A New Image Analysis Framework for Latin and Italian Language Discrimination

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

[1] University of Belgrade, Technical Faculty in Bor, V.J. 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
dbrodic@tf.bor.ac.rs, aamelio@dimes.unical.it,
zoran.milivojevic@vtsnis.edu.rs

Abstract. The paper presents a new framework for discrimination of Latin and Italian languages. The first phase maps the text in the given language into a uniformly coded text. It is based on the position of each letter of the script in the text line and its height, derived from its energy profile. The second phase extracts run-length texture measures from the coded text given as 1-D image, by producing a feature vector of 11 values. The obtained feature vectors are adopted for language discrimination by using a clustering algorithm. As a result, the distinction between the two languages is perfectly realized with an accuracy of 100% on a complex database of documents in Latin and Italian languages.

Keywords: Clustering, Document analysis, Image processing, Information retrieval, Italian language, Statistical analysis

1 Introduction

Information retrieval represents one of the areas of natural language processing. It finds the objects, which usually represent documents of an unstructured nature (usually text) that satisfy an information need from within large collections [11]. Typically, the vector space model is used for similarity distinction between the documents. However, the cross-language information retrieval is still a challenge. It is especially expressed between very similar languages or languages that evolved one from another.

The Latin language was originally spoken in the region around Rome called Latium. As a consequence of Roman conquests, Latin was quickly spread over a larger part of Italy and wider. Accordingly, it has begun the formal language of the Roman Empire. After its collapse, Latin language evolved into the various Romance languages. However, it was still used for writing. Furthermore, the Latin language was a lingua franca, which was used for scientific and political affairs, for more than a thousand years. Up to now, ecclesiastical Latin language has remained the formal language of the Roman Catholic Church. As a consequence, it is the official language of the Vatican. Although Latin language is not a live language, it is not a dead language. It is still partly in use.

Today, the Latin language is usually taught in order to translate Latin texts into modern languages. Because of this long tradition and of the influence on the modern languages, the study of Latin is extremely important for linguistic research. Italian language is one of the languages from the Romance language group, which is the closest to the Latin language. It comprises many dialects from the North to the South of Italy. However, the standard Italian language is virtually the only written language. Today, the standard Italian language is virtually the only dialect of culture in modern Italy, which is used as the language of intercommunication between different parts of Italy. To the very best of the author's knowledge, some aspects of evolving Latin into modern Italian language have been researched. Still, these aspects were completely linguistics in nature [5]. In contrast, we conducted the research in the direction of safe automatic differentiation of these languages in unsupervised manner.

In this paper, we propose a novel framework for the distinction between languages that evolved one from another. As an example, we use Latin and modern Italian languages. The framework includes the following stages: script coding, run-length texture analysis and clustering. The main novelty of the framework is the extension of a state-of-the-art clustering method and its application on document features for discrimination of languages evolved one into another. Because we deal with discrimination problem, unsupervised method is appropriate. The distinction between the two related languages is perfectly realized with an accuracy of 100%, which outperforms competitor methods.

The paper is organized in the following manner. Section 2 describes the proposed framework. Section 3 explains the experiment. Section 4 gives the results of the experiment and discusses them. Section 5 makes a conclusion.

2 The Proposed Framework

Our framework for Latin and modern Italian language discrimination is composed of the following three steps: (i) script coding, (ii) texture analysis, (iii) clustering. Script coding adopts the approach previously introduced by Brodić et al. [4]. In fact, it demonstrated to be successful for solving a critical task of closely related language discrimination [3]. In particular, given the text document as input, it maps each letter of the document to only four codes based on the corresponding position in the text line, representing the gray-level pixels of a 1-D image. Then, texture analysis is performed on the produced image in order to extract run-length texture features. In order to select the feature representation, three well-known types of texture features, run-length, co-occurrence and ALBP, have been evaluated on benchmark datasets of the same languages. Results demonstrated that run-length features obtain the best performances in language discrimination in this context. These features are discriminated by a new clustering method in order to detect classes representing documents written in two different languages.

2.1 Script Coding

Text documents can be divided into text lines. Furthermore, each text line can be segmented by considering the energy of the script signs [9] into the four virtual lines [20]: top-line, upper-line, base-line and bottom-line. These lines track the following vertical zones in the text line area [20]: upper zone, middle zone and lower zone. The letters can be categorized based on their position in vertical zones of the line, that represents their energy profile. The short letters (S) are located into the middle zone only. The ascender letters (A) occupy the middle and upper zones. The descendent letters (D) are spread into the middle and lower zones. The full letters (F) enlarge over all vertical zones. Consequently, all letters can be classified as belonging to four different script types [4]. Fig. 1 depicts the script characteristics according to their position in the baseline.

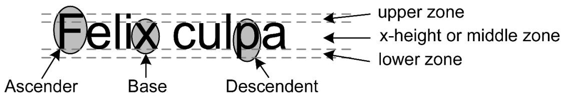


Fig. 1. Virtual lines and vertical zones in the text line.

Each script type can be mapped into a different number code. Because there are only four script types, mapping is performed to four number codes {0, 1, 2, 3}. Then, these codes are associated with four different gray levels to create an image. Fig. 2 illustrates the correspondence between script type number codes and gray levels.

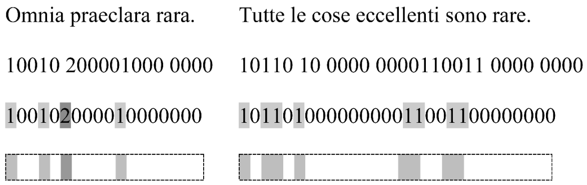


Fig. 2. Script type number codes and their corresponding gray levels of the 1-D image.

Consequently, each text document is translated into a set of number codes {0, 1, 2, 3} corresponding to pixels of only four gray levels. It obtains a textured 1-D image I, which can be analyzed by adopting the texture analysis.

2.2 Texture Analysis

Texture quantifies the intensity variation in the image area [16]. Hence, it is a powerful tool for the extraction of important properties like image smoothness, coarseness and regularity. Accordingly, the texture is useful to compute image statistical measures. Run-length statistical analysis is adopted to retrieve texture features and to evaluate texture coarseness [8]. A run is a set of consecutive pixels with the same gray-level value in the specific texture direction. The fine textures are characterized by long runs, while coarse textures include short runs.

Let I be an image of X rows, Y columns and L gray levels. The first step consists in building the run-length matrix $\mathbf{P}$. It is created by fixing a direction and then counting how many runs are encountered for each gray level and length in that direction. Accordingly, a set of consecutive pixels with identical intensity values identifies a gray-level run. The row number of $\mathbf{P}$ is equal to L, i.e. the number of gray levels, while the column number of $\mathbf{P}$ is equal to the maximum run length R. In our case, a single element of the run-length matrix $P(i, j)$ at position (i, j) represents the number of times a run of gray-level i and of length j occurs inside the image I (in our case, 1-D image).

Different texture features can be extracted from the $\mathbf{P}$ matrix [8]: (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). The extraction of texture features from $\mathbf{P}$ includes also the following two measures [6]: (i) Low gray-level run emphasis (LGRE) and (ii) High gray-level run emphasis (HGRE). In Dasarathy et al. [7], other four texture features are proposed, based on the joint statistical measure of gray level and run length. They are: (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).

In this way, run-length statistical analysis extracts a total of 11 feature measures, defining a 11-dimensional feature vector for language representation.

2.3 Clustering

The aforementioned run-length feature vectors, each representing a document in Latin or modern Italian languages, are subjected to unsupervised classification by a clustering technique. It is adopted for discriminating between documents written in Latin language and documents written in modern Italian language. In order to find the classes in the data, we adopt the Genetic Algorithms Image Clustering for Document Analysis algorithm (GA-ICDA), previously introduced by Brodić et al. [3], modified to be suitable for languages evolved one into another. We call the modified version of this algorithm *Genetic Algorithms Image Clustering for Document Analysis-Plus* (GA-ICDA$^+$). Next, we recall the main concepts underlying GA-ICDA and propose the modifications for GA-ICDA$^+$.

GA-ICDA is a bottom-up clustering method representing the set of documents written in different languages or scripts as a weighted graph $G = (V, E, W)$. Each node $v_i \in V$ is a document and each link $e_{ij} \in E$ connects two nodes v_i and v_j to each other. A weight $w_{ij} \in W$ associated to the link e_{ij} represents the similarity among the nodes v_i and v_j. For each node v_i, only a set of the other nodes $V \setminus v_i$ in G is considered. This set is called h-nearest neighborhood of v_i [1]. It represents the set of nodes whose corresponding documents are the most similar to the document associated to v_i. Similarity between two nodes v_i and v_j is calculated as:

$$w_{ij} = e^{-\frac{d(i,j)^2}{a^2}}, \tag{1}$$

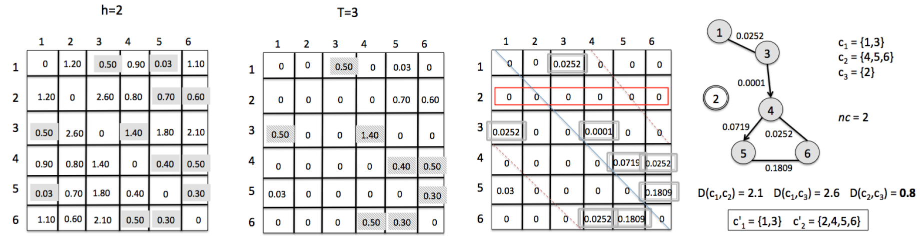
where a is a scale parameter and $d(i, j)$ is the distance between the document feature vectors of v_i and v_j. The L_1 norm is adopted as distance, while h is a parameter influencing the size of the neighborhood [1]. The h-nearest neighbor nodes of v_i are denoted as $nn^h_{v_i} = \{nn^h_{v_i}(1), ..., nn^h_{v_i}(k)\}$, where k is the number of h-nearest neighbors. Then, a mapping f is defined between each node in V and an integer label, $f : V \rightarrow \{1, 2, .., n\}$ $n = |V|$, realizing a node ordering. Finally, the difference is calculated between the label corresponding to the node $f(v_i)$ and the labels corresponding to the nodes in $nn^h_{v_i}$, $|f(v_i) - f(nn^h_{v_i}(j))|$ $j = 1...k$. Each node v_i in G is connected only to the nodes in $nn^h_{v_i}$ whose label difference is less than a given threshold value T. It implies that only similar and "spatially" close nodes are connected to each other in G. The obtained node connections, weighted by the similarity values, are represented in terms of the adjacency matrix $\mathbf{M}$ of G. Then, G is subjected to a genetic method for finding the connected components representing the clusters of documents. After that, for correcting the local optima, a merging procedure is applied on the found clusters. In particular, pairs of clusters having minimum mutual distance are selected and repeatedly merged, until a fixed cluster number is reached. The distance is computed as the L_1 norm between the two farthest document feature vectors, one for each cluster.

The first introduced modification in GA-ICDA$^+$ is the similarity computation among the graph nodes. The inner complex and variegate structure of the evolved language, like modern Italian, determines naturally higher distance values computed between the document feature vectors. Such a phenomenon may cause an anomaly in the similarity computation in Eq. (1). Consider v_i as a node in G with associated document feature vector d_i. If the distance $d(i, j)$ between the vectors d_i and d_j of the nodes v_i and v_j is particularly high, because of the power by 2, the numerator of the exponent $\frac{d(i,j)^2}{a^2}$ is very high, determining a similarity value which is zero. If it occurs much often for different pairs of document feature vectors, the adjacency matrix $\mathbf{M}$ corresponding to the similarity matrix will be unjustifiably very sparse. In order to overcome this problem, the exponent of $d(i, j)$ in Eq. (1) which is currently 2, is substituted by a parameter α for obtaining a more flexible and smoothed characterization of the similarity. Consequently, w_{ij} in Eq. (1) begins:

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

The second introduced modification is the graph construction. Specifically, consider the second step of the procedure where, for each node v_i, only the h-nearest neighbors are maintained, which are "spatially" close to v_i, given a node ordering f. It is clear that it determines a reduction in the number of neighbors, and consequently in the number of outgoing links, for each node v_i. It obtains in most cases a better characterization of the graph connected components. When the document graph is particularly complex, like in this task of capturing differences between languages evolved one into another, a low value of the threshold T is necessary for determining good components. However, it causes the presence of isolated nodes, for which all the nearest neighbors are removed by the threshold T. In GA-ICDA this situation is not considered, because we obtain good components even if the T value is higher. Here we relax this constraint, by managing the presence of isolated nodes. They are "singleton" nodes for the genetic procedure, which is not able to add them inside any connected component, because of the absence of node neighbors. At the end of the procedure, they will be considered as ”singleton” clusters and automatically managed by the final bottom-up strategy.

Fig. 3 shows an example of GA-ICDA$^+$ execution. From left to right, for each node in the distance matrix (6 nodes), the algorithm finds the 2-nearest neighbors (in grey). Then, for each node, the algorithm finds the neighbors with label difference smaller than $T = 3$ with respect to the label of that node (in dotted grey), making the node 2 isolated. The adjacency matrix is obtained by computing the similarity values from the distance values by adopting Eq. (2) ($\alpha = 1.5$). c_1, c_2 and c_3 are the clusters detected from the genetic algorithm. c'_1 and c'_2 are the final clusters detected from the bottom-up merging procedure, with fixed cluster number $nc = 2$. They are obtained by computing the distances of cluster pairs and merging the singleton cluster c_2 with c_3 exhibiting the minimum distance value of 0.8.



h=2

	1	2	3	4	5	6
1	0	1.20	0.50	0.90	0.03	1.10
2	1.20	0	2.60	0.80	0.70	0.60
3	0.50	2.60	0	1.40	1.80	2.10
4	0.90	0.80	1.40	0	0.40	0.50
5	0.03	0.70	1.80	0.40	0	0.30
6	1.10	0.60	2.10	0.50	0.30	0

T=3

	1	2	3	4	5	6
1	0	0	0.50	0	0.03	0
2	0	0	0	0	0.70	0.60
3	0.50	0	0	1.40	0	0
4	0	0	0	0	0.40	0.50
5	0.03	0	0	0	0	0.30
6	0	0	0	0.50	0.30	0

	1	2	3	4	5	6
1	0	0	0.0252	0	0	0
2	0	0	0	0	0	0
3	0.0252	0	0	0.0001	0	0
4	0	0	0	0	0.0719	0.0252
5	0.03	0	0	0	0	0.1809
6	0	0	0	0.0252	0.1809	0

Fig. 3. Example of GA-ICDA$^+$ execution.

3 Experimentation

As example of framework usage, an experiment is performed on a complex custom oriented database, publicly available at [10], composed of a set of 90 documents in Latin and modern Italian languages. Specifically, 50 out of 90 documents are given in Latin language and 40 out of 90 documents are given in modern Italian language. Documents count from 400 to 6000 characters each. 40 out of 50 Latin documents are extracted from Cicero’s works (106 BC - 43 BC), in particular from *De Inventione*, *De Oratore*, *De Optimum Genere Oratorum*, *De Natura Deorum* and *De Officiis*. 10 out of 50 Latin documents are extracted from Virgil’s *Aenead* (70 BC - 19 BC). The documents from the two different authors belong to a different historical period and the writing style of the two authors is also different. Consequently, recognition of common language is difficult. Modern italian documents are extracted from two well-known Italian newspapers, *Il Sole 24Ore* and *La Repubblica*, and from websites. In particular, 20 out of 40 modern Italian documents are excerpts from newspapers and 20 out of 40 modern Italian documents are excerpts from the web. The writing style of the newspapers excerpts is different, because more ”technical”, than the writing style of the excerpts from the web, which is more ”linear”.

4 Results and Discussion

Next, we demonstrate the efficacy of our framework as a combination of feature representation and clustering method, in correctly discriminating between Latin and modern Italian documents. Specifically, we show in Table 1 the clustering results obtained from our framework (named as GA-ICDA$^+$) on the custom oriented document database and compare them with the clustering results obtained from other five algorithms on the same database. They are three clustering methods, Hierarchical Clustering, K-Medians and Self-Organizing-Map (SOM), which are different well-known strategies for text document categorization [13],[15],[19]. In particular, we chose to adopt K-Medians instead of K-Means because the first one uses the same L_1 norm as our method GA-ICDA$^+$ and because it is more robust to outliers than K-Means. The other two algorithms are the GA-IC framework for image database clustering [1] and the GA-ICDA framework [3], which is the extension of GA-IC for document database clustering, without the modifications introduced for GA-ICDA$^+$. All the algorithms, K-Medians, hierarchical clustering, SOM, GA-IC and GA-ICDA adopt the same run-length feature vector representation used from GA-ICDA$^+$.

Clustering results are showed in terms of five methods for performance evaluation: precision, recall and f-measure indexes [2],[12], purity, entropy, Normalized Mutual Information (NMI) [2],[17],[18] and Adjusted Rand Index (ARI) [14]. Precision, recall and f-measure are reported separately for each language class (Latin and modern Italian) in correspondence to each algorithm. For the other performance measures, purity, entropy, NMI and ARI, a single overall value is reported for each algorithm. Purity, entropy, NMI and ARI are well-known performance measures for clustering evaluation. On the contrary, the computation of precision, recall and f-measure requires that the correspondence between each cluster detected from the algorithm and the true language class is known. Consequently, we associate each cluster with the true language class whose corresponding number of documents is in majority in that cluster. The number of clusters nc found from the algorithms is also reported.

A trial and error procedure has been adopted on benchmark documents, different from the documents in the considered database, for tuning the algorithms parameters. The parameter values providing the best possible results on the benchmark documents have been adopted for clustering the custom oriented document database. Consequently, in K-Medians algorithm, the number of clusters is fixed to 2. In SOM algorithm, the dimension of a neuron layer is 1×2. The number of training steps for initial covering of the input space is 100 and the initial neighborhood size is 3. The distance between two neurons is computed as the number of steps separating each other. Hierarchical clustering adopts a bottom-up agglomerative strategy using L_1 norm for distance computation. Average linkage is used for cluster distance evaluation. The obtained dendrogram is "horizontally" cut to obtain a number of clusters which is equal to 2. The h value of the neighborhood is fixed to 33 for GA-IC and GA-ICDA and to 43 for GA-ICDA$^+$ and the T threshold value to 9 for GA-ICDA and to 7 for GA-

ICDA$^+$. The α parameter for the similarity computation in GA-ICDA$^+$ is fixed to 1.5.

The algorithms have been implemented in MATLAB R2012a. Experiments have been run on a Desktop computer quad core 2.3GHz 4GB RAM and Windows 7. Each algorithm has been executed 100 times and the average values of each performance measure together with the standard deviation values (in parenthesis) have been reported. Our framework takes 55 s for each execution on the database of 90 documents.

Table 1. Results of Latin and modern Italian document clustering.

	classes	Precision	Recall	F-Measure	Purity	Entropy	NMI	ARI	nc
GA-ICDA$^+$	Latin	1.0000 (0.0000)	1.0000 (0.0000)	1.0000 (0.0000)	1.0000 (0.0000)	0.0000 (0.0000)	1.0000 (0.0000)	1.0000 (0.0000)	2
	modern Italian	1.0000 (0.0000)	1.0000 (0.0000)	1.0000 (0.0000)					
GA-ICDA	Latin	1.0000 (0.0000)	0.9000 (0.0000)	0.9474 (0.0000)	0.9444 (0.0000)	0.2237 (0.0000)	0.7428 (0.0000)	0.7878 (0.0000)	2
	modern Italian	0.8889 (0.0000)	1.0000 (0.0000)	0.9412 (0.0000)					
GA-IC	Latin	0.8113 (0.0000)	0.8600 (0.0000)	0.8350 (0.0000)	0.8111 (0.0000)	0.6215 (0.0000)	0.2967 (0.0000)	0.3803 (0.0000)	2
	modern Italian	0.8108 (0.0000)	0.7500 (0.0000)	0.7792 (0.0000)					
Hierarchical	Latin	0.5618 (0.0000)	1.0000 (0.0000)	0.7194 (0.0000)	0.5667 (0.0000)	0.4395 (0.0000)	0.0243 (0.0000)	0.0056 (0.0000)	2
	modern Italian	0.4382 (0.0000)	0.9750 (0.0000)	0.6047 (0.0000)					
SOM	Latin	0.8116 (0.0010)	0.8616 (0.0055)	0.8358 (0.0031)	0.8120 (0.0030)	0.6195 (0.0069)	0.2987 (0.0070)	0.3825 (0.0078)	2
	modern Italian	0.8126 (0.0061)	0.7500 (0.0000)	0.7800 (0.0028)					
K-Medians	Latin	0.8113 (0.0000)	0.8600 (0.0000)	0.8350 (0.0000)	0.8111 (0.0000)	0.6215 (0.0000)	0.2967 (0.0000)	0.3803 (0.0000)	2
	modern Italian	0.8108 (0.0000)	0.7500 (0.0000)	0.7792 (0.0000)					

We observe that our framework, which is the combination of run-length features and GA-ICDA$^+$ clustering method, performs successfully, overcoming all the other clustering methods (see Table 1). In fact, GA-ICDA$^+$ obtains the perfect distinction between Latin and modern Italian documents, with a number of clusters equal to 2, precision, recall and f-measure values of 1.00 for both Latin and modern Italian language classes, purity, NMI and ARI values of 1.00 and an entropy value of 0.00. Furthermore, standard deviation values are always zero, demonstrating the stability of the result. It is interesting to observe as GA-IC algorithm is not able to well discriminate the languages. Although the number of found clusters is exactly 2, the f-measure values are 0.83 for Latin and 0.78 for modern Italian, the purity value is 0.81, the NMI value is quite low and equal to 0.30, together with the value of ARI which is 0.38 and the high value of entropy which is 0.62. This means that the found clusters contain mixed Latin and modern Italian documents. The GA-ICDA procedure performs considerably better than GA-IC for this task. In fact, it exhibits f-measure values of 0.95 for Latin and 0.94 for modern Italian, a purity value of 0.94, a entropy value of 0.22 and NMI and ARI values of respectively 0.74 and 0.79. It indicates that GA-ICDA is more apt to deal with document data than GA-IC. However, the best result is given from GA-ICDA$^+$, demonstrating the efficacy of the performed modifications. About the other algorithms, we can observe that a pure bottom-up strategy like hierarchical clustering is not able to outperform the GA-IC, GA-ICDA and GA-ICDA$^+$ evolutionary strategies. In fact, it reaches f-measure values of 0.72 and 0.60 for respectively Latin and modern Italian, a purity value of 0.57, a entropy value of 0.44 and very low NMI and ARI values of respectively 0.02 and 0.006. It is also worth to note that the results of GA-ICDA, adopting together an evolutionary method and a bottom-up refinement procedure, are better than both the pure evolutionary procedure of GA-IC and the pure bottom-up strategy of hierarchical clustering. It demonstrates the efficacy of the combination of both the evolutionary and bottom-up methods in document clustering. The SOM results are very similar to the results obtained from GA-IC. In fact, the f-measure values are equal to 0.83 for Latin and 0.78 for modern Italian, the purity and entropy values are respectively 0.81 and 0.62, the NMI and ARI values are quite low and respectively 0.30 and 0.38. K-Medians also obtains results which are similar to the results of SOM and GA-IC, with a f-measure value of 0.83 for Latin and 0.78 for modern Italian, purity, NMI and ARI values of respectively 0.81, 0.30 and 0.38 and a very high entropy value of 0.62. It indicates that GA-IC, SOM and K-Medians are trapped into a recurrent solution consisting of mixed clusters of documents in Latin and modern Italian languages.

5 Conclusions

The paper introduced a new framework for the discrimination between documents written in Latin and modern Italian languages. It is characterized by the position of each script letter in the baseline, derived by its energy profile, for mapping into uniformly coded text. The statistical analysis of the coded text, represented as an image, is performed by the run-length matrix calculation for texture feature extraction. The obtained feature vectors revealed satisfactory dissimilarity of the documents in different languages. Such a dissimilarity is the basis for successfully document clustering by the extension of a state-of-the-art classification tool GA-ICDA$^+$. Experimental results demonstrated the superiority of the new framework with respect to the other clustering methods. Future work will extend the experiment to larger databases and multiple types of language feature representations.

Acknowledgments. This work was partially supported by the Grant of the Ministry of Education, Science and Technological Development of the Republic Serbia, as a part of the project TR33037. Authors are fully grateful to Ms. Zagorka Brodić, professor of French and Serbo-Croatian languages, for the helpful discussions about Latin and Italian languages.

References

1. Amelio, A. and Pizzuti, C.: A new evolutionary-based clustering framework for image databases. In: Image and Sign. Proc., June 30-July 2, Cherbourg, Normandy, France, 8509:322-331 LNCS, Springer, 2014.

2. Andrews, N. O. and Fox, E. A.: Recent Developments in Document Clustering. Technical report, Computer Science, Virginia Tech.
3. Brodić, D., Amelio, A., Milivojević, Z. N.: Characterization and Distinction Between Closely Related South Slavic Languages on the Example of Serbian and Croatian. In: Comp. Anal. of Images and Patterns, 2-4 September, Valletta, Malta, 9256:654-666 LNCS, Part I, Springer, 2015.
4. Brodić, D., Milivojević, Z.N., Maluckov, C.A.: Recognition of the script in serbian documents using frequency occurrence and co-occurrence analysis. The Scientific World Journal, 896328:1-14, 2013.
5. Calabrese, A.: On the Evolution of the short high vowel of Latin into Romance, in A. Perez-Leroux & Y Roberge (eds.) Romance Linguistics. Theory and Acquisition. Amsterdam, John Benjamins, 63-94, 2003.
6. Chu, A., Sehgal, C.M., Greenleaf, J.F.: Use of gray value distribution of run lengths for texture analysis. Pattern Recognition Letters 11(6):415-419, 1990.
7. Dasarathy, B.R., Holder, E.B.: Image characterizations based on joint gray-level run-length distributions. Pattern Recognition Letters, 12(8):497-502, 1991.
8. Galloway, M.M.: Texture analysis using gray level run lengths. Computer, Graphics and Image Processing 4(2):172-179, 1975.
9. Joshi, G.D., Garg, S., Sivaswamy, J.: A generalised framework for script identification. *IJDAR*, 10(2):55-68, 2007.
10. https://sites.google.com/site/documentanalysis2015/latin-italian-database.
11. Manning, C.D., Raghavan P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, 2008.
12. Powers, D. M. W.: Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation. Journal of Machine Learning Technologies 2(1):37-63, 2011.
13. Saarikoski, J., Laurikkala, J., Järvelin, K., Juhola, M.: Self-Organising Maps in Document Classification: A Comparison with Six Machine Learning Methods. In: 10th Int. Conf., ICANNGA, 14-16 April, Ljubljana, Slovenia, Part I 6593:260-269 LNCS, Springer, 2011.
14. Santos, J. M. and Embrechts, M.: 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, 14-17 September, Limassol, Cyprus, Springer-Verlag, Berlin, Heidelberg, 175-184.
15. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques. In: KDD Workshop on Text Mining, 20-23 August, Boston, MA, USA, 2000.
16. Tan, X.: Texture information in run-length matrices. IEEE Trans. Image Proc. 7(11):1602-1609, 1998.
17. De Vries, C.M., Geva, S. and Trotman, A.: Document clustering evaluation: Divergence from a random baseline. CoRR, abs/1208.5654, 2012.
18. Hu, X. and Yoo, I.: A comprehensive comparison study of document clustering for a biomedical digital library medline. In: 6th ACM/IEEE-CS Joint Conference on, 11-15 June, Chapel Hill, NC, USA, 220-229, 2006.
19. Zhao, Y., Karypis, G., Fayyad, U.: Hierarchical Clustering Algorithms for Document Datasets. Data Mining and Knowledge Discovery, 10(2):141-168, 2005.
20. Zramdini, A., Ingold, R.: Optical font recognition using typographical features. IEEE Trans. Pattern Analysis and Machine Intelligence, 20(8):877-882, 1998.