**Web Science Coursework – Topic Modelling**

1. **Introduction**

Topic Modelling is a popular technique in natural language processing that aims to uncover the underlying themes present in a collection of text documents. With the advent of social media platforms such as Twitter, vast amounts of data have been generated in the form of short text messages, or tweets. Analysing this data can be a challenging task due to the limited character count, non-standard vocabulary, and informal language used in tweets. The objective of this project is to apply topic modelling to a Twitter dataset consisting of around 10,001 tweets from the UK. The dataset includes columns such as Twitter ID, username, tweet text, newsworthy score, and quality score.

The project aims to investigate various challenges of topic modelling on the Twitter dataset, including the impact of data pre-processing, the number of topics used, and the determination of the optimal number of topics. Additionally, evaluation metrics are used to assess the quality of the generated topics. The coursework also explores the consequences of using a data grouped by single-pass clustering and how it impacts the quality of generated topics and evaluate using relevant metrics. The grouped Twitter dataset contains columns such as Grouped ID, username, tweet text, newsworthy score, and quality score. To perform topic modelling, the Latent Dirichlet Allocation (LDA) model is employed, the results obtained from the Twitter dataset and the grouped Twitter dataset are compared, and any observed differences in performances are discussed.

1. **Topic Modelling – Tweets Data**
2. **Generic outline for topic modelling of Twitter Data**
3. **Data Collection**

We have been provided with a Twitter dataset file named tweets.csv, which has already been collected and is ready for further processing. This dataset comprises 10,001 tweets from the UK, each containing information such as the Twitter ID, username, text content of the tweet, and corresponding nscore and qscore values of each tweet.

Furthermore, the Twitter dataset was loaded in the following manner to undergo pre-processing:

Graphical user interface, text, application, Word

Description automatically generated

1. **Exploratory Data Analysis (EDA) & Data Pre-processing**

To extract meaningful insights from the dataset, several steps were taken in the Exploratory Data Analysis (EDA) and Data Pre-processing stage.

1. Examined the total number of unique tweets and the total number of unique users who tweeted.

|  |  |
| --- | --- |
| Total number of unique Tweets | 10001 |
| Total number of unique Users | 8029 |

1. Analysed the top 10 users who have tweeted the most by extracting their tweet counts.

**Table

Description automatically generated**

1. Investigated the top 10 users’ data whose tweets text have both a high-quality score and are considered to be newsworthy.

**Application

Description automatically generated with medium confidence**

1. Calculated the total word count for each document by determining the length of the document in terms of the number of words.

Graphical user interface, table

Description automatically generated with medium confidence

1. Studied the Statistics of the Data for tweets/documents, newsworthy score, and the quality score.

**Tweets/Documents Statistics**

|  |  |
| --- | --- |
| Longest Tweet Length | 100 |
| Shortest Tweet Length | 4 |
| Average Tweet Length | 23.437 |

**Tweets Newsworthy Score Statistics**

|  |  |
| --- | --- |
| Average Newsworthy Score | 0.606 |
| Minimum Newsworthy Score | -7.033 |
| Maximum Newsworthy Score | 5.26 |
| Total Number of Newsworthy Score | 10001 |

**Tweets Quality Score Statistics**

|  |  |
| --- | --- |
| Average Quality Score | 0.597 |
| Minimum Quality Score | 0.374 |
| Maximum Quality Score | 0.821 |
| Total Number of Quality Score | 10001 |

1. To comprehend the types of hashtags and mentions present in the text, I extracted their counts along with the respective mentions and hashtags from each tweet/document.

Graphical user interface, application

Description automatically generated

1. To analyze the most popular hashtags among individuals, I have created a plot that showcases the top 10 hashtags that frequently occur in the tweets or documents.

Chart, line chart

Description automatically generated

The line graph displayed above illustrates the top 10 hashtags that occur most frequently in the given corpus of 10,001 tweets. The hashtags that are associated with London and the UK have the highest frequency, with approximately 60 and 45 occurrences respectively. This graph provides us with valuable insight into the most commonly discussed themes in the tweets, and it gives us an idea of the types of topics that we might encounter after applying topic modelling to the dataset. Moreover, the graph suggests that the resulting topics after performing topic modelling will be more closely related to subjects such as UK politics, government, elections, the Russia War, Boris Johnson, and Rishi Sunak.

1. I have plotted the frequency distribution of word counts in the documents to understand the frequency distribution of word counts in the documents which indicates the number of times different word counts appeared in the corpus.

Chart, bar chart, histogram

Description automatically generated

The above graph indicates that around 600-700 of documents have the word counts between 5 to 10.

1. Examined for the presence of null values in the Twitter dataset.

Text, table

Description automatically generated

The achieved results indicate that the dataset does not contain any null values, which is a positive indicator of its quality. This absence of null values not only helps to avoid bias and incorrect results, but also enhances the reliability and accuracy of any insights that can be drawn from the data.

1. The next crucial step in processing the Twitter dataset is to clean the text data. This step is necessary to ensure that the model doesn't learn from noisy data and generates meaningful topics from the tweets. To achieve this, several steps are involved in cleaning the text data. Firstly, HTML tags, URLs, hashtags, mentions, emails, newline characters, spaces, tabs, emojis, punctuations, all types of brackets, and regular expressions are removed. Additionally, special inverted commas are replaced with standard inverted commas, making them easier to detect and remove from the text. Finally, all words in the text are converted to lowercase, and the contractions library was used to expand English words.

Text

Description automatically generated

1. After cleaning the text data, the next step involved removing the stopwords, drop the duplicate tweets, filtering out words whose frequency in the corpus is less than 30, and tokenizing the words. Following this, the single letter words, which are deemed less important, were filtered out and the remaining words were stored in a single list. This list is commonly referred to as the corpus of the documents.

**Text

Description automatically generated**

**Corpus of words – Total length of the corpus is 9116**

**Background pattern

Description automatically generated**

1. To enhance the semantic understanding of the Twitter dataset, I built both Bigram and Trigram models to generate words that appear together frequently, with a minimum occurrence of 5 in the corpus, and a threshold of 100 to ensure the resultant phrases are stronger and more meaningful.

**Bigrams:**

**A screenshot of a computer

Description automatically generated with low confidence**

**Trigrams:**

**A picture containing rectangle

Description automatically generated**

1. After obtaining the bigram and trigram words, they are incorporated into the corpus, while simultaneously eliminating all other parts-of-speech words except for Nouns, Adverbs, Adjectives, and Proper Nouns.

Background pattern

Description automatically generated

1. Eventually, a dictionary was generated, and a new corpus was created, wherein every document was transformed into a bag-of-words format, which returns a tuple comprising of token ID and token count for each word in the document. The token ID is a unique identifier allocated to every word in the corpus, allowing them to be identified distinctly.

**Text, letter

Description automatically generated**

1. **Topic Modelling**

* To facilitate topic modelling, I began by implementing a baseline LDA model to determine the optimal number of topics that would produce the most favorable outcomes. During my experimentation, I varied several LDA parameters, including random state, chunk size, passes, and iterations. Ultimately, I arrived at a set of baseline parameters that proved most effective in generating pertinent topics:

1. Chunk Size - 500
2. Passes - 10
3. Iterations – 50
4. Random State – 50

* During my iteration process, I explored a range of 4 to 12 topics and printed the coherence and perplexity score outputs for each iteration. This selection was made to avoid excessive computational costs, as the system requires a significant amount of time to execute a larger range of topics. To visually represent the output of this process, I generated a graph plotting the number of topics against their respective coherence and perplexity scores, which can be observed in the accompanying images below.

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

Number of Topics vs Perplexity Scores

Number of Topics vs Coherence Scores

* By examining the perplexity and coherence scores, it is evident that as the number of topics increases, the perplexity scores decrease, and coherence scores improves. Thus, a higher number of topics is a more suitable option. Based on this, I have selected a range of 7 to 12 number of topics as the optimal number of topics. This range begins at topic number 7 because the coherence score decreases at topic number 6, indicating that the quality of topics will not be satisfactory at that number. Additionally, the graph indicates that after 11 topics, the coherence scores start to increase steeply, implying that the data is becoming overfitted with an increasing number of topics.
* To determine the optimal number of topics, I utilize a technique called hyperparameter optimization, which assists in identifying the most appropriate hyperparameters for the LDA model, allowing it to extract the most relevant topics from the dataset.
* In order to decide the optimal model hyperparameters, I have selected the following hyperparameter search space:

1. **no\_of\_topics** – [7, 8, 10, 12]
2. **alpha** – ['symmetric', 0.3, 0.5, 0.7]
3. **beta** – [0.3, 0.5, 0.7]

* To optimize the model's performance, I employed a technique known as Grid Search, which exhaustively tests all possible combinations of hyperparameters and reports relevant metrics, namely coherence and perplexity scores, for each iteration. Moreover, I conducted a qualitative analysis of the top-performing hyperparameter combinations to identify the set of hyperparameters that yield the most pertinent topics with high coherence scores and low perplexity scores. The final selection of the best hyperparameters was made based on a manual evaluation of the relevance and quality of the obtained topics for each parameter combination. Several combinations, along with their corresponding topics and metrics, are illustrated in the images below:

Text, letter

Description automatically generatedText, letter

Description automatically generated

Number of Topics - 8

Number of Topics - 7

Text, letter

Description automatically generatedText, letter

Description automatically generated

Text, letter

Description automatically generated

1. **Topic Interpretation & Metrics Used**

* Upon completing the aforementioned steps, I discovered that the following set of hyperparameters yielded the most optimal results:

1. **no\_of\_topics** - 7
2. **alpha** - 'symmetric'
3. **beta (eta)** - 0.3

* After computing the above combinations, the metrics - a coherence score of 0.343 and a perplexity score of -6.055 were obtained. The coherence score indicates the degree of semantic similarity between the top words in the topics and is generally used as a measure of topic coherence. A coherence score of 0.343 suggests that the topics generated have efficient level of coherence. On the other hand, the perplexity score measures how well the model can predict the holdout data, with lower scores indicating better performance. A perplexity score of -6.055 suggests that the model is performing reasonably well in predicting the holdout data.
* The resulting topics from the best hyperparameter scores are listed below.

*Text, letter

Description automatically generated*

It is apparent that there is a clear distinction between the topics, as illustrated below:

* Topic 0 delves into UK politics, specifically discussing the Labour and Conservative parties of the UK government, with the keywords pertaining to this topic being well-selected.
* Topic 1 is about conservative party, UK, London as shown above.
* Topic 2 focuses on the Ukraine war, as indicated by the keywords associated with this topic.
* Topic 3 is regarding High School Basketball.
* Topic 4 is related about the UK people and police.
* Topic 5 centres around football, with keywords such as Manchester United, a prominent football club, as well as mentioning an ongoing football game in England.
* Topic 6 delves into the former and current prime ministers of UK, namely Boris Johnson and Rishi Sunak, respectively.
* The topics identified are notably distinct and effectively encapsulate several essential themes discussed in the tweets dataset when compared to other topics, such as 1, 3, and 4.
* To better comprehend the effectiveness of the topics, a pandas dataframe was created using the results obtained, along with the percentage of dominant topic contribution in the documents. A higher contribution percentage indicates that the topics are more relevant in those documents and the topic modelling has worked more effectively in those instances. The image below exhibits the documents with the most optimal topic distribution, where the topic modelling has proven to be the most effective.

1. **Topic Evaluation**

**Good Topics**

Graphical user interface, application, table, Excel

Description automatically generated

* From Based on the image above, we can deduce that the first five rows in the Processed Text column pertain to the Labour and Conservative parties, demonstrating their relevance to the topics. The topic itself encompasses the Labour and Conservative parties, as well as UK politics, and the documents specifically discuss Jeremy Corbyn, who is both an MP and the leader of the Labour Party.
* Additionally, the subsequent five rows are indicative of the second topic, which pertains to the UK, news, tickets, and London, among other things. These documents likewise relate to the UK as a major economy and the availability of tickets in London.
* Moreover, rows 15 to 20 concern the topic of New York, London, and high school, with documents centering around basketball games of a New York-based team.
* Finally, beyond row 20, the topic of UK people comprises documents discussing UK women and staff.
* In contrast, the following diagram shows the documents where topic modelling has not worked as well:

**Bad Topics**

Calendar

Description automatically generated with low confidence

* The documents pertaining to topics such as labour, conservative party, and UK politics are irrelevant to topics such as love, good, and London.
* The next five rows of documents relate to topics such as UK, Ireland, America, and season, which are not correlated with the assigned topics.
* Documents 10-15 are about Edinburgh, year, street, and cold, which are not related to the assigned topics such as long, war, and Ukraine.
* Finally, the last five documents discuss topics such as economy, company, China, and world, which are not related to the assigned topics such as High, School, and New York.
* Upon observation of the above image, it is apparent that the topics bear little resemblance to the corresponding documents. However, in the diagram depicting good topic distributions, we can clearly see that the topics are closely related and relevant to the documents.

1. **Topic visualization**

* Furthermore, I utilized the pyLDAvis visualization library to create an inter-topic distance map that visually represents the relationship between the topics. This library allows for the interactive visualization of topic models.

Chart, bubble chart

Description automatically generated

* The top-left visualization presents the inter-topic distance map, which showcases the topics with reduced dimensionality, allowing us to discern the differentiation between the topics and their frequency in the corpus. It is evident that the topics have a significant distance between them, and the topic circles are nearly equal in size, indicating that they are all equally prominent in the corpus. Moreover, the top 30 important terms associated with each topic are displayed on the right, along with their respective term importance, to provide a better comprehension of the most influential terms within each topic.

1. **Topic Modelling – Grouped Tweets Data**
2. **Generic outline for topic modelling of Twitter Grouped Data**
3. **Data Collection**

We have been provided with a Twitter Grouped dataset file named groupedTweets.csv, which has already been collected and is ready for further processing. This dataset comprises 9968 tweets from the UK and it is grouped into 471 groups by using single pass clustering, each of the document contains information such as the group, Tweet ID, username, text content of the tweet, and corresponding nscore and qscore values of each tweet.

Furthermore, the Twitter Grouped dataset was loaded in the following manner to undergo pre-processing:

Graphical user interface, text, application

Description automatically generated

1. **Exploratory Data Analysis (EDA) & Data Pre-processing**

To extract meaningful insights from the dataset, several steps were taken in the Exploratory Data Analysis (EDA) and Data Pre-processing stage.

1. Examined the total number of unique tweets, unique users who tweeted, and the groups.

|  |  |
| --- | --- |
| Total number of unique Tweets | 9968 |
| Total number of unique Users | 7998 |
| Total Number of Groups | 471 |

1. Analysed the top user who have tweeted the most in each group by extracting their tweet counts.

**A picture containing table

Description automatically generated**

1. Investigated the top 10 groups data in which the users tweeted the most.

**Table

Description automatically generated**

1. Examined the top 10 groups with the highest number of tweets.

Table

Description automatically generated

1. Calculated the total word count for each document with respect to each group by determining the length of the document in terms of the number of words.

Text, application

Description automatically generated with medium confidence

1. Studied the Statistics of the Data for grouped tweets/documents, newsworthy score, and the quality score.

**Grouped Tweets/Documents Statistics**

|  |  |  |
| --- | --- | --- |
| Longest Tweet Length | Group 15 | 22971 |
| Shortest Tweet Length | Group 465 | 4 |

**Averaged Length of Grouped Tweets/Documents**

**Table

Description automatically generated**

**Tweets Newsworthy Score Statistics (Count, Minimum, Maximum, Average) for each Group**

**A picture containing table

Description automatically generated**

**Tweets Quality Score Statistics (Count, Minimum, Maximum, Average) for each Group**

**Table

Description automatically generated with medium confidence**

1. To comprehend the types of hashtags and mentions present in the grouped text, I extracted their counts along with the respective mentions and hashtags from each tweet/document with respect to each group.

Graphical user interface, text, application, email

Description automatically generated

1. To analyze the most popular hashtags among individuals, I have created a plot that showcases the top 10 hashtags that frequently occur in the tweets or documents with respect to each group.

Chart, line chart

Description automatically generated

The presented line graph showcases the 10 most frequently occurring hashtags in each group within the corpus of 9968 tweets. The hashtags linked to London and the UK have the highest frequency, occurring approximately 60 and 45 times, respectively. This graph offers crucial insight into the most frequently discussed themes in the tweets, enabling us to anticipate the types of topics that we may encounter after applying topic modelling to the grouped dataset.

1. Examined for the presence of null values in the Grouped Twitter dataset.

**Text

Description automatically generated with medium confidence**

The results obtained indicate that the dataset is devoid of any null values, which is a positive indication of its quality. This absence of null values not only prevents bias and incorrect results, but also improves the reliability and accuracy of any insights that can be extracted from the data.

1. The next essential step in processing the Twitter Grouped dataset is to clean the text data. This step is necessary to ensure that the model doesn't learn from noisy data and generates meaningful topics from the grouped tweets. To achieve this, several steps are involved in cleaning the text data. Firstly, HTML tags, URLs, hashtags, mentions, emails, newline characters, spaces, tabs, emojis, punctuations, all types of brackets, and regular expressions are removed. Additionally, special inverted commas are replaced with standard inverted commas, making them easier to detect and remove from the text. Finally, all words in the text are converted to lowercase, dropped the duplicate tweets and the contractions library was used to expand English words.

Graphical user interface, text

Description automatically generated

1. After cleaning the data, each row in the dataset was concatenated together as a list with respect to the group number (0-470) assigned to each row.

Text

Description automatically generated

1. I have plotted the frequency distribution of word counts with respect to each group in the documents to understand the frequency distribution of word counts in the documents which indicates the number of times different word counts appeared in the corpus.

Chart, bar chart

Description automatically generated Chart, histogram

Description automatically generated

1. The next step involves removing the stopwords, filtering out words whose frequency in the corpus is less than 5, and tokenizing the words. Following this, the single letter words, which are deemed less important, were filtered out and the remaining words were stored in a single list. This list is commonly referred to as the corpus of the documents.

**Text

Description automatically generated with medium confidence**

**Corpus of words – Total length of the grouped corpus is 449**

**Background pattern

Description automatically generated with medium confidence**

1. To enhance the semantic understanding of the Grouped Twitter dataset, I built both Bigram and Trigram models to generate words that appear together frequently, with a minimum occurrence of 5 in the corpus, and a threshold of 100 to ensure the resultant phrases are stronger and more meaningful.

**Bigrams:**

**Text, letter

Description automatically generated**

**Trigrams:**

**Text, letter

Description automatically generated**

1. After obtaining the bigram and trigram words, they are incorporated into the grouped corpus, while simultaneously eliminating all other parts-of-speech words except for Nouns, Adverbs, Adjectives, and Proper Nouns.

Map

Description automatically generated with medium confidence

1. Eventually, a dictionary was generated, and a new group corpus was created, wherein every document with respect to each group was transformed into a bag-of-words format, which returns a tuple comprising of token ID and token count for each word in the document. The token ID is a unique identifier allocated to every word in the corpus, allowing them to be identified distinctly.

**Scatter chart

Description automatically generated**

1. **Topic Modelling**

* To facilitate topic modelling for grouped dataset, I began by implementing a baseline LDA model to determine the optimal number of topics that would produce the most favorable outcomes. During my experimentation, I varied several LDA parameters, including random state, chunk size, passes, and iterations. Ultimately, I arrived at a set of baseline parameters that proved most effective in generating pertinent topics:

1. Chunk Size – 50 (The value 50 was taken, because there are 471 only groups in the dataset i.e size of the dataset is small)
2. Passes – 20 (The value 20 was taken, to improve the coherence score as there is small size of dataset)
3. Iterations – 50
4. Random State – 50

* During my iteration process, I explored a range of 6 to 21 topics and printed the coherence and perplexity score outputs for each iteration. This selection was made to avoid excessive computational costs, as the system requires a significant amount of time to execute a larger range of topics. To visually represent the output of this process, I generated a graph plotting the number of topics against their respective coherence and perplexity scores, which can be observed in the accompanying images below.

Chart, line chart

Description automatically generated

Number of Topics vs Coherence Scores

Chart, line chart

Description automatically generated

Number of Topics vs Perplexity Scores

* By examining the perplexity and coherence scores, it is evident that as the number of topics increases, the perplexity scores decreases, and coherence scores improves. Thus, a higher number of topics is a more suitable option. Based on this, I have selected a range of 6 to 9 number of topics as the optimal number of topics. This range begins at topic number 6 because the coherence score increases at topic number 6, indicating that the quality of topics will be satisfactory at that number. In the graph, the coherence score increases from 6 to 7 topics, and then starts decreasing till topic 8. Further, it was observed that the coherence score is increased drastically, which depicts the steep slope in the graph from 8 to 12 topics, and then it starts fluctuating till the end. This implies that the data is becoming strongly overfitted with an increasing number of topics.
* To determine the optimal number of topics, I utilize a technique called hyperparameter optimization, which assists in identifying the most appropriate hyperparameters for the LDA model, allowing it to extract the most relevant topics from the dataset.
* In order to decide the optimal model hyperparameters, I have selected the following hyperparameter search space:

1. **no\_of\_topics** – [6, 7, 8, 9]
2. **alpha** – ['symmetric', 0.3, 0.5, 0.7]
3. **beta** – [0.3, 0.5, 0.7]

* To optimize the model's performance, I employed a technique known as Grid Search, which exhaustively tests all possible combinations of hyperparameters and reports relevant metrics, namely coherence and perplexity scores, for each iteration. Moreover, I conducted a qualitative analysis of the top-performing hyperparameter combinations to identify the set of hyperparameters that yield the most pertinent topics with high coherence scores and low perplexity scores. The final selection of the best hyperparameters was made based on a manual evaluation of the relevance and quality of the obtained topics for each parameter combination. Several combinations, along with their corresponding topics and metrics, are illustrated in the images below:

Text

Description automatically generated Text, letter

Description automatically generatedText, letter

Description automatically generatedText, letter

Description automatically generated

Number of Topics - 8

Number of Topics - 9

Number of Topics - 7

Number of Topics - 6

1. **Topic Interpretation & Metrics Used**

* Upon completing the aforementioned steps, I discovered that the following set of hyperparameters yielded the most optimal results:

1. **no\_of\_topics** - 7
2. **alpha** - 'symmetric'
3. **beta (eta)** - 0.7

* After computing the above combinations, the metrics - a coherence score of 0.429 and a perplexity score of -8.685 were obtained. The coherence score indicates the degree of semantic similarity between the top words in the topics and is generally used as a measure of topic coherence. A coherence score of 0.429 suggests that the topics generated have good level of coherence. On the other hand, the perplexity score measures how well the model can predict the holdout data, with lower scores indicating better performance. A perplexity score of -8.685 suggests that the model is performing reasonably very well in predicting the holdout data.
* The resulting topics from the best hyperparameter scores are listed below.

*Text, letter

Description automatically generated*

It is apparent that there is a clear distinction between the topics, as illustrated below:

* Topic 0 delves into gaming hardware requirements, specifically discussing the type of graphics cards such as geforce, rtx. The keywords pertaining to this topic are very coherent and well-selected.
* Topic 1 talks about London, tickets, and a football game of the Manchester united football club, with the match possible happening within London and the topic of discussion of the tweets being about the tickets to the game.
* Topic 2 focuses on New York city and livestreaming of high school basketball games of a new York based team.
* Topic 3 is about the people of the United Kingdom with some good events happening around the time of the year.
* Topic 4 has topics about the former prime minister of UK, Boris Johnson, and his interview with Nadine. The tweets about these topics were possible regarding the topic of the Ukraine war and Afghanistan.
* Topic 5 centres around UK politics, with the tweets comprising of discussions about the labour party, the current prime minister Rishi Sunak, Corbyn, one of the political leaders in UK etc.
* Topic 6 talks about the Youth FA Cup which is a major football event happening within the UK.
* The topics identified are notably distinct and effectively encapsulate several essential themes discussed in the group tweets dataset.
* To better comprehend the effectiveness of the topics, a pandas dataframe was created using the results obtained, along with the percentage of dominant topic contribution in the documents. A higher contribution percentage indicates that the topics are more relevant in those documents and the topic modelling has worked more effectively in those instances. The image below exhibits the documents with the most optimal topic distribution, where the topic modelling has proven to be the most effective.

1. **Topic Evaluation**

**Good Topics**

Text

Description automatically generated

From the above image it is observed that the first 4 documents have the topics related to the former prime minister Boris Johnson, his interview with Nadine and topics associated with the Ukraine and Afghanistan war. The tweets are closely related to the topic of war with them comprising of discussion about the Ukraine war, Putin and EU to name a few.

Graphical user interface, text, application

Description automatically generated

The above topic revolves around the topic of the youth fa cup in football and the place Norfolk, and the most representative tweets also comprise of content around the Norfolk county and the other tweets having resemblance to winning and competition as associated with football.

Text, application

Description automatically generated

The topic comprises of keywords such as UK, people, good, well etc. which suggests that the topic talks about the people of the UK, some good events or happenings in UK at a certain time in the year and so on. The tweets associated with the topic have content related to England, Scotland, great season, wonderful etc. to name a few which are very relevant to the topic, whereas it has some other content which are not that relevant to the topic.

* In contrast, the following diagram shows the documents where topic modelling has not worked as well:

**Bad Topics**

Graphical user interface, application, Word

Description automatically generated

The first topic in the above dataframe contains keywords related to gaming and gaming related hardware such as graphics cards but the tweets do not have any relevance whatsoever to the topic content suggesting that the distribution of topics within the documents are poor.

Graphical user interface, text, application

Description automatically generated

The first 2 documents have been assigned to the topic related to the former prime minister Boris Johnson, his interview with Nadine and the Ukraine and Afghanistan war. The document themselves have content such as chocolate treat, hilarious sound etc. which are irrelevant with respect to the associated topics. The topic having keywords related to UK politics are shown to contain tweets about delicious food, Thomas association etc. which are again not relevant to the topic.

* Upon observation of the above image, it is apparent that the topics bear little resemblance to the corresponding documents. However, in the diagram depicting good topic distributions, we can clearly see that the topics are closely related and relevant to the documents.

1. **Topic visualization**

* Furthermore, I utilized the pyLDAvis visualization library to create an inter-topic distance map that visually represents the relationship between the topics. This library allows for the interactive visualization of topic models.

Chart

Description automatically generated

* The top-left visualization presents the inter-topic distance map, which showcases the topics with reduced dimensionality, allowing us to discern the differentiation between the topics and their frequency in the corpus. It is evident that the topics have a significant distance between them, and the topic circles are different in size. Moreover, the top 30 important terms associated with each topic are displayed on the right, along with their respective term importance, to provide a better comprehension of the most influential terms within each topic.

1. **Compare and analyse the performance differences between Tweets Data and Grouped Tweets Data.**
2. **Compare the performance of two specific chosen topic models in 1 and 2. Select appropriate metrics for comparison and justification with data statistics. In your opinion, which among 1& 2 tasks perform better? and why?**

|  |  |  |
| --- | --- | --- |
| **Model Name** | **Hyper-parameters** | **Metrics** |
| LDA – Twitter Dataset | Random state = 50  Number of topics = 7  Chunk size = 500  Passes = 10  Alpha = symmetric  Eta = 0.3  Iterations = 50 | Coherence Score = 0.3427  Perplexity Score = -6.055 |
| LDA – Twitter Grouped Dataset | Random state = 50  Number of topics = 7  Chunk size = 50  Passes = 20  Alpha = symmetric  Eta = 0.7  Iterations = 50 | Coherence Score = 0.4290  Perplexity Score = -8.6851 |

* The performance of the two topic models was compared, with the help of two important metrics: coherence score and perplexity score. The coherence score is used to measure the interpretability and quality of the topics generated by the model, while the perplexity score is used to measure the predictive power of the model like how well the model can predict on the unseen data.
* In the LDA- Twitter Dataset model (model 1), LDA was applied to the Twitter Dataset with random state 50, number of topics 7, chunk size 500, 10 passes, symmetric alpha, eta 0.3, and 50 iterations. It achieved a coherence score of 0.3247 and a perplexity score of -6.055.
* In the LDA-Twitter Grouped Dataset model (model 2), LDA was applied to the Twitter Grouped Dataset with random state 50, number of topics 7, chunk size 50, 20 passes, symmetric alpha, eta 0.7, and 50 iterations. It achieved a coherence score of 0.4290 and a perplexity score of -8.6851.
* Based on these metrics, it is very clear that the LDA- Twitter Grouped Dataset model has performed better than the LDA-Twitter Dataset model. This conclusion was made because the higher coherence score indicates that the topics in the 2nd model are highly interpretable, and the quality of topics are very good, the topics are observed to be at a far distance from each set of topics generated. On the other hand, the higher perplexity score suggests that the model 2 has better predictive power which means it will be able to effectively predict on the unseen data.
* The difference in the performance is due to the differences in the set of hyperparameters used to tune both the models in order to get the optimal result. For example, a higher eta value in Model 2 may result in more informative topics, while a smaller chunk size may improve the accuracy of the model but may take longer to run. One big advantage for model 2 is that it is a grouped dataset, which have helped to improve the interpretability of the topics. Additionally, the use of smaller chunk size and higher eta value helped to improve the performance of the model.

**Tag Cloud Comparison**

**LDA-Twitter Dataset Model**

**LDA-Twitter Grouped Dataset Model**

**Logo, company name

Description automatically generated**

Both the word clouds of the LDA model topics on the tweets and the grouped tweets dataset have 7 topics each. On first glance, we can observe the presence of terms like UK, New York, London and some topics related to UK politics present pervasively in both word clouds. From this, we can infer that many tweets within the corpus have themes around the United Kingdom, London, New York, and Politics. This seems to be the common theme within the given datasets. Additionally, we can observe the presence of certain topic keywords such as Boris/Ukraine/Nadine, Stock/Graphics and Youth/FA/CUP within the twitter grouped dataset model topics which are not only specific to the dataset itself but also not present in any other topics within the word cloud giving the impression of their coherence and uniqueness. On the other hand, certain topic keywords like Work/Public and News/London are specific to the grouped tweets word cloud and not present in any other topics within or in the other word cloud. Overall, we can conclude that both the word clouds have a mix of common thematic topics as well as specific topics and we would need to perform further analysis to assess the best topic model.

1. **Elaborating the comparison with examples; identify good topics and bad topics.**

**LDA-Twitter Dataset Model Topics (Model 1)**

*Text, letter

Description automatically generated*

**LDA-Twitter Grouped Dataset Model (Model 2)**

*Text, letter

Description automatically generated*

* The topics generated by the two models are quite different from each other. It is observed that model 1 is more focused on the political topics related to the UK, while Model 2 is focused on more general and diverse topics. Model 1 has topics related to political parties, politicians, issues such as labour and public services. On the other hand, Model 2 has topics related to cities, schools, graphics cards, and gaming.
* In model 1, topics 1,4, and 6 are related to UK politics, with keywords such as “conservative”, “tory”, “boris”, “minister”, “money”, etc. Topics 2 and 3 seems to be more general topics related to news and events in the UK, with keywords such as “uk”, “news”, “world”, “war”, “york”, “london” etc. Topic 5 seems to be related to sports and entertainment, with keywords such as “game”, “manchester”, “united”, etc.
* In contrast, Model 2 seems to have generated less coherent topics that are not directly related to UK news and politics. For example, topic 0 seems to be related to gaming and graphics cards, while topic 2 seems to be related to high school basketball and livestreams. Topic 4 seems to have some connection to UK politics with keywords such as "boris", "johnson", and "ukraine", but the overall topic is still quite random and not well-defined. Topic 5 seems to be related to UK politics with keywords such as "labour", "party", and "sunak", but again, the topic is not well-defined and has some random keywords mixed in.
* It is also worth noting that the top words in each topic generated by Model 1 appear to be more coherent and related to each other, whereas the top words in each topic generated by Model 2 appear to be more disparate and unrelated to each other. This may suggest that Model 1 is better suited for generating coherent and topical text, while Model 2 is better suited for generating diverse and exploratory text.
* Model 1 good Topics are topic 0,2,3,5,6. These topics are Labour-Conservative party, Ukraine War Putin, High School Basketball, Manchester United Football club, and Boris Johnson, Rishi Sunak. On the other hand, the bad topics are topic 1, and 4 in which it is difficult to identify the context.
* In model 2 all topics are considered to be good topics, they are gaming, Manchester united football, High school basketball, conservative party, Boris Johnson interview with Nadine, Labour Party, Rishi Sunak, youth fa cup.
* Overall, Model 2 – Grouped Twitter dataset is generated good and diverse topics as compared to model 1 – twitter dataset in which the context of the topics are not much clear.

1. **Identify documents with appropriate topic distribution and not so good topic distribution.**

**Model 1- Twitter Dataset (Good/Bad Topics Distribution)**

**Good Topics Distribution**

**Graphical user interface, table

Description automatically generated with medium confidence**

**Graphical user interface, application

Description automatically generated**

**Bad Topics Distribution**

**Graphical user interface, application

Description automatically generated**

**A picture containing graphical user interface

Description automatically generated**

**Model 2 – Twitter Grouped Dataset ( Good / Bad Topics Distribution)**

**Good Topics Distribution**

**Graphical user interface, text, application

Description automatically generated**

**Text

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**Bad Topics Distribution**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

1. **Identify and discuss topic formation issues, due to the nature of tweets.**

Twitter is a popular social media platform where users can post and read short messages known as tweets, which are limited to a maximum of 280 characters. Twitter is used to easily convey messages, news to large amount of crowd in a few minutes which can be very helpful to the local people. While this format has an advantage such as brevity and conciseness, it can also pose challenges for topic formation. Due to the maximum character limit, twitter users will likely not be able to introduce a lot of contexts into their tweets causing a lot of issues in topic formation. Some of the issues are mentioned in the below pointers.

Below are the issues observed in the Twitter dataset while pre-processing the data, and topic modelling on twitter data:

1. **Lack of Context:** Due to character limit, tweets lack the necessary context to fully convey the intended message this leads to the formation of data sparsity. The lack of context, and twitter users talking about a lot of specific topics, can make it difficult even for a normal user unfamiliar with certain ongoing happenings around the world to interpret and understand the theme of a tweet text. During qualitative analysis, it was difficult to interpret quite a few of the tweets and correlate them with the topics extracted due to their lack of context.

**For Example:**

**Text

Description automatically generated**

1. **Ambiguity:** The tweets usually rely on the context and cultural references. These tweets are difficult to understand what message the user want to convey. In below tweet, the statement ‘what’s on the toast?’ can have a few different meanings which cannot be interpret correctly without more knowledge of what the user is tweeting about. The statement listing different countries – Scotland, Seychelles, Syria is also just a generic list of countries and we have no understanding of relevance of the tweet.

**For Example:**



** 3. Hashtags and Mentions:** There is usually a presence of lot of hashtags and mentions in the tweets which might be relevant to twitter but is actually noise to the topic model. This can attested by the fact that on running the topic model with the hashtags, the model actually performed worse than without the hashtags even though intuitively the hashtags should have introduced some topic related context in the tweets. This resulted in introducing a pre-processing step to completely remove any hashtags in the dataset.

**For Example:**

**Text, application

Description automatically generated**

1. **Overfitting issues:** As observed in the LDA models on both the tweets and grouped tweets dataset, when the model starts reaching around 10 to 12 number of topics, it starts to produce redundant topics all over suggesting that the model has started to overfit on the dataset. Due to data sparsity, especially with bag of words representations, there is a high risk of topic models overfitting on the data.

**For Example:**

**Text, letter

Description automatically generated**

1. **Extensive pre-processing requirement:** There were more than 15 to 20 pre-processing steps that were required in order to process the tweets and get them ready to be input into the LDA topic model. This can be computationally expensive as well if there is a large dataset and extensive pre-processing can also backfire as it can remove essential context from the tweets as observed in some tweets in the dataset.

**For Example:**

**Graphical user interface, text

Description automatically generated with medium confidence**

1. **Lack of accommodation for diversity of tweets**: Due to the massive user-base of twitter and people from across the world, belonging to different regions, nationalities, race, and ethnicities use this platform, there are a variety of tweets which may consist of local slangs, abbreviations, dialects etc. which can be difficult for a model to understand unless explicitly trained on. Even while being in the UK, it was difficult for even myself to understand some of the tweets which seemed to be talking about some local event or topic with extensive use of slangs and abbreviations. Thus, it would be even more difficult to feed this understanding into the model.

**For Example:**

**A picture containing text

Description automatically generated**