Compression and Stylometry for Author Identification

D. Pavelec, L. S. Oliveira, E. Justino, F. D. Nobre Neto, and L. V. Batista

Abstract— **In this paper we compare two different paradigms for author identification. The first one is based on compression algorithms where the entire process of defining and extracting features and training a classifier is avoided. The second paradigm, on the other hand, takes into account the classical pattern recognition framework, where linguistic features proposed by forensic experts are used to train a Support Vector Machine classifier. Comprehensive experiments performed on a database composed of 20 writers show that both strategies achieve similar performance but with an interesting degree of complementarity demonstrated through the confusion matrices. Advantages and drawback of both paradigms are also discussed.**

Keywords: Author identification, Compression, Stylometry.

I. Introduction

The literature shows a long history of linguistic and stylistic investigation into author identification [10], [9] but the work published by Svartvik [14] marked the birth of term forensic linguistics, i.e., the linguistic investigation of authorship for forensic purposes. In it, he analyzed four statements that Timothy Evans, executed in 1950 for the murder of his wife and baby daughter, was alleged to have made following his arrest. Using both qualitative and quantitative methods Svartvik demonstrated considerable stylistic discrepancies between the statements, thus raising serious questions about their authorship. It was later discovered that both victims had actually been murdered by Evan's landlord, John Christie [4].

Since then, there has been a impressive growth in the volume with which lawyers and courts have called upon the expertise of linguists in cases of disputed authorship. Hence, practical applications for author identification have grown in several different areas such as, criminal law (identifying writers of ransom notes and harassing letters), civil law (copyright and estate disputes), and computer security (mining email content).

Author identification is the task of identifying the author of a given text, therefore, it can be formulated as a typical classification problem, which depends on discriminant features to represent the style of an author. In this context, we can cite two different paradigms.

The first one avoids defining features explicitly while describing the classes as a whole. In this vein, modern lossless data compression algorithms have been used as feature extractors due to their ability to construct accurate statistical models, with low or acceptable computational requirements. Those that speak in defense of this strategy argue that it yields an overall judgement on the document as a whole, rather than discarding information by pre-selecting features and it avoids the messy and rather artificial problem of defining word boundaries [6]. The contrary argument, on the other hand, relies on the fact that features are defined in a black box of which the inner workings are unclear because it does not follow well established forensic protocols.

The second paradigm takes into account the know-how developed by forensic examiners on stylometry to define discriminative features. The literature shows that several stylometric features that have been applied include various measures of vocabulary richness and lexical repetition based on word frequency distributions. As observed by Madigan et al [8], most of these measures, however, are strongly dependent on the length of the text being studied, hence, are difficult to apply reliably. Many other types of features have been investigated, including word class frequencies, syntactic analysis, word collocations, grammatical errors, number of words, sentences, clauses, and paragraph lengths [5], [7], [1].

In this work we compare both strategies for author identification. First we present the background about compression algorithms and introduce the PPM (Prediction by Partial Matching) algorithm [11], which is considered one of the best modern general-purpose compressing algorithms. Despite demanding much more computer resources than dictionary-based techniques, PPM typically yields substantially improved compression ratios. With modern digital technology, memory usage and processing time of PPM is acceptable. We also show how PPM can be used for pattern classification. Thereafter we discuss stylometry and present two sets of linguistic features of the Portuguese language which were used to train a Support Vector Machine (SVM) classifier. Comprehensive results on a database composed of short articles written in Portuguese by 20 different authors show that both strategies achieve similar performance, but make different mistakes, which indicates that they can be further combined to produce more reliable decisions. Finally, we also discuss some advantages and drawbacks of each paradigm.

This paper is organized as follows. Section II introduces the basics about compression algorithms and how they can be used for classification. Section III discusses the concept of stylometry and presents the stylometric features that have been used in this work. Section IV describes the database used in the experiments reported in Section V. Finally, Section VI concludes this work.

Daniel Pavelec, Luiz S. Oliveira and Edson Justino are with Pontifíca Universidade Católica do Paraná, Rua Imaculada Conceicao, 1155, Curitiba, Brazil, 80215-901; email: {pavelec,soares,justino}@ppgia.pucpr.br

Leonardo V. Batista and Francisco Dantas Nobre Neto are with Universidade Federal da Paraíba, Dpto de Informática, João Pessoa, PB, Brazil; email leonardo@di.ufpb.br

II. Compression Algorithm

A. Background

Let S be a stationary discrete information source that generates messages over a finite alphabet $A = \{a_1, a_2, \ldots, a_M\}$. The source chooses successive symbols from A according to some probability distribution that depends, in general, on preceding selected symbols. A generic message will be modeled as a stationary stochastic process $x = \ldots x_{-2}, x_{-1}, x_0, x_1, x_2$, with $x_i \in A$. Let $x^n = \{x_1, x_2, \ldots, x_n\}$ represent a message of length n. Since $|A| = M$, the source can generate M^n different messages of length n. Let $x_i^n, i = \{1, 2, \ldots, M^n\}$ denote the i^{th} of these messages, according to some sorting order, and assume that the source follows a probability distribution P, so that message x_i^n is produced with probability $P(x_i^n)$.

Let

$$G_n(P) = -\frac{1}{n}\sum_{i=1}^{M^n} P(x_i^n)\log_2 P(x_i^n)\frac{\text{bits}}{\text{symbol}} \tag{1}$$

It can be shown that $G_n(P)$ decreases monotonically with n [13] and the entropy of the source is given by Equation 2

$$H(P) = \lim_{n\to\infty} G_n(P)\frac{\text{bits}}{\text{symbol}} \tag{2}$$

An alternative formulation for $H(P)$ uses conditional probabilities. Let $P(x_i^{n-1}, a_j)$ be the probability of the sequence $x_i^n = (x_i^{n-1}, a_j)$, i.e., the probability of x_i^{n-1} concatenated with symbol $x_n = a_j$, and let $P(a_j|x_i^{n-1}) = P(x_i^{n-1}, a_j)P(x_i^{n-1})$ be the probability of symbol $x_n = a_j$ given x_i^{n-1}. The entropy of the n^{th} order approximation to $H(P)$ is given by Equation 3.

$$F_n(P) = -\sum_{i=1}^{M^n}\sum_{j=1}^{M} P(x_i^{n-1}, a_j)\log_2 P(a_j|x_i^{n-1})\frac{\text{bits}}{\text{symbol}} \tag{3}$$

$F_n(P)$ decreases monotonically with n [13] and the entropy of the source is given by Equation 4

$$H(P) = \lim_{n\to\infty} \log_2 P(a_j|x_i^{n-1})\frac{\text{bits}}{\text{symbol}} \tag{4}$$

Equation 4 involves the estimation of probabilities conditioned on an infinite sequence of previous symbols. In practice, finite memory is assumed, and the sources are modeled by an order-$(n-1)$ Markov process, so that $\mathrm{P}(a_j|\ldots x_{-1}, x_0, x_1, \ldots x_{n-1} = P(a_j|x_1, \ldots x_{n-1}).$). In this case, $H(P) = F_n(P)$.

The concept of entropy as a measure of information is central to Information Theory [13], and data compression provides an intuitive perspective to the concept. Define the coding rate of a coding scheme as the average number of bits per symbol the scheme uses to encode the output of a source. A lossless compressor is a uniquely decodable coding scheme whose goal is to achieve a coding rate as small as possible. The coding rate of any uniquely decodable coding scheme is always greater than or equal to the source entropy. Optimum coding schemes have a coding rate equal to the theoretical lower bound $H(P)$, thus achieving maximum compression.

For order-$(n-1)$ Markov processes, optimum encoding is reached if and only if a symbol $x_n = a_j$ occurring after x_i^{n-1} is coded with $-\log_2 P(a_j|x_i^{n-1})$ bits [15, 22]. However, it may be impossible to accurately estimate the conditional distribution $P(.|x_i^{n-1})$ for large values of n, due to the exponential growth of the number of different contexts, which brings well-known problems, such as context dilution.

B. The PPM Algorithm

Even though the source model P is generally unknown, it is possible to construct a coding scheme based upon some implicit or explicit probabilistic model Q that approximates P. The better Q approximates P, the smaller the coding rate achieved by the coding scheme.

In order to achieve low coding rates, modern lossless compressors rely on the construction of sophisticated models that closely follows the true source model. Statistical compressors, such as PPM, encode messages according to an estimated statistical model for the source.

For stationary sources, PPM algorithm learns a progressively better model during encoding. Many experimental results show that the superiority of the compression performance of PPM, in comparison with other asymptotically optimum compressors, results mainly from its ability to construct a good model for the source in very early stages of the compression process. In other words, PPM constructs ("learns") an efficient model for the message to be compressed faster than its competitors.

The PPM algorithm is based on context modeling and prediction. The PPM starts with a "complete ignorance model" (assuming independent equiprobable variables) and adaptively updates this model as the symbols in the uncompressed stream are coded. Based on the whole sequence of symbols already coded, the model estimates probability distributions for the next symbols, conditioned on a sequence of k previous symbols in the stream. The number of symbols in the context, k, determines the order of the model.

The next symbol, x, is coded by arithmetic coding, with the probability of x conditioned on its context. If x has not previously occurred in that specific context, no estimate for its probability is available. In this case, a special symbol ("escape") is coded, and PPM-C tries do code x in a reduced context, with $k-1$ antecedent symbols. This process is repeated until a match is found, or the symbol is coded using the independency-equiprobability model.

Experimentation shows that the compression performance of PPM increases as the maximum context size increases up to a certain point, after which performance starts do degrade. This behavior can be explained by the phenomenon of context-dilution and the increased emission of escape symbols. The context size in which compression performance is optimal depends on the message to be compressed, but typical values usually are in the interval 4 to 6.

III. Stylometry

Forensic stylistics is a sub-field of forensic linguistics and it aims at applying stylistics to the context of author verification. The stylistic is based on two premisses: a) Two writers (same mother-tongue) do not write in the same way and b) The writer does not write in the same way all the time.

The stylistic can be classified into two different approaches: qualitative and quantitative. The qualitative approach assesses errors and personal behavior of the authors, also known as idiosyncrasies, based on the examiner's experience. According to Chaski [3], this approach could be quantified through databasing, but until now the databases which would be required have not been fully developed. Without such databases to ground the significance of stylistic features, the examiner's intuition about the significance of a stylistic feature can lead to methodological subjectivity and bias. In this vein, Koppel and Schler [7] proposed the use of 99 error features to feed different classifiers such as SVM and decision trees. The best result reported was about 72% of recognition rate.

The second approach, which is very often refereed as stylometry, is quantitative and computational, focusing on readily computable and countable language features, e.g. word length, phrase length, sentence length, vocabulary frequency, distribution of words of different lengths. It uses standard syntactic analysis from the dominant paradigm in theoretical linguistics over the past forty years. Examples of this approach can be found in Tambouratzis et al [15], Chaski [3] and [16]. The latter addresses the problem of author verification for Turkish texts and reports an average success rate of 80%. Experimental results show that usually this approach provides better results than the qualitative one.

A. Linguistic Features

The literature suggests many linguistic features to be used for author verification. In [2], Chaski discusses about the differences between scientific and replicable methods for author verification. Scientific methods are based on empirical, testable hypotheses, and the use of these methods can be done by anyone, i.e., it is not dependent on a special talent. In the same work, nine empirical hypotheses that have been used to identify authors in the past are reported: Vocabulary Richness (number of distinct words), Hapax Legomena (numbers of words occurring once), Readability Measures, Content Analysis, Spelling, Errors, Grammatical Errors, Syntactically Classified Punctuation, Sentential Complexity, Abstract, Syntactic Structures.

Vocabulary Richness is given by the ratio of the number of distinct words (type) to the number of total words (token). Hapax Legomena is the ratio of the numbers of words occurring once (Hapax Legomena) to the total number of words. Readability Measures compute the supposed complexity of a document, and are calculations based on sentence length and word length. Content Analysis classifies each word in the document by semantic category, and statistically analyze the distance between documents. Spelling Errors quantifies the misspelled words. Prescriptive Grammatical Errors test errors such as sentence fragment, run-on sentence, subject-verb mismatch, tense shift, wrong verb form, and missing verb. Syntactically Classified Punctuation takes into account end-of-sentence period, comma separating main and dependent clauses, comma in list, etc. Finally, Abstract Syntactic Structures computationally analyzes syntactic patterns. It uses verb phrase structure as a differentiating feature.

In this work we have used conjunctions and adverbs of the Portuguese language. Just like other language, Portuguese has a large set of conjunctions that can be used to link words, phrases, and clauses. Such conjunctions can be used in different ways without modifying the meaning of the text. For example, the sentence "Ele é *tal qual* seu pai" (He is like his father), could be written is several different ways using other conjunctions, for example, "Ele é *tal e qual* seu pai", "Ele é *tal como* seu pai", "Ele é *que nem* seu pai", "Ele é *assim como* seu pai". The way of using conjunctions is a characteristic of each author, and for this reason we decided to use them in this work. Table I shows the 77 Portuguese conjunctions used in this work.

TABLE I

Conjunctions of the Portuguese language used as features

Conjunctions
e, nem, mas também, senão também, bem como, como também, mas ainda, porém, todavia, mas, ao passo que, não obstante, entretanto, porque senão, apesar disso, em todo caso, contudo, no entanto, logo, portanto, por isso, por conseguinte, porquanto, que, tal qual, tais quais, assim como, tal e qual, tão como, tais como, mais do que, tanto como, menos do que, que nem, tanto quanto, o mesmo que, tal como, mais que, consoante, segundo, conforme, embora, ainda que, ainda quando, posto que, por muito que, se bem que, por menos que, nem que, dado que, mesmo que, se, caso, contanto que, salvo que, a não ser que, a menos que, de sorte que, de forma que, de maneira que, de modo que, sem que, para que, fim de que, final, a proporção que, quanto menos, quanto mais, menos que, por mais que, a medida que.

In addition to conjunctions, we have used adverbs of the Portuguese language. An adverb can modify a verb, an adjective, another adverb, a phrase, or a clause. Authors can use it to indicate manner, time, place, cause, or degree and answers questions such as "how", "when","where", "how much". Table II describes the 94 adverbs used as features.

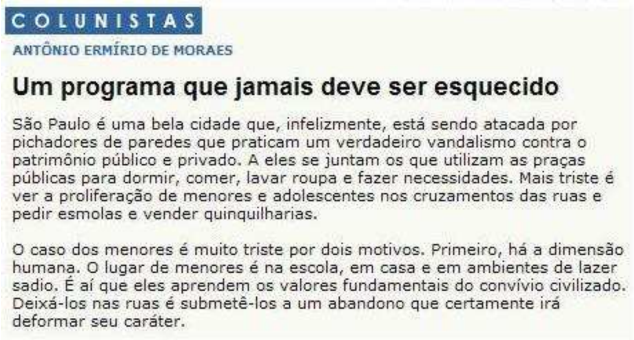
IV. Database

To build the database we have collected articles available in the Internet from 20 different people with profiles in Economics (7), Politics (4), Sports (2), Literature (3), Miscellaneous (3), Gossip (1), and Wine (1). Our sources were two different Brazilian newspapers, *Gazeta do Povo* and *Tribuna do Paraná*. We have chosen 30 short articles from each writer. The articles usually deal with polemic subjects and express the author's personal opinion. In average, the articles have 600 tokens and 350 Hapax. The option for short

TABLE II
CONJUNCTIONS OF THE PORTUGUESE LANGUAGE USED AS FEATURES

Adverbs
aqui, ali, aí, cá, lá, acolá, além, longe, perto, dentro, adiante,
defronte, onde, acima, abaixo, atrás, em cima, de cima, ao lado,
de fora, por fora, hoje, ontem, amanhã, atualmente, sempre, nunca,
jamais, cedo, tarde, antes, depois, já, agora, então, de repente,
hoje em dia, certamente, com certeza, de certo, realmente, seguramente,
sem dúvida, sim, ainda, apenas, de pouco, demais, mais, menos, muito,
pouca, pouco, quase, tanta, tanto, absolutamente, de jeito
nenhum, de modo algum, não, tampouco, embora, ainda que, ainda
quando, posto que, por muito que, se bem que, por menos que,
nem que, dado que, mesmo que, por mais que, todo, toda,
assim, depressa, bem, devagar, face a face, facilmente,
frente a frente, lentamente, mal, rapidamente, algo, alguém,
algum, alguma,bastante, cada, certa, certo, muita, nada,
nenhum, nenhuma, ninguém, outra, outrem, outro, quaisquer,
qualquer, tudo.

articles was made because in real life forensic experts can count only on short pieces of texts to identify a given writer. Another aspect worth of remark is that this kind of articles can go through some revision process, which can remove some personal characteristics of the texts. Figure 1 depicts an example of the article of our database.



COLUNISTAS

ANTÔNIO ERMÍRIO DE MORAES

Um programa que jamais deve ser esquecido

São Paulo é uma bela cidade que, infelizmente, está sendo atacada por pichadores de paredes que praticam um verdadeiro vandalismo contra o patrimônio público e privado. A eles se juntam os que utilizam as praças públicas para dormir, comer, lavar roupa e fazer necessidades. Mais triste é ver a proliferação de menores e adolescentes nos cruzamentos das ruas e pedir esmolas e vender quinquilharias.

O caso dos menores é muito triste por dois motivos. Primeiro, há a dimensão humana. O lugar de menores é na escola, em casa e em ambientes de lazer sadio. É aí que eles aprendem os valores fundamentais do convívio civilizado. Deixá-los nas ruas é submetê-los a um abandono que certamente irá deformar seu caráter.

Fig. 1. An example of an article used in this work.

All texts were preprocessed to eliminate numbers, punctuation and diacritics (cedilla, acute accents, etc). Spaces and end-of-line characters are not considered. All hyphenated words are considered as two words. In the example, the sentence "*eu vou dar-te um pula-pula e também dar-te-ei um beijo, meu amor*!" has 16 tokens and 12 Hapax. Punctuation, special characters, and numbers are not considered as tokens.

V. EXPERIMENTS

In this section we report the experiments performed using both paradigms described in this paper. Due to the rather small size of the corpus, cross-validation was adopted for computing classification rates. The articles of each author were randomly grouped in three sets (10 samples in each set). In the first cross-validation round, the first set of each author was used for training, and the remaining sets were used for classification. A similar procedure was done in the second and third rounds, with the second and third sets of each author, respectively, selected for training.

In the learning stage, the number N of classes (in this case $N = 20$ writers) is defined, and a training set T_i of text samples with fixed size known to belong to class $C_i, i = \{1, 2, \ldots, N\}$, is selected.

In the compression strategy, the feature extraction is done intrinsically by the PPM algorithm. It sequentially compresses the samples in T_i , and the resulting model M_i is kept as a model for the texts in $C_i, i = \{1, 2, \ldots N\}$.

In the classification stage PPM operates in static mode, i.e., the models generated in the training stage are used but not updated during the encoding process. Classification is done as follows: A text sample x from an unknown writer is coded by the PPM algorithm with static model M_i, and the corresponding coding rate $r_i, i = \{1, 2, \ldots N\}$ is registered. Then, the sample x is assigned to C_i if $r_i < r_j, j = \{1, 2, \ldots, N\}, j \neq i$. The rationale is that if x is a sample from class C_i, the model M_i probably best describes its structure, thus yielding the smallest coding rates. The average performance of this strategy on the three different partitions used as testing set (20 articles × 20 authors), was 84.3%. Table III shows the confusion matrix produced by this classification scheme.

Table III shows that some authors have very strong features for the compression strategy, even when writing about similar subjects, e.g., authors K, O, and Q. Others, like E and P, write about very specific subjects using a particular vocabulary and for this reason are not misclassified. On the other hand, this strategy achieve a very poor performance for other writers. The most critical case is author A, which writes about literature and had most of their articles confused with author O, which also is a literature critic. Other problems are related to those writes classified as "Misc", which are generalist and write just about everything. Their texts are confused with all sort of writers.

Regarding the experiments based on stylometry, the same formalism has been applied. However, a machine learning algorithm, the well-known Support Vector Machine (SVM), was used to model the N classes (authors). When using SVMs, there are two basic approaches to solve an N-class problems: pairwise and one-against-others. In this work both strategies have been tried out but the former produced better results. A Gaussian kernel was employed and its parameters (C and γ) were defined through a grid search. A modified version of the SVM [12], which is able to produce a estimation the the posterior probability $P(class|input)$ was considered in this work.

The feature vector used to train the SVMs is composed of 171 components, which is the number of occurrences of 77 conjunctions and 94 adverbs found in the text. Conjunctions and adverbs were assessed independently, but with no further improvements. LIBSVM were used in our experiments.

In the classification stage, a text sample x from an unknown author is assigned to C_i that maximizes the posterior probability, i.e., $C_i = \max P(C_i|x)$. The average performance of this strategy on the testing set was 83.2%. Table IV shows the confusion matrix produced by this classification

TABLE III

Confusion matrix produced by the PPM strategy.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
A (Literature)	**0.20**	0.13		0.20									0.03		0.03		0.03	0.70		
B (Politics)		**0.87**			0.07		0.03													0.03
C (Economics)			**0.94**					0.03									0.03			
D (Literature)	0.03	0.07		**0.84**											0.03			0.03		
E (Reviewer)					**1.00**															
F (Politics)						**0.97**	0.03													
G (Politics)		0.03					**0.94**			0.03										
H (Economics)		0.03	0.03		0.07			**0.81**							0.03		0.03			
I (Sports)		0.03							**0.97**											
J (Economics)		0.03					0.03			**0.84**										0.10
K (Economics)											**1.00**									
L (Misc)	0.10	0.03										**0.67**			0.03			0.03		0.14
M (Misc)		0.14			0.03				0.14				**0.57**		0.06	0.06				
N (Sports)									0.03					**0.97**						
O (Literature)															**1.00**					
P (Gossip)																**1.00**				
Q (Economics)																	**1.00**			
R (Politics)				0.03							0.03							**0.91**		0.03
S (Economics)											0.30								**0.70**	
T (Economics)		0.03		0.03						0.07	0.07							0.07		**0.74**

TABLE IV

Confusion matrix produced by the Stylometric strategy.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
A (Literature)	**0.97**							0.03												
B (Politics)	0.07	**0.84**							0.03			0.03							0.03	
C (Economics)		0.10	**0.84**																0.03	0.03
D (Literature)				**0.87**	0.03							0.03				0.03		0.03		
E (Reviewer)			0.03		**0.77**	0.03		0.07				0.07		0.03						
F (Politics)	0.03	0.03			0.03	**0.60**	0.03		0.03	0.03				0.07		0.03		0.03	0.07	
G (Politics)					0.03		**0.97**													
H (Economics)					0.03			**0.94**			0.03									
I (Sports)									**0.91**					0.03	0.03				0.03	
J (Economics)										**1.00**										
K (Economics)						0.03					**0.97**									
L (Misc)												**1.00**								
M (Misc)	0.03		0.10					0.07	0.03				**0.60**	0.03			0.10		0.03	
N (Sports)					0.07	0.10	0.03		0.07	0.03	0.03	0.07		**0.57**		0.03				
O (Literature)	0.03	0.03				0.03	0.07		0.07					0.03	**0.60**	0.14				
P (Gossip)																**1.00**				
Q (Economics)	0.03									0.03		0.03					**0.88**		0.03	
R (Politics)			0.03					0.03		0.03						0.03		**0.88**		
S (Economics)		0.03												0.07					**0.90**	
T (Economics)									0.14			0.03		0.10	0.03			0.07		**0.64**

scheme.

Table V reports the average performance of both strategies based on three different partitions of the database. Both achieve similar performances but PPM features a higher standard deviation.

Observing the confusion matrices we can notice that, in spite of the similar overall performance, the methods make different confusions. This is very clear for some writers,

TABLE V
COMPARISON BETWEEN PPM (COMPRESSION) AND SVM (STYLOMETRY) ON THREE DIFFERENT TESTING PARTITIONS. STANDARD DEVIATION IN PARENTHESIS.

Round	PPM (%)	SVM (%)
1	84.0 (22.1)	83.0 (15.2)
2	83.0 (23.4)	84.0 (14.8)
3	86.0 (22.1)	82.9 (15.6)
Average	84.3 (22.5)	83.3 (15.2)

such as writer A, where PPM performs very poorly (about 20%) and the stylometry-based classifier identify almost all samples of that writer correctly. The opposite situation also happens, e.g., author O is identified correctly 100% by PPM and only 60% by the stylometry-based classifier. Besides, even for similar performances, author D for example, the confusions are different. All that is a good indicative that both methods produce complementary results and can be further combined to build a more reliable identification system.

From the experimental point of view, both strategies have advantages and drawbacks. Compared to the traditional classification scheme used by the stylometric-based classifier, the PPM has some advantages, such as i) no definition of features, ii) no feature extraction and iii) no traditional learning phase. It is worth of remark that in spite of the apparent black box concept, compression algorithms like PPM are based on robust probabilistic frameworks. However, if the size of the text or the number of samples per writer cannot be augmented, the performance of the PPM strategy cannot be further improved. In other words, this is as good as it gets.

On the other hand, the traditional way of defining and extracting features to train a machine learning model gives us more perspective for improvements, since we always can explore new features, select the relevant and uncorrelated ones through feature selection, and try new classification algorithms.

VI. CONCLUSION

In this work we have discussed two different paradigms for author identification. The first one is based on the well-known compression algorithm, called PPM. In this case, the feature extraction is done in an implicity fashion where each writer of the database is modeled by compressing some samples of his writings. Then, during recognition those models are used to compress a given questioned samples which is assigned to the class that produces the lowest compression rate.

The second paradigm relies on the traditional pattern recognition framework, which involves steps such as definition of features, feature extraction, classifier training, etc. In this work we have used stylometric features (conjunctions and adverbs of the Portuguese language) to train an SVM classifier. Results using the same testing protocol show that both strategies produce very similar results, but making different confusions. This shows that both strategies are complementary to each other and can be combined to build a more reliable identification system. Besides, we believe that the PPM is an useful tool that can be used as parameter when designing new features for author identification. It is fair to expect that a discriminant feature set would at least achieve the same level of performance than PPM.

As future works, we plan to increase the database with more authors and also longer articles, which will enable us to assess the impacts of bigger databases on PPM. Moreover, different stylometric features used by forensic experts will be tried out, as well as strategies to combine these two different paradigms will be investigated.

ACKNOWLEDGES

This research has been supported by The National Council for Scientific and Technological Development (CNPq) grant 471496/2007-3. Francisco D. Nobre Neto was supported by scholarship from the Tutorial Education Programme (PET) from the Higher Education Secretariat of the Ministry of Education.

REFERENCES

[1] S. Argamon, M. Saric, and S. S. Stein. Style mining of electronic messages for multiple author discrimination. In *ACM Conference on Knowledge Discovery and Data Mining*, 2003.

[2] C. Chaski. A daubert-inspired assessment of current techniques for language-based author identification. Technical Report 1098, ILE Technical Report, 1998.

[3] C. E. Chaski. Who is at the keyboard. authorship attribution in digital evidence investigations. *International Journal of Digital Evidence*, 4(1), 2005.

[4] M. Coulthard. Author identification, idiolect, and linguistic uniqueness. *Applied Linguistics*, 25(4):431–447, 2004.

[5] R. S. Forsyth and D. I. Holmes. Feature finding for text classfication. *Literary and Linguistic Computing*, 11(4):163–174, 1996.

[6] E. Frank, C. Chui, and I. H. Witten. Text categorization using compression models. In *Data Compression Conference*, 2000.

[7] M. Koppel and J. Schler. Exploiting stylistic idiosyncrasies for authorship attribution. In *Workshop on Computational Approaches to Style Analysis and Synthesis*, 2003.

[8] D. Madigan, A. Genkin, D. D. Lewis, S. Argamon, D. Fradkin, and L. Ye. Author identification on the large scale. In *Joint Annual Meeting of the Interface and the Classification Society of North America (CSNA)*, 2005.

[9] C. Mascol. Curves of pauline and pseudo-pauline style i. *Unitarian Review*, 30:453–460, 1888.

[10] T. Mendenhall. The characteristic curves of composition. *Science*, 214:237–249, 1887.

[11] A. Moffat. Implementing the PPM data compression scheme. *IEEE Transactions on Communications*, 38(11):1917–1921, 1990.

[12] J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In A. Smola et al, editor, *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[13] C. E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. Journal*, 27:379–423, 1948.

[14] J. Svartvik. The evans statements: A case for forensic linguistics. *Acta Universitatis Gothoburgensis*, 1968.

[15] G. Tambouratzis, S. Markantonatou, N. Hairetakis, M. Vassiliou, G. Carayannis, and D. Tambouratzis. Discriminating the registers and styles in the modern greek language – part 2: Extending the feature vector to optimize author discrimination. *Literary and Linguistic Computing*, 19(2):221–242, 2004.

[16] T. Tas and A. K. Gorur. Author identification for turkish texts. *Journal of Arts and Sciences*, 7:151–161, 2007.