviews 1), and the second best result for second Dutch corpus (Reviews 2), being more than 5 times faster than the winner approach for this corpus. It obtains the best result for gender and the third best result for age prediction in Spanish blogs. As tweets are more similar to blogs, it obtains better results for blogs than for social media documents.

6. Conclusions

In this paper we addressed the task of author profiling using different compression-inspired approaches. We evaluated the overall performance of these techniques for predicting age, gender and personality traits of the author of a given text. We presented simple strategies, all of them ignoring any stylistic feature of the text. We analyzed their performance in terms of the time required to evaluate the model and the achieved accuracy over texts of different nature. We show that using compression-inspired classification obtains good quality predictions in efficient time.

The straightforward implementation of these strategies, combined with their good results, are of special interest for demanding systems, as they are scalable to very large datasets. In addition, due to their fast execution, their quick training time, and its applicability to different domains without adapting them in any form, may also be an advantage compared with other techniques.

As future work, we plan to extend this work by designing new compression-inspired approaches. In particular, we have only used zero-order statistical techniques. We will evaluate the performance of higher order techniques that might improve accuracy at the expense of efficiency. We will also evaluate the behavior of non statistical approaches, such as LZ77-type or grammar-based compression-inspired techniques. Moreover, we plan to adapt these strategies to new features or variations of the author profiling tasks at upcoming PAN competitions.

Acknowledgements

This work was supported by FONDECYT [grant number 11130104]; Ministerio de Economía y Competitividad (PGE and ERDF) [grant number TIN2016-77158-C4-3-R; TIN2015-69951-R; TIN2013-46238-C4-3-R], Centro para el Desarrollo Tecnológico e Industrial (co-founded with ERDF) [grant numbers IDI-20141259; ITC-20161074; ITC-20151247; ITC-20151305]; Xunta de Galicia (co-founded with ERDF) [grant numbers ED431C 2017/58; ED431G/01]. The authors thank Paolo Rosso for his comments on how to improve this paper, and Martin Potthast and Francisco Rangel for their support with the evaluation of the methods under TIRA platform.

References

1. D. Nguyen, R. Gravel, D. Trieschnigg, and T. Meder. "How old do you think I am?" A study of language and age in Twitter. In *Proc. ICWSM*, 2013, pp. 439–448.
2. D. Benedetto, E. Caglioti, and V. Loreto. Language trees and zipping. *Phys. Rev. Lett.*, 88 (2002) 2–5.
3. W. J. Teahan. Text classification and segmentation using minimum cross-entropy. In *Proc. CAIRS*, 2000, pp. 943–961.
4. E. Frank, C. Chui, and I. H. Witten. Text categorization using compression models. In *Proc. DCC*, 2000, pp. 200–209.
5. D. P. Coutinho and M. A. T. Figueiredo. Text classification using compression-based dissimilarity measures. *Int. J. Pattern Recognit. Artif. Intell.*, 29 (2015) 19 pages.
6. A. Bratko, G. Cormack, B. Filipic, T. Lynam, and B. Zupan. Spam filtering using statistical data compression models. *J. Mach. Learn. Res.*, 7 (2006), 2673–2698.
7. S. Bergsma, P. McNamee, M. Bagdouri, C. Fink, and T. Wilson. Language identification for creating language-specific twitter collections. In *Proc. LSM*, 2012, pp. 65–74.

8. D. Pavelec, L. S. Oliveira, E. Justino, F. D. N. Neto, and L. V. Batista. Compression and stylometry for author identification. In *Proc. IJCNN*, 2009, pp. 669–674.
9. W. Oliveira, E. Justino, and L. Oliveira. Comparing compression models for authorship attribution. *Forensic Sci. Int.*, 228 (2013) 100–104.
10. D. Cerra, M. Datcu, and P. Reinartz. Authorship analysis based on data compression. *Pattern Recognit. Lett.*, 42 (2014), 79–84.
11. J. Cleary and I. Witten. Data compression using adaptive coding and partial string matching. *IEEE Trans. Commun.*, 32 (1984) 396–402.
12. P. Ferragina, R. Giancarlo, V. Greco, G. Manzini, and G. Valiente. Compression-based classification of biological sequences and structures via the universal similarity metric: experimental assessment. *BMC Bioinformatics*, 8 (2007) 252.
13. P. Juola and E. Stamatatos. Overview of the author identification task at PAN 2013. In *CLEF 2013 Working Notes*, CEUR-WS.org, volume 1179, 2013.
14. M. Koppel, S. Argamon, and A. Shimoni. Automatically categorizing written texts by author gender. *Lit. Linguist. Comput.*, 17 (2002) 401–412.
15. J. D. Burger, J. Henderson, G. Kim, and G. Zarrella. Discriminating gender on twitter. In *Proc. EMNLP*, 2011, pp. 1301–1309.
16. J. Schler, M. Koppel, S. Argamon, and J. Pennebake. Effects of age and gender on blogging. In *Proc. AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006.
17. B. Y. Ryabko. Data compression by means of a "book stack". *Probl. Inf. Transm.*, 16 (1980) 265–269.
18. D. A. Huffman. A method for the construction of minimum-redundancy codes. *Proc. IRE*, 40 (1952) 1098–1101.
19. C. Cortes and V Vapnik. Support-vector networks. *Mach. Learn.*, 20 (1995) 273–297.
20. C. W. Hsu and C. J. Lin. A comparison of methods for multiclass support vector machines. *IEEE Trans. Neural Netw.*, 13 (2002) 415–425.
21. J. Shawe-Taylor and N. Cristianini. Kernel methods for pattern analysis. *Cambridge University Press*, 2004.
22. T. Gollub, B. Stein, and S. Burrows. Ousting Ivory Tower Research: Towards a Web Framework for Providing Experiments as a Service. *Proc. SIGIR*, 2012, pp. 1125–1126.
23. T. Gollub, B. Stein, S. Burrows, and D. Hoppe. TIRA: Conguring, Executing, and Disseminating Information Retrieval Experiments. *Proc. TIR*, 2012, pp. 151–155.
24. T. Gollub, M. Potthast, A. Beyer, M. Busse, F. Rangel, P. Rosso, E. Stamatatos, and B. Stein. Recent trends in digital text forensics and its evaluation. *Proc. CLEF*, 2013, pp. 282–302.
25. 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. *J. Mach. Learn. Res.*, 12 (2011) 2825–2830.
26. F. Rangel, P. Rosso, I. Chugur, M. Potthast, M. Trenkmann, B. Stein, B. Verhoeven, and W. Daelemans. Overview of the 2nd Author Profiling Task at PAN 2014. In *CLEF 2014 Working Notes*, CEUR-WS.org, volume 1180, 2014, pp. 898–827.
27. F. Rangel, F. Celli, P. Rosso, M. Potthast, and W. Daelemans. Overview of the 3rd Author Profiling Task at PAN 2015. In *CLEF 2015 Working Notes*, CEUR-WS.org, volume 1391, 2015.
28. F. Rangel, P. Rosso, M. Potthast, B. Verhoeven, W. Daelemans, M. Potthast, and B. Stein. Overview of the 4th Author Profiling Task at PAN 2016: Cross-Genre Evaluations. In *CLEF 2016 Working Notes*, CEUR-WS.org, volume 1609, pp. 750-784, 2016.
29. Y. Marton, N. Wu, and L. Hellerstein. On compression-based text classification. In

Proc. ECIR, 2005, pp. 300–314.