# **Topic Modelling with Latent Dirichlet Allocation**

## Introduction

Topic modelling is a technique derived from machine learning used within natural language processing in order to extract and identify topics from a corpus. It is considered a type of unsupervised learning, meaning it does not require labelling or previous knowledge of topics within a document. It instead utilises statistical patterns within the words of a document to group them into topics. This paper will be an exploration of what Latent Dirichlet Allocation (Blei et al., 2003), the most popular contemporary topic modelling algorithm is, how we got here, where we go from here, and also an example reference design on how to apply it to a dataset.

## Literature Review

### Evolution of Topic Modelling

The earliest examples of topic modelling can be dated back to the 1960s, elementary methods such as:

1. Cluster Analysis – this is perhaps the earliest method in which handcrafted similarity metrics were used to identify topics; one of the downsides of this methodology was that it required immense tweaking from a trained expert which prevented it from being scaled (Jain et al., 1999).
2. Term Frequency-Inverse Document Frequency (TF-IDF) – measures the frequency of a term within a document relative to the number of documents it is present in within the corpus. This method was mainly used for document retrieval and keyword extraction (Salton & McGill,1986).
3. Key-Term Extraction – this refers to a variety of methods such as heuristics, term weighting, and term frequency analysis were used to identify key phrases or terms within a document (Turney, 2000).

In the 90s, more advanced methods emerged which eventually evolved into LDA, however the evolution starts with Latent Semantic Indexing (LSI) to Probabilistic Latent Semantic Analysis (pLSA) and finally to Latent Dirichlet Allocation (LDA), each representing a progression in complexity and comprehension capabilities of probabilistic modelling for retrieving topics in a collection of documents. Here is a brief overview of this evolution:

1. Latent Semantic Indexing (LSI):

* LSI was first introduced in the 90s for document retrieval and topic modelling (Deerwester et al., 1990) – it utilises singular value decomposition or SVD (a factorisation of a real or complex matrix – essentially a mathematical method used to decompose a matrix into component matrices). In the case of LSI, it was used for dimensionality reduction of the term-document matrix.
* LSI specialised in capturing the semantic relationship between documents and terms however its main limitation was its inability to provide a clear probabilistic interpretation and ineffectiveness in handling words with multiple meanings (polysemy).

1. Probabilistic Latent Semantic Analysis (pLSA):

* pLSA is an evolution of LSI which was developed in order to do topic modelling within a probabilistic framework with topics represented as latent variables (unobserved or hidden variables used to explain the relationship between observed variables).
* pLSA provides more effective modelling, however, has some drawbacks such as the need for specifying the number of topics before training and a deficiency in the interpretability within the word-topic and document-topic distributions (Hofmann, 2001).

1. Latent Dirichlet Allocation (LDA):

* LDA was first introduced by Blein, Ng, and Jordan in 2003 (Blein et al., 2003), it represented a significant improvement in topic modelling. It improves the probabilistic modelling of topics using a generative process that depends on the Dirichlet distribution. LDA is built on the assumption that documents are mixtures of topics, and topics are mixtures of words.
* LDA improves on pLSA in many areas e.g. provides a clear probabilistic interpretation of word topic distributions (Griffiths & Steyvers, 2004). LDA is currently seen as the standard for topic modelling due to its effectiveness, simplicity relative to more advanced deep learning based methods, and interpretability. LDA models are commonly used in content recommendation system e.g., the YouTube recommendation algorithm and finding topics that are relevant to corporations or governments to automate the acquisition of what the general public thinks of their products or policies respectively.

### New Frontiers

Although LDA is essentially the industry standard, there have been newer models that have improved upon it in recent years in many specific attributes however not all have seen widespread use:

1. Evolutions of LDA – It is only natural that LDA has been used as the foundation for newer models. There are many variations and extensions of LDA such as sLDA (supervised LDA), topic models based on non-negative matrix factorisation (NMF) (Gokul & Sundararajan, 2021), Variational Inference for LDA which refines the estimation of topic distributions in documents, and Dynamic Topic Models (DTM) that extends LDA in order to capture how topics evolve over time (Blei & Lafferty, 2009).
2. Transformer Based Methods – for example BERTopic uses pre-trained large language models such as BERT (Bidirectional Encoder Representations from Transformers) for topic modelling. It provides the ability for advanced topic extraction from text while being able to comprehend the semantic meaning of words (Kiela & Conneau, 2020).
3. GSDMM (Gibbs Sampling for the Dirichlet Multinomial Mixture Model) – a dynamic topic modelling algorithm that uses Gibbs sampling for topic inference. It is specialised for extracting topics from short texts but is outperformed by models such as LDA and other advanced deep learning based methods for more complex and longer documents (Yan et al, 2013).

## How LDA Works

Latent Dirichlet Allocation is used to find unobserved, latent thematic structures from within a collection of documents (Blei et al., 2003). If you had a collection of news articles from a particular day, you could pass it through an LDA model to find the topics that were most relevant in the news cycle that day.

LDA uses a probabilistic framework, it processes each document as a bag of words – a collection of words drawn from a multinomial distribution over a set of latent topics (Blei et al., 2006). The algorithm runs on the assumption that each word in a document originates from a topic with a certain probability. It iteratively optimises a latent variable model in order to infer the topic-word and document-topic distributions (Griffiths, 2004).

The algorithm can be described in layman’s terms as:

* + - 1. Initialization: Randomly assigning the words in each document to a topic
      2. Expectation-Maximisation (EM) Algorithm:

1. Expectation (E): Estimate the probability of each document belonging to each topic (involves calculating the proportion of words in the document assigned to each topic – document-topic probability) and the probability of each topic generating each word (counting the occurrences of each word in each topic – topic-word probability).
2. Maximisation (M): Update the document-topic and topic-word distributions using the estimates from the expectation step.
   * + 1. Iteration: Cycle through steps 2a and 2b until convergence which is indicated by stabilisation of the distribution.

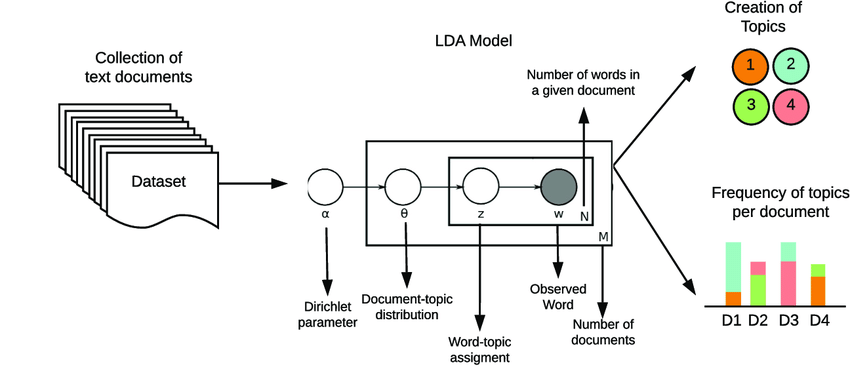


Figure 1 – Shows a basic diagram of the LDA model (Zhang et al, 2014).

## Methodology

### 1. Importing Libraries:

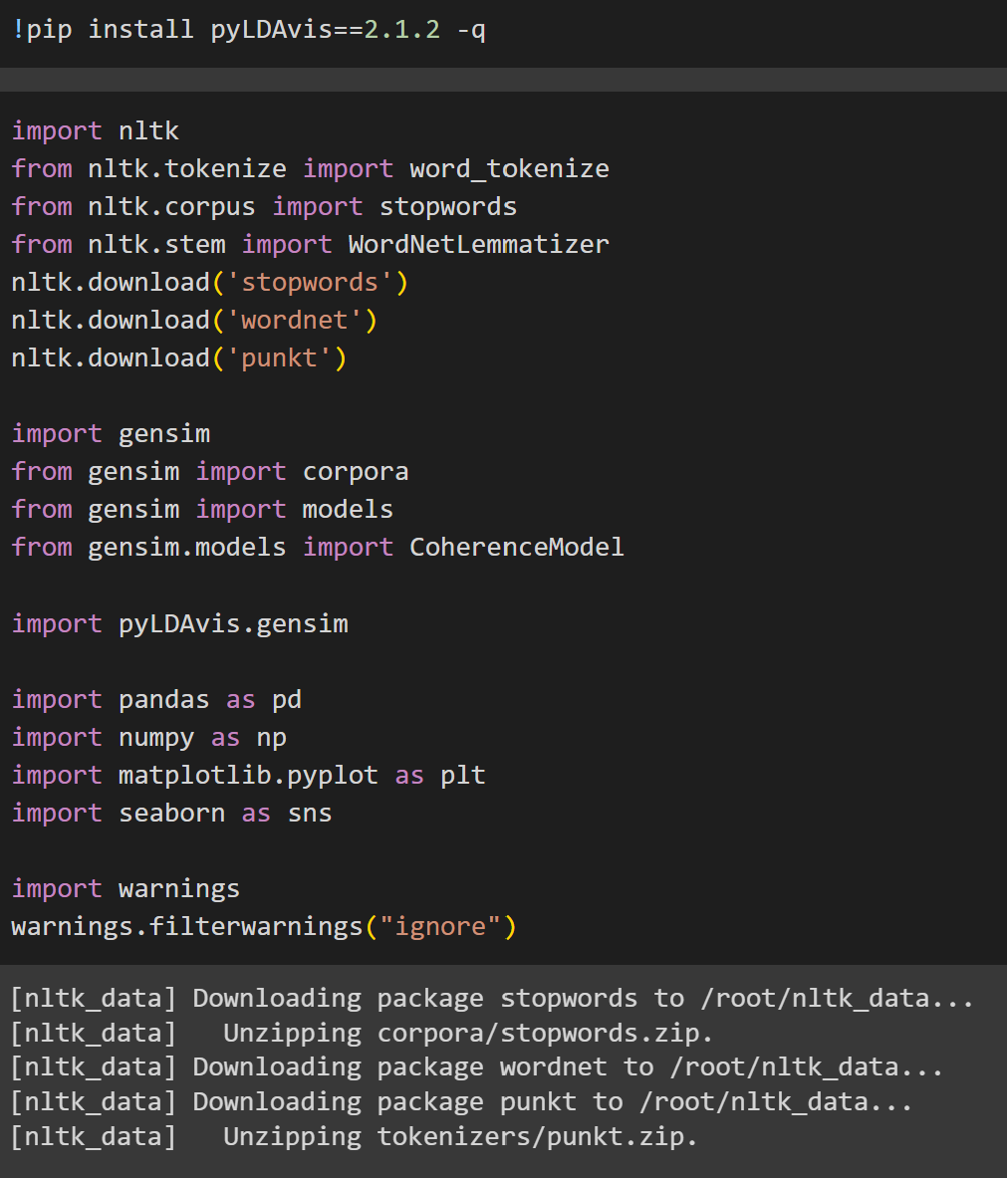


Figure 2

In this step all the libraries and corresponding packages are imported and downloaded. Natural Language Toolkit is a Python library used for preprocessing text data so it can be efficiently processed by an NLP algorithm. Gensim is the de facto library used for LDA as their implementation is much faster than their competition such as Scikit Learn. PyLDAvis is a visualisation tool that was originally developed for R but was ported to Python, it specifically allows you to visualise topics and word distributions for gensim’s implementation of LDA. Pandas, NumPy, Matplotlib, and Seaborn are used for data manipulation, exploratory data analysis, and visualisation.

### 2. Data Preprocessing Functions:

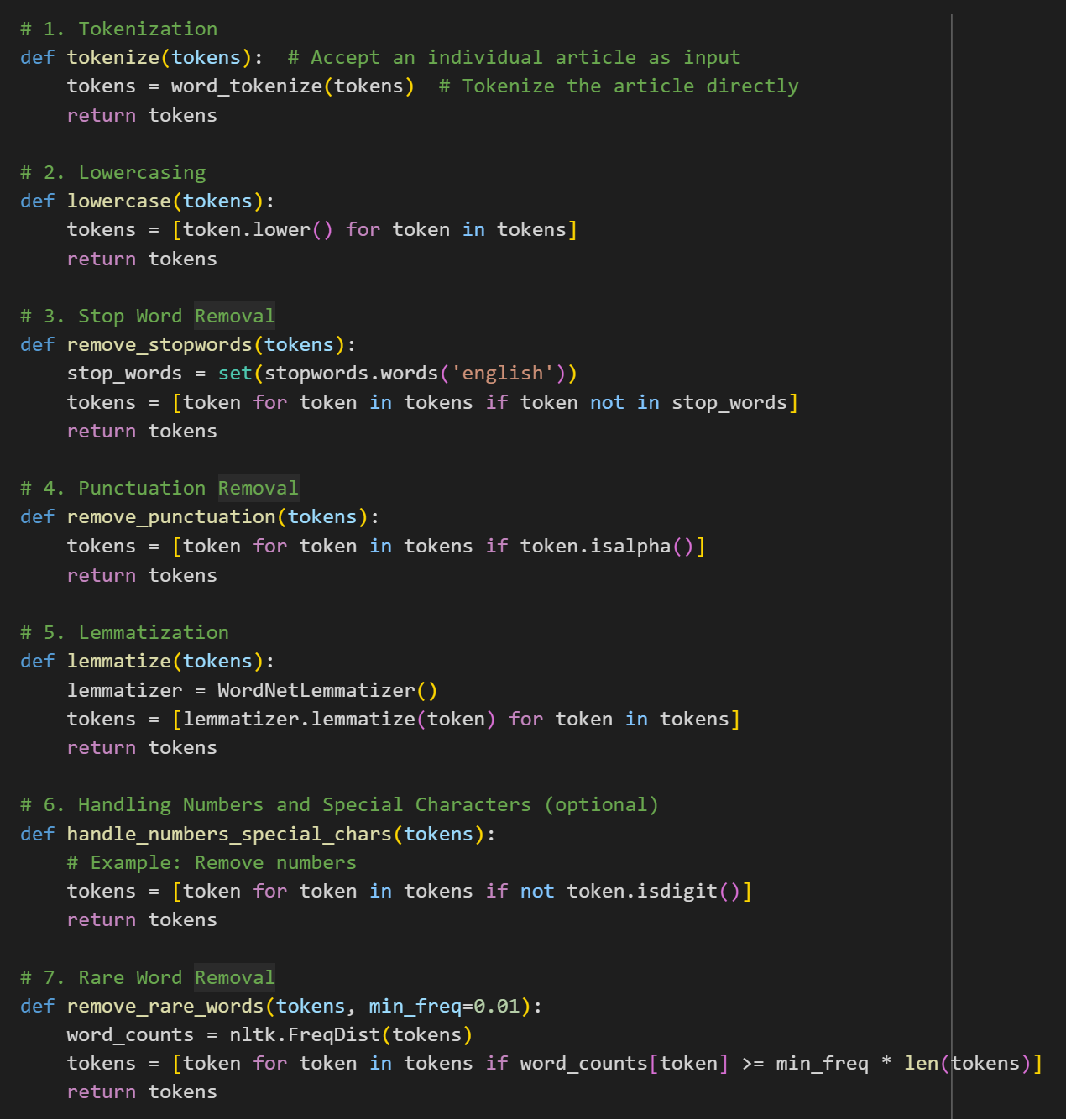


Figure 3

In this stage the data preprocessing functions are defined for use later on in the preprocessing pipeline. The tokenize function splits text into individual words or tokens, the lowercase function converts all tokens to lowercase, remove\_stopwords function removes common words e.g. “like”, “because”, “and” that don’t contribute meaning and are not topics. The remove\_punctuation function removes punction marks, the lemmatize function reduces a word down to its lemma or root form e.g. “flying” to “fly”. And the remove\_rare\_words function deletes words that do not appear frequently which in this case can be considered as a form of noise. The function calculates the frequency of each word in the corpus and if it falls below the min\_freq parameter the word is removed.

### 3. Loading Data:

A screenshot of a computer

Description automatically generated

Figure 4

This step reads TheHackerNews\_Dataset which is a dataset in excel form containing over 3000 news articles on cyber security. The data is read into a pandas data frame and the unnecessary columns like the title and https link columns are removed leaving behind a data frame with just the content of the articles.

### 4. Preprocessing Articles:

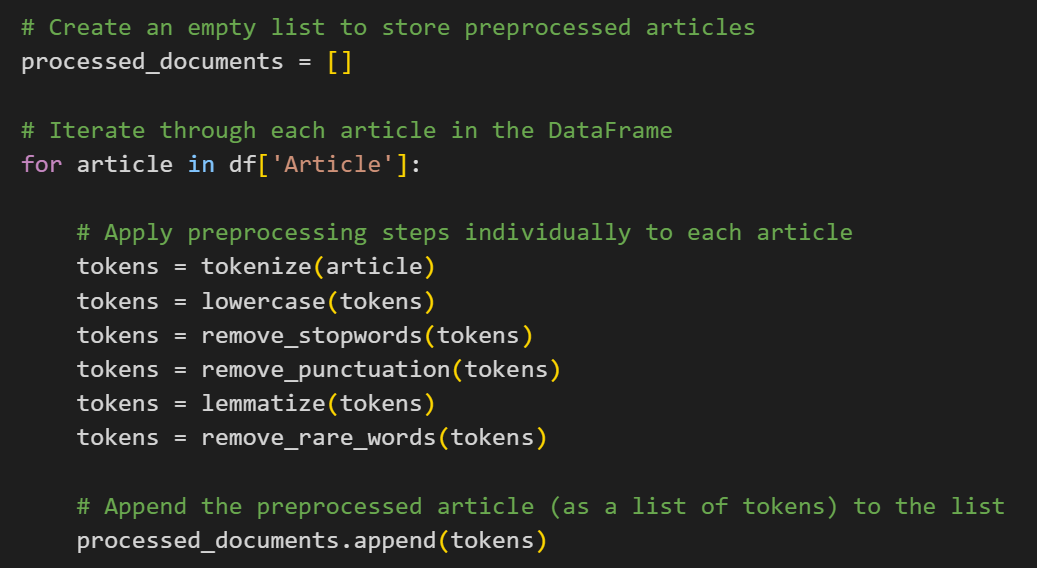


Figure 5

LDA requires each document or article in this case to be pre-processed and in a list. The processed\_document list is initialised and the for loop takes each article and tokenises it and then applies the preprocessing functions mentioned earlier, finally adding the document to the processed\_documents data structure. This is done for all the articles in the data frame leaving a list containing lists of tokens/words.

### 5. Creating Dictionary and Corpus:

A black background with colorful text

Description automatically generated

Figure 6

The dictionary constructs a mapping between unique IDs (integer identifiers) and unique words in the corpus, this is done by assigning a unique ID to each unique word, ultimately creating a word-ID mapping. The benefits of doing this is it enables an efficient numerical representation for model calculations and simplifies tracking word frequencies and relationships between topics.

The Corpus is created to transform the collection of processed documents into a format suitable for modelling by LDA; each document is converted into a list of word id and word count using the dictionary. This is done because it creates a structured input format for the LDA model to analyse the co-occurrence patterns and allows the model to ignore syntactic complexity so it can focus on word frequencies and relationships.

### 6. Optimizing LDA Model:

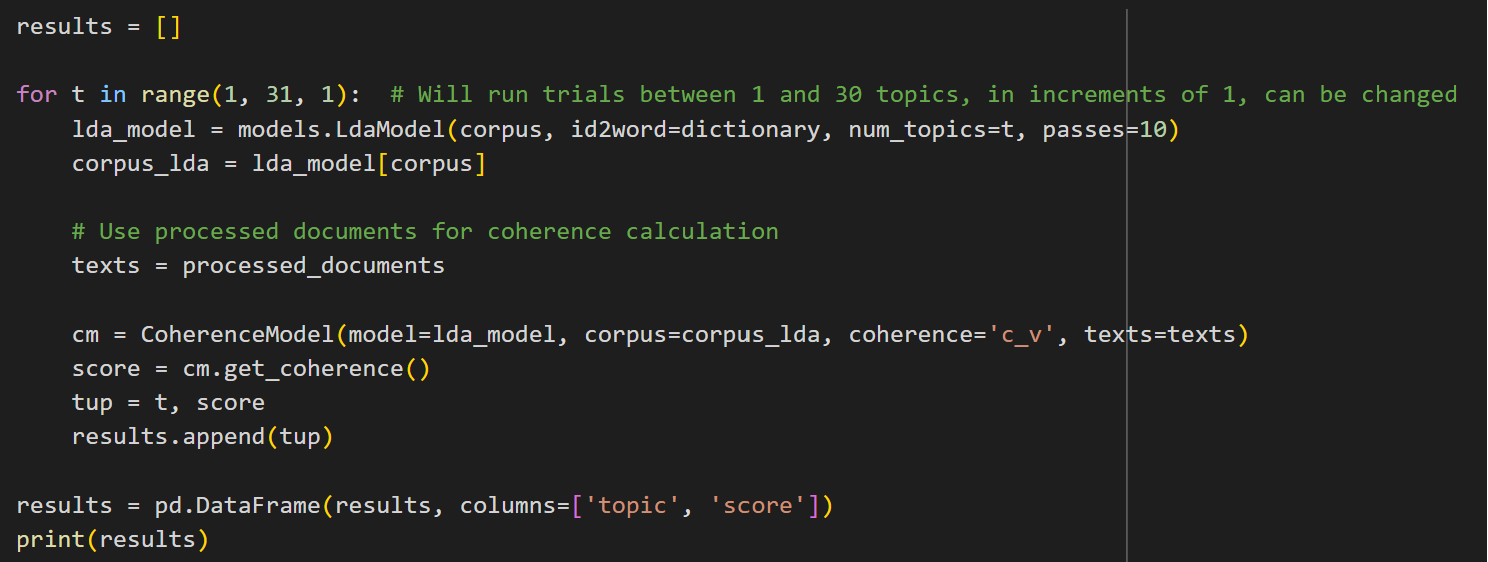


Figure 7

This is the hyperparameter tuning stage where we train the LDA model on various numbers of topics in this case between 1 and 30 topics in iterations of 1 topic with a standard 10 passes which should be enough to achieve convergence for the size of the dataset. The coherence value (in this case the ‘c\_v’ metric, other metrics such as ‘umass’ can also be used to measure coherence) refers to the interpretability and topical relevance of the words assigned to each topic. The coherence score is calculated and stored in a data frame.

A screenshot of a computer screen

Description automatically generated

Figure 8 – displays the output of the hyperparameter tuning trials.

A screen shot of a graph

Description automatically generated

Figure 9

The coherence scores are plotted in the graph above with number of trials on the x axis and coherence score on the y axis, additionally the optimal topics is calculated by finding the trial with the highest coherence score, in this case 7 topics provided the highest coherence score.

### 7. Training Optimal LDA Model:

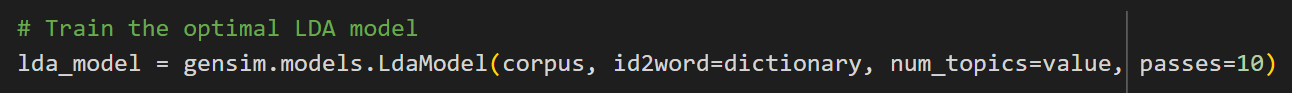


Figure 10

Finally, the finished model is trained and stored using the optimal number of topics.

### 8. Visualizing Topics:

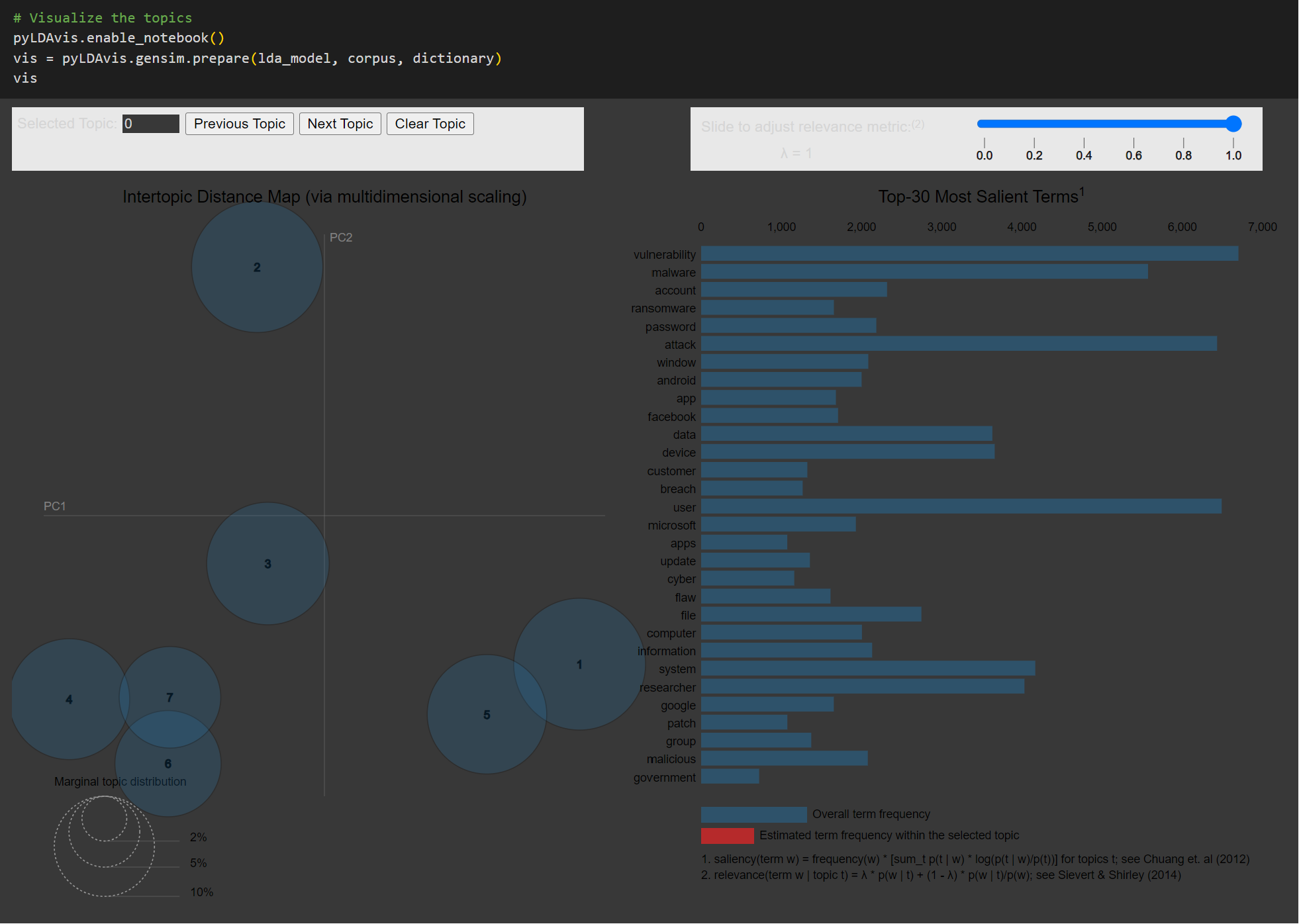


Figure 11

Here we use pyLDAvis to create an interactive visualization of the LDA model, which allows for the exploration of the topics and their associated words. The 30 most salient terms over the entire corpus are plotted on the right hand side. In this context, salience refers to the relative importance of the individual words of a topic, providing an insight to their overall contribution to the corpus.

The intertopic distance map is a visualisation that aids in understanding the relationship between the topics extracted from the corpus by the model. Each topic is represented by a circle, in this case 7 circles. The distance between the topics represents the similarity between the topics, the closer the topics, the more similar their word distributions are. The circle diameter is representative of the overall importance or salience of said topic and the words it contains to the overall corpus. The map is created using dimensionality reduction techniques such as MDS (multidimensional scaling).

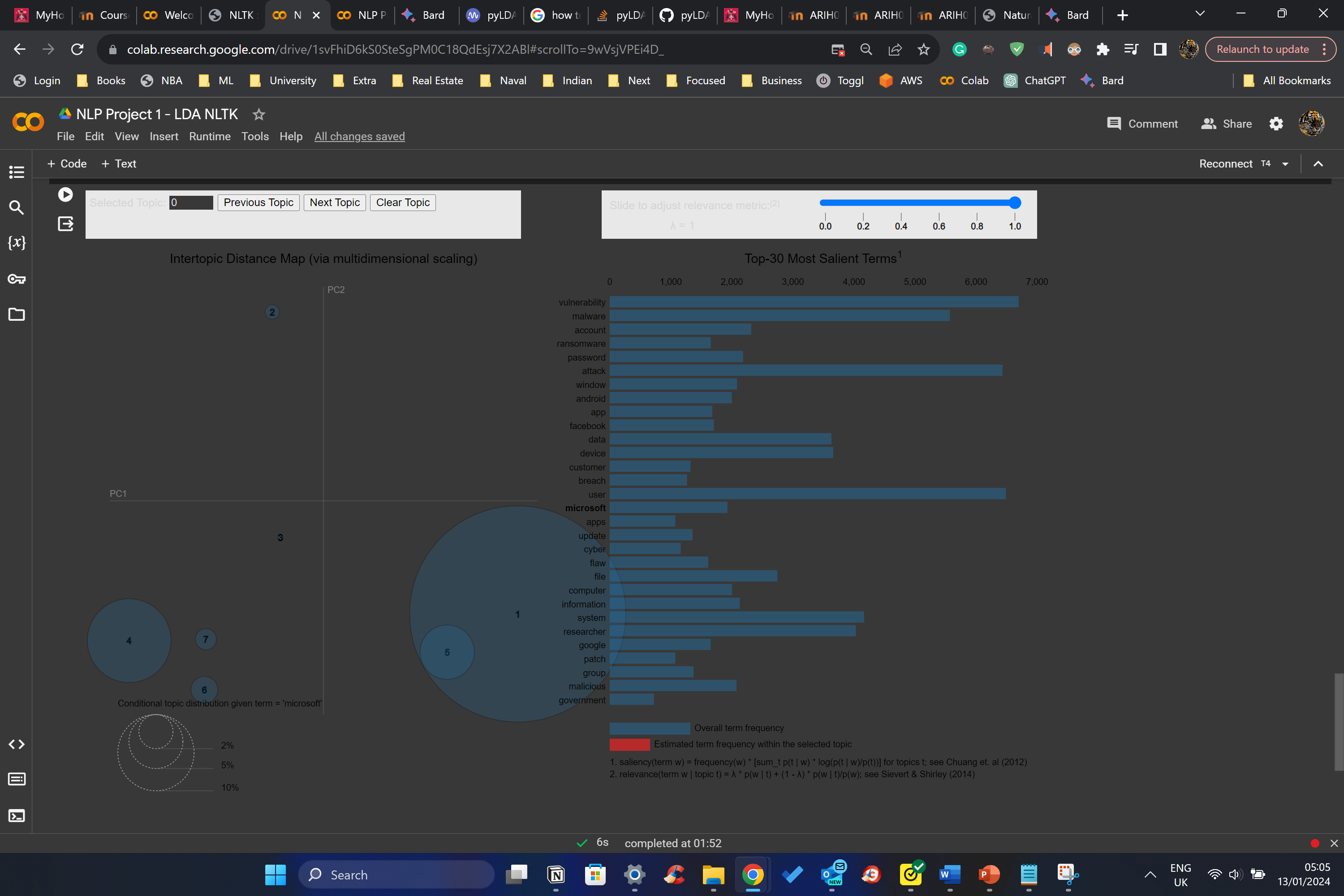


Figure 12

Furthermore, you can click on each word from the list on the right to find the intertopic distance map of each word.

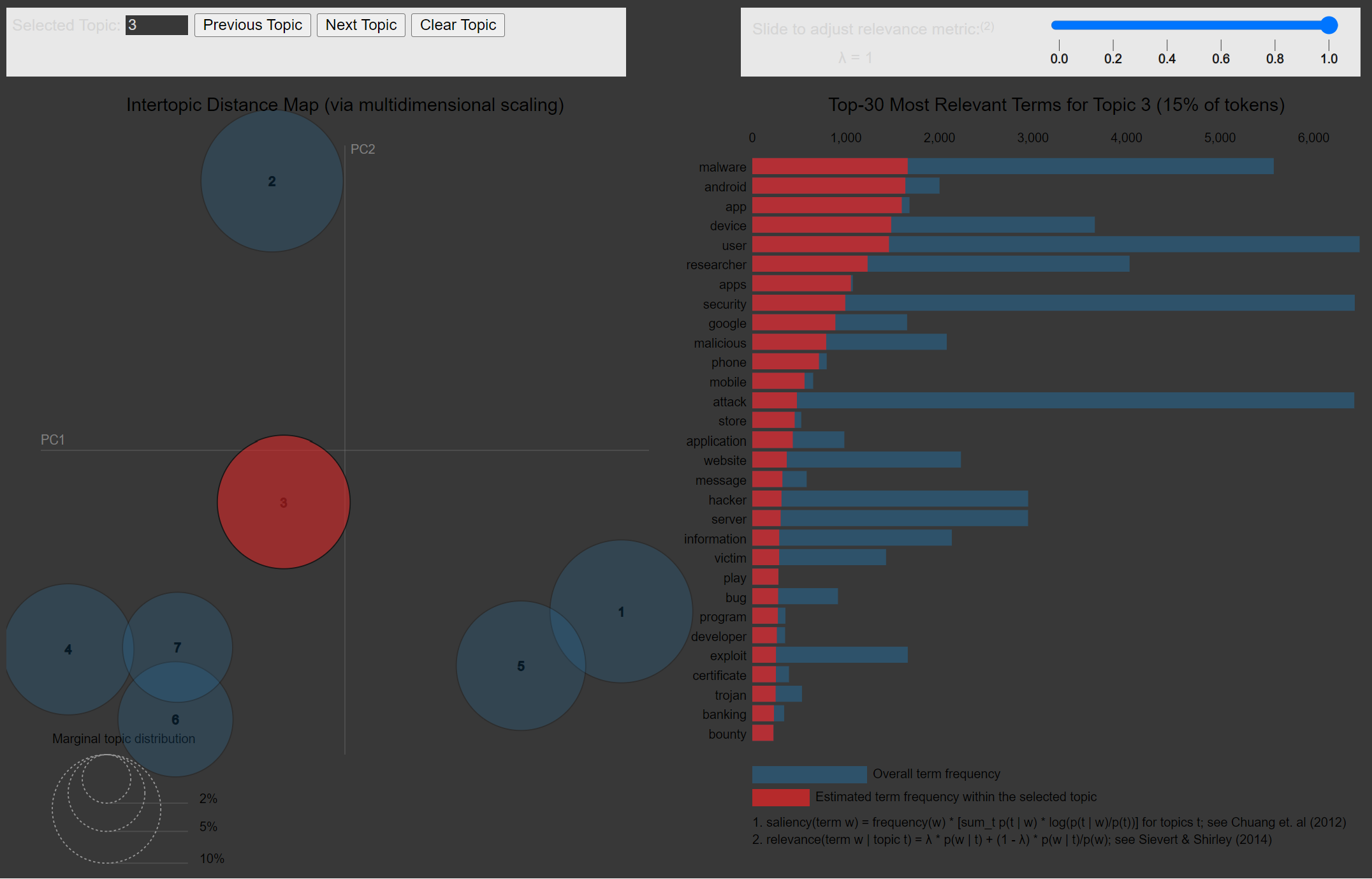


Figure 13

Additionally, if a topic is clicked it displays the most relevant terms of said topic and their frequency over the overall frequency of said words in the entire corpus.

## Conclusion

This paper has hopefully accomplished its purpose of outlining what LDA is, how it works, how academia and the industry has arrived at LDA, what the future of topic modelling is, and how to implement, optimise, and visualise Latent Dirichlet Allocation models. Hopefully, this paper has left you confident in your ability to be able to apply topic modelling to a multitude of real word tasks such as document clustering, text summarisation, market research and customer segmentation, trend analysis and forecasting, least information retrieval and search, etc.

## Figures

Figure 1 - Zhang, J., Lu, J., Zhu, X., & He, J. (2014). Text mining of open-ended questions in self-assessment of university teachers: An LDA topic modelling approach. https://www.researchgate.net/figure/Schematic-of-LDA-algorithm\_fig1\_339368709. (Accessed January 13, 2024)

Figures 2-13 – Can be found in NLP\_Project\_1\_LDA\_NLTK.ipynb

## References

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research, 3(4-5), 993-1022. https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf

Deerwester, S., Dumais, S. T., Furnas, G. W., Harshman, R., Landauer, T. K., Lochbaum, K., & Streeter, M. E. (1990). Indexing by latent semantic analysis. Journal of the American Society for Information Science, 41(6), 391-406. https://www.sciencedirect.com/science/article/pii/S0022000000917112

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. Proceedings of the National Academy of Sciences, 101(1), 5228-5235. https://www.pnas.org/doi/10.1073/pnas.0307752101

Salton, G., & McGill, M. J. (1986). Introduction to modern information retrieval. McGraw-Hill.

Turney, P. D. (2000). Learning algorithms for keyphrase extraction. National Research Council Canada.

Hofmann, T. (2001). Unsupervised learning by probabilistic latent semantic analysis. Machine Learning, 42(1-2), 177-196. https://doi.org/10.1023/A:1007617005950

Gokul, K. P., & Sundararajan, M. (2021). An efficient nonnegative matrix factorization topic modeling for business intelligence. In Proceedings of the First International Conference on Computing, Communication and Control System, I3CAC 2021 (pp. 127-132). EAI. https://eudl.eu/doi/10.4108/eai.7-6-2021.2308681

Blei, D. M., & Lafferty, J. D. (2009). Dynamic topic models. https://dl.acm.org/doi/10.1145/1143844.1143859

Kiela, D., Xu, C., & Conneau, A. (2020). BERT-topic: Pre-training for topic modeling with bert. https://arxiv.org/abs/2305.09329

Yan, B., Xu, H., & Yang, S. (2013). Gibbs sampling for the Dirichlet multinomial mixture model with informative priors on hyperparameters. https://www2.stat.duke.edu/~mw/MWextrapubs/West1992alphaDP.pdf

Blei, D. M., & Lafferty, J. D. (2006). Topic models. https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf