Appl Intell (2017) 46:916–933
DOI 10.1007/s10489-016-0878-8

Clustering documents in evolving languages by image texture analysis

Darko Brodić[1] · Alessia Amelio[2] · Zoran N. Milivojević[3]

Published online: 26 December 2016

Abstract This paper introduces a new method for clustering of documents, which have been written in a language evolving during different historical periods, with an example of the Italian language. In the first phase, the text is transformed into a string of four numerical codes, which have been derived from the energy profile of each letter, defining the height of the letters and their location in the text line. Each code represents a gray level and the text is codified as a 1-D image. In the second phase, texture features are extracted from the obtained image in order to create document feature vectors. Subsequently, a new clustering algorithm is employed on the feature vectors to discriminate documents from different historical periods of the language. Experiments are performed on a database of Italian documents given in Italian Vulgar and modern Italian. Results demonstrate that this proposed method perfectly identifies the historical periods of the language of the documents, outperforming other well-known clustering algorithms generally adopted for document categorization and other state-of-the-art text-based language models.

Keywords Coding · Clustering · Document analysis · Evolving languages · Image processing · Italian language · Language recognition · Statistical analysis

1 Introduction

The Latin language is a member of so-called Italic languages [1, 2]. It uses a Latin alphabet, which is based on the Old Italic alphabets derived from Greek and Phoenician scripts. Historically, Latin came from the prehistoric language used in the Latium region, specifically around the River Tiber, where Roman civilization developed. Essentially, Latin was brought to Italy around 1000 BC by Indo-European immigrants from Northern Europe. Firstly, it was used as an isolated local language on a small territory called Latium. In this territory, the city of Rome was founded in 753 BC. However, Latin Romans fell under the sway of Etruscan Kings. As a consequence, the Latin language was heavily influenced by non-Indo-European Etruscans. Furthermore, it was also affected by Celtic migrations and their dialects from Northern Italy as well as by Greek culture. The earliest Latin language inscription dated around the 6th century BC. It was written on a pin saying “Manios me fhefhaked Numasioi”.

From 250 to 100 BC, there was a period of Early Latin language, which incorporated many elements from the Greek language. The classical Latin language period dates from 100 BC up to 150 AC. This period was characterized mainly by the most known Latin literature pieces written by Cicero, Caesar, Virgil, and Tacitus. These works

✉ Darko Brodić
dbrodic@tfbor.bg.ac.rs

Alessia Amelio
aamelio@dimes.unical.it

Zoran N. Milivojević
zoran.milivojevic@vtsnis.edu.rs

[1] Technical Faculty in Bor, University of Belgrade, Vojske Jugoslavije 12, 19210, Bor, Serbia

[2] DIMES University of Calabria, Via P. Bucci Cube 44, 87036, Rende (CS), Italy

[3] College of Applied Technical Sciences, Aleksandra Medvedeva 20, 18000, Niš, Serbia

are considered Latin literature masterpieces. However, literary language was quite different from the Latin that was used by ordinary people. Furthermore, the historical period up to 550 AC represents the Late Latin language period. In this period, the Latin language incorporates some classical elements of Latin as well as Vulgar elements.

After the fall of the Western part of the Roman Empire, Latin evolved. It is commonly said that Latin evolved into Vulgar Latin language, including many of its varieties. Hence, the language common to the typical Roman was called Vulgar Latin language. In that sense, the term “vulgar” means “common”. Later, varieties of this language evolved into different Romance languages. By the beginning of the second millennium, the evolution of Romance languages has been completed. During this evolution process, the structure of the language was changed. However, literary and scientific documents were written in Latin across Europe. In the fourteenth century, Dante Alighieri wrote his famous work Commedia. It was the first literary work written in the Italian language, which has also been considered a masterpiece of world literature. It is considered very important, because Commedia was written in Italian Vulgar language, which was widespread in Sicily and Tuscany, representing an important aspect of the evolution of the Latin language in Italy, and easily understood by most Italians. Hence, Dante Alighieri has been considered as the Father of the Italian language [3].

The Italian language has many dialects. Some researchers regard them as separate languages. Still, literary Italian language is based on the Florentine dialect. In the nineteenth century, a standardized version of the Florentine dialect became the official language in Italy. Italian language is geographically and linguistically the closest successor to the Latin language [4, 5]. It is based on the assumption that Italian has retained some characteristic features of Latin, such as the distinction between short and long vowels, which is not present in the other Romance languages. Specifically, Pei [6] introduced a statistical method for comparison of the Latin language with French, Spanish, Italian, Portuguese, Rumanian, Old Provençal, and Logudorese Sardinian, based on different types of free and accented vowel changes. It concluded that Italian has the lowest change percentage of 12 % with respect to Latin, only followed by Sardinian with a change percentage of 8 %. Furthermore, Italian is the most similar to Latin in terms of vocabulary [7]. Accordingly, unlike other languages i.e. French, contemporary Italian has a limited number of phonetic and morphological differences with old Italian variants, directly derived from Latin. Hence, Italian exhibits a continuity in the old and the modern age, with a more conservative aspect than other languages. For such a reason, capturing the historical periods of the Italian language along time, from old to modern Italian, becomes the most meaningful scenario for analyzing the evolution of languages through time, and a more complex challenge than we might expect from other languages. Currently, the Italian language is spoken by approximately 60 million people. They are geographically spread in Italy, San Marino, Vatican City, Switzerland, Slovenia and Croatia. Italian is also spoken by Italian immigrants in the United States, Canada, Australia, Argentina and so forth.

This paper consists of the following sections. Section 2 presents the previous works on language discrimination. Section 3 introduces a new method for language discrimination based on text coding and analogy with image texture. Section 4 gives an experiment based on evolving language database. Section 5 shows the obtained results of the experiment and discusses them. Section 6 discusses the execution time of the procedure. Section 7 concludes the paper and outlines future directions to work on.

2 Previous works

Many approaches have been proposed for a language discrimination task. Some of them are specifically linked to certain languages or specific groups of languages. More complex techniques differentiate closely related languages. Up to now, the differentiation of a language in different historical periods has rarely been analyzed. A previous analysis of Latin language evolution into Italian Vulgar and then into modern Italian is performed in literature. However, only a linguistic approach has been employed in pursuing this analysis [8]. This incorporates elements of phonology, grammar, logic and rhetoric, which are mainly connected to the spoken language. From this point of view, our approach is more oriented toward the grammatology, statistics and pattern recognition in order to discriminate evolving languages. To the best of our knowledge, we are the first to automate a discrimination process of a language in different historical periods by adopting statistical analysis and data mining approaches.

Usually, language discrimination methods are classified in the following groups: (i) n-gram methods, (ii) Markov model methods, (iii) hybrid methods, (iv) other methods.

Historically, the n-gram method is one of the first methods used for language discrimination and recognition [9]. In the n-gram method, the text is divided into smaller text parts, i.e. substrings. They have a length equal to n (n-gram). Furthermore, this method calculates character occurrence in the substrings (frequency) and creates a language model. In this way, a language model is created for each language, which is stored further in the language database. Analyzed texts are compared to the language database in order to recognize their language. According to n value, n-gram method is typically defined as: (i) uni-gram method, (ii) bi-gram method, (iii) tri-gram method, etc.

In recent times, many of these methods have been expanded further. Modification of a uni-gram method called *letter base scoring* method has been proposed in [10]. Essentially, it calculates letter frequency in the analyzed text. Furthermore, the letter frequency is multiplied by average letter frequency of a certain language, which is obtained from the training set. Then, classification is performed to discriminate different languages. The test was conducted on documents written in English, French, German, and Turkish. The obtained results were respectable if the number of words in the sentences was high (above 21). However, the differences between analyzed languages are rather high. Consequently, good results are expected.

An extension to the typical bi-gram method is given in [11]. The proposed algorithm finds bi-grams in which at least one of the constituent words has a minimum document frequency. In the training phase, all documents are scanned to find all bi-grams where at least one of their constituent uni-grams is a seed. In the end, the bi-grams with high occurrences are extracted. The study also supports combinations of uni-grams and bi-grams to improve correctness. Furthermore, the best results are obtained by a complex classification method, which contains a two-stage classifier.

The combination of uni-gram and bi-gram methods is also proposed in [13]. The study also compares the combined method with the uni-gram and bi-gram methods. Extracted feature vectors are classified further. The classification is performed by the Naive Bayes method. The obtained results represent a slight improvement over the uni-gram and the bi-gram methods.

Still, it is worth noting that n-gram methods need a very large piece of text for the training process. Furthermore, if the analysis of the text cannot match any language from the language database, then the results cannot be valued. In other words, this language is considered as unknown.

The Markov model method is typically used in combination with Bayesian decision rules [14]. During the training process, it produces models for each language by segmenting the strings and entering them into a so-called transition matrix. It contains an occurrence probability of all character sequences. In the end, the system chooses the language that gives the largest probability. It is worth noting that the training process is computer time intensive, because it needs at least 50-100K words, successfully testing small pieces of text whose length is above 100 characters [15].

A hybrid approach that combines optical character recognition and language recognition techniques is proposed in [16]. Essentially, it converts a text document into six character shape codes. Furthermore, a tri-gram method is applied to the character shape coded document. In the experiment, training was performed with extracted tokens (tri-grams). The token length was at least 5000 in order to differentiate languages. This method was able to discriminate 23 languages with overall accuracy above 90 %. It represents a good result because initial documents were scanned documents.

Another interesting method, which can be considered as different from the other methods is the frequent word-based approach [12]. It creates a language model by using the most frequent words. These words represent a set of words, which have the highest frequency occurrence in the analyzed text document. Also, a modification of the given technique uses a certain group of the most frequent words of size equal to W, where W is set to 100 [17] or to 1000 [18]. The training phase of the given technique needs a high number of documents to create the language model. Hence, it is computer time intensive and language recognition results are good with a rate above 95 %.

Previously, a few methods on script and language recognition, which use a graphic based approach, were proposed. Typically, they use coding [16, 19] or texture analysis [20]. The graphic based approach is established according to letter characteristics. However, we have two approaches: local and global ones. In the global methods, the text document is divided into bigger blocks of text prior to analysis. Upon extraction, the text blocks are subjected to Gabor filters and texture analysis in order to extract their characteristics [21]. The main limitation of these methods is that they don't take into account the characters of the text. In the local methods, each specific letter is considered in the text. In some studies, it is mapped into another code according to some of its characteristics [16, 19]. In this way, the coding converts initial text documents into another space. Then, appropriate tools are used to extract the features from the documents. The main limitations of these methods are that they are sensitive to noise and that they need prior knowledge of the script type in the text. Finally, both methods need some classification technique to differentiate scripts or languages.

In this paper, we have proposed a new method for clustering of documents, which are written in a language that have evolved during different historical periods, with the example of the Italian language. The proposed method includes elements using both global and local approaches, to overcome their limitations. In fact, it takes into account the characters of the text but it does not need prior knowledge of the script type. This method is composed of three main steps related to (i): script coding, (ii) texture analysis and (iii) clustering. The main novelty in the proposed method is the use of script coding, which is typical of some local methods, and the analogy between coded text and the image, which is characteristic of the global method. In this way, we use coding according to the horizontal energy level of each letter typical of the local approach. Hence, the text document is converted into another space, by introducing an innovative coding method using only four codes instead of the previously proposed six ones [16]. In this way, a group of consecutive codes representing the amplitude values can be interpreted with a grayscale image. Then, from the global approach we use elements of texture analysis to extract features from the given image. Here the number of initial variables is considerably reduced and an advanced clustering technique is mandatory. Hence, a new clustering strategy is introduced and employed on the document feature vectors. Our aim is to find the natural groups of documents identifying different historical periods in the language, without any prior knowledge about them. Detection of the different historical periods in the language is a new and suitable task, which can be in general problematic, due to the common characteristics of the language through time. A complete experimentation using a database of documents given in Italian Vulgar language and modern Italian language is performed. It demonstrates that our method obtains an accuracy of 100 % in finding the two historical periods of the Italian language. Furthermore, a comparison with widespread text-based language models such as bi-gram and tri-gram is presented. Our algorithm outperforms all other techniques. Also, an exhaustive comparison with other clustering algorithms well-known for document categorization shows the superiority of our approach in discrimination of the documents from two historical periods of the language. In the end, it should be pointed out that this proposed technique can be used for differentiation of closely related languages as well.

3 The proposed framework

The methodology for discrimination of documents in different historical periods of a language consists of the following phases: (i) script coding, (ii) texture analysis, and (iii) GA-ICDA$^+$ clustering. Figure 1 illustrates the main phases of the proposed method.

The input represents any text document, which is subjected to text coding. Still, our coding scheme differs considerably from the previously proposed ones [16, 19]. Firstly, the script coding maps the text of the document to a given set of numerical codes, based on the height of the letters in the text line derived from their horizontal energy profile. Other approaches typically use coding to reduce the number of variables. However, they follow the text-based approaches for language analysis creating tokens [16]. In our case, we transform the set of codes into the image space. Hence, each code is associated to a gray level, such that the coded text is transformed into a long 1-D image. Secondly, texture analysis by co-occurrence, local binary pattern (LBP), adjacent local binary pattern (ALBP), run-length or a combination of them is applied to the obtained image to extract document features for retrieving the feature vector of the document. Similar types of analysis (Gabor filtering and co-occurrence) have been previously performed but only on the whole image (global method) [21], and not on the local coded elements. Finally, text documents represented as texture feature vectors are grouped into clusters which represent sets of documents in different historical periods of the same language. Clustering is performed by adopting a new algorithm on the document feature vectors. Essentially, we introduce a method for finding relevant clusters of documents in different historical periods of the language. Relevance is measured in terms of uniformity of the language in the given historical period. The algorithm is called *Genetic Algorithms Image Clustering for Document Analysis-Plus* (GA-ICDA$^+$). It is a refinement of Genetic Algorithms Image Clustering for Document Analysis (GA-ICDA) method [22–24], which is suitable for discrimination of languages with small variants and a common root.

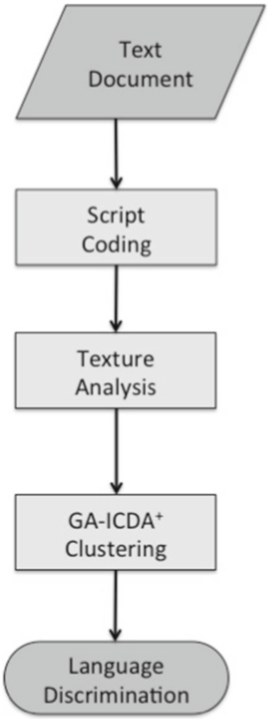


Fig. 1 Flow diagram of the proposed method

The proposed method performs favorably because each written language consists of different letters, which have some order in any text document. Essentially, we treat the text as a group of different letters with their accompanying horizontal energy level. According to that energy level, the letters are grouped into four different classes, each represented by a numerical code. Hence, the text is transformed into its codified version. Each document in any language is characterized by different occurring patterns, representing sequences of codes, which take different places in the text. The composition of these sequences and their occurrence in the text is different from language to language. Hence, script coding phase is crucial to differentiate documents written in different or evolving languages. Essentially, the *n*-gram technique uses only statistical combination and distribution of some sub-words (it depends on the level of *n*) in the text and its specific characteristics obtained by coding. Hence, it doesn't take into account the position and the order of these occurrences. On the contrary, our technique also uses the position of the pattern occurrence. A coded document consists only of these codes (numbers) and it can be seen as a signal. If we consider an analogy with the images, then the codes represent the gray levels of an image. Consequently, co-occurrence, run-length and ALBP texture features are adopted on the obtained image to capture the content, the position and the occurrence frequency of the patterns of coded information, determining the differentiation among the languages. Since the number of codes is considerably reduced (from six codes [16] to four codes of our approach), a strong feature extraction technique and a suitable classification tool are needed to differentiate initial documents given in different or evolving languages.

3.1 Script coding

The text document, which represents the input, can be: (i) a scanned printed document after preprocessing OCR stage (extracting each character by bounding boxes - character recognition is not needed), or (ii) a text given in unicode format. With this type of input, the script coding phase is applied. It is based on the differentiation of each character according to its horizontal energy profile in the text line. Energy profile technique is commonly used in a preprocessing stage of optical character recognition (OCR). Applying this technique, each character of the text document is classified differently based on its position in the text line. Figure 2 illustrates the differentiation of characters according to their energy profile in the text line.

Furthermore, the text document is segmented into text lines, composed of four different virtual lines [25], based on the energy of the script signs [26]. These virtual lines are called: (i) top-line, (ii) upper-line, (iii) base-line, and (iv) bottom-line. They correspond to the following vertical zones in the text line area: (i) upper, (ii) middle, and (iii) lower [25]. The letter categorization is performed by considering the position of the characters in vertical zones, which correspond to their energy profile. Hence, the short letters (S) are spread only along the middle zone, the ascender letters (A) include middle and upper zones, the descendent letters (D) occupy middle and lower zones, while the full letters (F) envelope all the vertical zones. Accordingly, they are classified into four categories corresponding to the four different script types [27]. The coding is performed from the script type to a numerical code. It is realized by considering the mapping from each of the four different script types to one of the four codes {0, 1, 2, 3}. Hence, each text document is given as a set of four codes, which are determined by the energy profile of each letter in the document. It considerably reduces the number of input variables, which will create a feature vector from each text document. Figure 3 shows the way of coding characters into different script codes.

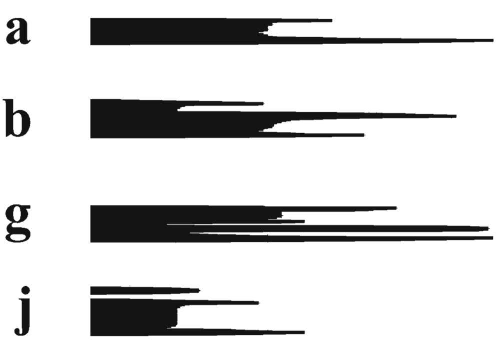


Fig. 2 Differentiation of characters based on their energy in the text line

Furthermore, the coded documents represent only a set of numbers. Considering an analogy with the image, these numbers can be seen as amplitude levels in a gray-scale image. Hence, each numerical code corresponds to a given gray level. Figure 4 illustrates two examples of mapping from numerical codes to gray levels, in two historical periods of the Italian language, which are Italian Vulgar language (top) and modern Italian language (bottom).

The text document is codified into a string of numerical codes, each corresponding to a gray level, then the document is translated into a 1-D gray level image **A**, enveloping a textural content. Hence, the texture analysis can be applied for extraction of textural features from the image **A**.

3.2 Texture analysis

The texture shows the image intensity changes. Hence, it can be used as a powerful discriminating feature. Texture is represented using spatial, frequency and perceptual characteristics [28]. Also, the texture attributes coarseness, smoothness, preferred direction, and periodicity which can be extracted. Hence, it is suitable for image retrieval using

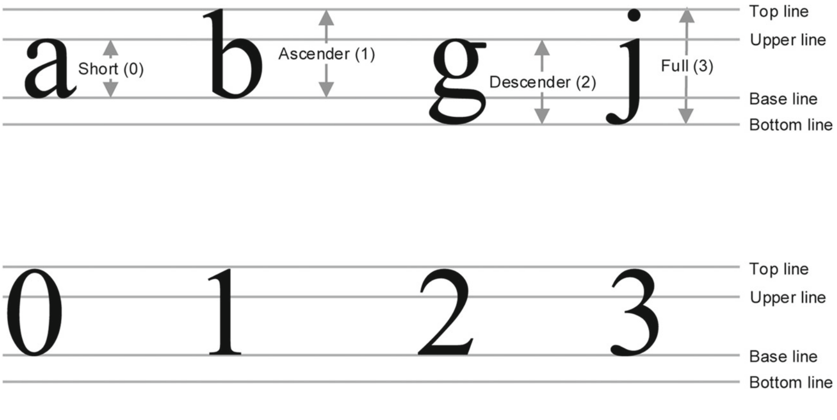
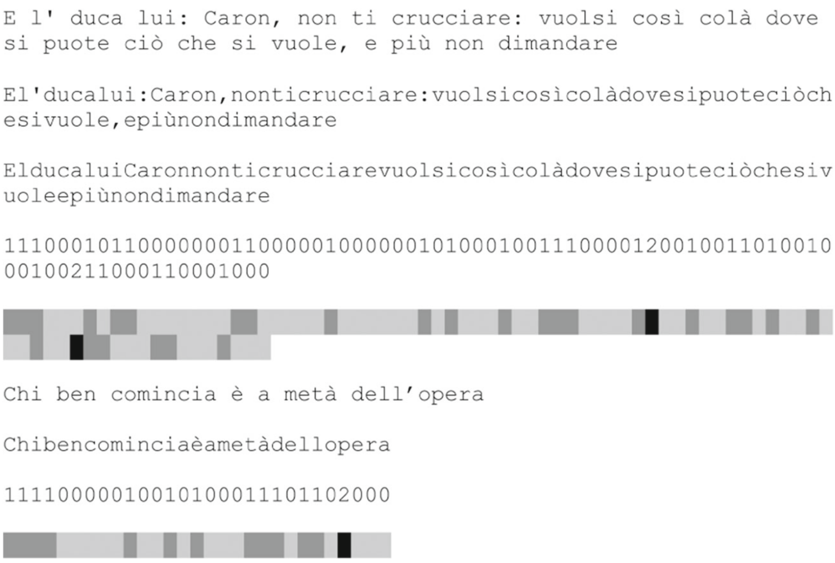


Fig. 3 Methodology of coding characters into different script codes

a texture (dis)similarity measure. Space-based techniques extracting texture descriptors among others are: (i) Co-occurrence matrix (COOC), (ii) Run-Length matrix (RLM), (iii) Local binary patterns (LBP), and (iv) Adjacent local binary patterns (ALBP). All aforementioned techniques use different ways of measuring textures by spatial distributions of gray level primitives. Let's suppose to have an image **A** with X rows, Y columns and L levels of gray. Our coded document represents a long 1-D set of four different numbers that corresponds to a gray level image. It is because the 2-D image of the document is transformed into 1-D image. In this process, the end of each row is continued at the beginning of the next row. Hence, an additional information of consecutive pixels is integrated into the newly created 1-D image. In our case, X is equal to 1 (one row), while Y representing the columns is equal to the number of characters in the initial document. Also, the number of grays given by L is equal to 4, because we have only 4 codes [29]. Furthermore, we need to extract the co-occurrence, run-length and (A)LBP characteristics. All these characteristics form a unique feature vector, which can be used for classification tasking and information retrieval.

3.2.1 Co-occurrence

Co-occurrence texture analysis is used to extract elements of the feature vector. The core of this analysis is the extraction of the gray level co-occurrence matrix **C**. It is a basis for the calculation of all primary or secondary co-occurrence characteristics. Co-occurrence matrix **C** represents a square matrix, whose dimension is given by the number of gray levels L, i.e. 4×4. To calculate the elements of **C**, a central pixel of the image $A(x, y)$ with a neighborhood of pixels has been taken into consideration. It creates the so-called window of interest (WOI), whose dimension is defined by inter-pixel distance d and orientation θ. Usually, d is set to 1, which means that only the first neighbor pixel is used. Furthermore, we have a 1-D image which incorporates only


E l' duca lui: Caron, non ti crucciare: vuolsi così colà dove
si puote ciò che si vuole, e più non dimandare

El'ducalui:Caron,nonticrucciare:vuolsicosìcolàdovesipuoteciòch
esivuole,epiùnondimandare

EllducaluiCaronnonticrucciarevuolsicosìcolàdovesipuoteciòchesiv
uoleepiùnondimandare

11100010110000000110000010000001010001001110000120010011010010
00100211000110001000

Chi ben comincia è a metà dell'opera

Chibencominciaèametàdellopera

11110000010010100011101102000

Fig. 4 Mapping of the letters to numerical codes and corresponding gray levels in Italian Vulgar language *(top)* and modern Italian language *(bottom)*. On the *top*, a sentence from Dante Alighieri's Commedia is depicted (distribution of script types: S = 64.63 %, A = 32.93 %, D = 2.44 %, F = 0.00 %), while at the *bottom*, a well-known Italian proverb is reported ("Chi ben comincia è a metá dell'opera") (distribution of script types: S = 55.17 %, A = 41.38 %, D = 3.45 %, F = 0.00 %)

$\theta = 0°$. Hence, it has a 2-connected neighborhood to be considered. From [30] we obtain the following:

$$C(i, j) = \sum_{x=1}^{X} \sum_{y=1}^{Y} \begin{cases} 1 & \text{if } A(x, y) = i, \text{ and } A(x + \Delta x, y + \Delta y) = j, \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where i and j represent the intensity values of the image $\mathbf{A}$, while $d = 1$ is the distance between the pixel of interest and its neighbor. In our case X is equal to 1. Furthermore, normalized matrix $\mathbf{P}$ obtained from the co-occurrence matrix $\mathbf{C}$ is calculated as [31]:

$$P(i, j) = \frac{C(i, j)}{\sum_i^L \sum_j^L C(i, j)}. \quad (2)$$

From matrix $\mathbf{P}$ we can extract all co-occurrence texture features. They are divided into first and second order statistical features. The first order texture features represent the first 4 elements of the co-occurrence feature vector, while the second order statistical features typically include 8 elements. Accompanying the first order statistical features, co-occurrence statistics extract a total of 12 features. They are defined in Table 1 [30, 31].

All these measures create a 12-dimensional co-occurrence feature vector.

Table 1 First and second order co-occurrence statistical measures

First order	$\mu_x = \sum_{i=1}^L i \sum_{j=1}^L P(i, j)$,
	$\mu_y = \sum_{j=1}^L j \sum_{i=1}^L P(i, j)$,
	$\sigma_x = \sqrt{\sum_{i=1}^L (i - \mu_x)^2 \sum_{j=1}^L P(i, j)}$,
	$\sigma_y = \sqrt{\sum_{j=1}^L (j - \mu_y)^2 \sum_{i=1}^L P(i, j)}$.
Second order	$\text{Correlation} = \sum_{i=1}^L \sum_{j=1}^L \frac{(i \cdot j) \cdot P(i,j) - (\mu_x \cdot \mu_y)}{\sigma_x \cdot \sigma_y}$,
	$\text{Energy} = \sum_{i=1}^L \sum_{j=1}^L P(i, j)^2$,
	$\text{Entropy} = -\sum_{i=1}^L \sum_{j=1}^L P(i, j) \cdot log P(i, j)$,
	$\text{Maximum} = \max\{P(i, j)\}$,
	$\text{Dissimilarity} = \sum_{i=1}^L \sum_{j=1}^L P(i, j) \cdot \lvert i - j \rvert$,
	$\text{Contrast} = \sum_{i=1}^L \sum_{j=1}^L P(i, j) \cdot (i - j)^2$,
	$\text{Invdmoment} = \sum_{i=1}^L \sum_{j=1}^L \frac{1}{1+(i-j)^2} P(i, j)$,
	$\text{Homogeneity} = \sum_{i=1}^L \sum_{j=1}^L \frac{P(i,j)}{1+\lvert i-j \rvert}$.

3.2.2 Run-length

To extract run-length characteristics, we calculate the run-length matrix $p(x, y)$. The elements of the matrix represent the number of runs with pixels of gray level $x = 1, ..., L$ and run length $y = 1, ..., N$. Accordingly, L represents the number of gray levels, while N is the maximum run length. Hence, each member of $p(x, y)$ represents the gray level run-length of image $\mathbf{A}$ that gives the total number of occurrences of gray level runs of length y and of intensity value x.

A sequence of consecutive pixels with identical or similar intensity values creates a gray level run. Firstly, the following five run-length features have been proposed [32]: (i) Short run emphasis (SRE), (ii) Long run emphasis (LRE), (iii) gray level non-uniformity (GLN), (iv) Run length non-uniformity (RLN), and (v) Run percentage (RP). Additional two run-length features proposed in [33] are: (i) Low gray level run emphasis (LGRE) and (ii) High gray level run emphasis (HGRE). Further extension to run-length features is given in [34]. It proposes the following new run-length features: (i) Short run low gray level emphasis (SRLGE), (ii) Short run high gray level emphasis (SRHGE), (iii) Long run low gray level emphasis (LRLGE), and (iv) Long run high gray level emphasis (LRHGE). All aforementioned run-length statistical measures are defined in Table 2. N_r represents the total number of runs, n_p is the number of pixels of image $\mathbf{A}$.

Run-length statistics extract 11 features, which can be given as a 11-dimensional feature vector.

3.2.3 LBP and ALBP

Local binary pattern (LBP) is a texture operator. It extracts features according to a local binary partition from the image texture. For each pixel $A(x, y)$ of the 1-D image, its two neighbors are considered. If the intensity of any of these pixels is greater than the intensity of the pixel under consideration $A(x, y)$, then LBP receives the value 1, otherwise 0. In this way, the obtained values are determined by a thresholding procedure in response to the intensity of the pixel $A(x, y)$. LBP is calculated as [35]:

$$b_i(y) = \begin{cases} 1, & A(x, y) < A(x, y + \Delta y_i) \\ 0, & \text{otherwise} \end{cases}, \quad (3)$$

where $i = 1, ..., N_n$, and N_n is the number of neighbor pixels equal to 2. Furthermore, Δy_i is the displacement vector equal to 1 from the position of center pixel y to neighbor pixels. Furthermore, the two neighbors are used as a basis to create a 2-digit binary label. As a result of the image analysis, the distributions of 2-digits labels are obtained. They represent the features of the analyzed image texture.

Table 2 Run-length statistical measures

Galloway run-length features	$SRE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}\frac{p(x,y)}{y^2}$,
	$LRE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}p(x,y)\cdot y^2$,
	$GLN = \frac{1}{n_r}\sum_{x=1}^{L}(\sum_{y=1}^{N}p(x,y))^2$,
	$RLN = \frac{1}{n_r}\sum_{y=1}^{N}(\sum_{x=1}^{L}p(x,y))^2$,
	$RP = \frac{n_r}{n_p}$.
Chu run-length features	$LGRE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}\frac{p(x,y)}{x^2}$,
	$HGRE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}p(x,y)\cdot x^2$.
Dasarathy run-length features	$SRLGE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}\frac{p(x,y)}{x^2\cdot y^2}$,
	$SRHGE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}\frac{p(x,y)\cdot x^2}{y^2}$,
	$LRLGE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}\frac{p(x,y)\cdot y^2}{x^2}$,
	$LRHGE = \frac{1}{n_r}\sum_{x=1}^{L}\sum_{y=1}^{N}p(x,y)\cdot x^2\cdot y^2$.

Adjacent local binary pattern (ALBP) is proposed as an extension to the LBP. It represents a co-occurrence LBP. The procedure for ALBP extraction consists of two stages [36]: (i) LBP(+) and (ii) LBP(×). LBP(+) considers only the direction given by sign +, i.e. two horizontal and two vertical pixels in the neighborhood of the center pixel, while LBP(×) considers only the direction given by sign ×, i.e. the four diagonal pixels in the neighborhood of the center pixel. In our case of 1-D image, LBP(+) only matters. The horizontal combination of two local 2-digit binary labels creates a 4-bit binary label. Hence, ALBP is calculated as follows:

$$a_i(y) = b_i(y)(+)b_{i+1}(y). \tag{4}$$

Furthermore, it gives a total number of $2^4 = 16$ different labels creating a feature vector of 16 elements.

3.3 GA-ICDA$^+$ clustering

Distinguishing variants of a language derived from a common root is a critical aspect in natural language processing and information retrieval. Next, we present GA-ICDA$^+$ which is a modified version of GA-ICDA whose aim is, specifically, the discrimination of language variants in different historical periods, with the example of the Italian language. Next, we further explain the main concepts underlying GA-ICDA clustering and introduce the modifications applied on GA-ICDA for defining GA-ICDA$^+$.

GA-ICDA represents a bottom-up clustering strategy, where the set of documents written in different languages or scripts is modeled as a weighted graph $G = (V, E, Z)$ and V is the set of nodes, E is the set of edges and Z is the set of weights. Let $v_i \in V$ be a node in G and $e_{ij} \in E$ an edge between two nodes v_i and v_j. Then, v_i represents a document written in the given language and e_{ij} is a connection between the corresponding documents of the nodes v_i and v_j. A weight $z_{ij} \in Z$ is related to the edge e_{ij}, quantifying the similarity between the corresponding documents of the nodes v_i and v_j. Each node v_i is linked to a restricted number of neighbor nodes in G, determining the h-nearest neighbors of v_i, denoted as $nn_{v_i}^h = \{nn_{v_i}^h(1), ..., nn_{v_i}^h(k)\}$, where h is not the number of neighbors but a parameter determining such a number for each node, and k is the number of h-nearest neighbors [37]. This set is composed of the k nodes of G whose corresponding documents have a similarity value included in the h-highest similarity values with v_i. Similarity between two nodes v_i and v_j is evaluated as:

$$z_{ij} = e^{-\frac{d(i,j)^2}{a^2}}, \tag{5}$$

where a is a scale parameter and $d(i, j)$ is a distance function, i.e. L_1 distance, between the corresponding document feature vectors of v_i and v_j.

Starting from $nn_{v_i}^h$, only a subset of h-nearest neighbors is considered, including those nodes which are spatially close to v_i, established a given ordering node. Specifically, considering a mapping f of each node in V to an integer label, $f : V \rightarrow \{1, 2, .., n\}$ $n = |V|$, the difference is evaluated between the label associated to the node v_i, $f(v_i)$, and the labels associated to the nodes in $nn_{v_i}^h$, $f(nn_{v_i}^h(1)), ..., f(nn_{v_i}^h(k))$. This label difference is expressed as:

$$|f(v_i) - f(nn_{v_i}^h(j))| \quad j = 1...k. \tag{6}$$

Each node v_i in G is linked only to the nodes in $nn_{v_i}^h$ whose label difference is less than an established threshold value T. The obtained edges between the nodes define the adjacency matrix $\mathbf{W}$ of G. A genetic algorithm is applied on $\mathbf{W}$ to detect node groups corresponding to document clusters. At the end of this procedure, the clusters are subjected to complete link clustering strategy for "correcting" the local optima. Specifically, each pair of clusters $< c_i, c_j >$ having the minimum distance to each other is merged, continuing repeatedly until a fixed number of clusters, nc, is established. The distance between two clusters is calculated as the distance (i.e. L_1 distance) between the two document feature vectors which are the farthest from each other, one for each cluster.

GA-ICDA$^+$ presents two main modifications with respect to the previously introduced GA-ICDA algorithm. In this context, the nodes of G represent the documents given in the same language but in different historical periods.

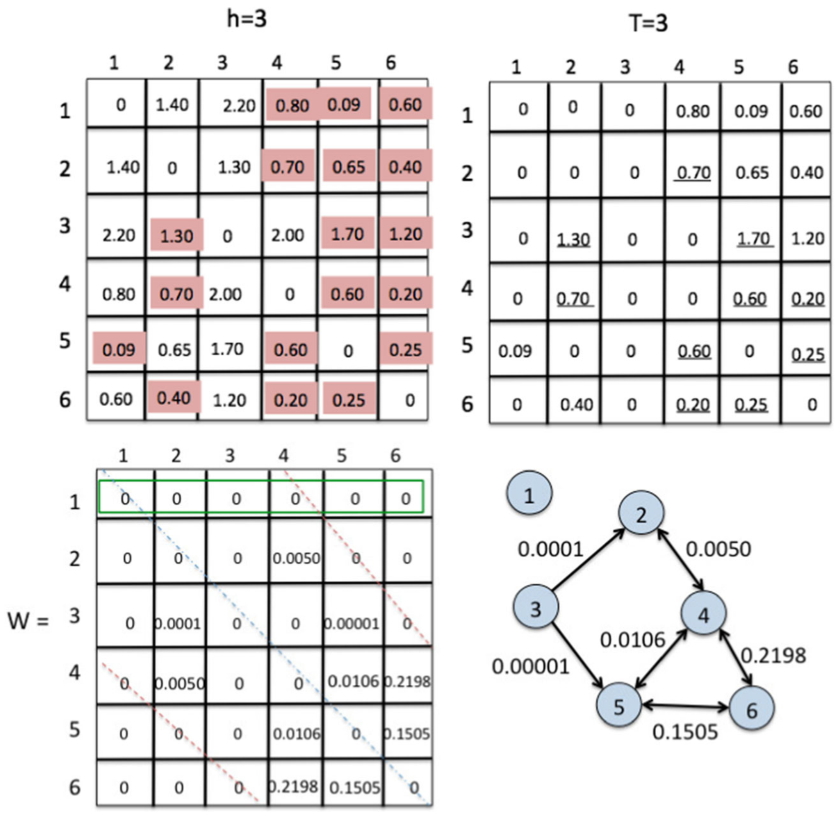
The first modification consists in a different characterization of the similarity between the nodes of G. Specifically, language variants in different historical periods exhibit a composite inner structure. Accordingly, the distance values calculated between the corresponding document feature vectors can unjustifiably be much higher. This fact determines a problem in the similarity computation in (5). In fact, let $v_i, v_j \in V$ be two nodes and d_i, d_j their associated document feature vectors. If the computed distance $d(i, j)$ between d_i and d_j is particularly high, because of the presence of the power of 2 at the numerator of the exponent and of the minus sign to the exponent in (5), the related similarity value z_{ij} will be zero. This phenomenon often may occur for multiple pairs of document feature vectors. The result is that $\mathbf{W}$ containing the similarity values for each pair of nodes associated to document feature vectors will be unjustifiably more sparse than normal. In order to overcome this problem, we substitute the exponent of the numerator $d(i, j)$ in (5) with a parameter α, instead of fixing its value to 2. It determines a more flexible and "softer" definition of the similarity values. Consequently, z_{ij} in (5) is defined as:

$$z_{ij} = e^{-\frac{d(i,j)^{\alpha}}{a^2}}. \tag{7}$$

The second proposed modification consists of a variation in graph construction. In particular, we start from the second phase of the method where, for each node v_i, only the h-nearest neighbors which are "spatially" near to v_i are considered as neighbors of v_i, for a given node ordering f. This phase generates a reduction in the number of neighbors for v_i and consequently in the number of edges outgoing v_i, easily characterizing the graph connected components. In the case of complex document graphs, such as in the context of documents written in the same language in different historical periods, it could be more suitable to have a low value of the threshold parameter T for defining satisfactory connected components. However, this choice may determine the presence of isolated nodes, for which all the associated h-nearest neighbors are excluded due to the low T value. GA-ICDA is not subjected to this situation. In fact, T is never fixed so low to determine isolated nodes in G. In GA-ICDA$^+$ this constraint is passed by managing the presence of isolated nodes inside the genetic procedure. An isolated node v_h is considered as a "singleton" node from the genetic algorithm. In fact, the algorithm cannot include v_h inside any connected component, because v_h doesn't have any neighbors. When the genetic algorithm ends returning graph connected components, v_h will also be returned as a cluster composed of a single node. Then, the complete link clustering procedure will try to automatically envelope v_h inside one of the other detected clusters, until the fixed cluster number is reached. Such a modification allows to choose a low T value when needed.

Figure 5 shows the graph construction procedure on a toy database of 6 documents. The first matrix represents the distance matrix, where each row i contains the L_1 distance values computed between the corresponding document and the other documents in the database. For each row i, the 3-nearest neighbors nn_i^3 are computed (in red), which are the documents associated to the 3-lowest distance values, i.e. in row 1, $nn_1^3 = \{4, 5, 6\}$. The second matrix illustrates the computed 3-nearest neighbors. For each row i, only a subset of neighbors is selected (underlined values), whose label difference with i is less than $T = 3$. For example, at row 3, we have the 3-nearest neighbors $nn_3^3 = \{2, 5, 6\}$ but we select only two 3-nearest neighbors $\underline{nn}_3^3 = \{2, 5\}$, because their label difference $|3-2| = 1 < 3$ and $|3-5| = 2 < 3$. Document 6 is not selected because the corresponding label difference is $|3-6| = 3$. Because of the low threshold value $T = 3$, document 1 remains isolated. In fact, it does not have connections with the other documents, having all the distance values equal to zero. The third matrix reports the similarity values computed from the previously selected distance values, by using (7) with $\alpha = 1$, defining the adjacency matrix $\mathbf{W}$. Finally, the corresponding graph G is depicted, where node 1, corresponding to row 1, remains isolated.

Figure 6 shows an example of GA-ICDA$^+$ execution. Genetic procedure, executed on the graph, determines three document clusters, c_1, c_2 and c_3. Because node 1 does not have connections with other nodes in G, it is returned as a singleton cluster c_1 from the procedure. We fix a number of clusters $nc = 2$. Starting from the detected clusters c_1, c_2 and c_3, in the first iteration of the algorithm, the L_1 distance among them is computed, for obtaining $D(c_1, c_2)$, $D(c_1, c_3)$ and $D(c_2, c_3)$. Distance between two clusters is evaluated as the distance between the two furthest documents, one for each cluster. For example, let's consider the two clusters $c_2 = \{2, 3\}$ and $c_3 = \{4, 5, 6\}$. Looking at the distance of every possible document pair, in particular, $<2-4>$, $<2-5>$, $<2-6>$, $<3-4>$, $<3-5>$, $<3-6>$, in the distance matrix in Fig. 5, we can observe that the maximum distance value of 2.00 is obtained for the document pair $<3-4>$. Consequently, $D(c_2, c_3)$ will be 2.00. Distance is computed also for clusters c_1, c_2 and c_1, c_3. In the end, the minimum cluster distance is selected, which is between clusters c_1 and c_3, equal to 0.8. Consequently, the two clusters c_1 and c_3 are merged into a new cluster c_1'. The final clusters of this iteration are $c_1' = \{1, 4, 5, 6\}$ and $c_2' = \{2, 3\}$. Since nc is fixed to 2, the algorithm ends.



h=3

	1	2	3	4	5	6
1	0	1.40	2.20	0.80	0.09	0.60
2	1.40	0	1.30	0.70	0.65	0.40
3	2.20	1.30	0	2.00	1.70	1.20
4	0.80	0.70	2.00	0	0.60	0.20
5	0.09	0.65	1.70	0.60	0	0.25
6	0.60	0.40	1.20	0.20	0.25	0

T=3

	1	2	3	4	5	6
1	0	0	0	0.80	0.09	0.60
2	0	0	0	0.70	0.65	0.40
3	0	1.30	0	0	1.70	1.20
4	0	0.70	0	0	0.60	0.20
5	0.09	0	0	0.60	0	0.25
6	0	0.40	0	0.20	0.25	0

W =

	1	2	3	4	5	6
1	0	0	0	0	0	0
2	0	0	0	0.0050	0	0
3	0	0.0001	0	0	0.00001	0
4	0	0.0050	0	0	0.0106	0.2198
5	0	0	0	0.0106	0	0.1505
6	0	0	0	0.2198	0.1505	0

Fig. 5 Graph construction. From *left* to *right*: detection of the 3-nearest neighbors for each node (row) of the distance matrix; selection of the closest nodes with threshold T = 3 from the 3-nearest neighbors of each node (row) of the distance matrix; similarity matrix computation from distance matrix (node 1 remains isolated); construction of the graph G from similarity matrix corresponding to adjacency matrix **W**

4 Experiment

Experiment is performed on a custom oriented database[1] composed of 50 documents in Italian language from two historical periods of Italian Vulgar and of modern Italian. All the documents are characterized by a number of characters ranging from 400 to 6000.

Modern Italian language counts 40 documents, composed of 20 text excerpts from two well-known Italian newspapers, *Il Sole 24Ore* and *La Repubblica*, and 20 text excerpts from online fiction books and websites discussing general topics. The difference in writing typology between documents extracted from newspapers and documents extracted from fiction books and websites is noticeable. In fact, the first ones exhibit a more “technical” writing style with respect to the second ones. In this way, we try to capture the most important facets of modern Italian language in both spoken and written styles.

Italian Vulgar language counts 10 documents. In particular, 7 documents are text excerpts from Dante Alighieri’s *Commedia*, 1 document is a text excerpt from Francesco Petrarca’s *Canzoniere*, 1 document is extracted from Stefano Protonotaro’s poem *Pir meu cori alligrari* and a last document is taken from Cecco Angiolieri’s rhymes *Or non é gran pistolenza la mia*. Documents cover the historical period from 1260 AC to 1374 AC. They involve the three important Tuscan writers in Vulgar language, Dante Alighieri, Francesco Petrarca and Cecco Angiolieri, and the well-known writer belonging to the Sicilian poetical school, Stefano Protonotaro, having its best characterization around

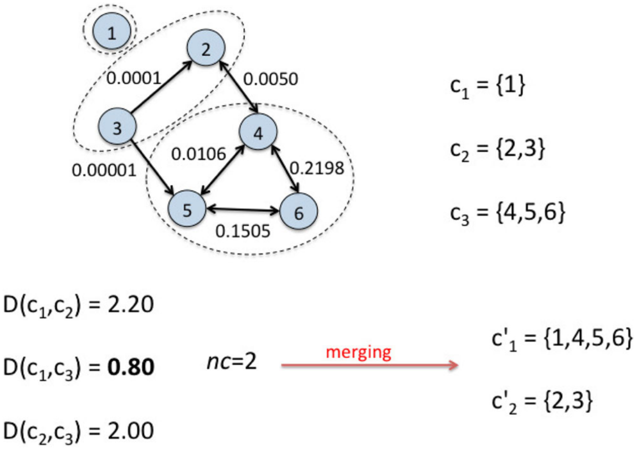


Fig. 6 GA-ICDA$^+$ execution. c_1, c_2 and c_3 represent the clusters detected on the graph G from the genetic procedure. $D(c_i, c_j)$, $i = 1 \ldots 3$, $j = 1 \ldots 3$, $i \neq j$, are the distance values computed between the clusters c_i, c_j. A fixed number of clusters nc=2, the merging procedure is applied, selecting the minimum cluster distance $D(c_1, c_3)$=0.8 and merging the corresponding clusters c_1 and c_3. c'_1 and c'_2 are the final clusters after a merging procedure

[1] The database is freely available at https://sites.google.com/site/documentanalysis2015/italian-italian-vulgar-database.

the court of Federich II of Svevia and determining Sicilian vulgar language [38]. Furthermore, the writing style of Dante Alighieri is defined as “stil novo”, that influenced also the style of Francesco Petrarca, characterized by a sweet, accurate, raffinate and noble choice of terms and sentences in writing [39], opposed to the “comic” and realistic writing style of Cecco Angiolieri [40]. Because of this composite model, Italian Vulgar language discrimination becomes a real challenge. In fact, the selected documents extracted from well-known poems of different authors, are chosen to maximize the coverage of the ancient historical period when Vulgar language had mainly been spreading.

5 Results and discussion

The framework, in terms of feature representation and clustering method is evaluated for detection of groups of documents in the two historical periods of Italian language from the aforementioned database. In our case, the class (Italian Vulgar or modern Italian) in the documents found in the database is only useful to evaluate in a post-processing phase the accuracy of solutions from the clustering algorithm. The algorithm does not know historical periods of the language in the documents. Hence, it has to find it by performing clustering.

Combinations of the proposed feature representations (co-occurrence, run-length, LBP, and ALBP) have been tested on a benchmark database different from the database adopted for experimentation. Clustering by GA-ICDA$^+$ has been employed on the documents given in these different feature combinations. Analysis has revealed that co-occurrence features are not able to obtain the perfect language discrimination in this context. Hence, the combination of feature representations, which is particularly suitable for this task is run-length statistics and ALBP patterns. Consequently, document feature vectors are composed of 11 run-length statistical features and 16 ALBP features, for a total of 27 features.

Figure 7 depicts the distribution of the feature values for the database of Italian Vulgar and modern Italian documents. In particular, the median and the quartiles (25 % and 75 %) are computed for each run-length and ALBP feature of the database. Then, values (y axis) of these statistical measures are reported in correspondence to each feature (x axis). ALBP features ’0110’, ’0111’, ’1110’, ’1111’ have zero values, consequently they are not reported inside the figure. Looking at the feature distribution, it is worth noting that some features determine an acceptable separation between the two variants of the Italian language, such as GLN, ’0101’ and ’1001’. However, perfect discrimination of Italian Vulgar and modern Italian remains a particularly complex task. In fact, looking at the overall distribution, it is observable that the median and quartiles of the two language classes exhibit some intersection points that don’t allow for an easy separation of the documents in the two variants of the Italian language by using a general-purpose clustering strategy.

The clustering solution obtained from our framework (run-length and ALBP features together with GA-ICDA$^+$) has been compared with: (i) the solution found from other clustering algorithms when applied on the same run-length and ALBP features, and (ii) the solution found from GA-ICDA$^+$ and other clustering algorithms on different text-based feature representations. Hence, multiple combinations of feature representations and clustering methods are under consideration. The adopted clustering algorithms are GA-ICDA [23], where isolated nodes are not considered and the concept of α parameter is not introduced, Genetic Algorithms Image Clustering (GA-IC) [37], an image database clustering algorithm adopted in this context for document image database clustering, and five other clustering methods well-known for document discrimination [41–45], which are: bottom-up hierarchical clustering (Hierarchical) [46], Self-Organizing-Map (SOM) [47], Spherical K-Means (SK-Means) [48] and Expectation Maximization (EM) [49]. The different text-based feature representation is based on the n-gram language model, which is the most cognate to our feature representation in literature. In particular, the documents are represented by bi-gram and tri-gram frequency vectors [50].

The performance measures employed for the evaluation are precision, recall and f-measure [51], purity, entropy, Normalized Mutual Information (NMI) [52] and Adjusted Rand Index (ARI) [53, 54], adopted in multiple contexts for document clustering evaluation [55–57]. Precision, recall and f-measure have been computed for each language class (Italian Vulgar and modern Italian), while purity, entropy, NMI and ARI have been calculated as a single value for all the language classes. Furthermore, precision, recall and f-measure require the correspondence between the clusters found from the algorithm and language classes, which is not known. Consequently, each cluster is associated to that language class whose corresponding documents are in majority inside that cluster.

A trial and error procedure has been adopted on benchmark documents in the same languages, different from the database, for selecting the GA-ICDA$^+$ parameter values giving the best possible solution [58]. This means that algorithm has been executed multiple times with different combinations of parameters and the performance measures computed on each found solution. Then, the parameter combination providing the best results in terms of performance measures has been employed in GA-ICDA$^+$ for clustering of the custom oriented database in Italian Vulgar and modern Italian languages. Consequently, the h parameter of the

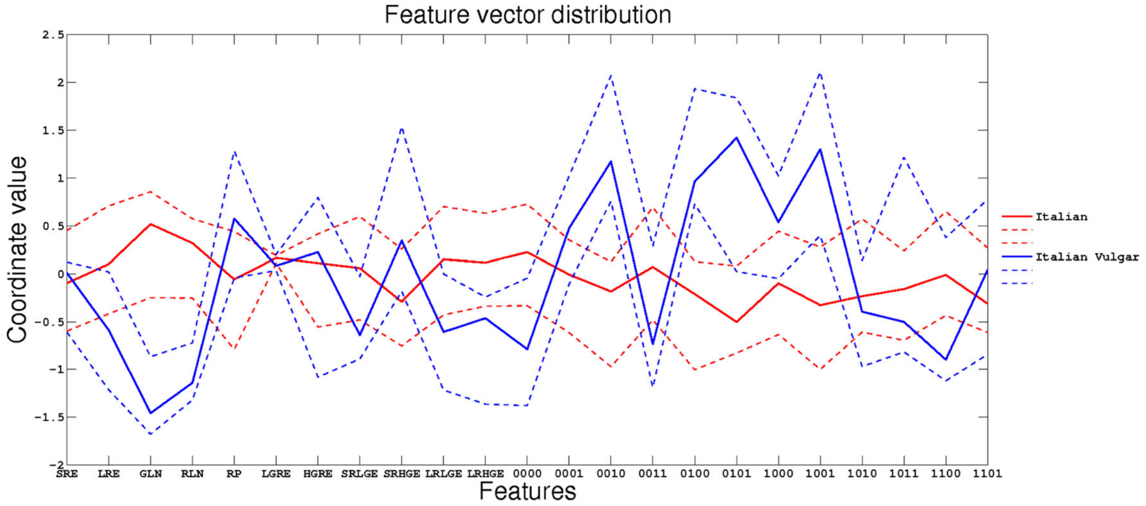


Fig. 7 Feature value distribution for the database of documents given in Italian Vulgar and modern Italian. Only ALBP features different from zero are reported. In particular for each language class, the values of median (*continuous line*) and quartiles, 25 % and 75 %, (*dotted line*) along the y axis are showed for each feature (x axis)

neighborhood is fixed to 33 and the T threshold parameter value is fixed to 7. Furthermore, the α parameter for similarity computation from the distance matrix is equal to 1, and the cluster number nc is equal to 2. The parameters of the genetic algorithm, population size is fixed to 700, number of generations is 200, probability of mutation is equal to 0.7 and probability of crossover is equal to 1. Furthermore, elite reproduction is fixed to 10 % of the population size and roulette selection function is adopted. Also, the trial and error procedure has established that cosine similarity is the most suitable measure for comparison of document feature vectors in the complete link procedure, while the L_1 norm is particularly apt to deal with distance computation in the graph construction phase. The same trial and error procedure has been employed for parameter tuning of GA-ICDA and GA-IC. For GA-ICDA, the h and T parameter values are fixed respectively to 10 and 30. In GA-IC, the h parameter value is also equal to 10. The same genetic parameter values adopted in GA-ICDA$^+$ have been selected for GA-ICDA and GA-IC. Furthermore, GA-ICDA uses the L_1 norm for distance computation in graph construction and in the complete link procedure. GA-IC also uses the L_1 norm for distance computation in the graph construction procedure.

In order to realize a fair comparison with GA-ICDA$^+$, hierarchical clustering employs the cosine similarity for document feature vector similarity. A horizontal cut is performed on the obtained dendrogram in order to detect a total of 2 clusters. SOM employs neuron layers of dimension 1 × 2. In order to initially cover the input area, the number of training phases is fixed to 100, while the initial neighborhood dimension is 2. Two neurons exhibit a distance which is equal to the number of steps dividing one neuron to the other. In K-Means algorithm, the cosine similarity is also used for document feature vector comparison (SK-Means). The number of clusters established for the SK-Means algorithm is 2. In EM, the number of clusters is also fixed to 2. All the algorithms, except that of hierarchical clustering, determine a solution which can be different depending on the execution, having been run 100 times on the same database. The average and standard deviation values of the performance measures have been computed and reported, together with the number of clusters detected from the algorithms.

Table 3 shows the results obtained from our feature representation as input to the different clustering algorithms. The proposed approach (run-length and ALBP features together with GA-ICDA$^+$) is reported as GA-ICDA$^+$. Looking at the results, it is worth noting that our framework obtains the best solution corresponding to the perfect discrimination of the Italian documents in the two historical periods of the language, outperforming all the other clustering algorithms in terms of performance measures. In fact, it reaches the value of 1.00 for precision, recall and f-measure in all language classes, the value of 1.00 for purity, NMI and ARI and the value of 0.00 for entropy. Furthermore, the standard deviation is always zero. It demonstrates that modifications introduced in GA-ICDA$^+$ determine local optimum solutions that are always successfully refined from the complete link procedure for all the runs. On the contrary, when modifications are not considered, like in the case of GA-ICDA, the complete link procedure is not always able to perfectly "correct" the local optimum solutions, failing in some runs to detect the perfect solution. In fact, the standard deviation of GA-ICDA is different from zero. Furthermore, GA-ICDA performs more poorly than GA-ICDA$^+$, obtaining f-measure values of 0.93 and 0.53 for respectively modern Italian and Italian Vulgar languages, a value of 0.88 for

Table 3 Clustering results in terms of average precision, recall, f-measure, purity, entropy, NMI and ARI for each class of documents written in modern Italian and Italian Vulgar languages, from the following methods: Genetic Algorithms Image Clustering for Document Analysis-Plus (GA-ICDA$^+$), Genetic Algorithms Image Clustering for Document Analysis (GA-ICDA), Genetic Algorithms Image Clustering (GA-IC), bottom-up hierarchical clustering (Hierarchical), Self Organizing Map (SOM), spherical K-Means (SK-Means) and Expectation Maximization (EM)

Algorithm	Class	Precision	Recall	f-meas.	NMI	Purity	Entropy	ARI	nc
GA-ICDA$^+$	Modern Italian	1.0000	1.0000	1.0000	1.0000	1.0000	0.0000	1.0000	2
		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	Italian Vulgar	1.0000	1.0000	1.0000					
		(0.0000)	(0.0000)	(0.0000)					
GA-ICDA	Modern Italian	0.8810	0.9875	0.9302	0.3398	0.8800	0.1565	0.4342	2
		(0.0502)	(0.0132)	(0.0222)	(0.1441)	(0.0422)	(0.0279)	(0.2210)	
	Italian Vulgar	0.5208	0.7500	0.5268					
		(0.3733)	(0.0527)	(0.2645)					
GA-IC	Modern Italian	1.0000	0.2550	0.4043	0.1899	0.8760	0.6578	0.0668	7
		(0.0000)	(0.0483)	(0.0595)	(0.0070)	(0.0084)	(0.1317)	(0.0190)	
	Italian Vulgar	0.7939	0.5400	0.6333					
		(0.0830)	(0.0843)	(0.0176)					
Hierarchical	Modern Italian	0.7959	0.9750	0.8764	0.0151	0.8000	0.1460	0.0001	2
		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	Italian Vulgar	0.2041	1.0000	0.3390					
		(0.0000)	(0.0000)	(0.0000)					
SOM	Modern Italian	0.9674	0.7415	0.8395	0.2488	0.8000	0.2407	0.2766	2
		(0.0005)	(0.0119)	(0.0079)	(0.0104)	(0.0000)	(0.0000)	(0.0190)	
	Italian Vulgar	0.4656	0.9000	0.6137					
		(0.0113)	(0.0000)	(0.0099)					
SK-Means	Modern Italian	0.7959	0.9750	0.8764	0.0151	0.8000	0.1460	0.0001	2
		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	Italian Vulgar	0.2041	1.0000	0.3390					
		(0.0000)	(0.0000)	(0.0000)					
EM	Modern Italian	0.8945	0.7625	0.8144	0.2103	0.8182	0.2195	0.1999	2
		(0.0966)	(0.1375)	(0.0940)	(0.0997)	(0.0274)	(0.0380)	(0.2193)	
	Italian Vulgar	0.4171	0.8840	0.5401					
		(0.1624)	(0.1152)	(0.1368)					

Standard deviation is reported in parenthesis. *nc* is the average number of document clusters found with the algorithms. Documents are represented by vectors of run-length and ALBP features of the coded text (proposed feature approach)

purity, a value of 0.16 for entropy, a value of 0.34 for NMI and a value of 0.43 for ARI. It demonstrates the utility of the modifications introduced in GA-ICDA$^+$ to find the perfect discrimination of the language in different historical periods. GA-IC doesn't include the concept of spatial closeness of graph nodes, the complete link procedure, the presence of isolated nodes and the modification of the α parameter, which are instead present in GA-ICDA$^+$ and GA-ICDA. Consequently, it reaches the worst performances. In fact, it exhibits an f-measure value of 0.40 for modern Italian language and of 0.63 for Italian Vulgar language, a value of 0.88 for purity, 0.66 for entropy, 0.19 for NMI and 0.07 for ARI. It demonstrates that GA-IC, which is currently adopted for image database clustering, is not apt to deal with the problem of document discrimination in multiple languages, especially in the context of language evolution. Although hierarchical clustering adopts a bottom-up agglomerative strategy which is similar to the complete link procedure of GA-ICDA$^+$ and GA-ICDA, it determines worse results than those obtained from the two evolutionary-based algorithms. In fact, in hierarchical clustering the f-measure value is 0.88 and 0.34 respectively for modern Italian and Italian Vulgar, the purity value is 0.80, the entropy value is 0.15, the NMI value is very low and equal to 0.01 and the ARI value is very low and equal to 0.0001. It indicates that cluster initialization determined from the genetic procedure is essential to obtain the perfect final solution. Again, SOM exhibits poor results, with an f-measure value of 0.84 for modern

Italian language and of 0.61 for Italian Vulgar language, a purity value of 0.80, an entropy value of 0.24, and NMI and ARI values respectively of 0.25 and 0.28. SK-Means is trapped into the same solution of hierarchical algorithm, reaching the same values of performance measures. Finally, EM reaches an f-measure value of 0.81 for modern Italian language and of 0.54 for Italian Vulgar language, a purity value of 0.82, an entropy value of 0.22, a NMI value of 0.21 and an ARI value of 0.20.

Tables 4 and 5 show the results obtained from the different clustering algorithms on the documents represented respectively as bi-gram and tri-gram frequency vectors. We can observe that a bi-gram model of language representation is not able to overcome our framework. In fact, GA-ICDA$^+$ reaches an f-measure value of 0.89 and 0.69 for respectively modern Italian and Italian Vulgar languages (see Table 4). The same is for tri-gram model of language representation, obtaining an f-measure value of 0.97 for modern Italian and of 0.90 for Italian Vulgar (see Table 5). On the contrary, GA-ICDA$^+$ obtains an f-measure of 1.00 for both language classes if it is employed on the run-length and ALBP feature representation (see Table 3). This result is confirmed also with the other computed performance measures. Furthermore, it is worth noting that the other clustering algorithms employed on the bi-gram and tri-gram frequency vectors are not able to overcome our framework (see Tables 4 and 5). Nonetheless, GA-ICDA$^+$, applied on the bi-gram and tri-gram frequency vectors, exhibits good values of

Table 4 Clustering results in terms of average precision, recall, f-measure, purity, entropy, NMI and ARI for each class of documents written in modern Italian and Italian Vulgar languages, from the following methods: Genetic Algorithms Image Clustering for Document Analysis-Plus (GA-ICDA$^+$), Genetic Algorithms Image Clustering for Document Analysis (GA-ICDA), Genetic Algorithms Image Clustering (GA-IC), bottom-up hierarchical clustering (Hierarchical), Self Organizing Map (SOM), spherical K-Means (SK-Means) and Expectation Maximization (EM)

Algorithm	Class	Precision	Recall	f-meas.	NMI	Purity	Entropy	ARI	nc
GA-ICDA$^+$	Modern Italian	0.9706	0.8250	0.8919	0.3386	0.8400	0.2360	0.4293	2
		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	Italian Vulgar	0.5625	0.9000	0.6923					
		(0.0000)	(0.0000)	(0.0000)					
GA-ICDA	Modern Italian	0.9632	0.8400	0.8971	0.3329	0.8460	0.2411	0.4412	2
		(0.0119)	(0.0242)	(0.0083)	(0.0092)	(0.0097)	(0.0082)	(0.0192)	
	Italian Vulgar	0.5784	0.8700	0.6933					
		(0.0255)	(0.0483)	(0.0016)					
GA-IC	Modern Italian	0.9905	0.8825	0.9314	0.5638	0.8980	0.1606	0.6137	2
		(0.0201)	(0.0708)	(0.0305)	(0.1041)	(0.0426)	(0.0565)	(0.1356)	
	Italian Vulgar	0.7084	0.9600	0.7981					
		(0.1641)	(0.0843)	(0.0638)					
Hierarchical	Modern Italian	0.8163	1.0000	0.8989	0.1104	0.8200	0.1376	0.1151	2
		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	Italian Vulgar	0.1837	0.9000	0.3051					
		(0.0000)	(0.0000)	(0.0000)					
SOM	Modern Italian	0.9524	1.0000	0.9756	0.7225	0.9600	0.0552	0.8142	2
		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	Italian Vulgar	1.0000	0.8000	0.8889					
		(0.0000)	(0.0000)	(0.0000)					
SK-Means	Modern Italian	0.9853	1.0000	0.9919	0.9288	0.9856	0.0110	0.9292	2
		(0.0501)	(0.0000)	(0.0276)	(0.2426)	(0.0491)	(0.0375)	(0.2413)	
	Italian Vulgar	0.9347	0.9920	0.9444					
		(0.2226)	(0.0273)	(0.1895)					
EM	Modern Italian	0.8528	0.8795	0.8574	0.1782	0.8304	0.1978	0.1939	2
		(0.0709)	(0.1496)	(0.0854)	(0.2194)	(0.0518)	(0.0836)	(0.2662)	
	Italian Vulgar	0.3370	0.8340	0.4303					
		(0.2622)	(0.1335)	(0.2019)					

Standard deviation is reported in parenthesis. *nc* is the average number of document clusters found with the algorithms. Documents are represented by the bi-gram frequency vectors

Table 5 Clustering results in terms of average precision, recall, f-measure, purity, entropy, NMI and ARI for each class of documents written in modern Italian and Italian Vulgar languages, from the following methods: Genetic Algorithms Image Clustering for Document Analysis-Plus (GA-ICDA$^+$), Genetic Algorithms Image Clustering for Document Analysis (GA-ICDA), Genetic Algorithms Image Clustering (GA-IC), bottom-up hierarchical clustering (Hierarchical), Self Organizing Map (SOM), spherical K-Means (SK-Means) and Expectation Maximization (EM)

algorithm	class	precision	recall	f-meas.	NMI	purity	entropy	ARI	nc
GA-ICDA$^+$	Modern Italian	0.9750	0.9750	0.9750	0.7132	0.9600	0.1022	0.8251	2
		(0.0007)	(0.0264)	(0.0135)	(0.1298)	(0.0211)	(0.0728)	(0.0865)	
	Italian Vulgar	0.9091	0.9000	0.9023					
		(0.0958)	(0.0000)	(0.0476)					
GA-ICDA	Modern Italian	0.9741	0.9425	0.9576	0.6038	0.9340	0.1508	0.7285	2
		(0.0011)	(0.0426)	(0.0224)	(0.1637)	(0.0341)	(0.0817)	(0.1297)	
	Italian Vulgar	0.8143	0.9000	0.8501					
		(0.1324)	(0.0000)	(0.0703)					
GA-IC	Modern Italian	0.9896	0.9300	0.9584	0.6525	0.9360	0.1451	0.7404	2
		(0.0135)	(0.0422)	(0.0225)	(0.1426)	(0.0337)	(0.0654)	(0.1230)	
	Italian Vulgar	0.7901	0.9600	0.8612					
		(0.1221)	(0.0516)	(0.0667)					
Hierarchical	Modern Italian	0.8163	1.0000	0.8989	0.1104	0.8200	0.1376	0.1151	2
		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	Italian Vulgar	0.1837	0.9000	0.3051					
		(0.0000)	(0.0000)	(0.0000)					
SOM	Modern Italian	0.8904	1.0000	0.9411	0.4465	0.8980	0.0955	0.5091	2
		(0.0575)	(0.0000)	(0.0326)	(0.2580)	(0.0600)	(0.0336)	(0.3008)	
	Italian Vulgar	0.6700	0.7700	0.6167					
		(0.4262)	(0.0949)	(0.2823)					
SK-Means	Modern Italian	0.9978	0.9970	0.9974	0.9901	0.9980	0.0027	0.9896	2
		(0.0222)	(0.0300)	(0.0263)	(0.0992)	(0.0200)	(0.0271)	(0.1044)	
	Italian Vulgar	0.9922	0.9980	0.9935					
		(0.0778)	(0.0200)	(0.0652)					
EM	Modern Italian	0.8370	0.9105	0.8657	0.1514	0.8246	0.1731	0.1399	2
		(0.0698)	(0.1228)	(0.0691)	(0.1801)	(0.0414)	(0.0598)	(0.2392)	
	Italian Vulgar	0.2986	0.8810	0.4019					
		(0.2265)	(0.1461)	(0.1726)					

Standard deviation is reported in parenthesis. *nc* is the average number of document clusters found with the algorithms. Documents are represented by the tri-gram frequency vectors

performance measures, comparable with other clustering methods. It indicates that GA-ICDA$^+$ algorithm is competitive with other state-of-the-art clustering methods even if different feature representations are used.

In general, we can observe from the f-measure values that Italian Vulgar language is more difficult to discriminate than modern Italian language from the competitor clustering algorithms. It is mainly due to the variegate and composite language characteristics in Italy around the thirteenth and fourteenth centuries. They are captured in a variety of different authors' writing styles in the database, determining solutions where some documents in Italian Vulgar are identified as modern Italian. This phenomenon is observed not so frequently in the documents given in modern Italian. Although the different facets of documents extracted from newspapers, fiction books and websites in the database, the modern Italian language is slightly more standardized than the Italian Vulgar language.

Finally, we have demonstrated the efficacy of run-length and ALBP feature combination as input to GA-ICDA$^+$ with respect to other feature combinations. Accordingly, Table 6 shows the clustering results obtained from GA-ICDA$^+$ when co-occurrence features are adopted for document representation (COOC, 12-dimensional vector), also in association with ALBP features (COOC-ALBP, 28-dimensional vector), and run-length statistical features (COOC-RL, 23-dimensional vector). It is worth observing that co-occurrence features, also in combination with

Table 6 Clustering results in terms of average precision, recall, f-measure, purity, entropy, NMI and ARI for each class of documents written in modern Italian and Italian Vulgar languages, from Genetic Algorithms Image Clustering for Document Analysis-Plus (GA-ICDA$^+$) when co-occurrence (COOC), a combination of co-occurrence and run-length (COOC-RL) and a combination of co-occurrence and ALBP features (COOC-ALBP) are adopted for document representation

features	class	precision	recall	f-meas.	NMI	purity	entropy	ARI
	Modern Italian	0.8670	0.9675	0.9143	0.3261	0.8720	0.1726	0.3808
COOC		(0.0577)	(0.0472)	(0.0515)	(0.2845)	(0.0641)	(0.1125)	(0.3255)
	Italian Vulgar	0.5010	0.7200	0.4795				
		(0.4308)	(0.1751)	(0.2713)				
	Modern Italian	0.9756	1.0000	0.9877	0.8363	0.9800	0.0331	0.9072
COOC		(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)
-RL	Italian Vulgar	1.0000	0.9000	0.9474				
		(0.0000)	(0.0000)	(0.0000)				
	Modern Italian	0.9179	1.0000	0.9571	0.5747	0.9280	0.0809	0.6633
COOC		(0.0214)	(0.0000)	(0.0113)	(0.0961)	(0.0193)	(0.0174)	(0.0907)
-ALBP	Italian Vulgar	1.0000	0.6400	0.7771				
		(0.0000)	(0.0966)	(0.0641)				

ALBP and run-length features, obtain worse results than run-length and ALBP. In fact, they are not able to obtain perfect discrimination of the documents in modern Italian and Italian Vulgar, which is obtained from the combination of run-length and ALBP features (See Table 3 for reference).

6 Execution time

Experiment has been performed in Matlab R2012a, on a Desktop computer with Quad-core CPU running at 2.6 GHz, 4GB RAM and operating system Windows 7. The execution time depends on feature extraction process and on clustering algorithm.

Our feature extraction process is computer time non-intensive, taking a CPU time of 1 s for processing of a text document of 2K characters. Differently, the feature extraction process by bi-gram model takes 5 s and by tri-gram model takes 6 s for processing of the same text document of 2K characters. Hence, our feature extraction process has a clear advantage wrt the text-based competitor methods.

The evolutionary-based clustering method has been implemented as a parallel application using the Matlab Global Optimization Toolbox. Hence, its execution can be automatically distributed on a set of cores. For this reason, it is not directly comparable with the competitor clustering algorithms, which intrinsically don't provide this feature. Parallelization determines a noticeable advantage in the execution time. Although the computational load of the genetic algorithm, the clustering method takes 52 s on 1 core, 32 s on 2 cores, and 24 s on 4 cores, for a database of 50 documents, each represented by our 27-dimensional feature vector.

7 Conclusions

The paper presented a new method for clustering of documents given in a language in different historical periods, with the example of Italian language evolved through time. Document is transformed into a uniformly coded text, composed of only four different numerical codes, derived from the location of each letter in the text-line, which is retrieved from its horizontal energy profile. Considering that each code is mapped into a given gray level, the obtained coded text becomes a gray level 1-D image, on which co-occurrence, (A)LBP, run-length statistics or a combination of them, can be applied for extraction of texture features. They define document feature vectors characterizing a strong distinction among the documents in a given language in different historical periods. Clustering of the feature vectors is performed by an extension of a state-of-the-art algorithm, called GA-ICDA$^+$, particularly apt to deal with evolving languages. Experiment has been realized on a custom oriented database of Italian documents given in modern Italian and Italian Vulgar languages. It has demonstrated the effectiveness of the new proposed technique in discriminating the documents in two historical periods of the language, when compared with other well-known clustering algorithms in the state-of-the-art and with the n-gram language model. Consequently, an extension of the method is very promising in information retrieval application.

Future work will provide a code for the system in C programming language, together with a complete documentation, to be freely available to download from the website, and will extend the method to the problem of discrimination of documents given in multiple writing styles of the same historical period in the same language.

Acknowledgments Authors are fully grateful to Ms. Zagorka Brodić, professor of French and Serbo-Croatian languages, for the helpful discussions about Italian language, and to Ms. Janet Newell, native professor of English language, for her precious editing support.

Compliance with Ethical Standards

Conflict of interests Author Darko Brodić declares that he has no conflict of interest. Author Alessia Amelio declares that she has no conflict of interest. Author Zoran N. Milivojević declares that he has no conflict of interest.

Funding This study was partially funded by the Grant of the Ministry of Education, Science and Technological Development of the Republic Serbia, as a part of the project TR33037 within the framework of Technological development program. The receiver of the funding is Dr. Darko Brodić.

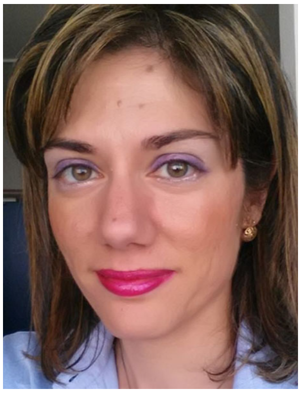
Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Janson T (2004) A natural history of latin. Oxford University Press, Oxford
2. History of Latin. Available at: https://en.wikipedia.org/wiki/History_of_Latin
3. Haller EK (2012) Dante alighieri. In: Matheson LM (ed) Icons of the middle ages: rulers, writers, rebels, and saints1, Santa Barbara, CA: Greenwood, p 244
4. Maiden M (1995) A linguistic history of italian. Longman, London
5. How Latin became Italian. Available at: https://damyanlissitchkov.wordpress.com/2013/03/23/how-latin-became-italian/
6. Pei MA (1949) New methodology for romance classification. WORD 5(2):135–146
7. Grimes BF (1996) Ethnologue: languages of the world. In: Pittman RS, Grimes JE (eds). 30th edn. Summer Institute of Linguistics, Academic Publisher, Dallas
8. Calabrese A (2003) On the Evolution of the short high vowel of Latin into Romance. In: Perez-Leroux A, Roberge Y (eds) Romance linguistics, theory and acquisition. John Benjamins, Amsterdam, pp 63–94
9. Cavnar W, Trenkle J (1994) N-gram-based text categorization. In: 3rd annual symposium on document analysis and information retrieval, April 11-13, Las Vegas, pp 161–175
10. Takci H, Sogukpimar I (2004) Letter based text scoring method for language identification. In: Advances in information systems, October 20-22, vol 3261, Izmir, pp 283–290
11. Tan CM, Wang YF, Lee CD (2002) The use of bigrams to enhance text categorization. Inf Process Manag 38(4):529–546
12. Grothe L, De Luca EW, Nurnberger A (2008) A comparative study on language identification Methods, Marrakech, Morocco
13. Braga IA, Monard MC, Matsubara ET (2009) Combining unigrams and bigrams in semi-supervised text classification. In: 14Th Portuguese conference on artificial intelligence (EPIA) - new trends in artificial intelligence, October 12–15, Aveiro, Portugal, pp 489–500
14. Goodman J (2006) A bit of progress in language modeling: extended version. Technical report MSR-TR-2001-72, machine learning and applied statistics group microsoft research. Redmond
15. Padro M, Padro L (2004) Comparing methods for language identification. In: XX Congreso de la Sociedad Espanola para el Procesamiento del Lenguage Natural, Barcelona, Spain, pp 155–161
16. Sibun P, Spitz AL (1994) Language determination: natural language processing from scanned document images. In: 4th applied natural language processing conference, October 13-15, Stuttgart, Germany, pp 15–21
17. Martino MJ, Paulsen RC (2001) Natural language determination using partial words. U.S. Patent No. 6216102 B1
18. Cowie J, Ludovic Y, Zacharski R (1999) Language recognition for mono and multilingual documents. In: Vextal conference, November 22-24, Venice, pp 209–214
19. Shijian L, Lim Tan C (2008) Script and language identification in noisy and degraded document images. IEEE Trans Pattern Anal Mach Intell 30(1):14–24
20. Tan TN (1996) Written language recognition based on texture analysis. In: Proceedings of ICIP'96, vol 2, Lausanne, Switz, pp 185–188
21. Peake GS, Tan TN (1997) Script and language identification from document images. In: Third Asian Conference on Computer Vision, January 8-10, Hong Kong, China, pp 97–104
22. Brodić D, Amelio A, Milivojević ZN (2016) Language discrimination by texture analysis of the image corresponding to the text. Neural Comput Appl 1–22
23. Brodić D, Amelio A, Milivojević ZN (2015) Characterization and distinction between closely related south slavic languages on the example of Serbian and Croatian. In: Comp. anal. of images and patterns, September 2-4, vol 9256, Valletta, Malta, pp 654–666
24. Brodić D, Milivojević ZN, Amelio A (2015) Analysis of the South Slavic scripts by run-length features of the image texture. Elektronika Ir Elektrotechnika 21(4):60–64
25. Zramdini A, Ingold R (1998) Optical font recognition using typographical features. IEEE Trans Pattern Anal Mach Intell 20(8):877–882
26. Joshi GD, Garg S, Sivaswamy J (2007) A generalised framework for script identification. IJDAR 10(2):55–68
27. Brodić D, Milivojević ZN, Maluckov CA (2013) Recognition of the script in Serbian documents using frequency occurrence and co-occurrence analysis. Sci World J 896328:1–14
28. Del Bimbo A (2001) Visual information retrieval. Morgan Kaufmann Publishers Inc., San Francisco
29. Brodić D, Milivojević ZN, Maluckov CA (2015) An approach to the script discrimination in the Slavic documents. Soft Comput 19(9):2655–2665
30. Eleyan A, Demirel H (2011) Co-occurrence matrix and its statistical features as a new approach for face recognition. Turkish J Electr Engin and Comp Sci 19(1):97–107
31. Clausi DA (2002) An analysis of co-occurrence texture statistics as a function of grey level quantization. Canadian J Remote Sens 28(1):45–62
32. Galloway MM (1975) Texture analysis using gray level run lengths. Comput Graphics Image Process 4(2):172–179
33. Chu A, Sehgal CM, Greenleaf JF (1990) Use of gray value distribution of run lengths for texture analysis. Pattern Recogn Lett 11(6):415–419
34. Dasarathy BR, Holder EB (1991) Image characterizations based on joint gray-level run-length distributions. Pattern Recogn Lett 12(8):497–502
35. Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of texture measures with classification based on feature distributions. Pattern Recogn 29:51–59
36. Nosaka R, Ohkawa Y, Fukui K (2011) Feature extraction based on co-occurrence of adjacent local binary patterns. In: Advance in image and video technology, November 20–23, vol 7088, Gwangju, South Korea, pp 82–91

37. Amelio A, Pizzuti C (2014) A new evolutionary-based clustering framework for image databases. In: Image and Sign. Proc., June 30-july 2, vol 8509, Cherbourg, Normandy, France, pp 322–331
38. Sicilian School. Available online: http://www.britannica.com/art/Sicilian-school
39. Dolce Stil Novo. Available online: http://www.britannica.com/art/dolce-stil-nuovo
40. Angiolieri C Available online: http://www.britannica.com/biography/Cecco-Angiolieri
41. Zhao Y, Karypis G, Fayyad U (2005) Hierarchical clustering algorithms for document datasets. Data Min Knowl Disc 10(2):141–168
42. Saarikoski J, Laurikkala J, Järvelin K, Juhola M (2011) Self-organising maps in document classification: a comparison with six machine learning methods. In: 10th international conference, ICANNGA, April 14-16, vol 6593, Ljubljana, Slovenia, pp 260–269
43. Steinbach M, Karypis G, Kumar V (2000) A comparison of document clustering techniques. In: KDD workshop on text mining, August 20-23, Boston, MA, USA
44. Zhong S (2005) Efficient online spherical k-means clustering. In: IEEE international joint conference on neural networks, 31 July-4 August, vol 5, Montreal, Canada, pp 3180–3185
45. Rigutini L, Maggini M (2005) A semi-supervised document clustering algorithm based on EM. In: IEEE/WIC/ACM international conference on web intelligence, September 19-22, Compigne, France, pp 200–206
46. Ward JH (1963) Hierarchical grouping to optimize an objective function. J Am Stat Assoc 58(301):236–244
47. Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 43(1):59–69
48. MacQueen JB (1967) Some methods for classification and analysis of multivariate observations. In: 5th Berkeley symposium on mathematical statistics and probability, June 21-July 18 and December 27-January 7, vol 1, Berkeley, USA, pp 281–297
49. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Ser B 39(1):1–38
50. Turney PD, Pantel P (2010) From frequency to meaning: vector space models of semantics. J Artif Intell Res 37(1):141–188
51. Powers DMW (2011) Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. J Mach Learn Technol 2(1):37–63
52. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd Edn. Morgan Kaufmann
53. Rand WM (1971) Objective criteria for the evaluation of clustering methods. J Am Stat Assoc 66(336):846–850
54. Santos JM, Embrechts M (2009) On the use of the adjusted rand index as a metric for evaluating supervised classification. In: 19th international conference on artificial neural networks: Part II, September 14-17, Limassol, Cyprus, pp 175–184
55. Andrews NO, Fox EA (2009) Recent developments in document clustering technical report, computer science, Virginia Tech
56. De Vries CM, Geva S, Trotman A (2012) Document clustering evaluation: Divergence from a random baseline. CoRR, abs/1208.5654
57. De Bie T, Cristianini N (2004) Kernel methods for exploratory pattern analysis: a demonstration on text data. In: Joint IAPR international workshops, SSPR 2004 and SPR 2004, August 18-20, vol 3138, Lisbon, Portugal, pp 16–29
58. Fodor JD, Sakas WG (2004) Evaluating models of parameter setting. Boston University conference on language development, Boston

Darko Brodić received his B.Sc. and M.Sc. in Electrical Engineering from the Faculty of Electrical Engineering, University of Sarajevo in 1987 and 1990, as well as Ph.D. in electrical engineering from the Faculty of Electrical Engineering, University of Banja Luka in 2011. Now he is Assistant Professor of computer science at the Technical Faculty in Bor, University of Belgrade, Serbia. His current research interests include different aspects of signal processing, natural image processing, document image processing, pattern recognition and magnetic field measurement. He is the author of more than 100 journal and conference papers (35 in journals indexed by Thomson SCI/SCIE JCR).



Alessia Amelio received her B.Sc. and M.Sc. in Computer Science Engineering from University of Calabria in 2005 and 2009, as well as Ph.D. in computer science engineering and systems from the Faculty of Engineering, University of Calabria in 2013. Now she is Research Fellow of computer science at the Department of Computer Science Engineering, Modeling, Electronics and Systems, University of Calabria, Italy. Her current research interests include different aspects of image processing, document classification, pattern recognition, social network analysis and data mining for magnetic field measurement and web applications. She is the author of more than 40 scientific papers.

Zoran N. Milivojević received his B.E., M.E., and Ph.D degrees in Electrical Engineering from the Faculty of Electronic Engineering, University of Niš, in 1984, 1994 and 2002, respectively. He is a Professor of College of Applied Technical Sciences in Niš, Serbia. His primary research interests are digital signal processing: algorithms and applications. He is the author and coauthor of over 260 journal and conference papers (24 in journals indexed by Thomson SCI/SCIE JCR).