IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A NOVEL MODEL TO CHARACTERIZE SEARCH QUERIES FOR SEARCH ENGINE

[1]Janardhan BV, [2]Dr. Chandrashekar B.H.

[1]Student, [2] Associate Professor
[1]Master of Computer Applications,
[1]RV College of Engineering, Bengaluru, India

Abstract: Search engines act as a vital role in information retrieval and knowledge discovery. Due to the huge amount of data on web and social sites it is a challenging task to find suitable information as requested by user efficiently even by using search engines. Also, the above frameworks specified will gather large amount of information, which could be helpful for end users in different manners yet would have neglected the original meaning, henceforth showing irrelevant results. In Order to reduce those problems, topic modelling plays a vital role in filtering data to improve efficiency in information retrieval. Topic modelling is an unsupervised machine-learning model which is one of the important tools for information retrieval as it plays a vital role in clustering the documents. In this modelling technique each document represents a set of / collection of bags of words and labelling appropriate word distributions to the defined topic is a challenging task. Traditional topic model such as Latent Dirichlet Allocation, which is a model that predicts topic space in a document using Dirichlet Distribution. The major drawbacks of LDA are that the topics are fixed and must be known before the time of running the model, Dirichlet topic distribution cannot capture correlations, etc. To overcome these problems, the model used Correlated Topic Modelling (CTM) to find the correlation between words and phrases, to speed up the processing and to increase the scalability. Correlated Topic Model (CTM) helps in finding the correlation between the hidden topics in the collection, and enables the construction of graphs that are related to topics collected and document that allow a user to go through the collection of topics in a standard manner. In this paper experimental results show that how CTM plays a important role than LDA in extraction of accurate information, and to enhance the optimization of the search query on a smaller scale and demonstrate its use as an major tool for large document collections.

Index Terms - Topic Modelling, Clustering, Correlation, Latent Dirichlet Distribution

I. INTRODUCTION

Search engines like Google, Bing help in retrieval of information. Due to rapid increase of the data and information on the web and media it is quite a bit of a challenging task to find relevant information efficiently as it may or may not show relevant results [3]. Data retrieval using web indexes that suit various different requirements for different set of users. So, to satisfy this current framework utilize a few informational portrayal strategies in PC frameworks including database models, particularly online database which empowers productive data querying [1]. Since most of the web applications are supporting relational schema concept in retrieval of information, they do not provide any feedback to users. Nowadays there is a need for huge amounts of data and users are looking for efficient access to the data they require. Since applications contain a large number of documents, handling and retrieving information from those documents must be efficient. Almost all search engines use text-based searching techniques in which the query string is matched with the text in files or a database [1]. Then the results are based on the number of times the string has occurred, more over it does not consider the real meaning i.e. the logical meaning of the query. Automatic extraction of topics from a large set of documents is a primary application of knowledge extraction. It is really hard to go through a large set of documents, provide relevant information to the users and to provide an efficient search query. Automation of this process of information retrieval is very much essential so that the algorithm reads through the text documents and automatically generates topics by considering the logical meaning of the topics [9]. This will enhance the results of search queries in search engines. Topic modelling is an unaided AI strategy which is equipped for examining a lot of records, recognizing words and expression designs inside them, and consequently bunching word gatherings and comparable articulations that best portray a lot of reports. Vector analysis utilized in the calculations encourages the web indexes to store or recover information dependent on points and their connections [2]. An investigation of the words related around a specific point assists with revealing the fundamental or significant subject of that report. Latent Dirichlet Allocation is a Bayesian rendition of PLSA [11]. Specifically, it utilizes Dirichlet priors for the record subject and word-theme circulations, fitting better speculation. The significant downsides of LDA are Fixed K, Uncorrelated points (Dirichlet theme dissemination cannot catch relationships), Non-progressive, Static, Bag of words and Unsupervised [2]. Correlated Topic Modeling for content or other discrete information that models connection between the event of various themes in an archive. Correlated Topic Model (CTM), which explicitly considers the correlation between the latent topics in the collection, and enables the user to construct topic graphs and document browsers that allow a user to navigate through the collection of topics in a guided manner.

II. Related Work: Correlated Topic Modelling

Correlated topic modelling helps in documenting set of hidden topics; each topic is extracted from the bag of words which in turn represents the sorted words (by removing the stop words, adjectives and finding out similar synonyms from the dictionary or vocabulary V). The below figure 1 illustrates the block diagram of correlated topic model along with the notations which represents each and every variable used in this document. Consider a given set of documents as A which represents the number of documents taken as input to the model for performing the topic modelling and extracting latent topic from those considered documents [5].

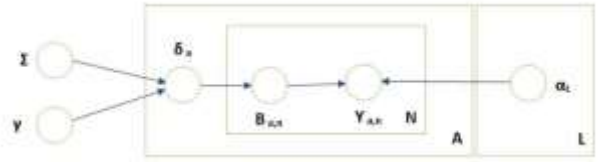

Fig 1: Block diagram of CTM

For the document A, number of latent topics L, (L x L) covariance matrix to implement correlation topic modelling. Taking an example of LDA model to find the probability between the topics generated from the bag of words [4]. Here, in CTM model provision is given to find the correlation between the hidden topics L with an additional feature of finding out the synonyms from the dictionary considered. Below, table 1 describes the different annotations used in this paper which is related to the document [13].

Table 1: Annotations for CTM

θ_d	Each document is associated with a set of topic proportions θ_d (Document specific topic proportion)
$Y_{d,b}$	Denote the n^{th} word in the document A^{th} which is the in a V-term vocabulary
α_L	Model which contains L topics - word distribution per topic
$B_{d,n}$	The topic assignment is associated with the n^{th} word and A^{th} document
γ	L dimensional positive vector
Σ	L * L covariance matrix – Logistic Normal Parameters
δ_d	Natural parameterization of this multinomial $\delta = \log(\theta_i / \theta_L)$ - topic distribution per document
A	Number of Documents
L	Number of latent topics
N_A	Total words in document A
V	Total words in vocabulary

As described above, for each document A, extract latent topics L from all the document A [7]. Later, tokenize each and every word from the vocabulary and remove the words which is un-necessary by using stop words [6]. Then denote the words $Y_{a,n}$ using a predefined vocabulary and generate a Covariance matrix Σ. Words with associated count is generated, then find the probability P between the topics and also find the correlation between them.

III. Proposed Methodology

In this section, the proposed methodology is been described to extract topics from the documents collected from the users on which LDA and CTM has to be performed.

Collection of Data

First and the foremost thing, collect huge set of documents from the users where it is difficult to read between the documents and extract information. The data collected can be of any format (e.g. .pdf, .xlsx) and can be of any huge data set. The data can be collected either by manually or by using the concept of web crawling. By using the crawling technique, data can be extracted from any social media platform and can consist of any sort of information later on for which the documents A has to be analyzed.

- Load the documents A on which topic modelling has to be performed.

Pre-Processing of Data

- Before passing data to the trained model, perform some data preprocessing where cleaning of the data has to be taken care of. Raw data collected from either manually or by web-crawling will have to reduce noise. Firstly, remove unwanted words from the document and this is done by tokenizing words from the document i.e. splitting each and every word in the document and adding it to the list or bag. By doing this it is helpful to segregate each and every word.
- After tokenizing, the next step is to remove unwanted characters from the document (characters – symbols of any form) and this can be achieved by removing single character from the document.
- The next step is to remove unwanted words by removing Stop Words, i.e. removing words like the, is, as but, etc. Since these words do not deliver any meaning to the sentence and cannot be considered for the topic collection as well.
- Using an online vocabulary V, decide what type of words to be considered for our model. Eliminate the words that are adjective, preposition, verbs, pronouns, etc. by considering the parts-of-speech library based on the level of efficiency required to the user as this can be varied every time based on the requirement.
- From the document A extract a bag of n words, which can be considered as a bag of n topics, where each and every word is considered as one topic which is compared from the vocabulary V. The bag of words consists of n topics considered from the document a which is denoted as N_A.

- In this paper, a small change incorporated for better efficiency is - After considering the words from the parts of speech library to make it much more efficient try to barge in the topics L with word synonyms S. From the topics/words considered, for each and every word from the bag extract the synonyms because any two words can have same/similar meaning [12]. By doing so, this can reduce the number of words with similar meaning and replace them with their original meaning and extract topics much more efficiently (L – latent topics). After performing each and every task a set of words with topics and each word is considered as a topic $B_{a,n}$ from the document A till the nth word $Y_{a,n}$.

Model

A pre trained model is used where the pre-processed data is passed into the model appropriately so as to get better and efficient results. Before doing that there are few prerequisites that has to meet before passing the data into the model.

- Once the topics are generated then prepare a term document matrix as each document is associated with set of topic proportion θ_a.
- A pre-trained model α_L consists of L topics which specifies each word distribution for a topic.
- From the term document matrix, the next step is to remove the repetition of topics from the bag of words considered. But before removing take a FLAG variable such that to match words whenever there is repetition of words
- Suppose, if words have a similar meaning then replace the original word with its synonym to get the count accurately. By doing so, our model can be much more efficient and get efficient topics which is relatable to the documents considered. A FLAG variable is used to match the words with repetition as well as match the words with their appropriate synonyms. This is an effective and improvisation which is noticeable and implemented in this paper.
- Once replacing of words with their actual synonyms is done, and before removing the repeated words take the count of words which is repeated in the document
- By having a L*L covariance matrix - Σ and by means of natural parameterization δ_a i.e. topic distribution per document $\delta = \log(\theta i/\theta L)$ and count of words generate the probability for the topics L.
- Since the document is huge there will be a greater number of topics generated. So, in order to reduce it further and get better results finding out the correlation between the topics is necessary and consider topics which have the higher probability comparatively.
- The below mathematical equation describes the overview of what will be performed throughout the course of extracting topics from a predefined model. Firstly, find out the probability between the words and then find the correlation between them. The topics from the set of huge number of documents is generated when passed through the model.

$$C\,(P\,(\theta_{1:a}, B_{1:a}, \alpha_{1:L}, \Sigma \mid A; \gamma_{1:a}, \delta_{1:a}))$$

Finally, now there are number of topics which represents the different documents that was passed through the model. By doing so, it will help to describe based on what topic the discussion is made in the entire document which makes tasks much easier and for better understanding.

IV. RESULT

The two data sets considered here is a random data/ document one of which consist information about Computer networks which is in a .PDF format and the other one is a random data set which consist information about title and authors which is a .xlsx format. When either of the data is passed to the model, a set of topics is generated which represents the entire document set. Firstly, let us consider a sample data which consist of information about computer networks and see the acquired result. The below table shows the sample data which was tokenized, filtered using parts of speech tags and with the help of online dictionary generating synonyms respectively which was considered for analysis at the later part.

Table 2: Synonyms for the tokenized words

Word	**Synonyms**		
computer	computer	Computing machine	Computing device
network	network	web	net
group	grouping	radical	Chemical group
communication	communicating		
protocols	protocol	Communications	

This will enhance the output when passed through a model with high classified words and in return get topic similarity much related to the document. Next step is when the words are retrieved with their synonyms – it's just that have to take the count of each and every word, find

the probability between the words and remove the words which is repeated. The below image shows the values which is considered after preprocessing or cleaning the dataset and those values which has to be passed through the model.

computer	10	0.028571429
network	57	0.162857143
group	1	0.002857143
computers	3	0.008571429
set	1	0.002857143
communic	4	0.011428571
protocols	3	0.008571429
interconne	2	0.005714286
purpose	7	0.02
sharing	3	0.008571429
resources	4	0.011428571

Fig 2: Output of the data after Data-Preprocessing

Finally, when the data is passed through the model, words associated to a particular topic is generated, to improve the output - control the number of times the model to be executed and occurrences as well. The below table shows the final result of words associated to a particular topic proportion for a particular document.

Table 3: Words associated to Topics (CTM model)

No.	Topic and Correlation Associated
1	(0, '0.243*"network" + 0.038*"computer" + 0.033*"information"')
2	(1, '0.060*"nodes" + 0.047*"example" + 0.034*"link"')
3	(2, '0.085*"overlay" + 0.050*"networks" + 0.050*"data"')

The below graph represents the topics along with their probabilities (probability on the y axis and names on the x axis).

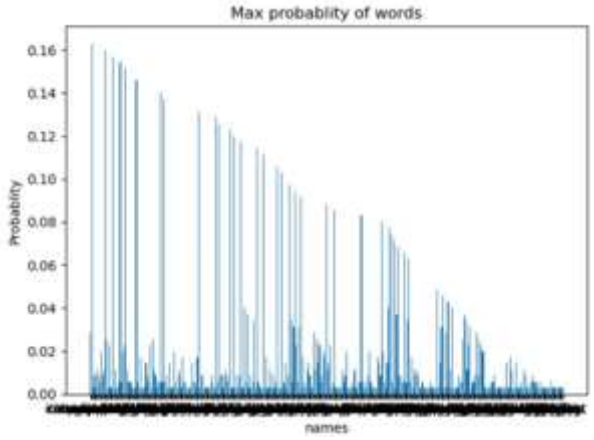

Fig 3: Graph (names * probability)

Now, consider the second scenario of data where it consists of title and author names when passed through the model will find the correlation between the topics and plot a graph accordingly. The values considered here are manually done by considering the values with higher correlation values accordingly. The below graph shows the topics and correlation values associated with it.

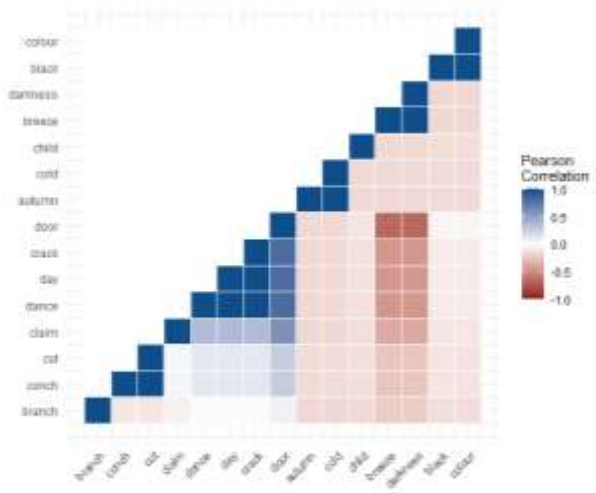

Fig 4: Graph - topics and correlation

After finding the correlation between the topics, those topics are considered with higher correlation values. The below table shoes the topics considered after passing the values through the CTM model.

Table 4: Terms with correlation

Topics	Correlation
Branch	11.8
Child	8.8
Day	29.4
Claim	14.7

The result is in turn helpful to analyze document when the data considered is very large. The similar functionality goes well with the search engines. When a key word is searched using either Google, Bing, etc. it has to scan all the documents i.e. millions of record set. Topic modelling comes into picture at that particular moment where topic proportions are considered and based on the topic ranking is done with highest probability and correlation and then the result is displayed. To enhance this feature synonyms and parts of speech tags are added for better results and later correlation is considered as well.

V. Conclusion

Search engines like Google, Bing use topic modelling (LDA) but implementing CTM which is an improvised version of LDA helps to retrieve word synonyms and with the help of parts of speech, topics can be extracted and perform a correlativity between them such that the topics generated are much more efficient and represent the document in a much efficient manner. Henceforth search engines must use CTM with parts of speech (vocabulary) and their respective synonyms and then find the correlation between the topics which is a enhanced feature for better results. By doing so when a key word is searched using the above frameworks search which is not related to the topic and their meaning does not appear. This will show better results when a key word is searched. CTM with POS and synonyms (vocabulary) perform much better comparatively than the LDA model. Thus, the algorithm specified in this paper will help in improvising the performance and efficiency in search engines.

REFERENCES

[1] M. K. Oo and M. A. Khine, "Correlated Topic Modeling for Big Data with MapReduce," 2018 IEEE 7th Global Conference on Consumer Electronics (GCCE), Nara, 2018, pp. 408-409.

[2] S. Suh and S. Choi, "Two-dimensional correlated topic models," 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, 2016, pp. 2559-2563.

[3] Duan, Y. Li, R. Li, R. Zhang and A. Wen, "RankTopic: Ranking Based Topic Modeling," 2012 IEEE 12th International Conference on Data Mining, Brussels, 2012, pp. 211-220.

[4] C. A and M. G. G, "An Enhanced Method for Topic Modeling using Concept-Latent," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 485-490.

[5] X. Wu and C. Li, "Short Text Topic Modeling with Flexible Word Patterns," 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 2019, pp. 1-7.

[6] D. A. Ostrowski, "Using latent dirichlet allocation for topic modelling in twitter," Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015), Anaheim, CA, 2015, pp. 493-497.

[7] B. Li, W. Xu, Y. Tian and J. Chen, "A Phrase Topic Model for Large-scale Corpus," 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China, 2019, pp. 634-639.

[8] P. M. Prihatini, I. Putra, I. Giriantari and M. Sudarma, "Indonesian text feature extraction using gibbs sampling and mean variational inference latent dirichlet allocation," 2017 15th International Conference on Quality in Research (QiR) : International Symposium on Electrical and Computer Engineering, Nusa Dua, 2017, pp. 40-44.

[9] C. Hsu and C. Chiu, "A hybrid Latent Dirichlet Allocation approach for topic classification," 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA), Gdynia, 2017, pp. 312-315.

[10] M. Rajapaksha and T. Silva, "Semantic Information Retrieval based on Topic Modeling and Community Interests Mining," 2019 Moratuwa Engineering Research Conference (MERCon), Moratuwa, Sri Lanka, 2019, pp. 60-65.

[11] P. Anupriya and S. Karpagavalli, "LDA based topic modeling of journal abstracts," *2015 International Conference on Advanced Computing and Communication Systems*, Coimbatore, 2015, pp. 1-5, doi: 10.1109/ICACCS.2015.7324058.

[12] S. Sendhilkumar, M. Srivani and G. S. Mahalakshmi, "Generation of Word Clouds Using Document Topic Models," *2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM)*, Tindivanam, 2017, pp. 306-308, doi: 10.1109/ICRTCCM.2017.60.

[13] P. Yang, W. Li and G. Zhao, "Language Model-Driven Topic Clustering and Summarization for News Articles," in *IEEE Access*, vol. 7, pp. 185506-185519, 2019, doi: 10.1109/ACCESS.2019.2960538.

[14] T. F. Kennedy, R. S. Provence, J. L. Broyan, P. W. Fink, P. H. Ngo and L. D. Rodriguez, "Topic models for RFID data modeling and localization," *2017 IEEE International Conference on Big Data (Big Data)*, Boston, MA, 2017, pp. 1438-1446, doi: 10.1109/BigData.2017.8258077.