Contents lists available at ScienceDirect

Egyptian Informatics Journal

journal homepage: www.sciencedirect.com

An effective dimension reduction algorithm for clustering Arabic text

A.A. Mohamed

Prince Sattam Bin Abdulaziz University, Kingdom of Saudi Arabia

ARTICLE INFO

Article history:
Received 7 August 2018
Revised 12 March 2019
Accepted 17 May 2019
Available online 31 May 2019

Keywords:
Clustering
Dimensionality reduction
PCA
SVD
NMF
Arabic NLP

ABSTRACT

Text clustering is a challenging task in natural language processing due to the very high dimensional space produced by this process (i.e. curse of dimensionality problem). Since these texts contain considerable amounts of ambiguities and redundancies, they produce different noise effects. For an efficient and accurate clustering algorithm, we need to extract the main concepts of the text by eliminating the noise and reducing the high dimensionality of the data. This paper compares among three of the famous dimension reduction algorithms for text clustering to show the pros and cons of each one, namely Principal Component Analysis (PCA), Nonnegative Matrix Factorization (NMF) and Singular Value Decomposition (SVD). It presents an effective dimension reduction algorithm for Arabic text clustering using PCA. For that purpose, a series of the experiments has been conducted using two linguistic corpora for both English and Arabic and analyzed the results from a clustering quality point of view. The experiments have shown that PCA improves the quality of the clustering process and that it gives more interpretable results with less time needed for the clustering process for both Arabic and English documents.

1. Introduction

Dimension reduction techniques are a very important step in the clustering process; it is not an easy problem due to the high dimensionality of the data. This high dimensions space damages the effectiveness of the clustering process (i.e., the curse of dimensionality problem) generally. The idea behind these techniques is to reduce dimensions by transforming existing features into a new low dimension features space [1–4]. The dimension reduction techniques such as PCA, NMF and SVD are the state of art in the field [5–9]. This paper compares among these three algorithms for text clustering to show the pros and cons of each one, namely PCA, NMF and SVD. In text mining and pattern recognition, Clustering analysis refers to the division of dataset into several topics. Each topic, also called a cluster contains data points that are similar to themselves and differ from points in other groups. Similarly, the document-clustering task collects a set of documents into groups or topics. In recent years, many researchers conducted research in this field due to its importance in the area of data extraction and the retrieval of a huge amount of text documents. In the clustering process, no information is provided on the previous label, so the task is to cluster a given data set into meaningful topics. The clustering divides a given set of parts into M topics (clusters) such that the parts of each topic are 'similar' to and 'different' from the objects of the other groups. Clustering approaches are divided as follows: Hierarchical techniques, partitioning techniques, grid based techniques, density based techniques, and model based techniques [10–15]. In the real world, there are many applications for clustering process such as search engines, community detection and recommendation systems. For example, in community detection system, the main challenge is to help users to uncover communities in complex network [16], whereas in news recommendation systems is to find interesting news articles to read. The clustering of news documents according to their themes is therefore an essential part because most readers only care about news on specific topics. In this paper, I used the partitioning approach. The partitioning approach constructs various partitions by decomposing the data points into a given number of mutual exclusive clusters. I used k-mean clustering algorithm for creating clusters of documents represented through various feature extraction methods. The k-means is a well-known clustering technique for document clustering. It is used frequently in many literatures due to its simplicity and efficiency. In this algorithm, there is a

E-mail address: dr_ashrafa@yahoo.com

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

ELSEVIER Production and hosting by Elsevier

https://doi.org/10.1016/j.eij.2019.05.002
1110-8665/© 2019 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information, Cairo University.

group of the cluster centers that labels each point into the nearest cluster and it updates the center position for each cluster [10–12,15,9]. This research is arranged as follows: II shows traditional machine learning verses deep learning, Section 3 introduces algebraic methods for dimension reduction. Section 4 shows Arabic language preprocessing. Section 5 evaluation metrics is described. In Section 6 explains data set and experimental setup. Section 7 shows obtained results and discusses these results. Finally, Section 8 gives conclusions and recommendations for future research.

2. Traditional machine learning versus deep learning

Learning algorithms can be classified into three main broad categories: (1) Supervised learning, (2) Unsupervised learning, and (3) Semi-supervised learning.

In supervised learning, the algorithm builds a mathematical model of a set of data that contains both the inputs and the desired outputs. Classification algorithms and regression algorithms are types of supervised learning. In the unsupervised learning, the algorithm builds a mathematical model of a set of data, which contains only inputs and no desired outputs. Unsupervised learning algorithms are used to find the clusters of data. Matrix decomposition algorithms such as SVD, NMF and PCA, which are called a blind source separation (BSS), are unsupervised learning algorithms [2–4]. The Semi-supervised learning algorithms develop mathematical models from incomplete training data, where portions of the sample inputs are missing the desired output. The Deep learning algorithm such as Word2vec is a natural language model which uses a semi supervised machine learning algorithm since it uses back-propagation to handle the feedback error to learn Neural Network (NN) [17–19]. From the computational point of view, the most important difference between deep learning and traditional machine learning is the nature of the performance of the former as the scale of data increases. When the data is small, deep learning algorithms do not perform that well. This is because deep learning algorithms need a large amount of data to understand it perfectly. In this research, I used unsupervised algebraic dimension reduction algorithms, which are the state of art in this field [5–9].

3. Algebraic methods for dimension reduction and text clustering

Building an accurate and efficient text-clustering algorithm needs to tackle the problem of high dimensionality by using an accurate dimension reduction technique. Since the high dimensionality produces a different, noise which affects the accuracy of the clustering algorithm. Any efficient clustering algorithm needs to extract the main concepts of the text by eliminating the noise and reducing the dimensionality of the data. The dimension reduction techniques decrease the dimensionality of data such that the non-characteristic features are eliminated from the building models by creating a new set of features from the original one, which is called, features transformation process. There are many algebraic methods used as dimension reduction techniques, which depend on matrix decomposition such as the SVD and the NMF or they may depend on eigenvalue decomposition such as the PCA [2–4,7,9]. In the following subsection, I will present three of the common algebraic dimension reduction techniques (i.e. the state of art in the field).

3.1. Singular value decomposition

SVD is a matrix decomposing algorithm and a feature transformation technique, in which new features are produced from the original one. The SVD is also used as a dimension reduction technique and features extractor. The SVD allows us to transform a matrix $X_{m\times n}$ to the diagonal form using unitary matrices. It factorizes the matrix $X_{m\times n}$ of the documents into three matrices as follows: $U_{m\times m}$ matrix, $V_{n\times n}$ matrix, and the diagonal matrix $\boldsymbol{\Sigma}$ $m\times n$ [2,20] thus:

$$X = U\sum V^T \tag{1}$$

$$\sum = diag(\sigma_1\sigma_2\cdots.\sigma_i\cdots\cdots.\sigma_r) \tag{2}$$

The $\boldsymbol{\Sigma}$ matrix (diagonal matrix) contains the singular values of X matrix and the singular values have the following property:

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq \sigma_{k+1} = \cdots = \sigma_r \cong 0 \tag{3}$$

The Features extracted from the SVD are based on using $\sum V^T$ as a base Matrix (i.e. the researcher used it as a reduced feature space instead of X matrix in the clustering algorithm).

3.2. Principal component analysis

PCA is a well-known method for dimension reduction. A mathematical technique uses an orthogonal linear transformation to convert the set of correlated variables into a set of new uncorrelated variables called principal components. The PCA linearly transforms the $X_{m\times n}$ matrix variables into a substantially smaller set of uncorrelated variables that contains much of the information of the original data set [3,21,22]. The method of computing the principal component from the data matrix X is done by computing the Eigen Decomposition of the covariance matrix of the data matrix X. But the fastest algorithm is by using the SVD to factorize the matrix X into U r$\sum$rV^Tr, which give the k sample principal components as the projection of U^Tk on the X data matrix. This projected space gives PCA after applying whiten transformation to normalize variances to one to remove any noises. In this paper the Features extracted using the PCA (i.e. the projected space after whiten transformation is used in the clustering algorithm) [3].

3.3. Non negative matrix Factorization

Many data have a non-negative nature and the hidden components of the data have physical meanings if they have positive values. The matrix decomposition methods such as the SVD and PCA can allow negative entries that gives negative values in the factorizing matrices, which has no physical interpretations. In the NMF algorithm the nonnegative matrix X $\in R^{m\times n}$ can be factorized into two matrix factors W and H by solving the optimization problem in Eq. 4 [23]. The semantic feature vectors resulted from the NMF are sparser than those resulted from other methods such as the SVD [16,24].

$$\begin{aligned} L(W,H) &= ||X - WH||_F^2 \\ \text{s.t.}W,H &\geq 0 \\ X &\approx WH \end{aligned} \tag{4}$$

where the Frobenius norm is denoted by ||.||. Notice that the objective function L(W, H) is convex in W only or H only. Let X is the term-document matrix that is decomposed into $W^{\boldsymbol{m}\times\boldsymbol{k}}$ (feature matrix), $H^{\boldsymbol{k}\times\boldsymbol{n}}$ (weighting or coefficient matrix) and k << min (m, n). In this piece of research the features extracted from NMF are based on weighting matrix H (i.e. using H matrix as the reduced feature space in the clustering algorithm).

4. Arabic language preprocessing

Arabic text preprocessing is the process of preparing the text for the computational analysis. It includes four main steps: segmentation, tokenizer, Stop list removal and stemmer. The process of stripping a given text down to its constituent sentences is called segmentation. After that, each sentence is split into its component words, which is called tokenization. In the Stop list removal step, all the common words from a list of 'common words' are removed from the text. The process of extracting the word stem for each term by eliminating suffixes and prefixes is called stemming [24,25]. The researcher used the light stemming (Light10 stemmer) which has outperformed other stemmers [26].

4.1. Representation phase

The researcher represents the corpus documents using term-document matrix. Each columns in this matrix stands for a given document whereas a row stands for a unique term. We represent a document of the corpus by a vector of terms in this document. The *j*th document is symbolized by vector Xj = [w_{1j}, w_{2j}, …, w_{mj}]^{T}, where w_{ij} is the term weight and the number of terms is m in the corpus.

4.2. 4.2 Term weight Formula (TFIDF)

In this paper, TFIDF is used as a term weight formula:

$$TFIDF_{ij} = TF_{ij}\log_2\left(\frac{N}{n_i}\right) \tag{5}$$

where the TF_{ij} is the term frequency as we have mentioned above in the preprocessing step. After that, the term weight can be computed according to the above formula. The term weight in document $_{\mathbf{i}}$ is equal to (wij) = $TFIDF_{\mathbf{ij.}}$ where N represents the number of the corpus documents,n_i is number of the documents that contains the term i. Using the vector model the dataset is represented by a matrix X $\in R^{M\times N}$, where the number of unique terms in the corpus is M and the elements of this matrix are represented by wij [24,25].

5. Evaluation metrics

Two evaluation criteria are used: the accuracy of clustering algorithm (AC) and the normalized mutual information (MI). Let a document di, the clustering label is li and the truth label is μi. We can define the AC as follows:

$$AC = \frac{\sum_{i=1}^{N}\pi(\mu_i, Image(l_i))}{N} \tag{6}$$

where the number of documents in the dataset is N, π(u, v) function takes a value of one when u = v and zero otherwise, and Image is a matching function that gives the best mapping for each clustering and the corresponding ground truth label. In this research, the Hungarian algorithm [27] is used to search for the optimal mapping.

The MI formula can be written as follows:

$$MI\left(K,K^{'}\right) = \sum_{k_i \epsilon K, k_j^{'} \epsilon K^{'}} pr\left(k_i, k_j^{'}\right).log_2\frac{pr\left(k_i, k_j^{'}\right)}{pr(k_i).pr\left(k_j^{'}\right)} \tag{7}$$

where the probabilities of the selected document that belongs to the clusters K or K′ $arepr(k_i)$ and $pr\left(k_j^{'}\right)$ respectively and the probability of a specified document that belongs to both K and K' is $pr\left(k_i, k_j^{'}\right)$. *The MI*($K$,$K'$) term takes a value between zero and max ($H(K)$, $H(K')$) while the entropies of K and K' are $H(K)$ and $H(K')$ respectively [20]. To simplify the MI values, I used the normalized mutual information $\widehat{MI}$ *formula instead of* ***MI***:

$$\widehat{MI}\left(K,K^{'}\right) = \frac{MI\left(K,K^{'}\right)}{\max\left(H(K),H\left(K^{'}\right)\right)} \tag{8}$$

6. Dataset and experiments

6.1. Dataset

In this research, I used two different data sets (i.e. two linguistic corpora) to conduct all experiments. The Arabic Corpus consists of 10 hand-made topics; the English Corpus consists of 10 topics from Reuters 21,578.

1) The Arabic dataset (Corpus) is a collection of about 7232 news documents (13,271 unique words) in 10 categories, and the topics range from computer, science to political talks.
2) The Reuters 21,578 dataset consists of a collection from the Reuters newswire. This corpus contains 21,578 documents in 135 topics. I selected 10 categories from this corpus and removed any document in more than one topic, leaving 7293 documents (18933 unique words) in 10 categories.

I applied common preprocessing methods for the Arabic corpus and English corpus such as stopword filtering, stemming using the stemmers (i.e. light stemmer for Arabic and Porter stemmer for English) and TFIDF term weighting formula is used to form a term document matrix for each corpus.

6.2. Experiments

In this paper, I used the K-Means algorithm as a clustering algorithm for the original data (without dimensionality reduction) and transformed data (after dimensionality reduction by using NMF, PCA and SVD. The number of clusters is considered equal to 10. I considered that the base matrix H resulted from NMF algorithm, the reduced space $\sum V^{T}$ resulted from SVD algorithm and the principal components resulted from PCA after applying whiting transform on these components (Zpc) as the latent reduced space. Then I applied the K-Means clustering algorithm on the transformed term-document matrix and I used cosine similarity as the distance metric.

The clustering process can be summarized in the following steps:

1. Do the pre-processing on the original document set and build the term-document matrix X for each corpus.
2. Perform NMF or PCA or SVD on X matrix to get the latent reduced space matrix (H, Zpc and $\sum V^{T}$).
3. Apply the K-means clustering algorithm on original space and on the latent reduced space. Which are produced from the above algorithms to get the cluster label for each document.
4. Use the bipartite matching algorithm (i.e. Hungarian algorithm) [27] to get the correct cluster labels for all documents.
5. Calculate the accuracy and normalized mutual information for each algorithm.

7. Results

In this section, I compared between the results obtained by applying K-means algorithm based on NMF, SVD and PCA as dimension reduction methods with results obtained by applying K-means algorithm without dimension reduction methods. As I mentioned above, I used two linguistic corpora (Reuters21578 Corpus and Arabic Corpus) to conduct these experiments. The results are shown in Tables 1 and 2 and Figs. 1 and 2 for English corpus and Tables 3 and 4 and Figs. 3 and 4 for Arabic corpus. Table 1 shows the performance of the different algorithms on Reuters 21,578.

Table 3 shows the performance of the different algorithms on Arabic corpus.

From the above tables it is clear that:

1) PCA results outperform K-means, SVD and NMF in terms of accuracy (AC = 0.667, AC = 0.352) and normalized mutual information $\left(\left(\widehat{MI}=0.700,\widehat{MI}=0.337\right)\right)$ using Arabic Corpus and Reuters 21,578 Corpus respectively.

Table 1
Comparison among the clustering algorithms performance using Reuters 21578.

	K-means	SVD	NMF	PCA	
				Without Whiten	With Whiten
MI	**0.337**	**0.313**	**0.304**	**0.311**	***0.337***
AC	**0.345**	**0.326**	**0.328**	**0.311**	***0.352***

Table 2
Comparison among the elapsed times of the clustering algorithms in milliseconds using Reuters 21578.

	K-means	SVD	NMF	PCA
Clustering Time in milliseconds	***650***	**77**	**70**	**85**

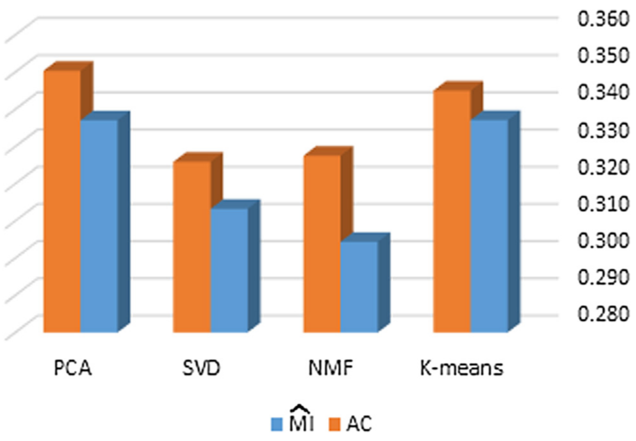

Fig. 1. Clustering Performance Comparisons using Reuters 21,578.

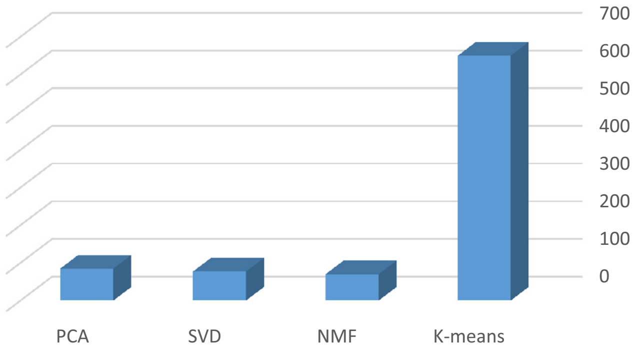

Fig. 2. Elapsed time in milliseconds for the clustering process using Reuters 21,578.

Table 3
Comparison among the clustering algorithms Performance using Arabic Document Dataset.

	K-means	SVD	NMF	PCA	
				Without Whiten	With Whiten
MI	**0.686**	**0.672**	**0.671**	**0.664**	***0.700***
AC	**0.649**	**0.646**	**0.640**	**0.642**	***0.667***

Table 4
Comparison among the elapsed times of the clustering algorithms in milliseconds using Arabic corpus.

	K-means	SVD	NMF	PCA
Clustering Time in milliseconds	***4610***	**47**	**38**	**43**

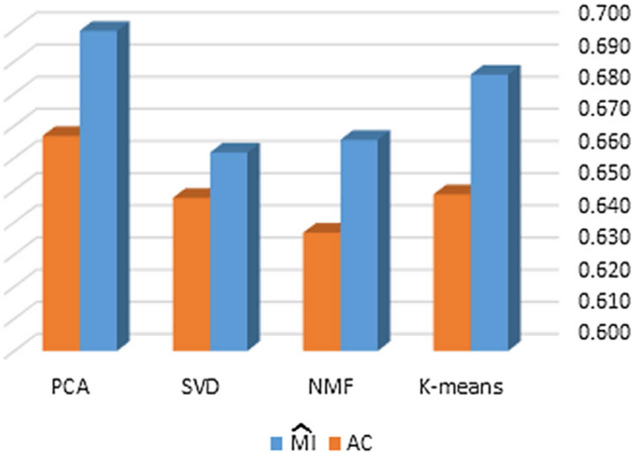

Fig. 3. Comparison among the clustering algorithms Performance using Arabic Document Dataset.

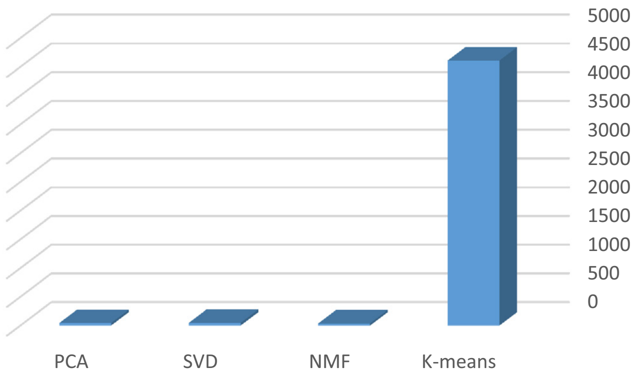

Fig. 4. Elapsed time in milliseconds for the clustering process using Arabic Corpus.

2) The experiments have shown that the PCA is better than the K-means algorithm for clustering Arabic and English texts with regards to clustering time in milliseconds (43 for Arabic and 85 for English Corpus) compared to time consumed for clustering in original space (4610 for Arabic and 650 for English Corpus).
3) SVD and NMF algorithms are approximately equal to each other in terms of accuracy, normalized mutual information and elapsed time for clustering using Arabic Corpus and Reuters 21,578 Corpus.

4) NMF is the faster clustering algorithm in terms of time in milliseconds (38 for Arabic corpus and 70 for English Corpus).

8. Conclusion

In this study, I introduced three famous dimension reduction algorithms (i.e. the state of art in this field namely, PCA, SVD and NMF) and compared among them in the clustering of Arabic texts. The experiments show that PCA gives better results as a dimension reduction technique in terms of accuracy and the normalized mutual information. The PCA yields better results in compact representation in the sense of semantic latent structure that indicates the ability of PCA to distinguish between texts. PCA is capable of easily identifying latent structures in the text data for both Arabic and English documents. From the researcher's point of view, the advantage of PCA is that it forces the principal component vectors to be orthogonal to each other, which is not achieved in NMF or SVD algorithms. The whiting transform accompanied with PCA algorithm enhances the algorithm performance by removing noises, which ultimately reduced the error value. Although there are many algebraic algorithms for text clustering, there has been, to the researcher's best knowledge, no study to date that includes both Arabic and English documents. Finally, perhaps the type and the size of the datasets and reprocessing techniques still influence the effects of the dimension reduction algorithms. However, I gave the readers some useful and helpful references for future research in clustering Arabic texts. For future work, the experiments may be extended to include other Language models like word embedded vector cooperated with the three algorithms (NMF, SVD and PCS) to enhance the clustering of Arabic texts.

References

[1] Liu Huan, Motoda Hiroshi. Computational Methods of Feature Selection. Chapman & Hall/CRC. Taylor & Francis Group, LLC; 2008.

[2] Zarzour Hafed, Al-Sharif Ziad, Al-Ayyoub Mahmoud, Jararweh Yaser. A new collaborative filtering recommendation algorithm based on dimensionality reduction and clustering techniques. 2018 9th International Conference on Information and Communication Systems (ICICS); 2018. p. 102–6.

[3] Kale Archana Pritam, Sonavane Shefali. PF-FELM: a robust PCA feature selection for fuzzy extreme learning machine. IEEE J Select Top Signal Process 2018:1.

[4] Austin W, Anderson D, Ghosh J. Fully supervised non-negative matrix factorization for feature extraction. IGARSS 2018–2018 IEEE International Geoscience and Remote Sensing Symposium; 2018. p. 5772–5.

[5] Allab Kais, Labiod Lazhar, Nadif Mohamed. A semi-NMF-PCA unified framework for data clustering. IEEE Trans. Knowledge Data Eng. 2017;29 (1):2–16.

[6] Kuang Da, Choo Jaegul, Park Haesun. Nonnegative matrix factorization for interactive topic modeling and document clustering. In: Partitional Clustering Algorithms. Springer; 2015. p. 215–43.

[7] Alghamdi Hanan, Selamat Ali. Topic Modelling used to improve Arabic Web pages Clustering. International Conference on Cloud Computing (ICCC); 2015. p. 1–6.

[8] Hosseini-Asl Ehsan, Zurada Jacek M. Nonnegative matrix factorization for document clustering: a survey. In: Rutkowski L, editor. ICAISC 2014, Part II, LNAI 8468. p. 726–37.

[9] Klinczak Marjori NM, Kaestner Celso AA. A Study on Topics Identification on Twitter using Clustering Algorithms. Latin America Congress on Computational Intelligence (LA-CCI); 2015. p. 1–6.

[10] Hennig C, Meila M, Murtagh F, Rocci R. Handbook of cluster analysis ISBN: 9781466551893. New York, USA: CRC Press; 2015.

[11] Christoph Thrun Michael. Projection-Based Clustering through Self-Organization and Swarm Intelligence. Published by Springer Vieweg; 2017. ISBN 978-3-658-20539-3.

[12] Aggarwal CC, Zhai C. A survey of text clustering algorithms. In: Mining Text Data. Springer; 2012. p. 77–128.

[13] Arbelaitz O, Gurrutxaga I, Muguerza J, Pérez JM, Perona I. An extensive comparative study of cluster validity indices. Pattern Recognit. Jan. 2013;46 (1):243–56.

[14] Andrews NO, Fox EA. Recent developments in document clustering Tech. Rep. TR-07-35. Computer Science. Virginia Tech; 2007.

[15] Pfitzner D, Leibbrandt R, Powers D. Characterization and evaluation of similarity measures for pairs of clusterings. Knowl. Inform. Systems 2009;19 (3):361–394p.

[16] Qin Yaoyao, Jia Caiyan, Li Yafang. Community Detection using Nonnegative Matrix Factorization with Orthogonal Constraint. 8th International Conference on Advanced Computational Intelligence Chiang Mai Thailand; 2016. p. 49–54.

[17] Collobert R, Weston J. A unified architecture for natural language processing: deep neural networks with multitask learning. In: Proceedings of the 25th International Conference on Machine learning. ACM; 2008. p. 160–7.

[18] Soliman Abu Bakr, Eissa Kareem, El-Beltagy Samhaa R. AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP. In: 3rd International Conference on Arabic Computational Linguistics. Dubai, United Arab Emirates: ACLing; 2017. p. 256–65.

[19] Salamaa Rana Aref, Youssefb Abdou, Fahmya Aly. Morphological Word Embedding for Arabic. In: The 4th International Conference on Arabic Computational Linguistics. Dubai United Arab Emirates: ACLing; 2018. p. 83–93.

[20] Taufik Fuadi Abidin, Bustami Yusuf, and Munzir Umran, "Singular Value Decomposition for Dimensionality reduction in Unsupervised Text Learning Problems" ICETC, 2010 IEEE International Conference on education Technology and Computer.

[21] Jolliffe I. Principal component analysis. Wiley Online Library; 2005.

[22] Combesa C, Azemab J. Clustering using principal component analysis applied to autonomy–disability of elderly people. Decis Support Syst 2013;55 (2):578–86.

[23] Hou Mi-Xiao, Gao Ying-Lian, Liu Jin-Xing. Comparison of non-negative matrix factorization methods for clustering genomic data. ICIC 2016 Part II, LNCS 2016;9772:290–9.

[24] Mohamed AA. Automatic Summarizition of the Arabic documents using NMF A preliminary study. 11th International Conference on Computer Engineering & Systems (ICCES); 2016. p. 235–40.

[25] Mohamed AA. An improved algorithm for information hiding based on features of Arabic text: a unicode approach. Egypt Inf J 2014;15:79–87.

[26] Larkey Leah S, Ballesteros Lisa, Connell Margaret E. Light stemming for Arabic Information retrieval. In: Arabic computational morphology knowledge-based and empirical methods, text, speech and language technology series. The Netherlands: Springer; 2007. p. 221–43.

[27] Kuhn HW. The Hungarian method for the assignment problem. Nav Res Logist Quat 1955;2:83–97.