Mehran University Research Journal of Engineering & Technology
Vol. 39, No. 1, 213-222, January 2020
p-ISSN: 0254-7821, e-ISSN: 2413-7219
DOI: 10.22581/muet1982.2001.20

An Efficient Topic Modeling Approach for Text Mining and Information Retrieval through K-means Clustering

JUNAID RASHID*, SYED MUHAMMAD ADNAN SHAH*, AUN IRTAZA*

RECEIVED ON 24.09.2018, ACCEPTED ON 09.10.2018

ABSTRACT

Topic modeling is an effective text mining and information retrieval approach to organizing knowledge with various contents under a specific topic. Text documents in form of news articles are increasing very fast on the web. Analysis of these documents is very important in the fields of text mining and information retrieval. Meaningful information extraction from these documents is a challenging task. One approach for discovering the theme from text documents is topic modeling but this approach still needs a new perspective to improve its performance. In topic modeling, documents have topics and topics are the collection of words. In this paper, we propose a new k-means topic modeling (KTM) approach by using the k-means clustering algorithm. KTM discovers better semantic topics from a collection of documents. Experiments on two real-world Reuters 21578 and BBC News datasets show that KTM performance is better than state-of-the-art topic models like LDA (Latent Dirichlet Allocation) and LSA (Latent Semantic Analysis). The KTM is also applicable for classification and clustering tasks in text mining and achieves higher performance with a comparison of its competitors LDA and LSA.

Key Words: Topic Modeling, Local term weighting, Entropy, Bag-of-words, Principal component analysis, K-means

1. INTRODUCTION

Topic modeling for text mining is used to identify the patterns in text documents. It is a statistical approach for discovering hidden topics form documents collections. The topic models are evolving from text documents that are modelled as weight mixture of the topics and each topic is a probability distribution over words. Topic modeling techniques are mainly used for finding themes from different documents such as news articles, web pages, and social news etc.

Topic modeling techniques are mainly used for searching, browsing and summarizing the text corpus. Topic modeling discovers hidden topics from documents. The documents are annotating with these topics and these annotations have been used for searching, summarizing and organizing from predictions.

Topic modeling is a probabilistic technique that is widely accepted in text mining and information retrieval fields

Authors E-Mail: (junaidrashid062@gmail.com, syed.adnan@uettaxila.edu.pk, aun.irtaza@uettaxila.edu.pk)
***Department of Computer Science, University of Engineering and Technology, Taxila, Pakistan

[1, 2]. The basic goal of topic modeling is shown in Fig. 1. Documents are about various topics at the same time and topics are associated with different words. The goal of topic modeling is to extract the thematical structure from the text corpus. The output of topic modeling is set of multi-distribution of topics. In topic modeling words are extracted from a collection of documents and words are belong to some topic.

The process of examining text documents to create new documents and transform unstructured text data to structure text data for more processing is called text mining. Several statistical and probabilistic topic-modeling approaches are used for text mining tasks such as temporal text mining [3], contextual text mining [4] and comparative text mining [5].

The obtaining of an information resource which is relevant to the information need form text corpus is called information retrieval. Topic models incorporate the framework for language model and achieve effective information retrieval results [2, 6-8].

The cross collection mixture model [5] for text mining is proposed that is based on probabilistic latent semantic analysis [9] discover the themes from documents collection.

The LDA [10] model generate topics from documents in term of the probability distribution of words for every topic. LDA technique is also used as a dimension reduction technique but does not achieve high classification accuracy [11].

Documents
Topics
Words

FIG. 1. THE GOAL OF TOPIC MODELING

The LSA based documents are usable for many applications of information retrieval [12, 13] for discovering lexical semantics and simple word co-occurrence approach perform better [14,15].

Topics extraction from documents is performed which clusters text documents in groups with the similarity of semantic terms [16,17]. The models are good for classification use but the limitation of these models is that every document is associated with one cluster only.

The PLSA (Probabilistic Latent Semantic Analysis) [9] topic model is an extension of LSA to fix some issues of LSA. PLSA improves the LSA in a probabilistic sense and use the generative model.

The aspect model is based on the statistical model as an extension of PLSA [18]. Aspect model is also called latent variable model for data co-occurrence and associate unobserved calls with every observation [19].

The correlated topic model uses normal logistic distribution to generate a relationship with topics and allow words to occurrences in every topic [20]. Limitation of this method is that it needs numerous calculations.

Common semantic topic model [21] is used for filtering of noise in short text topic modeling. Word co-occurrence network-based topic model [22] extracts topics from short news and apply to cluster on these topics.

In this research, we propose an efficient K-means topic modeling approach that discovers the semantic hidden topics from text documents. KTM is used for text classification and clustering tasks in text mining. Experimental results on real-world datasets show that KTM performance is better than LDA and LSA which are state-of-the-art topic models.

2. MATERIALS AND METHODS

2.1. Proposed k-means Topic Modeling Approach

The proposed K-means topic modeling approach consists of the following steps.

Step 1

Preprocessing of text documents is performed in this step. Different preprocessing steps are performed as shown in Fig. 2. The punctuation is eliminated from raw text and transformed into lowercase. The text may contain stop words [29] such as is, then, are, of etc. After that, stop words and short words are removed. However, removing these words are normalized through porter stemmer and documents are cleaned.

Step 2

After the preprocessing step Bag of Words (BOW) model is used on preprocessing text documents. BOW model [23] is used for words occurrences in text documents. In natural language processing, a document is usually represented by a BOW that is a word-document matrix. BOW example is shown in Table 1. There are six documents (d1, d2, d3, d4, d5, d6) and four words (bank, account, customer, manager). The word bank occurs three times in document one and four times in document three. The different words occurrence in documents is different as shown in Table 1.

Step 3

Local term weighting has been calculated in this step. Term frequency [24] method has been used for finding the local term weighting. This method estimates that how much a term is appearing in documents collection. Term frequency has been described in equation 1. The typical weight term that uses the vector length of the normalization factor i is shown in equation 1. Documents are represented by d and k for terms in documents. The most important term is represented by one and less important term represented by zero. The weight w is used for the term k.

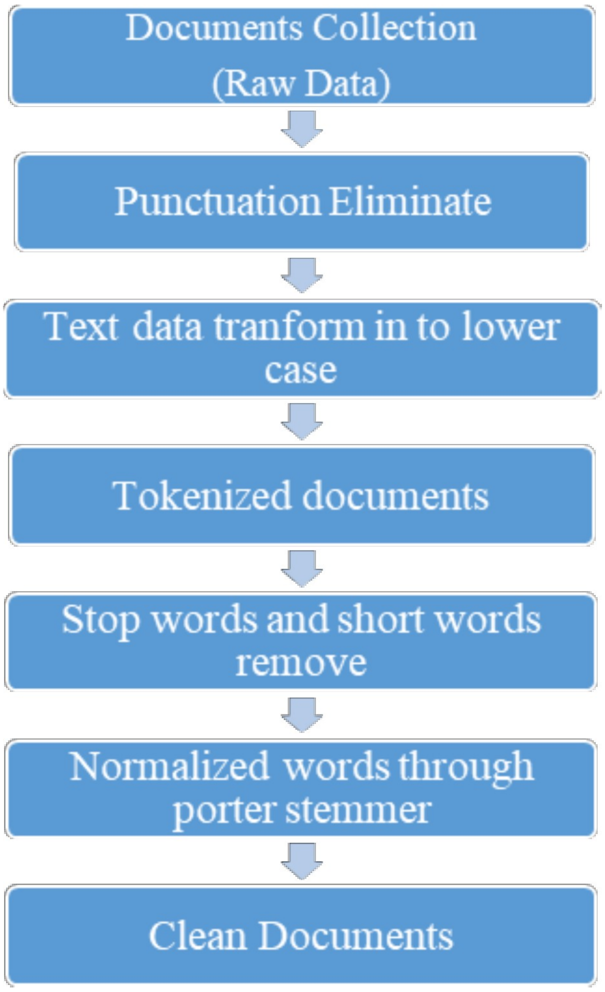
Documents Collection (Raw Data)
Punctuation Eliminate
Text data tranform in to lower case
Tokenized documents
Stop words and short words remove
Normalized words through porter stemmer
Clean Documents

FIG. 2. PREPROCESSING STEPS FOR TEXT DOCUMENTS COLLECTION

TABLE 1. EXAMPLE OF BOW

	D1	D2	D3	D4	D5	D6
Bank	3	0	4	0	0	0
Account	0	6	0	3	0	0
Customer	0	0	0	0	3	2
Manager	0	0	4	7	0	0

$$\text{Term Weight} = {w_{dk}} \Big/ \sqrt{\sum_{vector} (w_d i)^2} \qquad (1)$$

Step 4

Global term weighting has been calculated in this step by using the Entropy method. The global term weighting through entropy is calculated by finding $b(tf_{ij})$ and P_{ij} through equation 2, 3.

$$b(tf_{ij}) = \begin{Bmatrix} 1 & if & tf_{ij} > 0 \\ 0 & if & tf_{ij} = 0 \end{Bmatrix} \qquad (2)$$

$$P_{ij} = \frac{tf_{ij}}{\sum_j tf_{ij}} \qquad (3)$$

The GTW (Global Term Weighting) with entropy is calculated by using equation 4. The *N* is amount documents and n_i documents amount in which *i* term appear. The entropy assigns a higher weight to a term that is of lower frequency in documents [25].

$$\text{Entropy} = 1 + \sum_{j=1}^{N} \frac{\frac{f_{ij}}{F_i} \log \frac{f_{ij}}{F_i}}{\log N} \qquad (4)$$

The output of this step is the entropy term matrix.

Step 5

To avoid high dimensionality negative impact on global term weighting of entropy in step 4 principal component analysis (PCA) [26] is used. The PCA objective is to reduce the large set of variables to a small set of variables that even holds information the information in a large set. In PCA to make a process fast, we select two dimensions which are minimum dimensions.

Step 6

In this step, K-means clustering [27] algorithm is used that clusters the documents, which represents in global term weighting method of entropy. K-means clustering finds that how many k clusters exist in data. The algorithm iteratively moves k-centers and data points are selected which are the closest centroid. K-means clustering algorithm minimizes the objective function as shown in equation 5.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 \qquad (5)$$

Where chosen measure distance is shown in equation 6 and measure between the $x_i^{(j)}$ data points and c_j is cluster centers that is distance indicator for n data point for the particular cluster center.

$$\left\| x_i^{(j)} - c_j \right\|^2 \qquad (6)$$

Step 7

Documents term matrix are used with GTW method in step 3 (Words × Documents matrix) for finding the probability of documents $P(W)$. The W represent words and D represent the documents in equation 7.

Here *i* represents a number of documents. *P(D$_j$)* is calculated by equation 7.

$$P(D_j) = \frac{\sum_{i=1}^{m} (\mathrm{W_i, D_j})}{\sum_{i=1}^{m} \sum_{j=1}^{n} (\mathrm{W_i, D_j})} \qquad (7)$$

Step 8

The probability of documents *j* in topics *k* to find $P(D_j \mid T_k)$ is found by using the $P(D_j)$ and $P(T_k \mid D_j)$ through equation 8. The T represents the topics and D represents documents in equation 8.

$$P(D_j, T_k) = P(T_k \mid D_j) \times \mathrm{P(D_j)} \quad (8)$$

Step 9

The probability of words *i* in documents *j* and $P(D_j \mid T_k)$ is calculated by the document-term matrix and GTW method through equation 9.

$$P(\mathrm{W}_i \mid \mathrm{D}_j) = \frac{P(\mathrm{W_i}, D_j)}{\sum_{i=1}^{m} P(\mathrm{W_i}, D_j)} \quad (9)$$

Step 10

The probability of words i in topics k is calculated using and through equation 10.

$$P(\mathrm{W}_i \mid T_k) = \sum_{j=1}^{n} P(\mathrm{W}_i, D_j) \times \mathrm{P(D_j} \mid T_k) \quad (10)$$

2.2. Datasets

We used two datasets including Reuters 21578 news dataset and BBC news articles dataset. The Reuters 21578 datasets contain many classes, in this research two big classes including acq and earn have been selected for experiments. Reuters 21578 dataset is used for the classification and BBC news dataset is used for clustering. Table 2 describes the basic statistics of these two datasets.

TABLE 2. DATASETS BASIC STATISTICS

Dataset	Documents	Terms	Description
Reuters 21578	2070	18381	News
BBC	2225	9635	News articles

3. EXPERIMENTAL RESULTS AND DISCUSSION

3.1. Classification Results

First evaluation of classification is performed on Reuters 21578 with Bayesian optimization. The classification has been performed on the probability of topics over documents P(T|D). The fit function in Matlab is used for Bayesian optimization. The classification has been performed through Fit binary classification decision tree (fitctree) to achieve the high-performance ratio.

The KTM performance has been checked with LDA and LSA model on tenfolds cross-validation method. This technique evaluates the predictive model by subdividing the original sample into training and testing set. Through this method, data is divided into 10 subsets for 10 iterations. Each of the subsets is selected for training and others for testing. The training datasets percentage is 70 and testing dataset percentage is 30 for classification.

Fit binary classification decision tree method is used with 25, 50, 75, 100, 125, 150,175 and 200 topics for the input features of documents in classification.

Precision, recall, and accuracy are used to check the performance of KTM in comparison with LDA and LSA topic models. Equations 11, 12 and 13 show the precision, recall and accuracy formula.

True negative (TN) is correct predictions that instance is negative.

False positive (FP) is incorrect predictions with the positive instance.

False negative (FN) is incorrect predictions and the instance is negative.

True positive (TP) is correct predictions with the positive instance.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (11)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (12)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

Classification results on Reuters 21578 dataset are shown in Table 3 in terms of accuracy, precision, and recall. Results indicate that the proposed topic model KTM gives a higher performance as compared to LSA and LDA model on different numbers of topics. The KTM classification results accuracy is 89.21, 82.45, 82.61, 81.16, 78.74, 78.17, 80.52 and 76.01 with numbers of topics 25, 50, 75, 100, 125, 150, 175 and 200 which is increased and higher than LDA and LSA topic models.

TABLE 3. CLASSIFICATION RESULTS OF REUTERS 21578 DATASETS WITH 25, 50, 75, 100, 125, 150, 175 AND 200 TOPICS

Method	Accuracy	Precision	Recall	#Topics
LDA	55.23	0.4105	0.3966	25
LSA	52.81	0.3833	0.3882	25
KTM	89.21	0.8602	0.8565	25
LDA	53.14	0.3953	0.4304	50
LSA	53.94	0.3939	0.3840	50
KTM	82.45	0.7602	0.7890	50
LDA	56.03	0.4151	0.3713	75
LSA	52.97	0.3810	0.3713	75
KTM	82.61	0.7841	0.7511	75
LDA	56.84	0.4367	0.4515	100
LSA	52.49	0.3717	0.3544	100
KTM	81.16	0.7727	0.7173	100
LDA	51.04	0.3696	0.4008	125
LSA	53.30	0.3936	0.4135	125
KTM	78.74	0.7397	0.6835	125
LDA	47.50	0.3169	0.3249	150
LSA	52.33	0.3766	0.3797	150
KTM	76.17	0.6862	0.6920	150
LDA	48.95	0.3425	0.3671	175
LSA	53.78	0.3832	0.3460	175
KTM	80.52	0.7377	0.7595	175
LDA	50.72	0.3532	0.3502	200
LSA	52.33	0.3806	0.3966	200
KTM	76.01	0.6803	0.7004	200

3.2. Time Execution of Classification

KTM time execution of classification on Reuters 21578 dataset is compared with LDA and LSA. In this experiment famous method Gibbs sampling has been used that need multiple numbers of iterations and increases the computational cost. KTM performance is stable with the increasing numbers of topics and better than state-of-the-art topic models LDA and LSA as shown in Fig. 3.

3.3. Clustering Results

The second performance of clustering has been checked on unlabeled BBC dataset. Documents clustering performed on P(T|D) and K-means clustering technique is used. Different numbers of topics and clusters are evaluated using Calinski-Har-abasz index [28] internal validation method. Calinski-Har-abasz index calculates the validity of clusters based on the average of the sum squared error clusters. Calinski-Harabasz index criterion values are used to estimate an optimal number of clusters. The higher Calinski-Har-abasz index shows better clustering results.

The performance of KTM is compared with LSA and LDA model with 2 to 10 numbers of clusters on 25, 50, 75, 100,125,150, 175 and 200 numbers of topics. The result of CH index indicates that the proposed technique performance is higher than LDA and LSA models with different features and clusters.

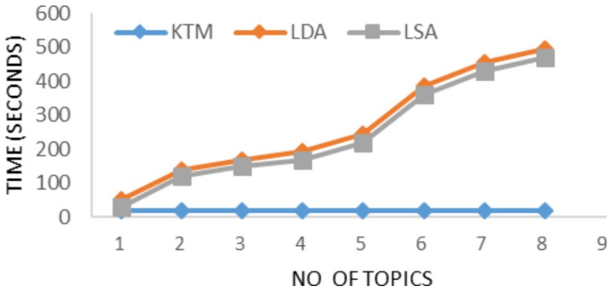

FIG. 3. KTM TIME EXECUTION OF CLASSIFICATION WITH A COMPARISON TO LDA AND LSA

Fig 4. shows that KTM CH-index is higher than the LDA and LSA topic models with 25 numbers of topics. Therefore, KTM clustering results are better than LDA and LSA on 25 numbers of topics.

Fig. 5 shows that KTM CH-index is also higher than the LDA and LSA topic models with 50 numbers of topics. So, KTM clustering results are better than LDA and LSA on 50 numbers of topics.

Fig. 6 shows that KTM CH-index is higher than the LDA and LSA topic models with 75 numbers of topics. So, KTM clustering results are better than LDA and LSA on 50 numbers of topics.

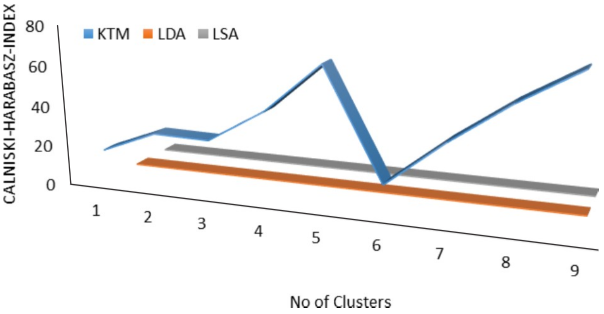

FIG. 4. CALINSKI-HARABASZ FOR 25 TOPICS OF BBC DATASET

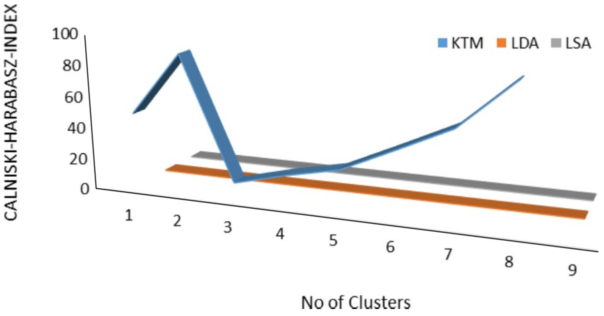

FIG. 5. CALINSKI-HARABASZ FOR 50 TOPICS OF BBC DATASET

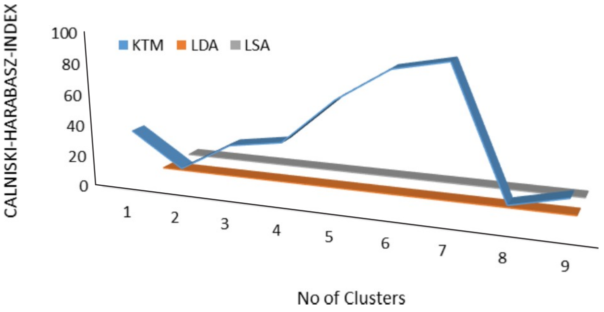

FIG. 6. CALINSKI-HARABASZ FOR 75 TOPICS OF BBC DATASET

Fig. 7 shows that KTM CH-index is higher than the LDA and LSA topic models with 100 numbers of topics. So, KTM clustering results are better than LDA and LSA on 100 numbers of topics.

The Fig. 8 shows that KTM CH-index is higher than the LDA and LSA topic models with 125 numbers of topics. So, KTM clustering results are better than LDA and LSA on 125 numbers of topics.

Fig. 9 shows that KTM CH-index is higher than the LDA and LSA topic models with 150 numbers of topics. So, KTM clustering results are better than LDA and LSA on 150 numbers of topics.

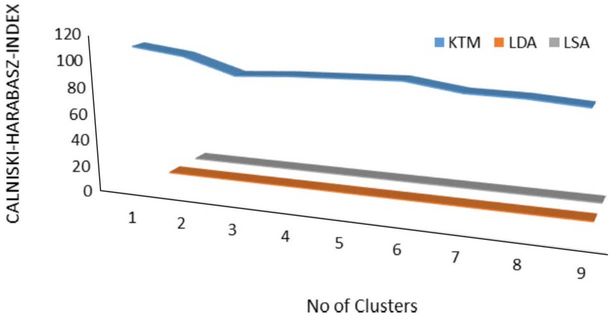

FIG. 7. CALINSKI-HARABASZ FOR 100 TOPICS OF BBC DATASET

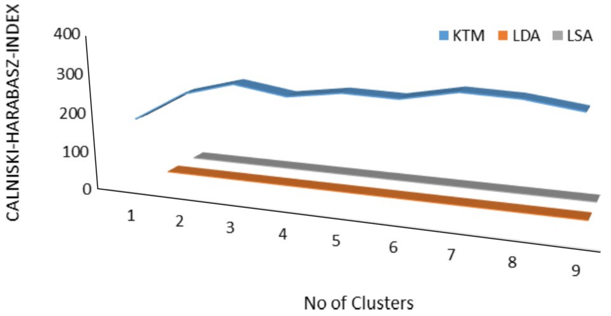

FIG. 8. CALINSKI-HARABASZ FOR 125 TOPICS OF BBC DATASET

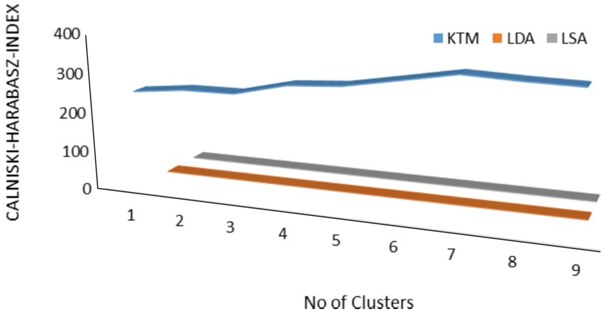

FIG. 9. CALINSKI-HARABASZ FOR 150 TOPICS OF BBC DATASET

Fig. 10 shows that KTM CH-index is higher than the LDA and LSA topic models with 175 numbers of topics. So, KTM clustering results are better than LDA and LSA on 175 numbers of topics.

Fig. 11 shows that KTM CH-index is higher than the LDA and LSA topic models with 200 numbers of topics. So, KTM clustering results are better than LDA and LSA on 200 numbers of topics.

3.4. Time Execution of Clustering

KTM time execution of clustering on BBC news dataset is compared with LDA and LSA. KTM performance is stable with the increasing number of topics and better than state-of-the-art topic models LDA and LSA as shown in Fig. 12.

3.5. EXAMPLE OF TOPIC MODELING FOR BBC NEWS DATASET

An example of a sports topic from BBC news datasets through LDA, LSA, and KTM topic models are shown in Table 4. The example shows that KTM discovers the more relevant words for sports topic as compared to LDA and LSA topic models.

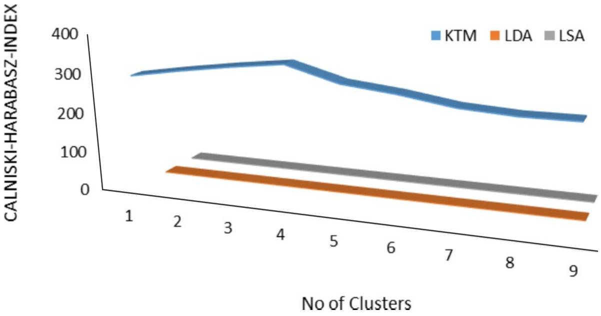

FIG. 10. CALINSKI-HARABASZ FOR 175 TOPICS OF BBC DATASET

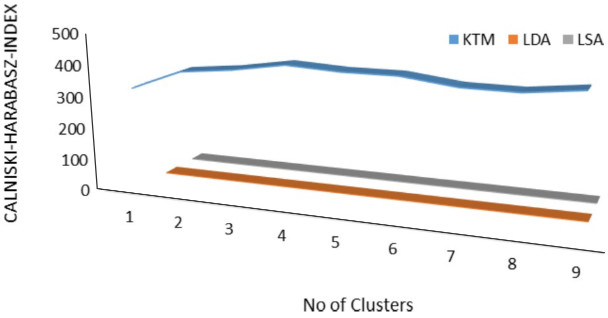

FIG. 11. CALINSKI-HARABASZ FOR 200 TOPICS OF BBC DATASET

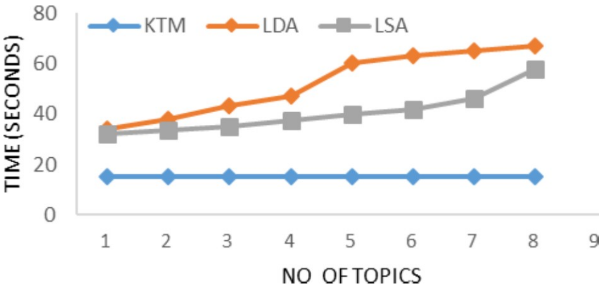

FIG. 12. KTM TIME EXECUTION OF CLUSTERING WITH A COMPARISON TO LDA AND LSA

TABLE 4. EXAMPLE OF THE EXTRACTED TOPIC (SPORT) FROM BBC NEWS DATASETS

KTM	LDA	LSA
Cricket	Delight	Lost
Cycling	Golf	Squash
Sportsman	Computer	Rob
Game	Cheer	Pupil
Golf	Attitude	Ballet
Athletics	Soccer	Game
Tennis	Lost	Tennis
Squash	Ballet	Charity
Soccer	Tennis	Floppy
Lost	Squash	Play

4. CONCLUSION

In this paper, we proposed a K-means topic modeling approach for news text documents. Topic modeling for news text is a challenging task due to the increase of news content on the web. Here we proposed a new K-means topic modeling approach for news text documents. KTM discovered more precise topics from documents and topics are multi-distribution of words.

The experimental results on two real-world datasets indicate that KTM can learn more coherent topics and it has been competitive against state-of-the-art topic models such as LDA and LSA for classification and clustering tasks.

The classification results are calculated with different numbers of topics from 25 to 200.Classification results show that KTM performance in terms of precision, recall, and accuracy is higher than state-of-the-art topic model LDA and LSA with numerous numbers of topics.

The KTM performance also measures for clustering with CH-index on 25 to 200 numbers of topics. Clustering results of KTM are also better than LDA and LSA with different numbers of topics.

Overall performance of KTM for classification and clustering is better than LDA and LSA topic models. KTM can be utilized in text mining and information retrieval field.

ACKNOWLEDGMENTS

This research work is supported by the University of Engineering and Technology, Taxila, Pakistan. The authors are very thankful to the University of Engineering and Technology, Taxila, Pakistan for providing all the required resources to perform this research.

REFERENCES

1. D. Ramage, C. D. Manning, and S. Dumais, "Partially labeled topic models for interpretable text mining," Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 457-465, 2011.

2. X. Wei and W. B. Croft, "LDA-based document models for ad-hoc retrieval, "Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 178-185, 2006.

3. Q. Mei and C. Zhai, "Discovering evolutionary theme patterns from the text: an exploration of temporal text mining," Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pp. 198-207, 2005.

4. Q. Mei and C. Zhai, "A mixture model for contextual text mining," in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 649-655,2006.

5. C. Zhai, A. Velivelli, and B. Yu, "A cross-collection mixture model for comparative text mining", Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 743-748,2004.

6. L. Azzopardi, M. Girolami, and C. Van Rijsbergen, "Topic-based language models for ad hoc information retrieval ", IEEE International Joint Conference on, pp. 3281-3286, 2004.

7. H.-Y. Lee and L.-S. Lee, "Improved semantic retrieval of spoken content by document/query expansion with a random walk over acoustic similarity graphs ", IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), pp. 80-94, 2014.

8. X. Yi and J. Allan, "A comparative study of utilizing topic models for information retrieval ", European conference on information retrieval, pp. 29-41,2009.

9. T. Hofmann, "Probabilistic latent semantic analysis ", Proceedings of the Fifteenth Conference on Uncertainty in artificial intelligence, pp. 289-296, 1999.

10. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation ", Journal of Machine Learning Research, pp. 993-1022, 2003.

11. S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories ", In null, pp. 2169-2178, 2006.

12. T. K. Landauer and S. Dumais, "Latent semantic analysis ", Scholarpedia, p. 4356, 2008.

13. W. Kintsch, D. S. McNamara, S. Dennis, and T. K. Landauer, "LSA and meaning: In theory and application ", Handbook of latent semantic analysis, pp. 467-479, 2007.

14. J. A. Bullinaria and J. P. Levy, "Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming, and SVD ", Behavior research methods, pp. 890-907, 2012.

15. T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge ", Psychological review, p.211, 1997.

16. A. Popescul, G. W. Flake, S. Lawrence, L. H. Ungar, and C. L. Giles, "Clustering and identifying temporal trends in document databases ", Advances in Digital Libraries Proceedings IEEE, pp. 173-182, 2000.

17. A. McCallum, K. Nigam, and L. H. Ungar, "Efficient clustering of high-dimensional data sets with application to reference matching ", Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 169-178, 2000.

18. T. Kakkonen, N. Myller, E. Sutinen, and J. Timonen, "Comparison of dimension reduction methods for automated essay grading ", Journal of Educational Technology & Society, pp. 275-288, 2008.

19. S. Tarapiah, S. Atalla, and B. Alsayid, "Smart onboard transportation management system Geo-Casting featured ", Computer Applications and Information Systems (WCCAIS), pp. 1-6, 2014.

20. J. D. Lafferty and D. M. Blei, " Correlated topic models ", Advances in neural information processing systems ", pp. 147-154, 2006.

21. X. Li, Y. Wang, A. Zhang, C. Li, J. Chi, and J. Ouyang, "Filtering out the noise in short text topic modeling ", Information Sciences, pp. 83-96, 2018.

22. A. Sahni and S. Palwe, "Topic Modeling on Online News Extraction ", Intelligent Computing and Information and Communication, pp. 611-622, 2018.

23. R. J. Y. Zhang, and Z.-H. Zhou, "Understanding bag-of-words model: a statistical framework " International Journal of Machine Learning and Cybernetics, pp. 43-52, 2010.

24. G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval ", Information processing & management, pp. 513-523, 1988.

25. S. Dumais, "Enhancing performance in latent semantic indexing (LSI) retrieval ", 1992.

26. H. Abdi and L. J. Williams, "Principal component analysis ", Wiley interdisciplinary reviews computational statistics, pp. 433-459, 2010.

27. K. Alsabti, S. Ranka, and V. Singh, "An efficient k-means clustering algorithm ", 1997.

28. T. Cali?ski and J. Harabasz, "A dendrite method for cluster analysis ", Communications in Statistics-theory and Methods, pp. 1-27, 1974.

29. Wilbur, W. John, and Karl Sirotkin. "The automatic identification of stop words ", Journal of information science , pp. 45-55,1992.