Journal of Theoretical and Applied Information Technology
31st January 2024. Vol.102. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL HEURISTIC FOR GRAPH-BASED TOPIC MODELING USING SPECTRAL CLUSTERING

P.K.PATTANAYAK [1], R.M.TRIPATHY[2], S.PADHY[3]
[1]Asst Professor, Silicon Institute of Technology, Department of Computer Science and Engineering, India
[2]Associate Professor. XIM University, School of Computer Science and Engineering, Harirajpur, India
[3]Professor. Institute of Mathematics and Applications, India
E-mail: [1]ppattanayak@silicon.ac.in, [2]rudramohan@xim.edu.in, [3]spadhy07@gmail.com

ABSTRACT

Topic modeling is one of the popular techniques for identifying the latent topic from a large corpus of text data. Various topic modeling techniques have been studied for managing short and long texts that consider different kinds of interactions and constraints within a dataset. Most researchers use Latent Dirichlet Analysis (LDA) and an extension of the LDA algorithm for topic modeling. While these algorithms are flexible and adaptive, they are occasionally a poor choice for modeling increasingly complex data relationships. Topic modeling has used various encoding strategies, many of which do not adequately represent the semantic relationships between the words. This study proposes a novel heuristic for the graph-based topic modeling technique and applies it to a benchmark dataset, which outperforms the current LDA model for short text. The proposed heuristic is based on graph-splitting methods. We used it on the TripAdvisor hotel review dataset, a sizable collection of huge text corpora. Our suggested strategy has been demonstrated to outperform several current methods for concept extraction and effective topic. The detailed result was compared based on the coherence score. We also employed word cloud and compared the outcome to user reviews, demonstrating that our performance is superior to many of the already used methods.

Keywords: *Topic Modeling, Spectral Clustering, Word2Vec, Graph-Splitting, Linear Dirchlet Allocation*

1. INTRODUCTION

A method to extract hidden topics from enormous amounts of text is called topic modeling as in [1]. A huge number of documents can be analysed statistically to discover the underlying semantic structure. Without already understanding the themes, topic modeling seeks to identify the topics or clusters within a corpus of texts, such as emails, news articles, tweets, etc. Normally tagged or annotated data are not available for topic modeling. We just have raw text data, and topic modeling algorithms will identify the topics from this corpus of data. The overall workflow of the topic modelling is given in Figure1.

The most commonly used topic modeling techniques are Latent Dirichlet Allocation (LDA) [2] Non-Negative Matrix Factorization (NMF) [3], Latent Semantic Analysis (LSA) [4], Probabilistic Latent Semantic Analysis (PLSA) [5], Gibbs Sampling Dirichlet Mixture Model (GSDMM) [6], and Graph-based topic modelling [7]. We proposed a novel heuristic for Graph based Topic Modeling using Spectral clustering which is being compared with some popularly used in recent research on topic

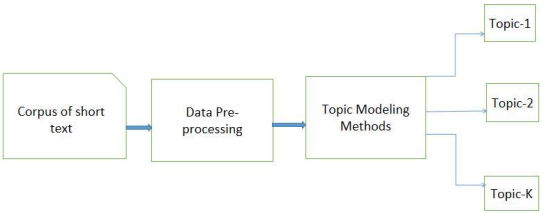


Figure 1 Workflow of Topic modelling

modeling that is Latent Dirichlet Analysis (LDA) and Gibbs Sampling Dirichlet Mixture Model (GSDMM). To have a clear understanding of our techniques, the basic overview of some basic topic modeling techniques are given in Section 1.1

1.1. Topic Modeling Techniques

Latent Dirichlet Allocation (LDA). LDA is a probabilistic graphical model which is used to obtain relationships between multiple documents in a corpus [2]. This technique works on two basic assumptions: documents are a mixture of topics, and the topics are a mixture of words. Here each word is related to a latent topic, which is represented by Z (say). The word distribution of Z is represented by θ. The two important

parameters, α (document-topic distribution), and β (topic-word distribution) control the LDA model. This model efficiently represents the data into two matrices: *document* × *topic* and *topic* × *word* as in [8], [9]. The mathematical formulation of LDA is:

$$P(W, Z, \theta, \psi, \alpha, \beta) = \prod_{i=1}^{T} P(\psi_i; \beta) \prod_{j=1}^{D} P(\theta_j; \alpha) \prod_{i=1}^{N} P(Z_{j,t}|\theta_j) P(W_{j,t}|\psi Z_{jt})$$

Where, T is the number of topics, D is the number of documents, N is the number of words, $\psi_{i=1..T}$ is the distribution of words in topics T, $\theta_{d=1..D}$ is the distribution of topics in document d, $Z_{d=1..D,\ \omega=1..N}$ is the identity topic of words ω in document d, Z_{ij} is the topic for the j[th] word in document i, W is the vector of words in the corpus, α and β is the Dirichlet prior parameter on the per-document topic distributions

Gibbs Sampling Dirichlet Mixture Model (GSDMM). This technique describes the method of iterating through and reassigning clusters based on a conditional probability. It works like the Naive Bayes Classifier, where the documents are assigned to clusters based on the highest conditional probability. Feifei et al [10] suggested a Bayesian inference of class-specified topic model, with the supervised scenario being a specific instance. This model's basic assumption is, a single topic sample for each text as in [6]. This model claim to address the sparsity problem of short text clustering and also displaying word topics, but it has not captured the semantics of the words in the process of model generation.

Word2Vec. Word2Vec is a technique [11] which is used to generate fixed length distributed vector representation of each word of the corpus. The basic objective of using Word2Vec has two important reasons, first one is fixed-size vectors, meaning that the size of the vector is independent of the number of distinct words in the corpus and the second one is, incorporating semantic information in the vector representations. This technique is highly efficient at grouping similar words together. The algorithm can make strong estimates based on the position of the word in the corpus. The most crucial aspect of Word2Vec is that it preserves the context information while maintaining the semantic meaning of various words in a document, and at the same time it does not lose the context information. The extremely short size of the embedding vector is another fantastic benefit of the Word2Vec technique [12], [13], [34]. The embedding vector provides data on one feature of the word for each dimension. We have used the Continuous Bag-of-Words (CBOW) architecture of Word2Vec techniques. CBOW model [14] is able to predict the probability of the central word based on *n-1* words around the input and it used the surrounding words to predict the current target word. The network structure has three layers, input, output, and projection layer. The input layer used one-hot encoding. This model tries to predict the target word by using the context of the surrounding words.

Graph Based Topic Modelling. Textual documents are represented as graphs of words. It is an alternate weighting technique for graph theory-based topic models [15], [31]. It is an alternative representation of a document that captures the relationships between the terms using the graph of terms nodes corresponding to the terms *t* of the document edges capture co-occurrence relations between terms within a fixed-size sliding window of size *w*. There are many different techniques used to generate graphs from documents or words and graph partitioning techniques [7] are used to find the latent topic of the graph. Graph partitioning techniques are also being used for short text topic modeling [16, 17]. Tripathy et al. [18] uses Wikipedia taxonomy graph to group all of the relevant tweets into a single cluster.

Spectral clustering [19, 20] is advantageous for non-convex clusters as well as when clusters are nested circles on the 2D plane. For clear understanding of spectral clustering, we represent a flowchart as Figure 2, which describe the workflow of spectral clustering. As the latent topics are non-convex, so it is suitable to apply spectral clustering. Instead of using KMeans clustering inside spectral clustering, A Density-Based Spatial Clustering (DBSCAN) technique has better applications with noisy data [21]. In case of DBSCAN we need to specify the number of clusters rather it can automatically identify the number of clusters [21]. Graph-based topic modelling offers the potential for progress despite the development of several methodologies. In essence, we may use graph partitioning approaches to enhance the efficiency of the graph-based topic modelling. We have developed a heuristic for graph splitting because we are aware that the approach is NP-Hard in nature.

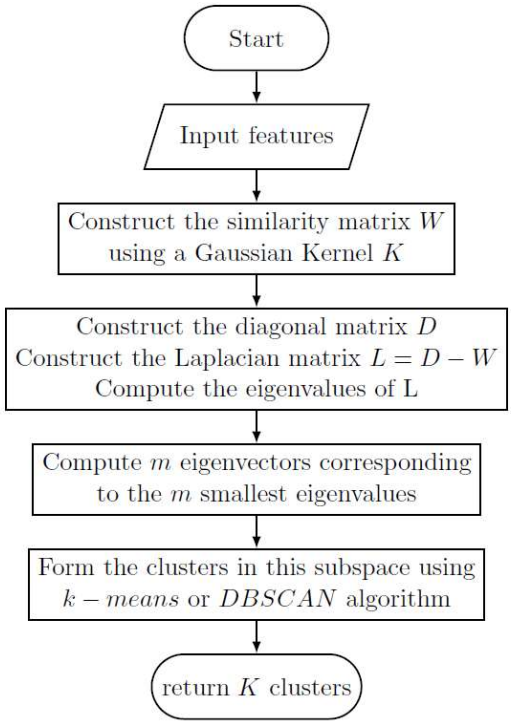


Figure 2 Flowchart of Spectral Clustering

2. RELATED WORK

Manuscripts Topic modeling is a method to extract hidden topics from enormous amounts of text and gained popularity in the field of computer science, particularly with regard to text mining, text summarization, information retrieval etc. Topic modeling has been received a lot of attention among researchers and gained widespread interest among them in many research fields since it was initially proposed. Deerwester et al. [22] has developed latent semantic analysis (LSA) for topic modeling, which has served as the basis for the development in this area. Hofmann et al. 2001 [5] proposed a probabilistic latent semantic analysis (PLSA) for topic modeling. This technique adds a probabilistic approach to topics and words on top of LSA. As compared to LSA, PLSA is a better and more flexible model, but it has some limitations. In the case of PLSA we have no parameters to model for computing the probability of documents, and also the technique has not given any idea to assign probabilities to new documents, and another major drawback of PLSA is that it leads to overfitting because the number of parameters grows linearly with the number of documents. Blei et al. 2003 [23] have proposed Latent Dirichlet Allocation (LDA), which is an extension of probabilistic latent semantic analysis (PLSA). LDA is a powerful technique for discovering and exploiting latent topics in large document collections. The literature suggests that the basic LDA model can be easily modified for a more complicated application [23]. Therefore, this technique has been extended and applied in many ways since its development. However, it performs poorly for shorter text, sparse text, and high-dimensional text data, such as user reviews, Twitter, and Reddit. Yin et al. 2014 [6] proposed Gibbs Sampling Dirichlet Multinomial Mixture (GSDMM), which can be applied to the sparse and high-dimensional problem of short texts, and can be used to obtain the representative words of each cluster. Mazarura et al [24] show that the GSDMM model tends to outperform the LDA model on short and sparse text when using coherence scores as an evaluation metric. Their findings demonstrate that the GSDMM model outperforms the LDA model for short and sparse text. Weisser et. al [25] in 2022 also show that GSDMM model is better than LDA for short text. GSDMM model is based on the assumption that a single topic samples each text. However, the model claims to solve the sparsity problem of short text clustering while also displaying word topics like LDA, but it has not captured the semantics of the words in the process of model generation. In this paper, we propose a heuristic that overcomes the limitation of GSDMM. Our heuristics captures the semantics of the text documents by building a graph-based topic model. Topic modeling usually involves clustering natural language embedding's, which combines words with similar semantics together, in order to discover the semantic structure of the underlying corpus [26 - 29]. According to their research, the weighted and unweighted embedding clustering approach that used Word2Vec could beat conventional methods. Sahlgren et. al [27] have com- pared document-based topic modeling with word-based topic modeling. The document-based model employed document embedding's, [30] while the word-based topic models used word embedding's for each important word. According to the study, word-based topic modeling produced subjects with less or no over- lap, more distinct topics, and greater average topic coherence. Additionally, Wang et al. [28] recently applied embedding clustering to evaluate the efficacy of several topic modeling algorithms on Twitter data. The study shows that less sophisticated models, which may not always outperform methods for distributed embedding's. Recently many researchers used graph-based topic modeling, they used clustering techniques to create semantically related word groups. Their basic

objective is to motivate research in using graph connectivity for topic modeling, whereas the common clustering techniques require strict hyper parameter tuning. Altuncu et, al. [34] used graph connectivity and document embedding's to extract latent topics. The vertex of the graph represents documents, and the weighted edges are the cosine similarity of the document pair. They used minimum spanning tree (MST) to extract the group of documents that represented the topics of the corpus. The research found that graph connectedness surpasses many conventional clustering methods. Graph-based clustering techniques have been effectively applied in a variety of applications, including the investigation of crime patterns [31] and the finding of cohesive sub graphs for social networks [30, 32].

Based on the above studies we have proposed an effective heuristic for graph-based topic modeling which captures the semantic meanings of the text and at the same time, it reduces the computational complexity. The graph has been generated, and then it has been broken down into various components, each of which can be thought of as a topic. Although graph partitioning techniques are desirable, they are NP-Hard. Therefore, we have proposed a heuristic that will solve the problem in polynomial time, and at the same time, it has a better coherence score and also be clearly interpreted by the users.

In our heuristic, we proposed a novel graph partitioning technique to generate a subgraph. Then we applied clustering techniques to the generated sub-graph and found that its performance is better than some of the popular topic modeling techniques. To validate our model, we compared our heuristic with LDA as well as GSDMM, based on certain metrics, and we found that our model performance is better than the above techniques. This paper is organized as follows. In section 3 the methodology is presented followed by the result and discussion in section 4. Finally, the conclusion is given in section-5.

3. METHODOLOGY

3.1 Proposed Model. In this section, we describe our proposed heuristic for graph-based topic modeling approach. The main goal of our strategy is to identify the most popular topics stated in the corpus of user reviews. By using the fundamental data cleaning and pre-processing techniques, we were able to extract the distinct words from the corpus, which are represented as $W = \{w_1, w_2, ..., w_n\}$, where each w_i represents a word in the corpus. Each unique word is being considered as a feature vector and is being generated by Word2Vec model. The goal of using the Word2Vec deep learning approach to create a feature vector is to capture words in a text while also capturing their semantics. The Word2Vec model is being trained by using four threads as parameters. We eliminated the stop words and also removed the words which are not in Wordnet, after converting each word to vectors.

eliminate and also remove the words which are not available in Wordnet. This resulted in significant reduction from 2126905 words to 48834 words. A graph $G = (V, E)$, where V is the set of vertices which is being created by considering all unique 48834 words. There will be an edge $e(v_i, v_j) \in E$ if the cosine similarity between the words, which is greater than or equal to a threshold value (T). The mean distance between the pair of vertices in the graph G is used to determine this threshold value. The key idea behind considering cosine similarity measure for assigning weight to the edge is a low cosine value suggests that the neighbouring words are not semantically linked, whereas a larger cosine score implies greater semantic closeness. While working on high-dimensional data, dimensionality reduction plays an important role in topic modeling. Therefore, for reducing the dimension we remove the words having a low degree of centrality because these words are not contributing much to topic modeling and also reduce the computational complexity. After getting the subgraph by removing these words, we applied spectral clustering because this clustering technique has been known to perform well in data that follows non-convex distributions. The proposed heuristic is explained in detail in the flowchart given in Figure 3 and also in the Algorithm 1

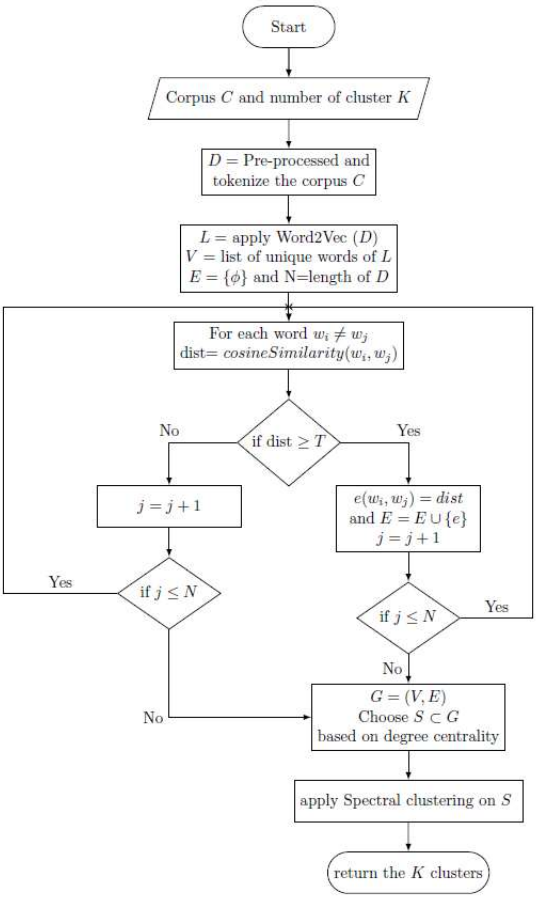


Figure 3 Flowchart of Effective Graph Based Topic Modeling ***(EGBTM)***

Once the word has been converted to a vector, we eliminate and also remove the words which are not available in Wordnet. This resulted in significant reduction from 2126905 words to 48834 words. A graph $G = (V, E)$, where V is the set of vertices which is being created by considering all unique 48834 words. There will be an edge $e(v_i, v_j) \in E$ if the cosine similarity between the words, which is greater than or equal to a threshold value (T). The mean distance between the pair of vertices in the graph G is used to determine this threshold value. The key idea behind considering cosine similarity measure for assigning weight to the edge is a low cosine value suggests that the neighbouring words are not semantically linked, whereas a larger cosine score implies greater semantic closeness. While working on high-dimensional data, dimensionality reduction plays an important role in topic modeling. Therefore, for reducing the dimension we remove the words having a low degree of centrality because these words are not contributing much to topic modeling and also reduce the computational complexity. After getting the subgraph by removing these words, we applied spectral clustering because this clustering technique has been known to perform well in data that follows non-convex distributions. The proposed heuristic is explained in detail in the flowchart given in Figure 3 and also in the Algorithm 1

3.2 Data Set. We used the benchmark dataset from Github for the suggested model, which contains information gathered from Tripadvisor hotel ratings. 20K reviews were crawled from Tripadvisor for the dataset. Travellers can make trip preparations with the help of Tripadvisor, the world's largest travel planning and booking

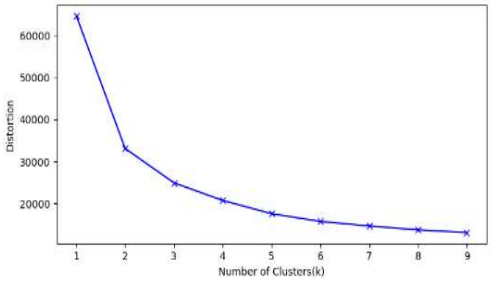


Figure 5 Estimate Number Of Clusters Based On Elbow Method

platform. It can assist travellers to determine if a hotel is good or poor, we must first determine the most significant topics and keywords that have been started been stated in the discussion Fundamentally, the dataset comprises two attributes: review text and review rating. For topic modeling, we have just taken review text into consideration as in [33]. The frequency distribution of the top 20 most frequently occurring words is shown in Figure 4(A). The majority of these words, as you can see, have to do with describing hotels. Such as room, great, staff etc. In order to provide a clear picture of the hotel reviews dataset, we have also included a word cloud of the words used in the reviews in Figure 4(B) The frequency distribution of the top 20 most frequently occurring words is shown in Figure 4(A). The majority of these words, as you can see, have to do with describing hotels. Such as room, great, staff etc. In order to provide a clear picture of the hotel reviews dataset, we have also included a word cloud of the words used in the reviews in Figure 4(B).

4. RESULTS AND DISCUSSION

We have used Python programming language and several associated libraries for our experimentation and implemented it, using core i7

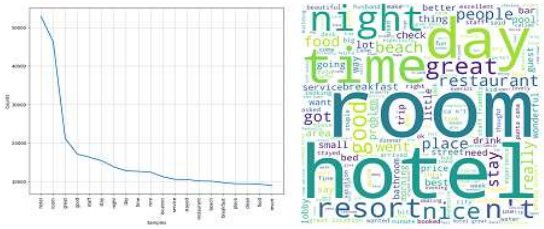
Algorithm 1. Effective Graph Based Topic Modeling **(EGBTM)**

Input : The corpus *C* and number of topics *K*
Output : *K* clusters of topics where each topic has related words
1 : *D*=Apply the basic text cleaning and preprocessing steps on the corpus *C*
2 : vect = Apply Word2Vec (*D*)
3 : List = List of unique words of *D* and the words that are not available in Wordnet
4 : For each word i of the List do:
5 : For each word j of the List do:
6 : dist=ApplyCosineSimilarity(vect[wordi],vect[wordj])
7 : If (wordi ≠ wordj and dist ≥ Threshold (*T*)) then
8 : AdjMat[i, j] = dist
9 : Else:
10 : AdjMat[i, j] = 0
11 : End if
12 : End for
13 : End for
14 : Set graph *G* = AdjMat
15 : Set *S* =*G*
16 : For each vertex v_i of the graph *G*, which as low degree centrality do
17 : Remove the edges E that incident on the graph *S*
18 : Remove the vertex v_i from the graph *S*
19 : Remove the word w_i from the list
20 : Topics = Call function clusterData (*S*, *K*)
21 Return Topics

Algorithm 2. Procedure for clusterData

Input : The graph *G* and the number of topics *K*
Output : *K* clusters of topic and each topic has related words
1 : *K* clusters of topic and each topic has related words
Let $G \in R^{n\times n}$ $defined\ by\ G_{ij} = e^{\frac{-\|s_i - s_j\|}{2\sigma^2}}$ $if\ (i \neq j)\ and\ G_{ij} = 0\ otherwise$
2 : Construct the matrix $L = D^{-1/2}GD^{-1/2}$
3 : Construct $W = L - D$
4 : Find $x_1, x_2, \ldots x_k$ the eigenvectors of the top k eigenvalues of W
5 : Form the matrix $X = \{x_1, x_2, \ldots x_k\}, R^{n\times k}$ by stacking the eigenvectors in columns.
6 : topics=Apply miniBatch K-Means Clustering(*X*)
7 : Return topics

computer with 16 GB of RAM. We have chosen the number of clusters *K* as 3. Since we are dealing with benchmark data, we have the domain knowledge of the dataset. There are three major



(A) Frequency Distribution (B) Wordcloud

Figure 4 Frequently used words

types of topics in the dataset. Therefore, we have chosen *K* as 3. We have also performed an Elbow test on the number of clusters to further strengthen its authenticity. The result is shown in Figure 5. As you can see in Figure 5 the distortion decreases rapidly at first then slowly flatten forming an "elbow", near the value of $k = 3$. The distortion is computed as the average of the squared distance between each data point and its closest centroid. Following the application of our heuristic, the word clouds for three themes are shown in Figure 6. From the word cloud in Figure 6 we can clearly segregate the words into three different topics such as excellent, terrible, and normal reviews. The chart demonstrates how clearly the clusters have been made. The result of the word cloud encourages us to carry out a user study on the results in order to evaluate the quality of the cluster. We have used graduate students and faculty of engineering institute for the user study. A Google form has been created for the survey. There are twelve-word pairs in the Google form. Each pair has two words, and they are organized so that the first six pairs of words belong to the same cluster while the words from the following six pairs belong to different clusters. In that form, we ask the user whether or not the two terms in each pair are connected or related. The user has the option to answer" Yes" if they believe the two words in the pair aren't sure. The results are stored in an Excel file. From the file, we have computed the four parameters shown in Table-I, we have created the contingency table shown in Table II. Please note that the percentage is computed taking into account the total number of questions answered" Yes," or "No,".

Table 1. Parameter description

Figure 6 Wordcloud of Topics

Parameters	Descriptions
True Positive (TP)	Words in a pair are related and from the same cluster
True Negative (TN)	Words in a pair are unrelated and from different clusters
False Positive (FP)	Words in a pair are related and from different clusters
False Negative (FN)	Words in a pair are unrelated and from the same cluster

Table 2. Contingency table based on user reviews

	Words from same cluster	Words from different clusters
Words are related	46% (TP)	7% (FP)

Figure 7 Coherence score of different models

We have used the following formula to determine F-score.

$$F - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Were,

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

Solve the above formula, we have obtained Precision as 0.868, Recall as 0.92, and F-score as 0.893.

Table 3. Topic coherence score

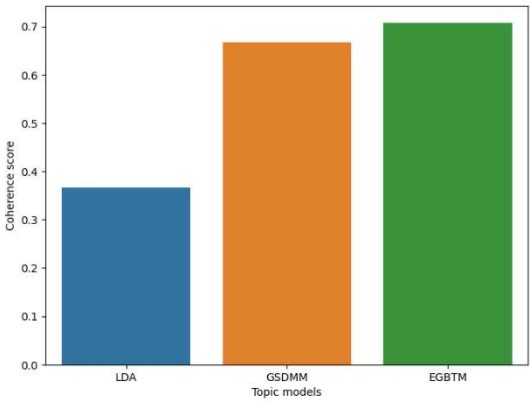
	Top 20 words	Top 30 words
Coherence Score	0.712	0.723

We have compared the coherence score of two baseline models LDA (Linear Dirchlet Allocation), and GSDMM (Gibbs Sampling Dirichlet Mixture Model) with our proposed heuristic EGBTM (Effective Graph Based Topic Modeling). In all the models we keep the number of topics k as 3

Table 4. Model coherence score

Topic Models	C_v(Coherence Score)
LDA(k=3)	0.366
GSDMM(k=3)	0.668
EGBTM(k=3)	**0.708**

The result in Table 4 shows that our proposed heuristics, EGBTM is out- performed the other two models. The bar chart in Figure 7 shows that the proposed model has high coherence score as compare to the bassline model.



5. CONCLUSION

The topic model produced by our proposed heuristic, EGBTM, is superior as compared to the two significant baseline models, LDA, and GSDMM. The proposed algorithm was applied to the benchmark dataset of Tripadvisor hotel reviews, and it yielded a somewhat improved cluster coherence score as well as a satisfying outcome based on the user study. In our research, based on the user survey, the clusters are being clearly identified by the users. In spite of this, there is space for improvement in the topic's interpretability, according to third-category ratings that are considered to be neutral. According to the study, more neutral user reviews may have been included in the dataset, which may have improved the clarity of the neutral reviews. Due to the NP-Hard nature of the graph partitioning technique, there is room for future development of efficient heuristics, that are both computationally faster and have a better coherence score

REFERENCES:

[1] L. Xia, D. Luo, C. Zhang, Z. Wu, *A survey of topic models in text classification,* 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD) (2019) 244–250.

[2] D. M. Blei, A. Y. Ng, M. I. Jordan, *Latent dirichlet allocation*, Journal of machine Learning research 3, Jan. 2003, pp 993–1022.

[3] C. Meaney, M. Escobar, R. Moineddin, T. Stukel, S. Kalia, B. Aliarzadeh, T. Chen, B. O'Neill, M. Greiver, Non-negative matrix factorization temporal topic models and clinical text data identify covid-19 pandemic effects on primary healthcare and community health in toronto, Canada, Journal of Biomedical Informatics 128 (2022)

[4] T. Hofmann. (1999) Probabilistic latent semantic analysis, In Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence, pp.289_296.Morgan Kaufmann Publishers Inc..

[5] T. Hofmann, Unsupervised learning by probabilistic latent semantic analysis, Machine learning 42 (1-2) (2001) pp.177.

[6] J. Yin, J. Wang, A dirichlet multinomial mixture model-based approach for short text clustering, in: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 233–242.

[7] L. Chen, J. Jose, H. Yu, F. Yuan, D. Zhang, A semantic graph-based topic model for question retrieval in community question answering, in: Ninth ACM International Conference on Web Search and Data Mining, 2016, pp. 287–296.

[8] D. Blei, L. Carin, D. Dunson, Probabilistic topic models, IEEE Signal Processing Magazine 27 (6) (2010) pp.55–65.

[9] G. A. Martin Gerlach, Tiago P. Peixoto, A network approach to topic models, Science Advances 4 (7) (2018) pp.1360.

[10] F. Wang, J. L. Zhang, Y. Li, K. Deng, J. S. Liu, Bayesian text classification and summarization via a class-specified topic model, The Journal of Machine Learning Research 22 (2021) pp.3971–4018.

[11] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Advances in neural information processing systems 26 (2013) pp.3111–3119.

[12] S. Park, C. Liu, A study on topic models using LDA and word2vec in travel route recommendation: Focus on convergence travel and tours reviews, Personal Ubiquitous Computing 26 (2) (2020) pp.429–445.

[13] Z. Yuan, L. Congrui, L. Hao, W. Junjie, Topic modeling of short texts: A pseudo-document view with word embedding enhancement, IEEE Trans- actions on Knowledge and Data Engineering 35 (1) (2023) pp.972–985.

[14] Deerwester, Scott and Dumais, Susan T. and Furnas, George W. and Landauer, Thomas K. and Harshman, Richard, Using topic modeling and word embedding for topic extraction in twitter, Procedia Computer Science 207 (2022) pp.790– 799.

[15] B. R. Chandra, R. Sawan, G. Atul, A new graph-based extractive text summarization using keywords or topic modeling, Journal of Ambient Intelligence and Humanized Computing 12 (10) (2021) pp.8975–8990.

[16] Y. Wang, J. Liu, J. Qu, Y. Huang, J. Chen, X. Feng, Hashtag graph based topic model for tweet mining, in: 2014 IEEE International Conference on Data Mining, 2014, pp. 1025–1030.

[17] R. Albalawi, T. H. Yeap, M. Benyoucef, Using topic modeling methods for short-text data: A comparative analysis, Frontiers in Artificial Intelligence 3 (2020) 42.

[18] R. M. Tripathy, S. S. Sharma, S. Joshi, S. Mehta, A. Bagchi, Theme based clustering of tweets, in: CODS, 2014.

[19] Q. Wu, A. Hare, S. Wang, Y. Tu, Z. Liu, C. G. Brinton, Y. Li, Bats: A spectral biclustering approach to single document topic modeling and segmentation, ACM Transactions on Intelligent Systems and Technology (TIST) 12 (2021) pp.1–29.

[20] P. Favati, O. Menchi, A two-phase strategy for nonconvex clusters

integrating a spectral clustering with a merging technique, Expert Systems with Applications 214 (2023).

[21] G. Tolegen, A. Toleu, R. Mussabayev, A. Krassovitskiy, *A clustering-based approach for topic modeling via word network analysis, in*: 2022 7th International Conference on Computer Science and Engineering (UBMK), IEEE, 2022, pp. 192–197.

[22] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, R.Harshman, *Indexing by latent semantic analysis*, Journal of the American society for information science 41 (6) (1990) pp.391–407.

[23] H. Jelodar, Y. Wang, C. Yuan, X. Feng, X. Jiang, Y. Li, L. Zhao, *Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey*, Multimedia Tools and Applications 78 (2019) pp.15169–15211.

[24] J. Mazarura, A. de Waal, A comparison of the performance of latent dirichlet allocation and the dirichlet multinomial mixture model on short text, 2016, pp. 1–6.

[25] C. Weisser, C. Gerloff, A. Thielmann, A. Python, A.Reuter, T. Kneib, *Pseudo-document simulation for comparing lda, gsdmm and gpm topic models on short and sparse text using twitter data, Computational Statistics,* 38 (2023),pp.647-674.

[26] S. A. Curiskis, B. Drake, T. R. Osborn, P. J. Kennedy, *An evaluation of document clustering and topic modelling in two online social networks*: Twitter and reddit, Information Processing & Management 57 (2020).

[27] M. Sahlgren, Rethinking topic modelling: *From document-space to term-space, in: Findings of the Association for Computational Linguistics*: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 2250–2259.

[28] L. Wang, C. Gao, J. Wei, W. Ma, R. Liu, S. Vosoughi, *An empirical survey of unsupervised text representation methods on twitter data, in*: Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020), Association for Computational Linguistics, Online, 2020, pp. 209–214.

[29] Z. Peng, W. Suge, L. Deyu, L. Xiaoli, X. Zhikang, *Combine topic modeling with semantic embedding: Embedding enhanced topic model*, IEEE Transactions on Knowledge and Data Engineering 32 (12) (2020) pp.2322–2335.

[30] A. Meddeb, L. B. Romdhane, *Using topic modeling and word embedding for topic extraction in twitter*, Procedia Computer Science 207 (2022) pp.790–799.

[31] P. Das, A. K. Das, J. Nayak, D. Pelusi, W. Ding, *A graph based clustering approach for relation extraction from crime data*, IEEE Access 7 (2019) pp.101269–101282.

[32] W. Yanping, Z. Jun, S. Renjie, C. Chen, W. Xiaoyang, *Efficient personalized influential community search in large networks*, Vol. 6, 2021, pp. 310–322.

[33] V. Nguyen, T. Ho, *Analyzing customer experience in hotel services using topic modeling*, Journal of Information Processing Systems 17 (2021) pp.586–598.

[34] A. M. Tarik, S. N. Yaliraki, M. Barahona, Graph-based topic extraction from vector embeddings of text documents: Application to a corpus of news articles, in: Complex Networks & Their Applications IX: Volume 2, Proceedings of the Ninth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2020, Springer International Publishing, 2021, pp. 154–166.