**ML1030 Machine Learning Capstone**

**Research paper-citation analysis**

**Viraja Ketkar**

**Sumaira Afzal**

*School of Continuing Education*

*York University*

**Abstract—Topic modeling aims to discover hidden topics or themes across documents that capture semantic information beyond individual words. It aims to address a key challenge in building a machine learning algorithm that learns from text data by going beyond the lexical level of what has been written to the semantic level of what was intended. Topic models permit the extraction of sophisticated, interpretable text features that can be used in various ways to extract useful information from large collections of documents. The identified topics appear quite representative of the data. Navigating through the research literature is becoming a challenge for a regular reader especially if a particular research paper is cited many times it becomes difficult to know what other papers are saying about the cited paper. To address this we have implemented Latent Dirichlet Allocation(LDA) model to retrieve the latent topics of the citation papers using semantic scholar’s citation dataset. A preliminary evaluation of implementing LDA shows that it can be useful to extract what other people are saying in their citations about a particular research paper to establish an opinion about the research paper.**

1. **Introduction**

Navigating through a large collection of research documents – whether that collection is some digital repository of documents or the internet – is an increasingly difficult task to carry out in a fast and effective manner. The analysis of the research citation with growing number of research papers is a difficult task.If we were to obtain a good summary of a certain paper, instead of looking at that paper itself, we could look at a list of later papers that cites that certain paper.

How can you measure the quality of a research paper? More importantly, how can you determine whether your research is making an impact and is considered important? An objective way is through citation analysis.Citation analysis should be performed to supplement, not replace, a robust system of expert review to determine the actual quality and impact of published research.

The fundamental part of any research is list of references directing readers to prior relevant research. For extracting useful summary of what other papers are talking about academic research papers instead of reading all citations one by one we use Natural Language Processing techniques to find meaning of large collections of citation papers.Topic modelling makes this more achievable.

Topic models are machine-learning algorithms to uncover hidden or latent thematic structures (i.e. topics) from large collections of documents. The latent thematic structures automatically emerge from the statistical properties of the documents and, as such, no prior labeling or annotation is necessary. The result of these thematic structures can be used for automatic categorization or summarization of documents up to a scale that would be impossible to do manually.

With the ability to identify the underlying themes of documents, topic models may provide a better way to represent and navigate among documents. One popular method for topic modeling is Latent Dirichlet Allocation (LDA) , which defines a generative model for documents where each word is determined by a latent underlying topic structure. It is the most popular topic model because it tends to produce meaningful topics that humans can relate to, can assign topics to new documents, and is extensible. The implications of such a model allows for a greater interaction with documents (and similarly words and topics).

For better analysis of text we have built different topic models (LSA, LDA, HDP) and clustering algorithm(k-means, Hierarchical Clustering) to classify research papers based on topic and cluster. We have further compared the models following to pick the model that presents best choice topics for human inference. We used GridSearch algorithm to find the best LDA model. We have benefited from LDA’s topic modeling by generating topics with their keywords and sentences.

Since we are discussing a mechanism that involves inference, visualization is clearly a very important element. The proper visualisation not only illustrates the output of the topic model, but also very helpful for interpreting the topics in a way humans can interpret.

1. **RELATED WORK**
2. **Topic Modeling**

With the advent of linking tools and digital archives of research papers, scientific literature is more easily retrievable than ever before. Therefore, it is only to be expected that the population of researchers turning to citation data will continue to grow. In such a scenario, researchers cannot afford to undermine the importance of citation analysis.There are applications with the citation count feature offered by online databases like Web of Science. It offers most cited papers of an author, citation count for a publication,find the correct author, perform a cited reference search etc. Again the importance of citation analysis can supplement the assessment of quality of research.This project can support the research community to assess the publication.

1. **Visualization**

The task of exact inference is not tractable, there are also many variations to the method of inference, in this project we have implemented simple visualization techniques.There has been a large and diverse amount of visualization techniques for topic model output, especially with extensions to the basic LDA model.With the standard LDA model, it is relatively simple to display many different types of information beyond document topic labels: similar documents (or topics or words), most relevant documents based on a particular topic, most relevant words from a topic, among many other things.However, despite the various types of information and clean interface, we believe that quantitative information is important for understanding the data.

1. **Inference**

There is less work on complete analysis of cited research paper.The Web of Science and Google scholar can be/are data source for citation analysis in one or the other way.Most people use Google scholar to support their analysis because it is freely available to any one with an internet connection whereas The Web of Science is only available to those academics whose institutions are able and willing to bear the (quite substantial) subscription costs of the Web of Science. Given this it is further costly to analyze the literature of research publication using the citations. Using the standard LDA model, we explore a simpler solution of finding the latent meaning of citations and finding top topics and their keywords along with their sentences.

1. **BACKGROUND**
2. Latent Dirichlet Allocation

LDA is a generative probabilistic topic model that aims to uncover latent or hidden thematic structures from a corpus. The latent thematic structure, expressed as topics and topic proportions per document, is represented by hidden variables that LDA posits onto the corpus. The generative nature of LDA describes an imaginary random process based on probabilistic sampling rules from which we assume that the documents come from. However, we only observe the words within documents and need to infer the hidden structure, that is, the topics and topic proportions per document, by applying statistical inference techniques. This process aims to answer the question: Which hidden structure or topic model is most likely to have generated these documents? In doing so, we obtain the posterior distribution that captures the hidden structure given the observed documents.

According to studies topic models based on the LDA have the following parameters:

* *α*: The parameter of the prior Dirichlet distribution for “documents-topics”;
* *β*: Parameter of the prior Dirichlet distribution for “topics-words”;
* *tn*: The number of topics;
* *b*: The number of discarded initial iterations according to Gibbs sampling;
* *n*: The number of samples;
* *si*: Sampling interval.

We have fine tuned the value of *α, β* and *tn* toget the optimal value of these parameters

1. **Topic Coherence Measurement**

After approximating LDA’s posterior distribution, the K topics are represented as multinomial distributions over V. Each topic distribution contains every word but assigns a different probability to each of the words. The words within topics with high probability are words that tend to co-occur more frequently. These high-probability words, usually the top 10 or top 15, are used to interpret and semantically label the topics. However, LDA outputs as many topics as are defined by K: a low K results in too few or very broad topics, whereas a high K results in uninterpretable topics or topics that ideally should have been merged. Choosing the right value of K is thus an important task in topic modeling algorithms, including LDA. Measures such as the predictive likelihood of held-out data have been proposed to evaluate the quality of generated topics. However, such a measure correlates negatively with human interpretability, making topics with high predictive likelihood less coherent from a human perspective.As a result, researchers have proposed topic coherence measures, which are a qualitative approach to automatically uncover the coherence of a topic and the underlying idea is rooted in the distributional hypothesis of linguistics; words with similar meanings tend to occur in similar contexts. The topics are considered to be coherent if all or most of the words, for example, the topic’s top N words, are related. We have adopted the CV coherence measure for topic coherence calculations.

1. **METHODOLOGY**

We have explored latent topics and their topic coherence score, a proxy for topic quality when applying LDA on text data.

**The Dataset**

We experimented on three datasets, BERT ,GNN and Elmo. Each of these papers contain 900+ citations. We only downloaded the PDFs available on ArXive.org. The following table shows the count of PDFs, their respective extracted sentences and documents made from those sentences.

**TABLE. I**

**OVERVIEW OF DATASETS BERT,GNN,ELMO**

|  |  |  |  |
| --- | --- | --- | --- |
|  | BERT | GNN | ELMO |
| Total Citations urls | 998 | 1000 | 990 |
| PDFs | 647 | 647 | 605 |
| Sentences | 1002 | 1245 | 751 |
| Documents(corpus) | 1002 | 1245 | 751 |

The choice of these papers is based on the factors: 1) The paper is highly cited; 2) The paper has arXiveId as we have to extract it from semantic scholar api; 3) The papers which have cited this paper are downloadable PDF files.

All papers citation papers were downloaded from semantic scholar api in PDF format and then converted to XML using GROBID(a tool to process PDF files to retrieve xml content.

**Data gathering/engineering**

We extracted the sentences from the pdf of citation paper using Semantic Scholar API. The data was collected from citation papers which required some data engineering steps to build data pipeline.

We obtained the PDF content of the citing papers using a simple python script to download them from the API. For papers behind a paywall (i.e. IEEE and ACM), we ignore the paper altogether.We implemented the following algorithm to extract sentences from pdfs.

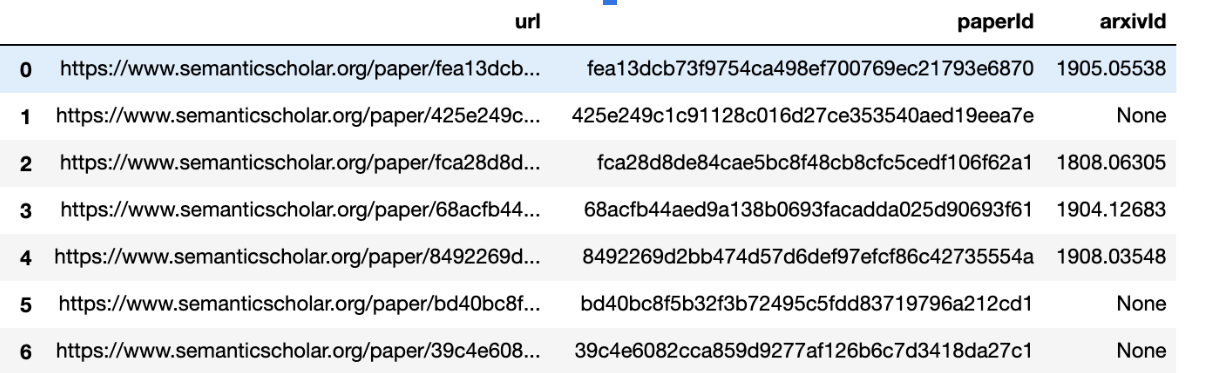
|  |
| --- |
| Algorithm for data pipeline     1. Choose a paper of interest (In our case BERT, Elmo, GNN) 2. Obtain a list of all the papers that cite this paper 3. Download the pdf of citation papers from API 4. Convert the PDF file into XML/TEI format with full text option. (We used the recommended Grobid tool) 5. Get XML nodes inserted into sentences that denotes all the references from the paper 6. Find the correct reference ID for paper A with simple fuzzy string matching on the title (We chose [SequenceMatcher](https://stackoverflow.com/a/17388505)) 7. Generate the output of representative sentences 8. Convert the sentences as Pandas DataFrame and save as CSV file |

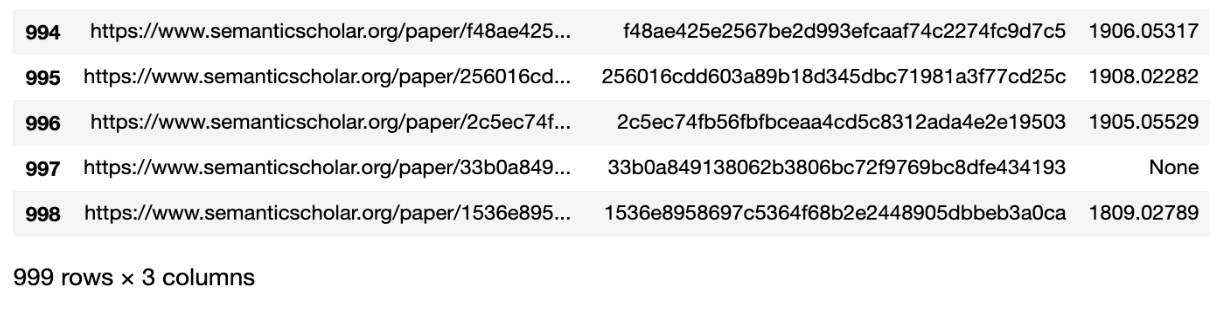
We downloaded the files as pdf and stored in input folder.The input folder of PDF files is uploaded to Grobid server to create xml files and stored in output folder. We pass the files to get the reference where the paper name is mentioned. We get reference as ref tag. We stored 5 ref sentences from each paper. Finally we converted the sentences as dataframe .

The extracted text data .csv format.

**OVERVIEW OF CITATIONS OF BERT**

***url, paperId, arxivId* : FEW DICTIONARY ELEMENTS OF JSON FILE.**





**TABLE.II**

**File download/GROBID xml conversion process**

|  |  |  |
| --- | --- | --- |
| **File Download from archive.org** |  | **File processing on GROBID server** |
|  |  |  |

**The GROBID Python Service is run with this command:**

**python3 grobid-client.py --input ../input1 --output ../output1 processFulltextDocument --force**

**Data pre-processing**

All text data was tokenized, and single-character words, numbers, and punctuation marks were removed. Furthermore, we removed words that belonged to a standard English stop word list. Apart from grouping lowercase and uppercase words, normalization method such as stemming or lemmatization were applied.

Using NLTK and gensim libraries data preprocessing was done in three steps:

1. **Cleaning:** remove punctuation, remove stopwords
2. **Tokenization:** Split the sentences into words. We used Gensim utilities for tokenization
3. **Normalization:**

* Lowercase the words
* Lemmatization — words in third person are changed to first person and verbs in past and future tenses are changed into present.
* Stemming — words are reduced to their root form.

After every preprocessing step, it is a good practice to check the most frequent words in the data.

Prior to topic modelling we create the bigrams of the tokenized and lemmatized text. After normalizing the data set we created Bigrams and Trigrams. The Dictionary of processed documents is created with BOW model. All the text documents combined as the corpus.To run any mathematical model on text corpus, it is a good practice to convert it into a matrix representation. LDA model looks for repeating term patterns in the entire **Document-Term matrix**. Python provides many great libraries for text mining practices, “gensim” is one such clean and beautiful library to handle text data.

1. **Creating LDA Models**

For datasets we created LDA models by varying the K parameter i.e. k = { 2 ...40 } and repeating this process three papers.The LDA models are created using the Python library Gensim.The Dirichlet parameter alpa α = 0.01. By keeping α < 1, the modes of the Dirichlet distribution are close to the corners, thus favoring just a few topics for every document and leaving the larger part of topic proportions very close to zero. The LDA models are created using the Python library Gensim. The convergence iteration parameter for the expectation step (i.e. E-step) is set to 100; the part where perdocument parameters are fit for the variational distributions.

1. **Model Evaluation**
2. **Topic Coherence**

For every LDA model created (120 in total for 3 papers, presenting results of only one paper), we calculated the CV coherence score.From the table below we can see the highest topic coherence is 0.6113 where k=10. The LDA model with the optimal coherence score, obtained with an elbow method (the point with maximum absolute second derivative), was additionally analyzed. A model can be evaluated with log-likelihood and perplexity.With higher log-likelihood and lower perplexity (exp(-1. \* log-likelihood per word)) is considered to be good.One drawback of perplexity is it is not near human ranking thus not giving enough insight about topics.So keeping in view we chose Coherence value to evaluate the model.

1. **GridSearch**

We used GS to find best LDA model for that we have to build different models with these parameters tuning parameters(n-component and learning decay)

The most important tuning parameter for LDA models is n\_components (number of topics). In addition, learning\_decay (which controls the learning rate) as well.

Besides these, other possible search params could be learning\_offset (downweigh early iterations. Should be > 1) and max\_iter. These could be worth experimenting if you have enough computing resources.

Be warned, the grid search constructs multiple LDA models for all possible combinations of param values in the param\_grid dict.

1. **RESULTS**

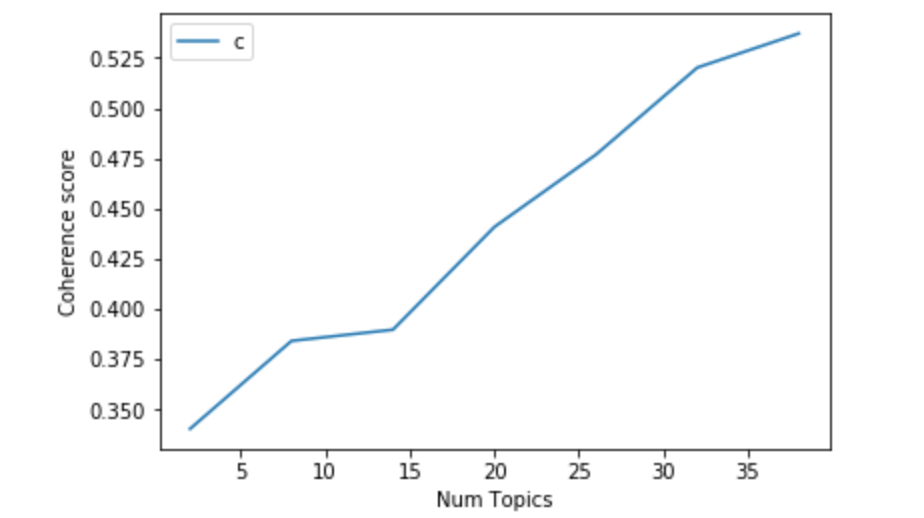
**TABLE. III**

**CALCULATED COHERENCE SCORE FOR BERT (No of Documents=1002)**

***K= NUMBER OF TOPICS, Cv= COHERENCE VALUE***

|  |  |  |  |
| --- | --- | --- | --- |
| ***K*** | ***Cv*** | ***K*** | ***Cv*** |
| 5 | 0.3227 | 22 | 0.4941 |
| 6 | **0.5006** | 23 | 0.4865 |
| 7 | **0.5695** | 24 | 0.4646 |
| 8 | 0.4951 | 25 | **0.5849** |
| 9 | 0.4186 | 26 | **0.5258** |
| 10 | **0.6113\*\*** | 27 | **0.5766** |
| 11 | 0.4799 | 28 | 0.4854 |
| 12 | **0.5184** | 29 | 0.4772 |
| 13 | **0.5094** | 30 | **0.5392** |
| 14 | 0.4668 | 31 | **0.5790** |
| 15 | **0.5076** | 32 | **0.5648** |
| 16 | **0.5728** | 33 | **0.5026** |
| 17 | **0.5335** | 36 | **0.5427** |
| 18 | **0.5824** | 37 | **0.5394** |
| 19 | **0.5431** | 38 | **0.5103** |
| 20 | **0.5599** | 39 | 0.4686 |
| 21 | **0.5097** | 40 | **0.5062** |

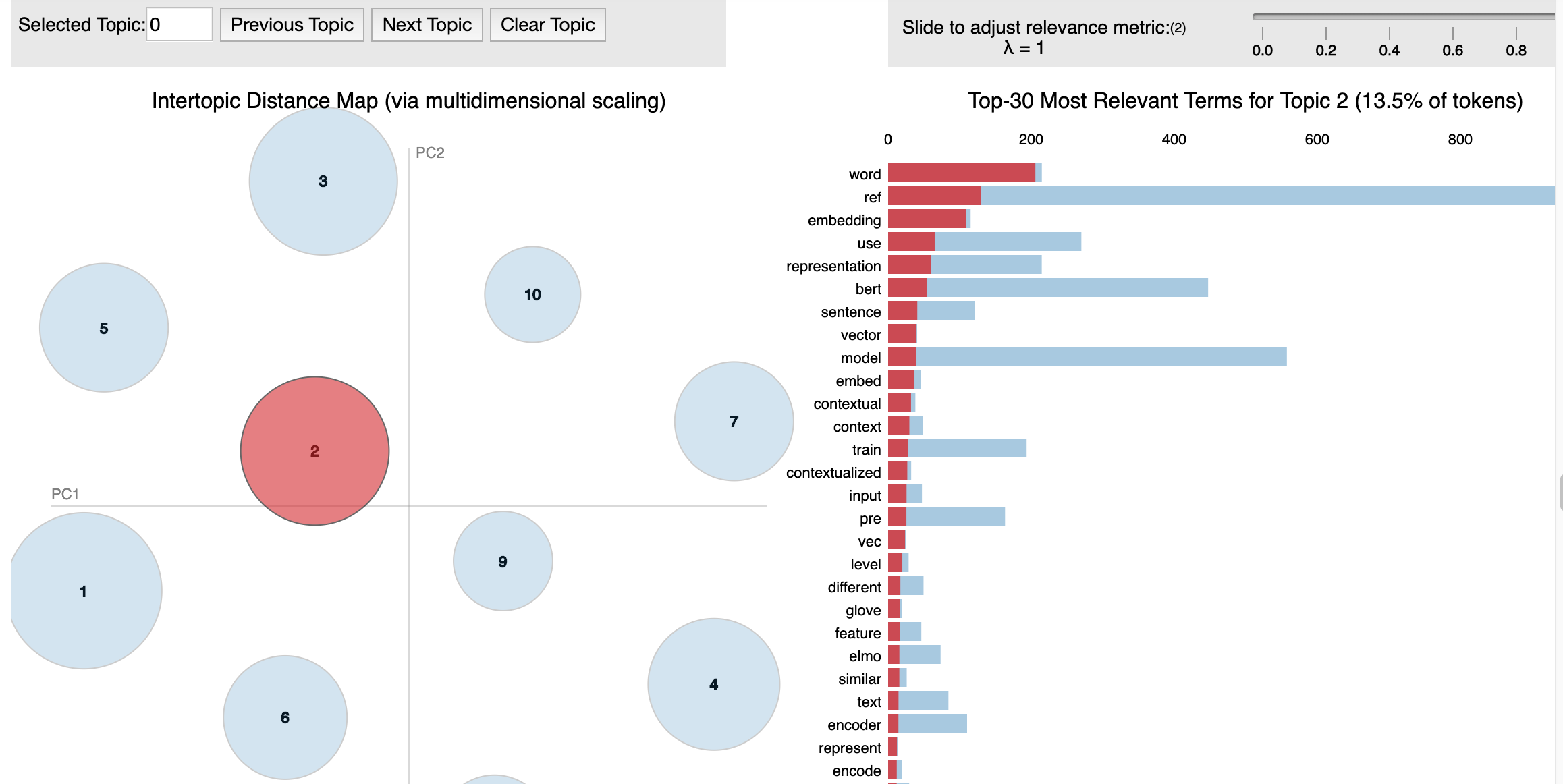
**FIG 1.TOPIC EVALUATION WITH ELBOW METHOD**

****

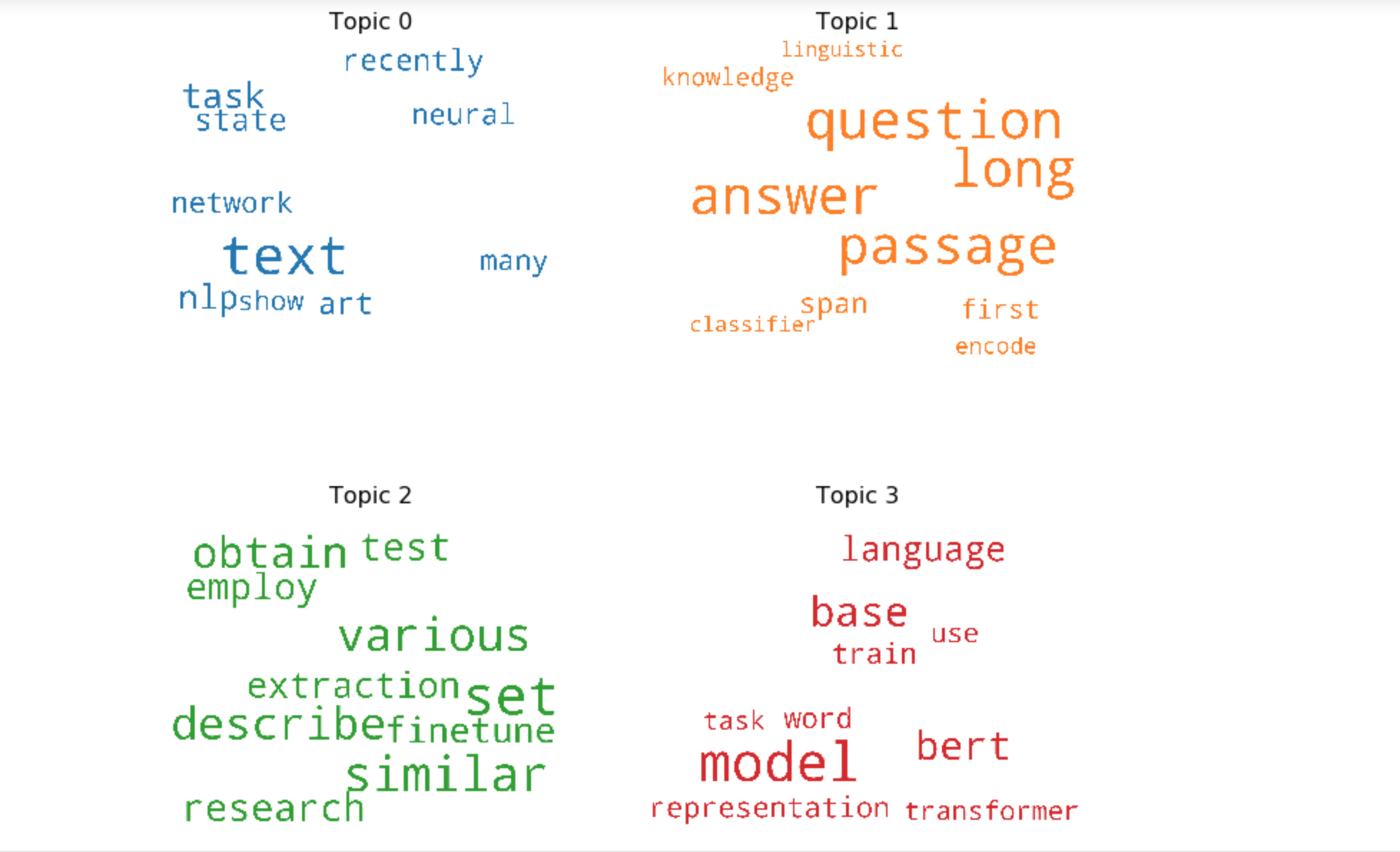
**LDA Topic Visualisation**

We visualized the topics using pyLDAvis Python library for interactive topic model visualization. pyLDAvis is designed to help users interpret the topics in a topic model that has been fit to a corpus of text data. The package extracts information from a fitted LDA topic model to inform an interactive web-based visualization.

**FIG. 2 TOPIC-WORD VISUALIZATION FROM BERT CITATION PAPERS**

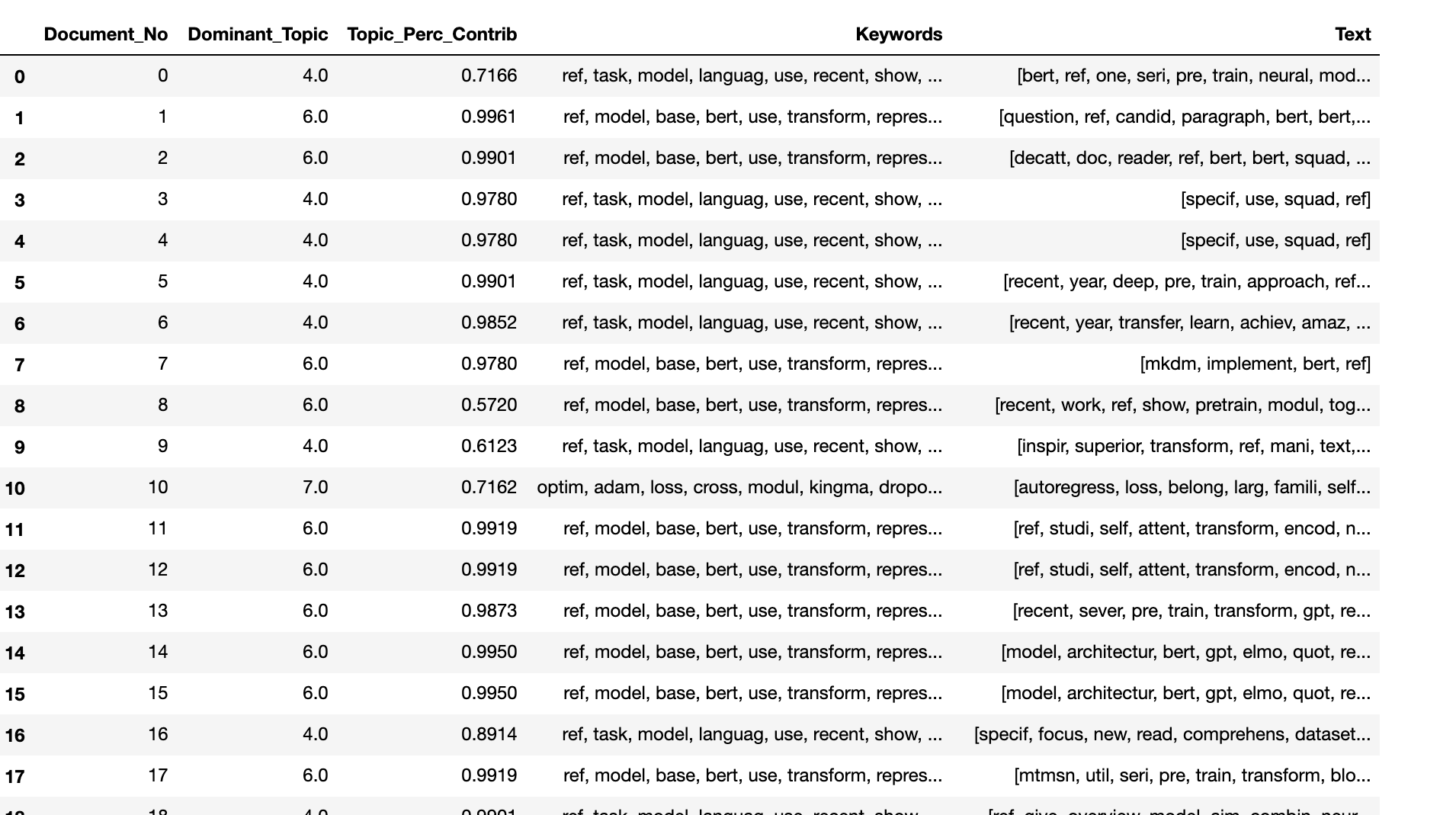


**FIG 3. WORD CLOUD OF TOP N WORDS IN EACH TOPIC**

****

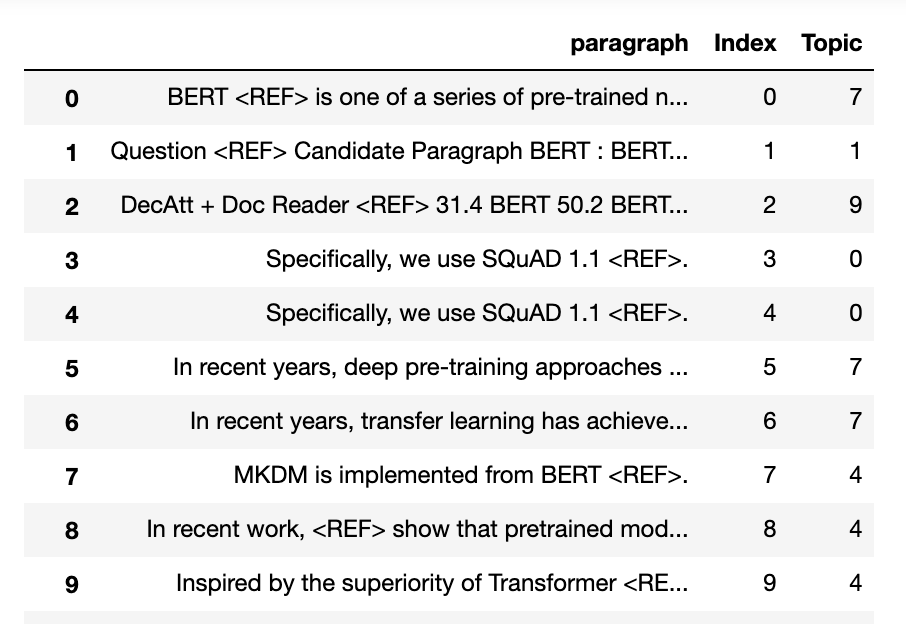
**TABLE.IV DOMINANT TOPICS,TOPIC % CONTRIBUTION,**

**KEY WORDS AND ORIGINAL TEXT**

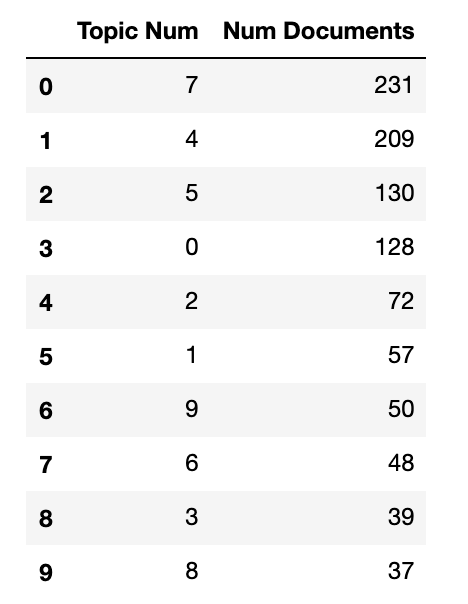
****

We have added a column to the original data frame that will store the topic for the text.We have used LDA.Transform() method which assign the probability of all the topics to each document.

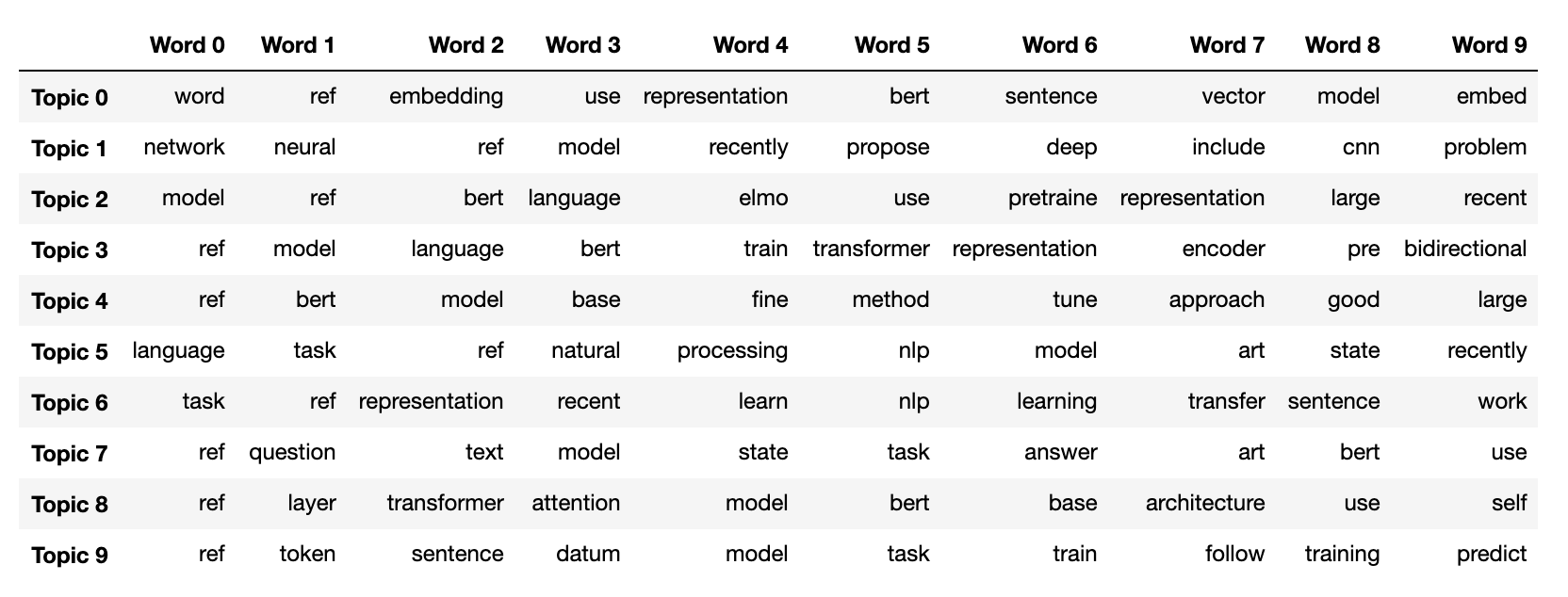
**TABLE V. PARAGRAPH SENTENCES FROM CITATION PAPERS FOR BERT**



**TABLE VI. REVIEW OF TOPIC DISTRIBUTION ACROSS DOCUMENTS**

****

**TABLE VII . TOP 10 TOPIC-KEYWORD REPRESENTATION OF TOPIC**



1. **DISCUSSION and EXPERIMENT**

The coherence of a topic is based on the topic’s top 10 - 15 words and shows how strongly pairs of these top words support each other within the corpus. Such an approach, drawing on the philosophical premise that a set of statements or facts is said to be coherent if its statements or facts support each other, informs us about the understandability and interpretability of topics from a human perspective. Secondly the quality of topics from a human perspective was lowered by the inclusion of incorrect terms in the top 15 words. Such terms, however, are not related to the biological, ecological, or socio-ecological meanings of those topics but can be seen as noise terms: using, optim, use, ref, task and kingma as seen in the keywords results. There is little to no specific semantic meaning behind these terms, and although they are important in written text, they are less important when uncovering latent semantic structures (i.e. topics) from documents. This issue may be potentially rectified by a part-of-speech (POS) tagger to eliminate the verbs or prepositions that crop up as noise among the topic’s top words. However, one should proceed carefully in cases where verbs are important cues for understanding the semantics of the top words.

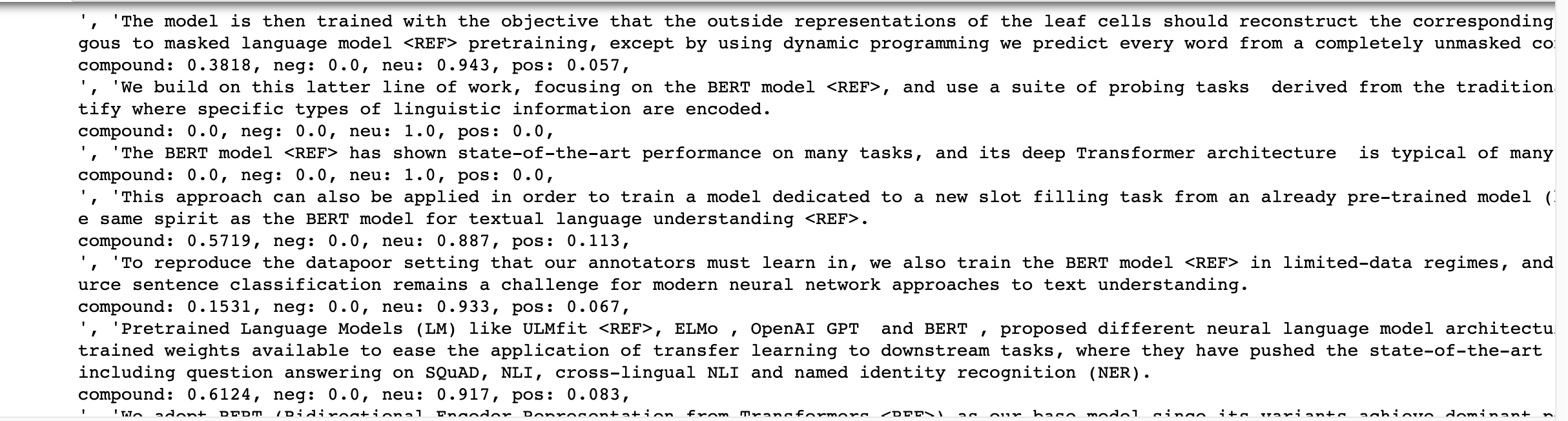
A comparison between other LDA models (not presented) shows word limitation, with a relatively small number of documents, has practical effects on the level of detail (i.e. granularity) of uncovered LDA topics.

Another comparison between LDA and Clustering Algorithm(not presented) shows the LDA can achieve better results than k-means to find the latent meaning where as clustering requires more time and computation time to fine tune the clusters. Topic labeling is usually performed by inspection of the topic words with the highest probabilities (top 10 or 15). Such an approach might up-weight terms that have high probability under all topics on the other hand cluster labelling can not be done manually but again it requires deep domain knowledge to be labelled and make it useful.Finally, data, being restricted by the limited number of words, fail to adequately convey the heterogeneity of research ideas or topics that are part of a document. Uncovered latent topics might thus not completely resemble the document collection and, as a result, provide a limited or even incorrect view of the underlying thematic structure.

**Sentiment analysis**

Sentiment analysis can shed light on the emotions expressed when discussing a given topic.we have imported tools for the NLTK: the VADER lexicon, which calculates negative, positive, and neutral values for our text, and a word tokenizer, which splits our large text file into sentences or words.

**FIG. 4 SENTIMENT ANALYSIS**



Though sentiment analysis can be a powerful tool for quickly determining the emotions expressed through text, there are limitations to what sentiment analysis can provide. Additionally, like all text analysis, we need to be cautious in interpreting the results. Sentiment Analysis can be helpful if user want to analyze citation papers who has given negative comments about the original paper.

1. **CONCLUSION**

In this project, we presented an LDA model with topic coherence scores of LDA topics from citation text data. Three datasets were compared, BERT consisting of a 647 research citation papers, ELMO consisting of 629 research citation papers , and GNN with 647 research citation papers. Topics were statistically compared by adopting the CV coherence measure that shows the highest correlation .The LDA models with the optimal coherence scores were manually and statistically (elbow method) inspected

Our results show that uncovering LDA models from a paper with few citation papers with, relatively speaking, a low number of documents are very prone to noise terms that crop up into the topic’s top words—words that are often used to capture the semantics of the topic. Such noise terms require special attention when dealing with small data with, e.g. an increased cleaning phase, POS filtering, or a domain-specific stop word list.Our results show that larger data(containing 1000+ documents) seems less affected by such words, thus increasing the coherence. Increasing the number of documents (e.g. BERT 1000+ ) results in fewer noise terms, thus an improvement in coherence.

**With implementation LDA Model we achieved following:**

* + **Dominant Topics**
  + **Dominant-Topic with Words**
  + **Dominant Topic with Original sentences**

From the above information we can easily make inference and detect citations context in scientific papers.

1. **Future Work**

A detailed analysis of the reasons behind these differences would yield interesting results and would be a possible direction for future research. We can further experiment for determining the optimal number of topics will be experimented based on the following principles

1. Using dense vector representation (GloVe, FastText, Word2Vec) Word2vec produces one vector per word which is great for digging into documents and identifying content and subsets of content
2. Build topic model on a large collection of scientific documents.
3. Prototype App [ML Flask App](http://localhost:5500/index1.html)