Developing & Comparing Various Topic Modeling Algorithms on a Stack Overflow Dataset

RAPHAEL IBRAIMOH[1], KWAME OFOSU DEBRAH[2], EMMANUEL NWAMBUONWO[3]
[1, 2, 3] *School of Science, Engineering & Environment, University of Salford*

Abstract- ***This research extracts and compares coherent and instructive topics from Stack Overflow, a tech and programming community. Scraping questions, summaries, and tags, exploratory data analysis, rigorous pre-processing, and topic models to find latent topics are the study's main steps. LSA, LDA, and BERTopic are popular topic models. To achieve the best models for each algorithm, base model hyperparameters were tweaked and refined. Then, each algorithm's models were compared for performance and accuracy using coherence score, topic distinctiveness, and different visualization techniques to examine semantic separation. Each technique was tested to see how well it handled different data dimensions. The comparison study showed that BERTopic was the best topic model, achieving more granular and semantically meaningful categorizations through improved semantic comprehension, topic distinguishability, and topic extraction coherence. This research shows how advanced topic modelling may extract nuanced insights from text data, giving a complete process from data acquisition to subject categorization. The results demonstrate BERTopic's ability to decipher complicated textual relationships and generate coherent words for varied themes. Thus, this research improves information retrieval and user experience on online community platforms like Stack Overflow by using advanced natural language processing models.***

Indexed Terms- ***BERTopic, Latent Semantic Allocation, Latent Dirichlet Allocation***

I. INTRODUCTION

The emergence and steady rise of community-based user-generated question-answering sites where users gain and share knowledge has resulted in a vast and ever-growing pool of data. Some examples of these websites include Stack-overflow, Quora, Reddit, and Pinterest where users are encouraged to learn from one another in a crowd-sourcing manner where a question is answered by multiple users and votes are used to rate the relevance of the answer to the question asked.

These data are a valuable resource that can be leveraged to enhance user engagement and further encourage knowledge exchange. However, the growing volume of users, questions, and answers posed on these platforms makes it increasingly cumbersome and time-consuming for users to locate the content they are interested in. Considering Stack overflow as an example, As of November 2022 It had obtained a total of 23 million questions for the year 2022 [3]. This necessitates a robust system to organize and manage this content, a function often fulfilled by topic modelling and question tagging.

While LSA and LDA have been instrumental in the foundational development of topic modelling, these models often face limitations in terms of coherence, and interpretability. In numerous instances, these models exhibit high levels of topic overlap and struggle with distinguishing between subtle nuances in semantic contexts, thereby impairing the clarity and richness of derived insights. The emerging question is the exploration of whether more advanced models, incorporating state-of-the-art NLP techniques, can surmount the inherent limitations of traditional models and render more coherent, diverse, and interpretable topics in varied text datasets.

This project's relevance and applications, primarily in automated content management, information retrieval, user experience enhancement, and the broader field of natural language processing cannot be overemphasized.

The following are the key reasons why this study is important:

I. By comparing these models, this study can unveil new insights into the capabilities and limitations of different topic modelling techniques .
II. The study will contribute to a deeper understanding of how well different models capture the semantic essence of documents and represent them coherently, impacting the development of more semantically aware models in the future.
III. Enhanced Information Retrieval: Correctly assigning topics to questions or statements will improve the efficiency and effectiveness of information retrieval. Topics serve as a concise summary of the content, query, or question, enabling users to find the information relevant to their searches or closely related to the information they are looking for.
IV. Improving User Experience: User satisfaction and engagement on these websites can be significantly improved when the time spent searching for relevant information is greatly reduced due to a streamlined process of content discovery.
V. Reducing human error and bias: Enhanced topic modelling systems can reduce the inconsistencies, errors, and biases that arise from human judgement, providing a more objective and consistent tagging or topic assigning method.

II. LITERATURE REVIEW

[11] describe topic modelling as a probabilistic based statistical approach that serves as a valuable tool for deciphering latent themes or topics within the content of documents. Finally, [4] describes topic modelling as a common statistical technique for extracting latent variables from voluminous datasets.

Topic modelling isn't confined to academic exercises. It shows significant potential for diverse professionals - from social scientists to business analysts, serving to extract useful information from the text datasets that are already available. Transforming them into insights and actionable knowledge. The overarching objective of these researchers is to gain a deeper knowledge of various things in the world through the written words of others [12].

At the core of topic modelling lies a probabilistic framework, built upon the premise that every document is a mixture of various topics, but these topics manifest in differing proportions within a given document and aims to use topic modelling to discover the underlying set of topics that make up the documents in an efficient and accurate manner [1]. The topic model does this by revealing the topic distribution in each document and the word distribution in each topic [1][10]

According to [7], the use of topic modelling has been employed widely to read and understand themes of data embedded in texts and literature. As there are more and more electronic document archives, it is necessary to employ new techniques or tools that deal with automatically categorizing, searching, indexing, and viewing enormous collections to preserve them successfully.

Topic modelling, initially a subset of generative probabilistic modeling, originated in the 1980s, as described by [7]. The foundational approach of topic modelling is rooted in the use of probabilistic models, constructed under the presumption that terms within a corpus of text occur independently. Consequently, a term's relevance is weighed by the frequency of occurrences of that term in response to a specific query. This method of weighing term relevance is vital for evaluating the importance of a text within a document.

Subsequently, it was realized that words used frequently in the texts of documents could not be used to efficiently differentiate these documents from the other texts of a corpus if their occurrence frequency were also high in the other documents. Specifically, words that appear repeatedly within individual documents but are also pervasive across the collection lose their distinguishing value. This realization catalyzed the inception of the term frequency-inverse document frequency (TF-IDF) paradigm. TF-IDF is formulated as the product of two components: *tf* and *idf*. which is the product of the term count or frequency specific to a given document and the inverse document frequency given by the logarithmically scaled inverse fraction of the total documents that contain the term. '*t'*.
The *idf* is expressed mathematically as:

$$idf(t) = \log\left(\frac{N}{1+df(t)}\right) \quad (1)$$

Where N is the total number of documents in the collection, and $df(t)$ is the number of documents containing the term t. Through this mathematical rendition, terms that are prevalent in one document but sparse across the entire corpus are given higher weights, thereby making them important in characterizing the document's content [9]

The *tfidf* was successful in identifying words that are important and distinctive within a collection of documents and differentiating documents in a collection based on the words they contain, but faced certain challenges such as Synonymy and Polysemy, Contextual Ambiguities, and its propensity for surface level analysis focusing predominantly on term frequency and global distribution.

Considering the limitations of the *tfidf*, Latent Semantic Indexing (LSI) or Latent Semantic Analysis (LSA) was Developed by Deerwester et al. in 1990. LSA ventured beyond raw term-document associations, leveraging singular value decomposition to reduce the dimensionality of the term-document space and thereby capture latent semantic structures. This was pivotal in alleviating challenges like synonymy, polysemy, and other issues ingrained in TF-IDF, paving the way for more sophisticated and nuanced models in the realm of text analysis [4].

Latent Semantic Indexing works by factorizing the *tf-idf* matrix using Singular Value Decomposition (SVD), a mathematical technique that is used to decompose a matrix into three separate matrices. The resulting matrices from the SVD are used to find a linear subspace within the original high dimensional space of the *tf-idf* matrix. This linear subspace represents the most significant variations present in the corpus

Mathematically, given a term-document matrix A, which is of dimensions $m \times n$ (where m is the number of terms and n is the number of documents), we aim to decompose A into three matrices:

$$A = U\Sigma V^T \quad (2)$$

Where:

- U is an $m \times r$ orthogonal matrix. The columns of U are called the left singular vectors (representing terms).
- Σ is an $r \times r$ diagonal matrix. The diagonal elements, known as the singular values, are non-negative and are usually presented in decreasing order.
- V^T is an $r \times n$ orthogonal matrix. The rows of VT are the right singular vectors (representing documents).

The approximated matrix A retains most of the important semantic information from the original matrix A, but with reduced dimensionality.

While LSI was effective for compressing large corpora and capturing semantic relationships, a combination of limitations such as handling complex semantics, scalability, and interpretability.

Considering the challenges of the LSI, there was a push towards alternative approaches that could overcome these hurdles. One such notable shift was towards generative probabilistic models. Hofman, in 1999, proposed utilizing generative models which offer a different perspective: instead of transforming the data into a new space (as with LSI), generative models aim to describe the underlying data generation process. In essence, the objective is to reverse engineer the generation of a document, discerning its thematic elements. The probabilistic LSA (pLSA) model also referred to as the 'aspect model', Introduced as an improvement on the previous LSI model. incorporates probabilistic principles and focuses on representing documents as mixtures of topics. The goal of pLSI is to address some of the limitations of traditional LSI by providing a more flexible and probabilistic approach to understanding the underlying structure of text data.[6]. This marked the beginning of applying probabilistic methods to topic identification. Like all models, generative probabilistic models have their own set of limitations which includes scalability, overfitting, absence of a generative process for document-topic distributions, and the lack of a prior [3].

In 2003, Blei, Ng, and Jordan developed the Latent Dirichlet Allocation (LDA) which addresses and builds upon the shortcomings of the pLSI. The evolution from pLSI to the Latent Dirichlet Allocation (LDA) represents a significant

advancement in topic modeling and probabilistic modeling for natural language processing. LDA overcomes the limitations of the pLSI by introducing a probabilistic mechanism for determining the mixture of proportions of topics in each document using the Dirichlet distribution. The Dirichlet distribution is a probability distribution over multinomial distributions (or mixture proportions), which makes it ideal for modelling topic proportions in documents. In LDA, the data is structured into three levels – document, topic, and word. It models each document as a mixture of topics, and each topic as a mixture of words. This structure allows LDA to capture intricate word co-occurrences across documents and elucidate underlying topics. The key insight is that the mixture proportions of topics in documents are generated from a Dirichlet distribution. LDA introduced collapsed Gibbs sampling as an efficient method for inference.[2]

Researchers continued to refine and extend this technique, and variational inference methods were developed to approximate the posterior distribution of latent variables in a more scalable One of the strengths of LDA is the interpretability of its output. By capturing the thematic structure of a corpus, it provides meaningful topics, each represented as a collection of words with associated probabilities. Some of the limitations of the LDA include manually determining the number of topics, handling short texts, and it requires a lot of pre-processing.

Subsequently, with the advancements of neural networks, researchers were able use embeddings that serve as dense vector representations to capture the essence of words, sentences, or entire documents, unlike traditional methods that utilize sparse representations like one-hot-encoding. Some examples of these models are Word2vec by [8] Doc2Vec, Glove (Global Vectors for Word Representation) by [10] BERT (Bidirectional Encoder Representations from Transformers) by [5] employs a multi-layer bidirectional Transformer architecture. It creates contextually rich embeddings by considering the full context of a word within a sentence, using bidirectional attention to capture both the preceding and subsequent words' meaning and relationships. [5].

While context free models like word2vec and GloVe generate single word embeddings that are context independent, BERT model generates embeddings that enable us to have multiple vector representation for the same word depending on the context in which the word is used.

In comparison, embeddings are an improvement upon traditional modelling methods like LDA. They not only provide a way to map word distribution across documents but also enhance computational efficiency by reducing dimensionality.

The neural topic models are also not without limitations themselves as they are limited by their reliance on the volume and quality of data required, the training time and computational resources necessary for these models are quite much, and the model size of embedding based models can be large and memory intensive.

III. METHODOLOGY

Data Extraction

The target site was identified as a rich source of questions relevant to the research objectives. The scope of the scraping included the questions, along with any associated metadata, such as existing tags or categories, that might contribute to the analysis. Before proceeding with web scraping, careful attention was paid to the target site's terms of service and applicable laws and regulations. The scraping was conducted in a manner that respected the site's policies and legal requirements, ensuring responsible data collection.

Specific web scraping tools and libraries mentioned below were employed to extract the required data. The methodology was designed to minimize the impact on the site's performance, including considerations like rate limiting and respectful user-agent declaration.

Beautiful Soup from BS4: Used for parsing HTML and XML documents

Requests: This library was employed to handle the HTTP requests to the website

Pandas: Pandas' library was used to clean, convert the data to a parquet file, and then a dataframe, and finally read the data.

Data Analysis

Data Structure and Format: The data collected using Beautiful Soup and Requests was structured into a parquet format for efficiency.

Quality Assurance: Quality checks were performed to ensure that the scraped data was accurate, consistent, and free from duplication or corruption. This involved validation routines and manual inspections as necessary.

Documentation: Finally, the entire data collection process, including the tools, parameters, and challenges encountered were thoroughly documented to ensure transparency, replicability, and adherence to best practices in research methodology.

Data Description

This section will provide a comprehensive insight into the structure and the components of the collected dataset.

The scraped data consists of four main columns, each of which plays a vital role in the analysis and modelling phases.

	votes	summary	question	tags
0	27071	\nWhy is processing a sorted array faster than...	\r\n In this C++ code, sorting ...	java
1	25851	\nHow do I undo the most recent local commits ...	\r\n I accidentally committed t...	git
2	20256	\nHow do I delete a Git branch locally and rem...	\r\n Failed Attempts to Delete ...	git
3	13671	\nWhat is the difference between 'git pull' an...	\r\n What are the differences b...	git
4	12704	\nWhat does the "yield" keyword do in Python?\n	\r\n What is the use of the yie...	python

Figure 1:The first five rows of the dataset

The figure 1 explains as follows:

I. Votes: The votes represent a quantifiable measure of the community's response to each question. They reflect the popularity or relevance of the questions and can be considered as an indirect measure of the importance or interest level of the underlying topics. This quantitative attribute adds a valuable dimension to the analysis, enabling potential insights into correlations between topics and community engagement.

II. Question Summaries: Alongside the complete questions, the dataset also includes concise summaries of each question. These summaries encapsulate the essence of the questions, offering a compact representation of the content. The inclusion of summaries adds an additional layer to the analysis, allowing for comparisons between detailed and summarized views of the data.

III.Questions: The main body of the questions constitutes the core of the dataset. This includes the full text of the questions posted on the target site. These texts are rich in content and present the primary source for extracting topics and themes for auto-tagging.

IV. Tags: The tags are the topics related to the questions in that column

Data Format: The data has been structured in a tabular format, with each row representing a unique question, and columns corresponding to the votes, question summary, question text, and tags. This structured representation ensures a streamlined pre-processing and analysis phase.

Data Quality: The quality of the scraped data was assessed to ensure accuracy, completeness, and consistency. The dataset was found to be well-aligned with the research objectives, providing a balanced combination of textual and numerical information necessary to achieve the aims and objectives set out to be achieved.

Data Preparation

The following will be carried out on the dataset using pandas:

- Removing the escape sequence characters
- Removing all the stop words and punctuations
- Lowercasing all the words in the text

Data Transformation and Feature Engineering

The following NLP feature engineering techniques were carried out on the data to effectively carry out exploratory data analysis and to create a pre-processing pipeline to prepare the data for the LDA and LSA algorithms.

- Tokenization: This involves splitting the text into individual words or tokens. It simplifies the text by reducing it to its most basic components.
- N-grams: N-grams are sequences of adjacent words with length 'N' e.g. bigrams are 2-word sequences, and trigrams are 3 – word sequences,

and quad grams are 4-word sequences etc.. This helps to capture more context and help the model better understand the meaning of the text. It identifies patterns and relationships in the text that may not be obvious when considering individual words or phrases.

- Named Entity Recognition: This involves identifying and categorizing named entities in text such as locations, items, companies. This helps understand the text better during the exploratory data analysis and can also be used to build more advanced features for the model.
- Count Vectorizer: This is a technique used to convert a collection of documents into a matrix of token counts. It is a useful step in machine learning algorithms that require numerical input. For each document, we count the occurrences of each word in the and these counts form the rows of the matrix, .
- TF-IDF: Term Frequency Inverse Document Frequency(TF-IDF) is a technique that helps to identify the most important words in a corpus by assigning weights to words based on their frequency in the document and in the corpus. This helps the model identify important and distinguishing terms, words or keywords within the corpora.
- Part-of-Speech (POS) tags: These are labels assigned to words that indicate their grammatical categories. Checking POS tags helps to understand the structure and meaning of a text based on the surrounding words

Exploratory Data Analysis

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 4 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   votes     10000 non-null  object
 1   summary   10000 non-null  object
 2   question  10000 non-null  object
 3   tags      10000 non-null  object
dtypes: object(4)
```

Figure 2: Data description

Figure 2 describes the dataset to be used for the initial model. It shows the number of columns, the column names, the number of non-null values and the data types.

10,000 unique observations were collected from the website and there were no null values.

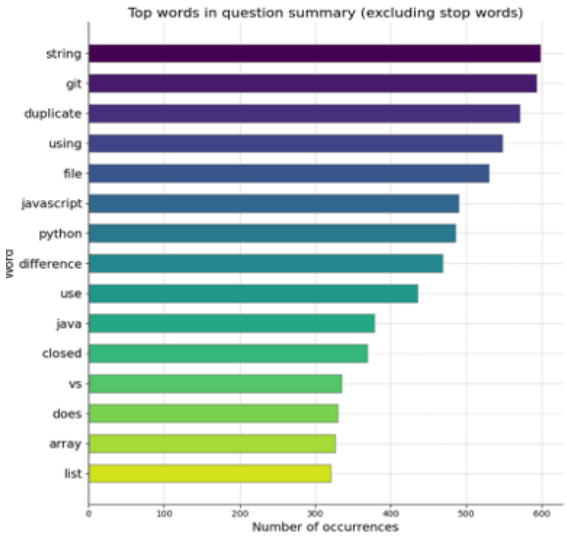

Figure 3: Wordcount of the top 15 words in the summary column

Figure 4: Wordcloud of the summary column

Figure 3 shows the most frequent words after removing the stop words from the documents. From the data above, the most frequent in descending order are:String, git, duplicate, using, file, python, javascript, difference, use, java, closed, vs, does, array, and list. This is further supported by the wordcloud in figure 4

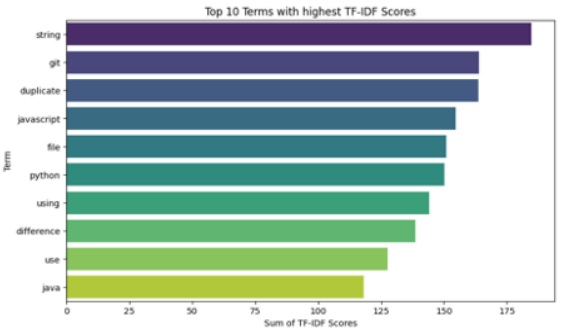

Figure 5: Top 10 terms with the highest *dfidf* score

Figure 6: Wordcloud of the terms with the highest *tfidf* scores

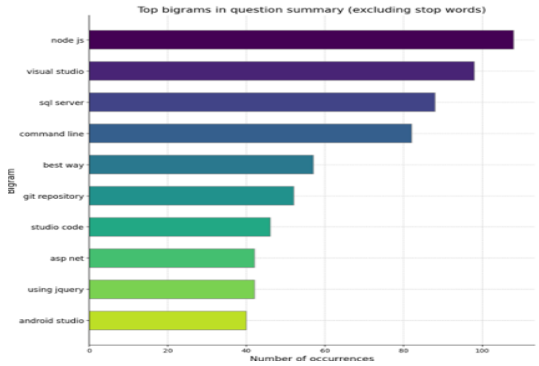

Figure 7: 10 most frequent bigrams in the corpus

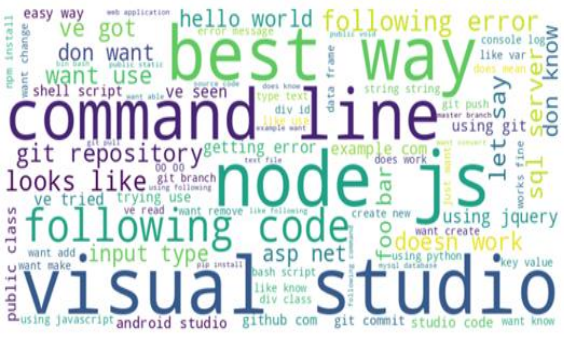

Figure 8: Wordcloud of the bigrams in the corpora

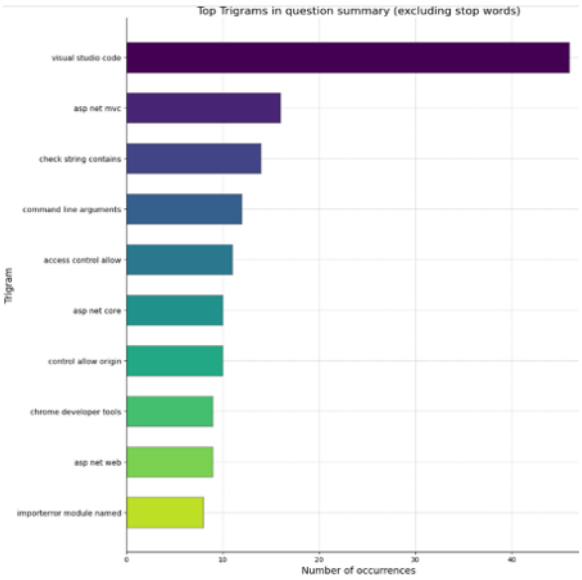

Figure 9: Most common trigrams

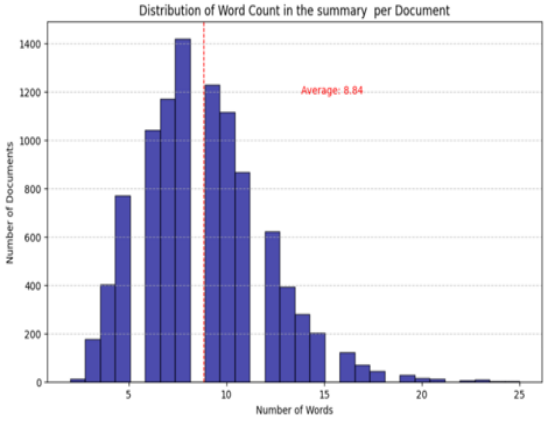

Figure 10: Words count distribution of the question summary

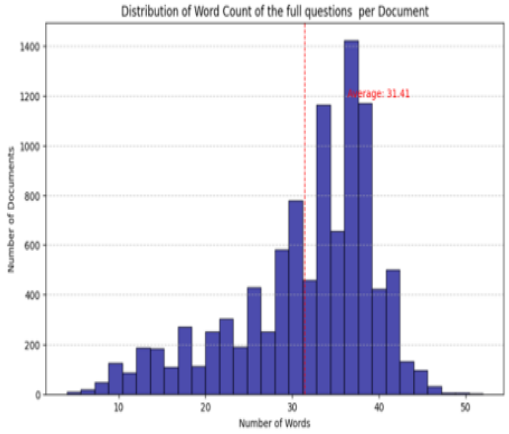

Figure 11: Word count distribution of full questions

Figure 10 indicates that the summary column has an average of 8.8 words per sentence across the corpora and figure 11 shows that the question column has an average of 31.41 words per document. The summary provides a succinct encapsulation of the full question column which is more detailed and involves more complex constructions.

TAGS

To get an idea of the distribution of the topics, the tags column will be observed to determine how the questions are distributed.

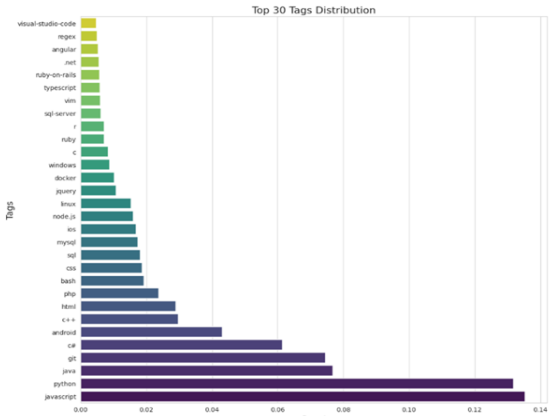

Figure 12: Barchart of the normalized distribution of the tags

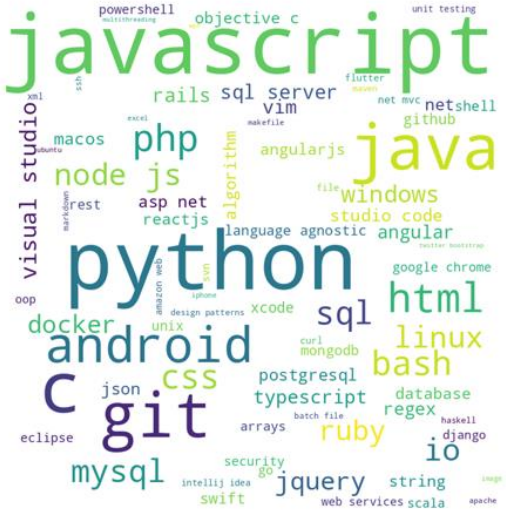

Figure 13: Wordcloud of the tag's column in the dataset

Figures 12 and 13 show the prevalence of tags within the corpora. From figure 12, It is observed that python and javascript account for over 13% each of all the questions, and the top 6 tags which are javascript, python, git, java, git, and c# account for roughly 50% of the dataset. It can thus be inferred that the dataset exhibits a significant degree of concentration, with a substantial proportion of the observations dominated by a small number of tags. This showcases a strong focus on a few topics within the dataset.

IV. LIMITATION AND BIASES

Given the existence of 434 tags in total within this dataset of 10,000 observations, the pronounced concentration of observations within a few tags suggests that many tags are likely associated with a relatively small number of observations. This could imply a long-tail distribution of tags, where the majority are infrequent, and a minority are prevalent, reflecting the varied and specialized nature of topics within the corpora.

V. SELECTING THE RIGHT TOPIC MODELLING TECHNIQUE

Selecting the appropriate topic modelling procedure is crucial for the extraction of useful statistics and characteristics from a dataset. Recent topic modelling techniques perform noticeably better than earlier algorithms, they however still need to be tuned and optimized to deliver accurate results. This will involve considering several factors related to the data, goals of the project, and the characteristics of the different algorithms. Multiple topic modelling techniques have been developed for use with more specialized data relationships and structures, such as brief texts, long-term sequential data, strongly correlated data, and data with complicated structural links, as was previously mentioned. In order to create a topic modelling technique that best fits the needs of a specific project, researchers starting a text analysis project must comprehend the differences between different algorithms as they relate to the project goals [7].

Findings

After training and tuning the hyperparameters of the three algorithms, the following results were obtained.

Algorithm	Coherence score	Topic Diversity	Noise	Interpretability
LSA best	0.32	0.17	A lot of Unrelated words are clustered within a topic	A lot of topics are not interpretable
LDA best	0.40	0.86	Some unrelated words are clustered within a topic	Quite interpretable
BERTopic best	0.51	0.99	Little noise within topics, Noise/unrelated words are all captured within one cluster	Very interpretable

Table1 1: The evaluation results of the best model from each of the algorithms.

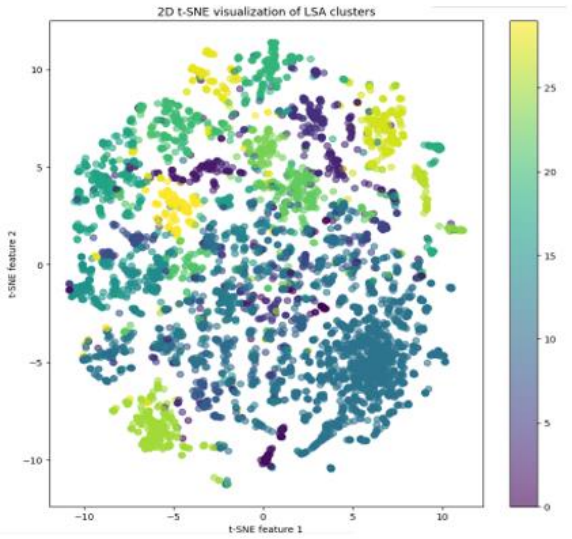

Figure 14: t-SNE 2D visualization of the LSA topics

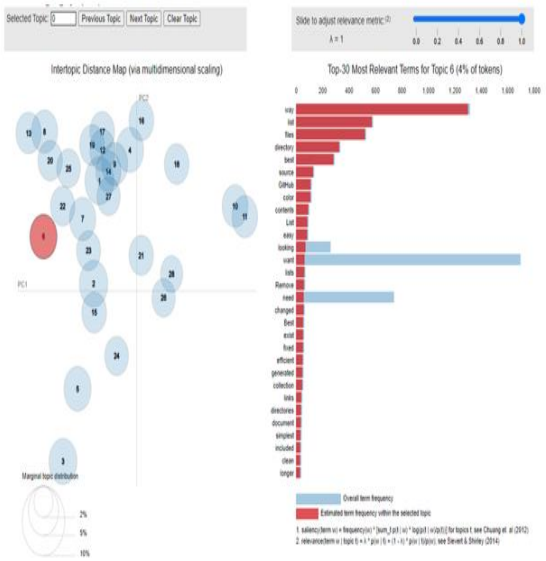

Figure 15: Interactive visualization with pyLDAvis

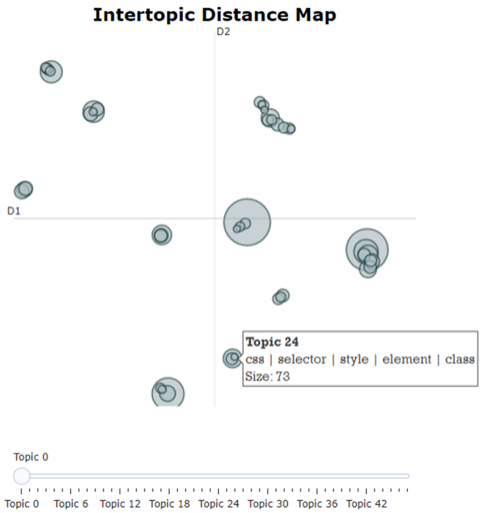

Figure 16: Interactive Intertopic Distance Map showing the visualization of the BERTopic model

VI. DISCUSION

1. Coherence Score:

Based on the coherence scores, BERTopic is the most coherent, followed by LDA and then LSA. This suggests that BERTopic is likely generating the most cohesive and semantically meaningful topics.

2. Topic Diversity:

Based on the topic diversity scores, BERTopic has the highest topic diversity, indicating that it can distinguish between a wider range of themes or

subjects. LDA has a high diversity score at 0.86, whereas LSA has a very low diversity score indicating that the LSA algorithm has difficulty distinguishing between different topics.

3. Noise:

- LSA has the most noise mixed within coherent topics words, which means that the topics it generates contains more unrelated or noisy terms than the others, making it harder to interpret and use for downstream tasks.
- LDA has some noise within topic words, suggesting that while it is an improvement over LSA, it still generates topics with some noise or less precision.
- BERTopic has little noise within topic words, but however clusters all the noise into the '-1' row making it easier to categorize, and separate from other topics, indicating that it produces topics that are relatively clean and more focused.

4. Granularity:

- Based on the topic words, LSA generates more generalizable topics, which means the topics tend to be broader and less specific.
- As observed from the topic words, LDA provides more precise topics but still contains some noise, making it somewhat less precise compared to BERTopic.
- BERTopic is the most precise in terms of topic granularity, meaning it generates highly specific and well-defined topics. The words are well within the topic environment

5. Intertopic Distance and Overlap:

As observed in figures 14, 15, and 16,

- LSA: The t-SNE showed high overlap and less clear boundaries between topics, reflecting the low topic diversity score.
- LDA: Despite having decent topic diversity, there still exists a notable amount of overlap of topics, as seen in the pyLDAvis visualization.
- BERTopic: The intertopic distance map shows a decent amount of space between topics with some occurrence of topic bubbles within each other, potentially indicating hierarchical relationships or some degree of overlap.

The comparative results indicate that BERTopic outperformed both LDA and LSA when the base model was trained and after hyperparameter optimization as indicated. The BERTopic not only has a higher coherence score, but also makes more sense semantically to the trained eye that can place the words into their necessary topics.

The BERTopic model has the highest coherence and diversity scores, indicating it is the best model among the three for extracting meaningful and distinct topics from this dataset. LDA also performs reasonably well, especially regarding topic diversity. In contrast, LSA has the lowest scores for both coherence and diversity, suggesting that it might be less effective at generating meaningful and distinct topics in comparison to LDA and BERTopic.

This superiority can be attributed to the underlying architecture, which leverages transformer based embeddings which allow them to produce more nuanced and context-aware embeddings.

CONCLUSION

In conclusion, the evaluation of LSA, LDA, and BERTopic reveals substantive distinctions in their performance and approach towards topic modeling. LSA, despite its streamlined and efficient algorithm, shows limitations in coherence, diversity, and clear delineation of topics, yielding high overlap and less precise boundaries between topics as shown in the t-SNE visualization. This is corroborated by the low coherence and topic diversity scores, highlighting its weakness in achieving high semantic relation and distinct topic segregation. LDA, on the other hand, employs more intricate methodologies, utilizing sophisticated Dirichlet priors and tuned alpha and beta values, leading to more coherent and diverse topics. Nonetheless, it still experiences some degree of overlap in topics, suggesting room for improvement in optimizing intertopic distinctions and clarities.

BERTopic, with its density-based clustering approach, outshines the other models in most aspects, achieving high coherence, nearly perfect topic diversity, and demonstrating discerning intertopic distances. Its utilization of specific parameters

controlling granularity and the minimum size of topics reflects a meticulous and refined approach to address the subtleties and complexities in topic modeling. However, some occurrences of topic bubbles within each other hint at potential hierarchical relationships or overlaps, indicating an intricate interplay of topics within the model. The comparative analysis of these models underscores the imperative of considering the balance between model complexity, interpretability, and the precision of topic delineation based on the specific requirements and constraints of individual analytic endeavors.

REFERENCES

[1] Alghamdi, R. and Alfalqi, K. (2015) 'A Survey of Topic Modeling in Text Mining', *International Journal of Advanced Computer Science and Applications*, 6(1), pp. 147–153.

[2] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research 3, 3*(2), 993-1022.

[3] David, C. (2023, January 17). *Stack Overflow Growth and Usage Statistics (2023)*. Retrieved from usesignhouse: https://www.usesignhouse.com/blog/stack-overflow-stats

[4] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990, Sep). Indexing by Latent Semantic Analysis. *Journal for the American Society for Information Science, 41*(6).

[5] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *North American Chapter for the Association of Computational Linguistics*

[6] Hofmann, T. (1999). Probabilistic latent semantic indexing. *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, (pp. 50–57).

[7] Kherwa, P. and Bansal, P. (2020) 'Topic Modeling: A Comprehensive Review', *EAI Transactions on Scalable Information Systems*, 7(24), pp. 1–16.

[8] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space.

[9] Salton, G. (1986). Recent trends in automatic information retrieval. *Wikisym05: Int'l Symposium on Wikis Palazzo dei Congressi.* Pisa Italy: Association for Computing Machinery New York, United States.

[10] Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (pp. 1532–1543)

[11] Prabha, S., & Sardana, N. (2023). Question Tags or Text for Topic Modeling: Which isbetter. *International Conference on Machine Learning and Data Engineering*, 2172–2180.

[12] Ramage, D., Hall, D., Nallapati, R., & Manning, C. D. (2009). Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. *Association for Computational Linguistics* (pp. 248–256).