Semantic Similarity Measure for Topic Modeling Using Latent Dirichlet Allocation and Collapsed Gibbs Sampling

Micheal Olalekan Ajinaja (ajinajalekan@gmail.com)
Federal University of Technology Akure, Ondo
Olusola Adebayo Adetunmbi
Federal University of Technology Akure, Ondo
Chukwuemeka Christian Ugwu
Federal University of Technology Akure, Ondo
Popoola Olugbemiga Solomon
Osun State College of Education

Research Article

Keywords: semantic similarity, topic modelling, LDA, collapsed gibbs sampling

Posted Date: August 22nd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1968318/v1

SEMANTIC SIMILARITY MEASURE FOR TOPIC MODELING USING LATENT DIRICHLET ALLOCATION AND COLLAPSED GIBBS SAMPLING

Ajinaja Micheal Olalekan[1], Adetunmbi Adebayo Olusola[2], Ugwu Chukwuemeka Christian[3], Popoola Olugbemiga Solomon[4]

[1]Department of Computer Science, Federal University of Technology, Akure, Nigeria

[2,3]Department of Computer Science, Federal University of Technology, Akure, Nigeria

[4]Department of Computer Science, Osun State College of Education, Ila-Orangan, Nigeria

ajinajalekan@gmail.com, aoadetunmbi@futa.edu.ng, uchristian407@gmail.com, popsol7@yahoo.com

Abstract

One of the key applications of Natural Language Processing (NLP) is to automatically extract topics from large volumes of text. Latent Dirichlet Allocation (LDA) technique is commonly used to extract topics based on word frequency from the pre-processed documents. A major issue of LDA is that the quality of topics extracted are poor if the document do not coherently discuss a single topic. However, Gibbs sampling uses word by word basis which changes the topic assignment of one word and can be used on documents having different topics. Hence, this paper proposed a hybrid based semantic similarity measure for topic modelling using LDA and Gibbs sampling to exploit the strength of automatic text extraction and improve coherence score. Unstructured dataset was obtained from a public repository to validate the performance of the proposed model. The evaluation carried out shows that the proposed LDA-Gibbs had a coherence score of 0. 52650 as against LDA coherence score 0.46504. The proposed multi-level model provides better quality of topics extracted.

Keywords- *semantic similarity, topic modelling, LDA, collapsed gibbs sampling*

1.0 INTRODUCTION

Topic modelling (TM) is a text processing technique, which is aimed at overcoming information overload by seeking out and demonstrating patterns in textual data, identified as topics [1]. It allows data analysts to quickly navigate through a corpus of text or collection using identified topics to improve user experience. Unsupervised learning is often used to perform TM, and the results of running the models are an overview of the topics that were found. TM involves the task of identifying underlying concepts which are discussed within a collection of documents and determining which topics each document is addressing [2]. TM has numerous applications in NLP; for example, topic models may be used for information retrieval (IR) [3], to identify influential individuals on a social media platform [4], or to detect signs of depression [5]. The algorithms conventionally used to tackle the problem of TM include LDA [6] and probabilistic latent semantic analysis (pLSA) [7]. Large companies like Jumia, Amazon, Netflix can use TM to identify the topics of a set of customer reviews by detecting patterns and recurring words and by identifying frequently used words and expressions such as awesome, free to use, fee, charging or 2.5% plus 99 cents transaction fee. Hence, TM can group this review with other reviews that talk about similar things [8]. Each of the topic generated can be measured to determine quality of the topics using semantic similarities.

Semantic similarity finds out the degree of semantic equivalence between two items, which can be concepts, sentences, or documents [9]. The semantic similarity measure (SSM) is the ability to determine the similarity between various terms such as words, sentences, documents, concepts or instances. SSM has great importance in many computer applications related field such as information retrieval, educational system, text summarization and NLP [10]. Estimating the semantic similarity on large text data is one of the challenging and open research problems in the field of NLP. The versatility of natural language makes it difficult to define rule-based methods for determining SSM [12]. Measuring the semantic similarity between various text components like words, sentences, or documents plays a significant role in a wide range of NLP tasks like information retrieval [13], text summarization [14], essay evaluation [9], text classification [15] and among others. During the early days, two text snippets were considered the same if they contain the same words/characters. Techniques like Bag of Words (BoW), Term Frequency – Inverse Document Frequency (TF-IDF) were used to represent text, as real value vectors to aid calculation of semantic similarity. However, these techniques failed to attribute to the fact that words have different meanings and different words can be used to represent a similar concept. The methods captured the lexical feature of the text and were easy to implement. However, [11] overlooked the semantic and syntactic properties of text. For example, considering two sentences “Tunde and Ayo studied English and Agric.” and “Tunde studied English and Ayo studied Agric.”. Though these two sentences have exactly the same words they do not carry the similar meaning. Also, similarly the sentences “Mary is allergic to dairy products.” and “Mary is

lactose intolerant." carry the same meaning but different words. A better way of understanding how semantic can be measured using coherence score is discussed.

Therefore, in this paper, we focused on developing a multi-level based SSM for topic modelling using a combination of LDA and Gibbs sampling and show comparison using the coherence score as evaluation.

The remaining part of this paper is organized as follows: Section 2 presents the review of related works, section 3 provides the methodology, data source, text pre-processing, model architecture used in the study. Results and discussion are provided in Section 4 and Section 5 presents the conclusion.

2.0 REVIEW OF RELATED WORKS

In [18], the paper focused on exploring LDA, topic modelling, its' application and a detailed survey. The paper investigated highly scholarly articles (between 2003 to 2016) related to topic modelling based on LDA to discover the research development, current trends and intellectual structure of topic modelling. In addition, it summarized the various challenges and also introduced tools such as user behaviour modelling, topics visualization based on LDA. In [26] a comprehensive review of TM methods was discussed extensively which included classification hierarchy, posterior inference techniques, different evolution models of LDA and its applications in different areas of technology including Scientific Literature, Bioinformatics, Software Engineering and analysing social network was presented.

In [16], empirical prior Dirichlet allocation (epLDA) model was used for latent semantic indexing framework to derive the priors required for topics computation from data. The parameters of the priors obtained were related to the parameters of the conventional LDA model using exponential function. The model was implemented and tested with benchmarked

data and it achieved a prediction accuracy of 92.15%. It was observed that the epLDA model consistently outperforms the conventional LDA model on different datasets with an average percentage accuracy of 6.33% which clearly demonstrated the advantage of using side information obtained from data for the computation of the mixture. [14] worked on a text summarization approach based on semantic role labelling and explicit semantic analysis using two well-established text semantic representation techniques; Semantic Role Labelling (SRL) and Explicit Semantic Analysis (ESA). Experimental results indicated that the proposed summarizer ROUGE-SU4 outperforms all state-of-the-art related comparators in the single document summarization based on the ROUGE-1 and ROUGE-2 measures, while also ranking second in the ROUGE-1 (0.499) and ROUGE-SU4 (0.286) scores for the multi-document summarization. The paper in [19] focused on extracting topics from software engineering data. The research then illustrated how to employ LDA on a textual data set to create

different topics. It showed topic-topic correlation matrix had 95% confidence Intervals of the correlation amount.

[2] worked on evaluation of TM techniques for twitter dataset. For each experiment, the optimal models were chosen for a setting of K=100 topics, as it is equal to the number of search queries that was used when the data was collected. The result of the research indicated that biterm topic model (BTM) was superior to all other models when working with short documents.

SSM receives considerable attention in recent years due to its numerous potential applications in NLP. In SSM, there are several techniques of measurement. The paper in [20] presented a review of semantic similarity. The result displayed and classified various semantic similarity methods and metrics with their advantages and limitations using different documents.

[22] was to enable the use of topic modelling for researchers by presenting a step-by-step framework which consisted of three steps; pre-processing, topic modelling, and post-processing, where the topic model LDA is used. The full run of 650 papers considered for 20 topics took 3.5 h to compute and the outcome of the method is a 650 by 20 matrix of topic probabilities. Also, topic 16 had the highest probability distribution for one document.

The research in [23] examined current work series of metrics from literature on a quantitative basis by performing benchmarks against a generated dataset with a known value of k and evaluate the ability of each metric to recover the true value, varying over multiple levels of topic resolution in the Dirichlet prior distributions. The new metric proposed in the paper suffered much less overfitting at low values of k, and maintained good performance throughout the tested range of k. All the three metrics displayed initial signs of underfit as K exceeds 80. Also, the new metric displayed a clear kink at the true number of topics. The conclusion was that the new metric is able to both identify the correct value of k here, as well as provide evidence for potential overfit.

[24] considered the computational complexity of probabilistic inference in LDA by studying the problem of finding the maximum a posteriori (MAP) assignment of topics to words. [25] considered using LDA for improving the topic modelling of the official bulletin of the Spanish state (BOE). The results of the analysis showed that more than 89% of the documents cannot be recommended because they are not well described at the documentary level, some of their key meta-data are empty.

[27] worked on LDA and t-Distributed Stochastic Neighbour embedding to enhance scientific reading comprehension of articles related to enterprise architecture. The result showed that documents ‘sustainability’ had the highest distribution and assignment of the examined documents to the identified topics. The work [28] provided a simple and efficient learning procedure that was guaranteed to recover the parameters for a wide class of topic models, including LDA. [29] worked on online inference of topics with LDA. The paper [30] explored a variety of methods for applying the LDA automated topic modelling algorithm to the modelling of the structure and behaviour of virtual organizations found within modern social media and social networking environments.

In [31], three algorithms for parameter estimation of the LDA model were reviewed. They were experimentally compared to determine their time complexity and performance and found out that the online variational Bayesian inference converges faster than the other two inference techniques, with comparable quality of the results. The inference in [32] for the number of topics was calculated in the LDA Model via Bayesian Mixture Modelling. In [33], the research provided a multiple-corpora LDA (mLDA) model that assumed the document topic proportions follow a symmetric Dirichlet distribution. The result showed mLDA allowed the power of TM to be applied to a huge range of fields with diverse data by incorporating more information into a single topic model. It also enhances the applicability of TM to information retrieval. [34] used LDA to measure trend topic analysis.

3.0 METHODOLOGY

Under this section, four cardinal approaches were used in our topic modelling and discussed.

3.1 Data Source

11,000 newsgroups posts were sourced from an online repository [34] consisting 20 different topics ranging from ICT topics in sport, entertainment, politics, health section of different newspaper in Nigeria. Among the news post considered were punch Nigeria, the Sun newspaper, metro news among others.

3.2 Text Preprocessing

Text preprocessing is used to clean and normalize the text data to improve the accuracy and reduce the data redundancy as well as training time of the models [33]. Each sentence was tokenized into a list of words using regular expression patterns, removing punctuations and unnecessary characters altogether. The next stage was forming the bigram and trigrams. Bigrams are two words frequently occurring together in the document. Trigrams are 3 words frequently occurring. This was done to find out words that frequently occur together and predict the conditional probability of the next word. The LDA model was built with 20 different topics where each topic is a combination of keywords and each keyword contributes a certain weightage to the topic. The step-wise process for initial text cleansing is stated in Table 1 while the output of the pre-processed text is as shown in Fig 1.

Table 1: Text pre-processing process carried out on newsgroup dataset

Step	Description
Punctuation Removal	Remove punctuation
Word tokenization	Tokenize sentences to sets of words
Lowercase Conversion	Convert words to lowercase
Stopwords Removal	Remove stopword
Lemmatization	Lemmatize the words

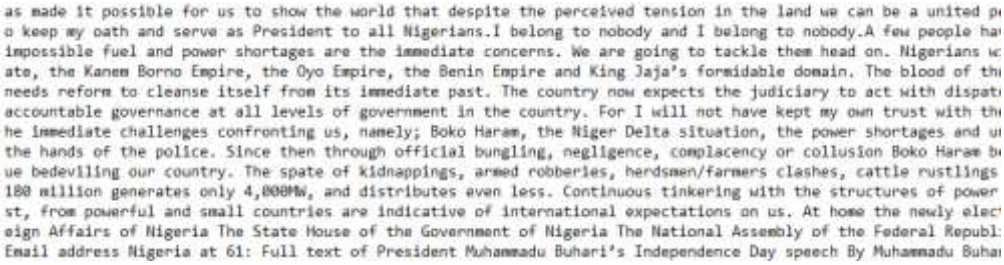

Figure 1: Snippet of the pre-processed text

3.3 Feature Extraction Concepts

Feature extraction is a key factor for the performance of the topic modelling. The text document is represented by set, $d_i = \{W_1, W_2, W_3 \ldots W_n\}$, where w_i represents a word in document d. The words are subjected to feature extraction process to map each word to a topic and each topic to a document. The feature extraction can be generally classified into two classes: documents-topics and word-topic. We provided a brief description of these techniques in equation 1 for topic-document term frequency-inverse document frequency (TF-IDF) for feature extractions.

$$\text{IDF(t)} = \log \frac{1+ n_d}{1+df(t,d)} + 1 \quad (1)$$

Where

n_d is the total number of documents

df(t, d) is the number of documents that contain the word t.

3.4 Methods I - Latent Dirichlet Allocation

Given a number of documents d_i that comprises of different words W_i belonging to several topics k_i. LDA assumes that documents are composed of words that help determine the topics and maps documents to a list of topics by assigning each word in the document to different topics and is expressed as follows:

$$d_i = \{W_1, W_2, W_3 \ldots W_n\} \quad (2)$$

LDA then finds the probability of a word w_j belonging to topic t_k where 'j' and 'k' are the word and topic indices respectively. Once the probabilities are estimated, finding the collection of words that represent a given topic is done by either by picking top 'r' probabilities of words or by setting a threshold for probability and picking only the words whose probabilities are greater than or equal to the threshold value. Assuming there were 3 topics and 3 words, LDA finds the probability by the expression below:

$$d_i = (w_{1i} * \text{Topic} - 1) + (w_{2i} * \text{Topic} - 2) + (w_{3i} * \text{Topic} - 3) \quad (3)$$

In the above representation, there are three weights for topics: topic-1, topic-2 and topic-3 respectively for a given document d_i. $(w_{1i} * \text{Topic} - 1)$ indicates the proportion of words in document that represent topic-1, $(w_{2i} * \text{Topic} - 2)$ indicates the proportion of words in document that represent topic-2 and so on.

LDA assumes that each document is generated by a statistical generative process. That is, each document is a mix of topics, and each topic is a mix of words. Documents are mixed of topics and each topic, in turn, is a mix of different collections of words. In the process of generating this document, first, a topic is selected from the document-topic distribution and later, from the selected topic, a word is selected from the multinomial topic-word distributions. LDA algorithm starts with assuming K number of topics, it then loops through each number of documents and randomly assign each word in the document to one of the K topics.

For each document loop through, it loops through each word w and compute the following below:

- P(Topic t | Document d) = the proportion of words in document d that are currently assign to topic t and expressed as $p(t_k \mid d_i)$, that is, proportion of words in document d_i that are assigned to topic t_k
- P(Word w | Topic t) = the proportion of assignments to topic t over all documents that come from this word w and expressed as $p(w_j \mid t_k)$, that is, proportion of all documents assigned to a topic t_k given word w_j assigned assignments to topic t over all documents that come from this word w

The proportion of words w_j in document d_i that are assigned to topic t_k tries to capture how many words belong to the topic t for a given document d and expressed as follows:

$$p(w_j \mid t_j, d) \tag{4}$$

The final step of LDA is to reassign or update the $p(w_j \mid t_j, d)$ to a new topic where Topic T with probability P(Topic T | Document D) * P(Word W | Topic T) is chosen which is essentially that Topic T generated word w and is expressed as follows:

$$p(w_j \mid t_k, d_i) = p(w_j \mid t_k) * p(t_k \mid d_i) \tag{5}$$

LDA topic modelling is a sampling-based algorithm that endeavours to collect samples from the posterior to approximate it with an empirical distribution. The model technique was used to generate topics based on word frequency from the pre-processed documents. The LDA topic model accepts the created dictionary and the corpus as inputs and works as follows:

i. choosing the number of topics (k) that are in the corpus
ii. randomly assign each word in each document to one of the ‘k’ number of topics
iii. go through every word and its topic assignment in each document.
iv. looks at the first topic and how often the topic occurs in the document, then how often the words occur in ‘k + 1’ where k = 20.
v. based on step (iii), assign the word to a new topic

The model can be written in equation 5 as:

$$p(W, Z, \theta) = \sum_{n \epsilon N} p(Wd_n|Zd_n)p(Zd_n|\theta_d) \tag{6}$$

where W is specific word, Z represents specific topic, θ_d is topic distribution for document *d,* $p(Wd_n|Zd_n)$ is probability of words in topics and $p(Zd_n|\theta_d)$ is probability of distribution of topics in documents. LDA's approach to topic modelling is that it considers each document as a collection of topics in a certain proportion and each topic as a collection of words. LDA topic model technique was used to generate topics based on word frequency from the pre-processed documents. The number of topics was provided into the algorithm and all it does it to rearrange the topics distribution within the documents and keywords distribution within the topics to obtain a good composition of topic keywords distribution. For obtaining good segregation of topics, the following key factors were considered in the research;

i. The quality of text processing.
ii. The variety of topics the text talks about.
iii. The choice of topic modeling algorithm.
iv. The number of topics fed to the algorithm.
v. The algorithms tuning parameters.

3.5 Methods II: Gibbs Sampling

Gibbs sampling uses word by word basis which changes the topic assignment of one word. It works on the assumption that topic assignment is not known of the given word but the assignment of all other words in the text is known which will then be used to infer what topic will be assigned to this word. Considering 'm' as the corpus of a document and 'k' as the number of topics, the model randomly assign k topics to all the words in 'm' number of documents and be represented as follows:

$$m_i = \{w_1, w_2, w_3, w_4 \ldots w_n\} \tag{7}$$

Equation 6 above tells us the number of words in the corpus of a document from word-1 to word-n. Assuming there are 3 topics to be considered:

$$k = \{t_1, t_2, t_3\} \tag{8}$$

The model then randomly assigns k topics to all the words in m_i and can be represented as:

$$\begin{Bmatrix} w1 = t3 \\ w2 = t2 \\ w3 = t1 \\ w4 = t3 \\ . \\ . \\ . \\ . \\ . \\ wn = ti \end{Bmatrix}$$

The model then counts the total number of words in the i^{th} document belonging to the k^{th} topic. For example, $n_{(1,2)}$ means the total number of words in 1^{st} document belonging to 2^{nd} topic and can be represented as:

$$n_{(i\,,\,k)} = \text{total number of words in } i^{th} \text{ document in } k^{th} \text{ topic} \quad (9)$$

After creating a document wise topic count, it the model creates a topic wise assignment of word count from all documents as said earlier, that is, it counts each word belongs to a particular topic for all the documents. To explore the entire space, a small number alpha α is added to $n_{(i\,,\,k)}$ and known as Dirichlet parameter for document to topic distribution represented as:

$$n_{(i\,,\,k)} + \alpha \quad (10)$$

The model continues by decrementing the count for the respective topic allocated from the document-topic matrix by subtracting 1 from the number N_i of in the i^{th} document which is added to the product of number of topics k and the hyper parameter α introduced earlier. This can be represented as:

$$(N_i - 1) + k\alpha \quad (11)$$

To indicate how much document d_i likes topic t_k or finding the probability of how much document d_i likes topic t_k can be represented as

$$p(t_k \mid d_i) = \frac{n(i,k) + \alpha}{(Ni-1)+k\alpha} \quad (12)$$

To care for word to topic assignment, the model introduces β to explore the entire space of the corpus. The corpus wide assignment of word w_j to k^{th} topic is added to the hyper parameter as done earlier for document to topic assignment and is represented as:

$$m_{(j,\,k)} + \beta \quad (13)$$

For example $m_{(3,\,4)}$ represent the 3^{rd} word that belongs to the 4th topic which is then added to the hyper parameter. Recall a decrement had occurred earlier by decrementing the count for the respective topic allocated from the document-topic matrix. The model looks through all the vocabulary 'V' of the corpus from the 1^{st} word to the last word in the vocabulary and sums it with product of the vocabulary and β represented as:

$$\sum_{j \epsilon V} m(j,k) + V\beta \quad (14)$$

To indicate how much topic t_k likes word w_j or finding the probability of how much word w_j likes topic t_k is represented as:

$$p(w_j \mid t_k) = \frac{m(j,k)+V\beta}{\sum_{j\epsilon V} m(j,k)+V\beta} \quad (15)$$

To calculate for the word w_i, the product of probability of how much document d_i likes topic t_k and probability of how much word w_j likes topic t_k is represented as:

$$p(w_j \mid t_k, d_i) = p(t_k \mid d_i) * p(w_j \mid t_k) \quad (16)$$

In expansive form, it is represented as:

$$p(w_j \mid t_k, d_i) = \left(\frac{n(i,k) + \alpha}{(Ni-1)+k\alpha}\right)\left(\frac{m(j,k)+V\beta}{\sum_{j\epsilon V} m(j,k)+V\beta}\right) \quad (17)$$

For a given word w_i in a document d_i, the model finds the topic 'k' for which $p(w_j \mid t_k, d_i)$ is maximum and reassigns the word to the 'k^{th} topic. The process is repeated over and over again as against the number of iterations. Below is the flowchart of the LDA/Gibbs which was used in the research;

3.6 Method 3: Proposed Hybrid Model

The proposed hybrid model uses both LDA and Gibbs to produce output. Gibbs sampling uses word by word basis which changes the topic assignment of one word. It works on the assumption that topic assignment is not known of the given word but the assignment of all other words in the text is known which will then be used to infer what topic will be assigned to this word. The conditional probability equation for a single word w in document d that belongs to topic k is given below:

$$P(W, Z, \theta, \Phi\ \alpha, \beta\ ) = \prod_{j=1}^{M} P(\theta_j; \alpha) \prod_{i=1}^{K} P(\Phi_i; \beta) \prod_{t=1}^{N} P\left(Z_j, i \mid \Theta_j\right) P(W_j, t \mid \Phi Z_j, t) \quad (18)$$

where W is specific word, Z represent specific topic, θ is topic distribution for document, Φ is word distribution for topic k, α is Dirichlet parameter for document to topic distribution, β is Dirichlet parameter for topic to word distribution, M represents number of documents, N is number of words in a given document and K is number of topics. There are two parts to equation (2). First part tells how much each topic is present in a document and the second part tells how much each topic is present in a word. For the multi-level system, the output for LDA was considered first, then the output for LDA/Gibbs algorithm was considered and then compared. The coherence score for LDA, LDA/Gibbs (hybrid - multi-level concept) was obtained and compared. This modelling procedure is as shown in Figure 2.

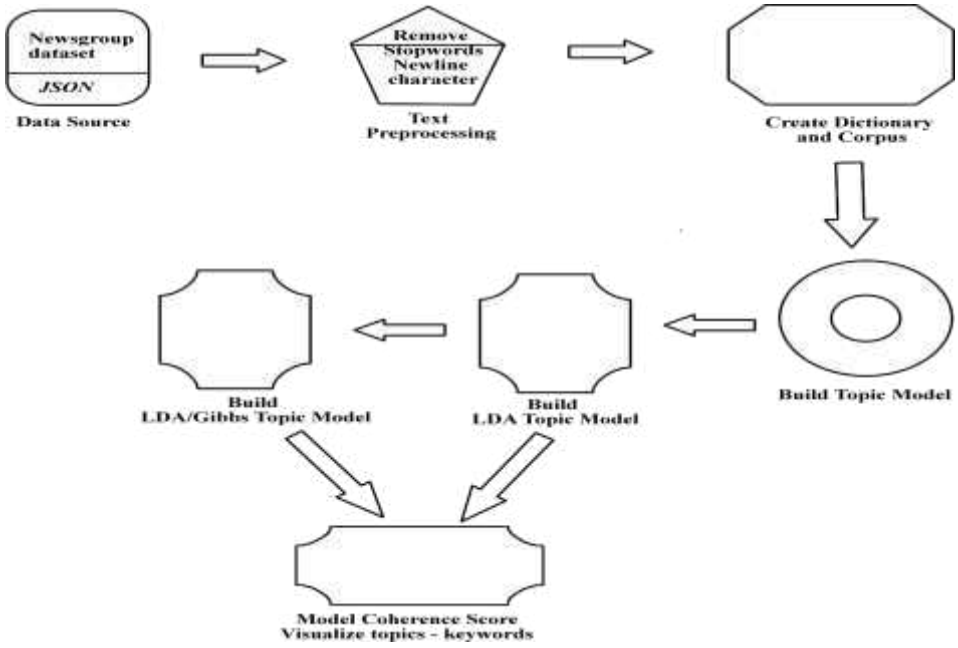

Figure 2: Architecture of the proposed hybrid model

3.7 Coherence Score

Topic coherence provide a convenient measure to judge how good a given topic model is and used in evaluation in topic models. It is computed taking cognisance of the top n words by frequency. For one topic, the words *i, j* being scored in equation 18:

$$\text{Coherence} = \sum_{i<j} score(w_i, w_j) \text{ where}$$

w_i and w_j are the top words of the topic

$$score(w_i, w_j) = \log\left(\frac{p(w_i, w_j)}{p(w_i)\, p(w_j)}\right) \tag{19}$$

Where

p(w) = the probability of seeing w_i in a random document

$p(w_i, w_j)$ = the probability of seeing both w_i and w_j co-occurring in random document

4.0 RESULTS AND DISCUSSION

The experiment setup for topics modelling was implemented on Anaconda Jupyter with GPU capability using the following packages: Numpy, Pandas, pyLDAvis and the NLTK in python 3.6.

4.1 LDA Topics

The LDA model is built with different topics where each topic is a combination of keywords and each keyword contributes a certain weightage to the topic. One can see the keywords for each topic and the weightage (importance) of each keyword as shown in table 2. The output of the different topics where

each topic is a combination of keywords and each keyword contributes a certain weightage to the topic as shown in Table 2:

Table 2 – output of topic 0 to 4 showing the combination of keywords and how each keyword contributes to the weightage of the topic

Topic 1				
0.023*"people	0.022*"say	0.014*"may"	0.013*"reason	'0.013*"believe
0.011*"evidence	0.010*"make	0.010*"think	'0.010*"would	0.010*"many"

Topic 2				
'0.147*"price	0.118*"player	0.107*"sale	0.079*"tape	0.065*"sell
0.041*"offer	0.018*"brother	0.017*"purchase	0.017*"super	'0.015*"duo"

Topic 0				
0.058*"screen	0.058*"memory	0.056*"disk"	0.051*"display	'0.048*"instal
0.036*"cpu	0.034*"error	0.033*"problem	'0.031*"port"	0.031*"board

Topic 3				
'0.039*"space	0.028*"patient	0.024*"power	0.022*"drug"	'0.016*"ground
0.016*"launch	0.014*"food	0.014*"low	0.013*"cool	0.013*"treatment

Topic 4				
'0.036*"law	0.030*"child	0.030*"government	0.029*"gun"	'0.027*"people
0.027*"kill	0.026*"state	0.023*"death	'0.021*"right	0.020*"die

Topic 0 is represented as (0, '0.036*"law" + 0.030*"child" + 0.030*"government" + 0.029*"gun" + "0.027*"people" + 0.027*"kill" + 0.026*"state" + 0.023*"death" + "0.021*"right" + 0.020*"die"'). It means the top 10 keywords that contribute to this topic are: 'law', 'child', 'government.. and so on and the weight of 'law' on topic 0 is 0.036 law. The weights reflect how important a keyword is to that topic. Looking at these keywords, one can you guess what this topic could be related to "terror" or "negativity".

4.2 Visualize the topics-keywords

After building the LDA model, the produced topics and the associated keywords are examined by Intertopic distance map which shows the proportion to the number of words that belong to each topic. As shown in figure 6, it can be seen that topic 4 has the most intersection with all other topics meaning its contained more relevant keywords and meaning. Also, topic 1 had the most probability of been talked about if words are picked at random from the datset, followed by 2, 3 and 4. All other topics were quite

distance from the man topics of 1, 2, 3 and 4. This can also be interpreted in word prediction in the result by the scale of the circle. The bigger the circle, the higher the topic prediction in accordance with the size of the circle.

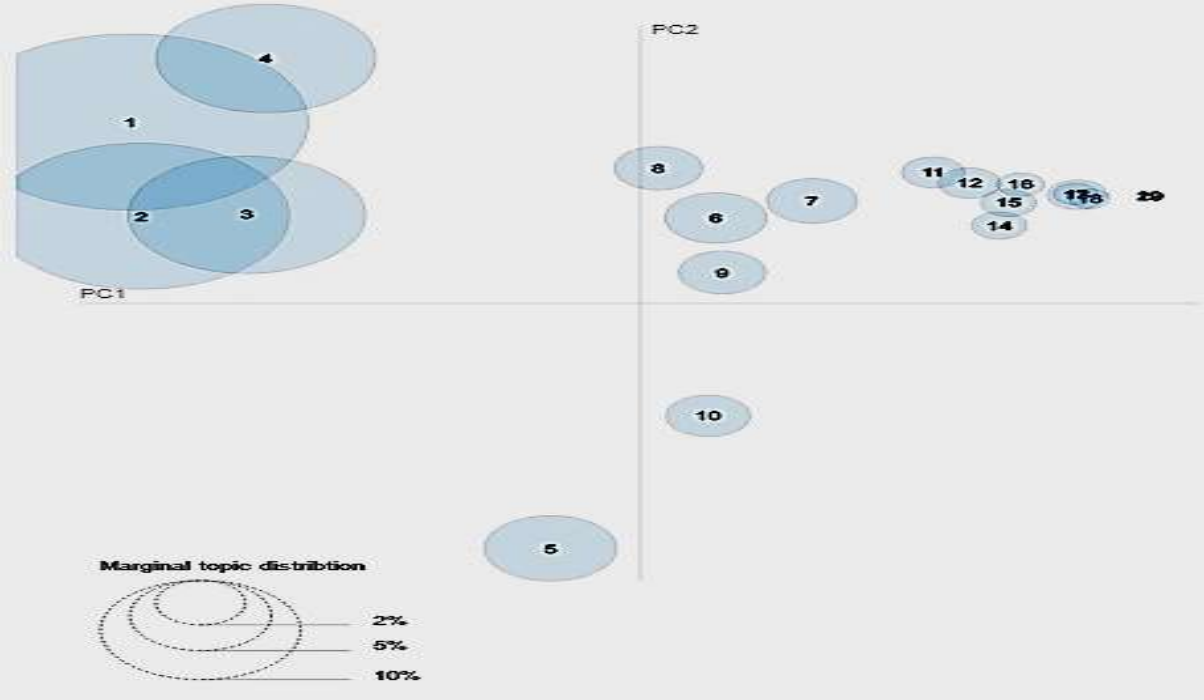

Figure 3 – Topic visualization of topics generated by LDA

4.3 Frequency Distribution of Word Counts in Documents

Working with a large number of documents requires to know how big the documents are as a whole and by topic. Plotting the document word counts distribution.

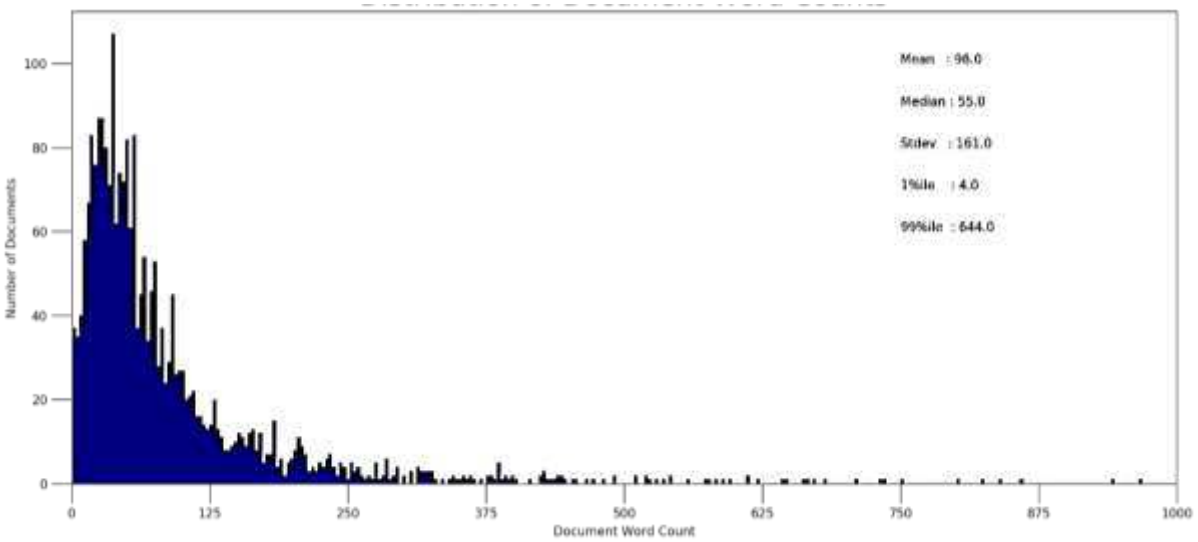

Figure 4 – Distribution of documents Word Counts

4.4 Distribution of Document Word Counts by Dominant Topic

In Fig 5, the document words count distribution is analysed by topic that is dominant in that dataset. As mentioned above, different topics were identified, and below, in fig. 8, the distribution for the topics are plotted for Collapsed Gibbs Sampling since it has higher coherence score. It can be observed that Topic 2 and 1 has a higher concentration of words in them that are under consideration. The distribution of each of the topics is similar.

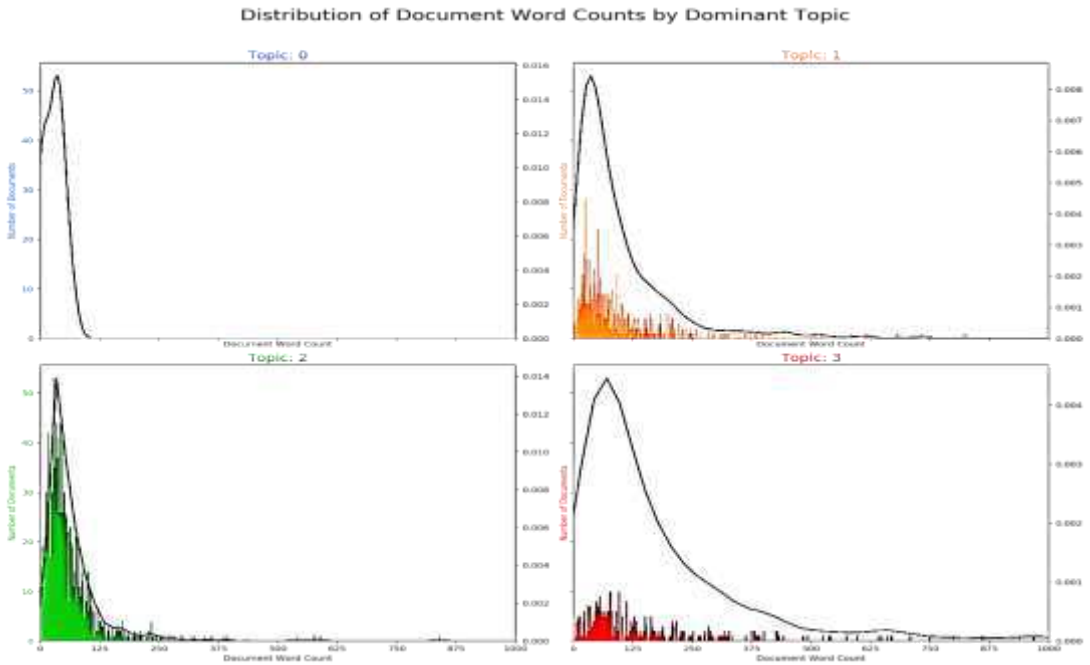

Figure 5 - Distribution of Document Word Counts by Dominant Topic

4.5 Word Count and Importance of Topic Keyword

The significance of the keywords in the topics as indicated by the weights is plotted in figure 8. Also, the frequency with which the words have occurred in the pre-processsed datasets is plotted. As can be seen, some words are shared among the topic. The common words to stop words can be added to make sure they are not considered, but the reduction of several topics is a better solution as there is overlapping. Topics keywords is plotted as shown in fig. 6.

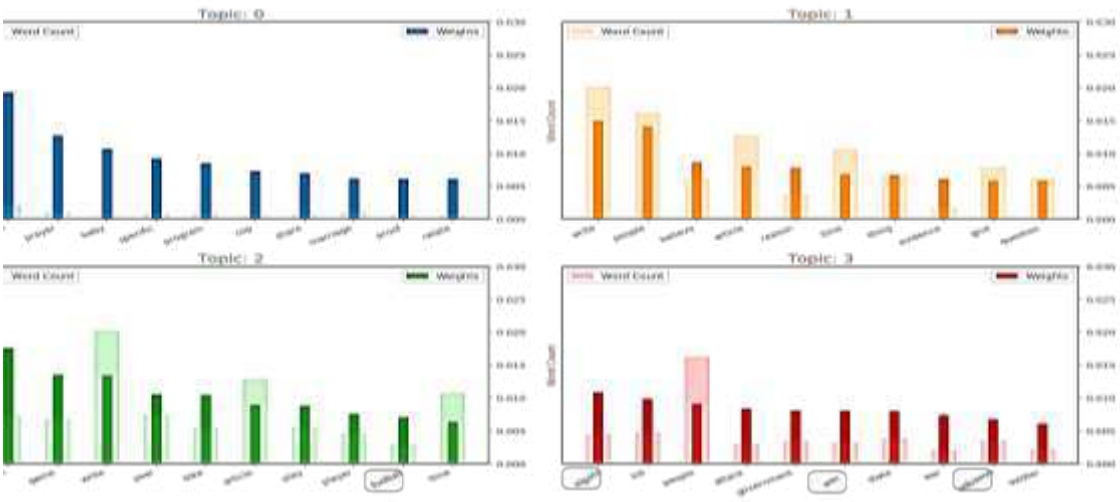

Figure 6 - Word Count and Importance of Topic Keyword

4.6 LDA Model Coherence Score

Topic coherence provide a convenient measure to judge how good a given topic model is. LDA output for coherency is shown in table 3:

Table 3 – output to compute coherence score that provide a convenient measure to judge how good the topic model is

LDA	
Coherence Score	*0.4650389083419528*

4.7 LDA/Gibbs Model Topics

Gibbs sampling uses word by word basis which changes the topic assignment of one word. It works on the assumption that topic assignment is not known of the given word but the assignment of all other words in the text is known which will then be used to infer what topic will be assigned to this word. The aim is to know show that the proposed multi-level model gives better quality of extracted topics. After generating the topics for LDA-Gibbs sampling, the coherence can be calculated with the output given in table 4:

Table 4 – output of the proposed Multi-level model coherence score

Topic 1				
'people', 0.0231	'kill', 0.0116	'nigeria', 0.009	'attack', 0.009	'war', 0.00959
'government', 0.0092	'arm', 0.009	'unknown', 0.008347	'land', 0.00798	'child', 0.00792
Topic 2				
'drug', 0.01279	'study', 0.01138	'food', 0.009110	'doctor', 0.00906	'effect', 0.00825
'problem', 0.007498	'patient', 0.00718	'eat', 0.0067655	'disease', 0.00635	'show', 0.006228

4.8 Proposed Model Coherence Score

Topic coherence provide a convenient measure to judge how good a given topic model is. The table 5 shows the proposed hydrid model coherency score.

Table 5 – output of the proposed Multi-level model coherence score

Proposed Model	
Coherence Score	0.5265013017532258

4.9 Proposed Model Comparison with LDA

The evaluation carried out shows that the proposed model had a coherence score of 0. 5265013 as against LDA which had a coherence score of 0. *465038* indicating the proposed model had extracted better quality of topics. The proposed Multi-level model provided a better and convenient measure to judge how good a given topic than LDA.

Table 6 – Proposed Model comparison with LDA

Topic Model	**Coherence Score**
LDA	0. *465038*
LDA/Gibbs	0.5265013

5.0 CONCLUSION

This study provided with understanding of topic modelling and how basic topic model is built. One major issue of LDA is that the quality of topics extracted is poor if the document does not coherently discuss a single topic. However, Gibbs sampling with LDA can be used on documents having different topics. Hence, the proposed multi-level based SSM for topic modelling using LDA and Gibbs sampling has been exploited to underpin the strength of generating a better coherence score. This was demonstrated by feeding the output of LDA into Gibbs sampling to improve coherence score (which is used to determine the quality of extracted topics). For future work, other conditional distribution techniques like variational inference can be used with LDA as earlier mentioned in [34] to compare with Gibbs sampling. Also worthy of note is visualization of the output of Gibbs sampling and variational inference.

Author contributions The research's conception and design were influenced by the work of all contributors. Data gathering, material preparation, and analysis were completed by Micehal Olalekan Ajinaja. The first draft of the manuscript was written by Adetunmbi Adebayo Olusola, Ugwu Chukwuemeka Christian, Popoola Olugbemiga Solomon. All the authors read and approved the final manuscript and also agreed to all the content of the article including the author list and contributions.

Funding The research did not enjoy any external funding.

Declarations

Conflict of interest as far as the content of this article is concerned, the authors have no conflicts of interest to declare.

REFERENCES

[1] Lazarina (2021, July 1) Topic Modelling: A Deep Dive into LDA, hybrid-LDA, and non-LDA Approaches.https://lazarinastoy.com/topic-modelling-lda/

[2] Jonsson E. and Stolee J. An Evaluation of Topic Modelling Techniques for Twitter. An evaluation of topic modelling techniques for Twitter. (n.d.). Retrieved August 7, 2022, from https://www.cs.toronto.edu/jstolee/projects/topic.pdf

[3] Xing Yi and James Allan. A comparative study of utilizing topic models for information retrieval. In Advances in Information Retrieval, pages 29–41. Springer, 2009.

[4] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. Twitterrank: finding topic-sensitive influential twitterers. In Proceedings of the third ACM international conference on Web search and data mining, pages 261–270. ACM, 2010

[5] Philip Resnik, William Armstrong, Leonardo Claudino, Thang Nguyen, Viet-An Nguyen, and Jordan Boyd-Graber. Beyond lda: exploring supervised topic modeling for depression-related language in twitter. NAACL HLT 2015, page 99, 2015.

[6] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. the Journal of machine Learning research, 3:993–1022, 2003.

[7] Thomas Hofmann. Probabilistic latent semantic indexing. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pages 50–57. ACM, 1999.

[8] Topic modeling: An introduction. MonkeyLearn Blog. (2019, September 26). Retrieved August 7, 2022, from https://monkeylearn.com/blog/introduction-to-topic-modeling/

[9] Sunilkumar P. and Athira P.S. A survey on semantic similarity - researchgate.net. (n.d.). Retrieved August 7, 2022, from https://www.researchgate.net/profile/Athira-Shaji-2/publication/339975566_A_Survey_on_Semantic_Similarity/links/603e15224585154e8c6e5b64/A-Survey-on-Semantic-Similarity.pdf?origin=publication_detail 2019

[10] A. Ali, F. Alfayez & H. Alquhayz. Semantic similarity measures between words: A brief survey, Journal of Science International, vol. 30(6), pp. 907-914, 2018

[11] Gaurav V and Balaji V.S. A Lexical, Syntactic, and Semantic Perspective for Understanding Style in Text, BigData Experience Lab, Adobe Research, 2012

[12] D. Chandrasekaran and V. Mago. Evolution of Semantic Similarity - A Survey. Journal of Association of Computing machinery, vol. 37(4), 2020

[13] Y. Jiang, X. Zhang, Y. Tang and R. Nie. Feature-based approaches to semantic similarity assessment of concepts using Wikipedia. Information Processing & Management, vol. 51(3), pp. 215–234, 2015

[14] M. Mohamed and M. Oussalah. 2019. SRL-ESA-TextSum: A text summarization approach based on semantic role labeling and explicit semantic analysis. Information Processing & Management, vol. 56(4), pp. 1356–1372, 2019

[15] Thabet S. Description and Evaluation of Semantic similarity Measures Approaches. ArXiv.org e-print archive. (n.d.). Retrieved August 7, 2022, from https://arxiv.org/ftp/arxiv/papers/1310/1310.8059.pdf

[16] Adegoke M.A, Ayeni J.O and Adewole P.A. Empirical Prior Latent Dirichlet Allocation Model. Nigerian Journal of Technology (NIJOTECH). Vol. 38(1), 2019, pp. 223 – 232

[17] M. Röder, A. Both nd A. Hinneburg. Exploring the Space of Topic Coherence Measures, Association for Computing Machinery, 2015

[18] Hamed J., Yongli W., Chi Yuan, Xia F., Xiahui Jiang, Yanchao Li and Liang Zhao. Latent Dirichlet Allocation (LDA) and Topic modeling: models, applications, a survey. ArXiv.org e-print archive. (n.d.). Retrieved August 7, 2022, from https://arxiv.org/ftp/arxiv/papers/1310/1310.8059.pdf

[19] Joshua C.C., Abram H. and Eleni S. Latent Dirichlet Allocation: Extracting Topics from Software Engineering Data. January, 2016

[20] Akila D. and Jayakumar C. Semantic Similarity- A Review of Approaches and Metrics. International Journal of Applied Engineering Research. Vol 9 (24), 2014 pp. 27581-27600

[21] Vivek Kumar Rangarajan Sridhar. Unsupervised topic modeling for short texts using distributed representations of words. In Proceedings of NAACL-HLT, pages 192–200, 2015.

[22] Asmussen, C.B., Møller, C. Smart literature review: a practical topic modelling approach to exploratory literature review. J Big Data 6, 93 (2019). https://doi.org/10.1186/s40537-019-0255-7

[23] Jason Hou-Liu. Benchmarking and Improving Recovery of Number of Topics in Latent Dirichlet Allocation Models. Retrieved January 4, 2018 from https://vixra.org/pdf/1801.0045v1.pdf

[24] David S. and Daniel M.R. Complexity of Inference in Latent Dirichlet Allocation. Retrieved from https://people.csail.mit.edu/dsontag/papers/SontagRoy_nips11.pdf

[25] J.C. Bailón-Elvira J.C., Cobo, M.J., Herrera-Viedma, A.G. and López-Herrera. Latent Dirichlet Allocation (LDA) for improving the topic modeling of the official bulletin of the Spanish state (BOE). Procedia Computer Science, Vol 162, 2019, pp 207-214

[26] Pooja Kherwa and Poonam Bansal. Topic Modelling: A Comprehensive Review. Journal of EAI Endorsed Transactions on Scalable Information Systems, 2019.

[27] Horn, N., Gampfer, F., Buchkremer, R. Latent Dirichlet Allocation and t-Distributed Stochastic Neighbor Embedding Enhance Scientific Reading Comprehension of Articles Related to Enterprise Architecture. Institute of IT Management and Digitization Research (IFID), April 2021

[28] Anima A. et al. A Spectral Algorithm for Latent Dirichlet Allocation. Retrieved from https://www.cs.columbia.edu/~djhsu/papers/lda-nips.pdf

[29] Lei Shi, Thomas L. Griffiths and Kevin R.C. Online Inference of Topics with Latent Dirichlet Allocation. 2th International Conference on Artificial Intelligence and Statistics (AISTATS) 2009 Vol. 5.

[30] Alexander Gross and Dhiraj Murthy. Modeling virtual organizations with Latent Dirichlet Allocation: A case for natural language processing, Journal of Neural Networks, Vol. 58, 2014, pp 38-49,

[31] Jaka Špeh, Andrej Muhiˇc and Jan Rupnik. Parameter Estimation for the Latent Dirichlet Allocation. Retrieved from https://ailab.ijs.si/dunja/SiKDD2013/Papers/Speh-ldaAlgorithms.pdf

[32] Zhe Chen and Hani Dossaca. Inference for the Number of Topics in the Latent Dirichlet Allocation Model via Bayesian Mixture Modelling. Retrieved from https://users.stat.ufl.edu/~doss/Research/lda-ntopics.pdf

[33] Adam Foster, Hangjian Li, Georg Maierhofer and Megan Shearer. An extension of standard latent dirichlet allocation to multiple corpora. Retrieved on April 2016 from http://evoq-eval.siam.org/Portals/0/Publications/SIURO/Vol9/AN_EXTENSION_STANDARD_LATENT_DIRICHLET_ALLOCATION.pdf?ver=2018-04-06-152049-177

[34] Aulia R.D, Isnandar Slamet and Sri Subanti. Trend Topic Analysis using Latent Dirichlet Allocation (LDA) (Study Case: Denpasar People's Complaints Online Website). Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI) Vol. 5(1), June 2019, pp. 50~58

[33] Hew Z.J., Olanrewaju V.J., Chew X.Y. and Khaw K.W. Text Summarization for News Articles by Machine Learning Techniques. Journal of Applied Mathematics and Computational Intelligence. July, 2022.

[34] Newsgroup Master dataset. Retrieved from https://raw.githubusercontent.com/selva86/datasets/master/newsgroups.json

[35] Guifen Zhao, Yanjun Liu, Wei Zhang and Yiou Wang. TFIDF based Feature Words Extraction and Topic Modeling for Short Text. International Conference on Management Engineering, Software Engineering and Services (ICMSS), January, 2018