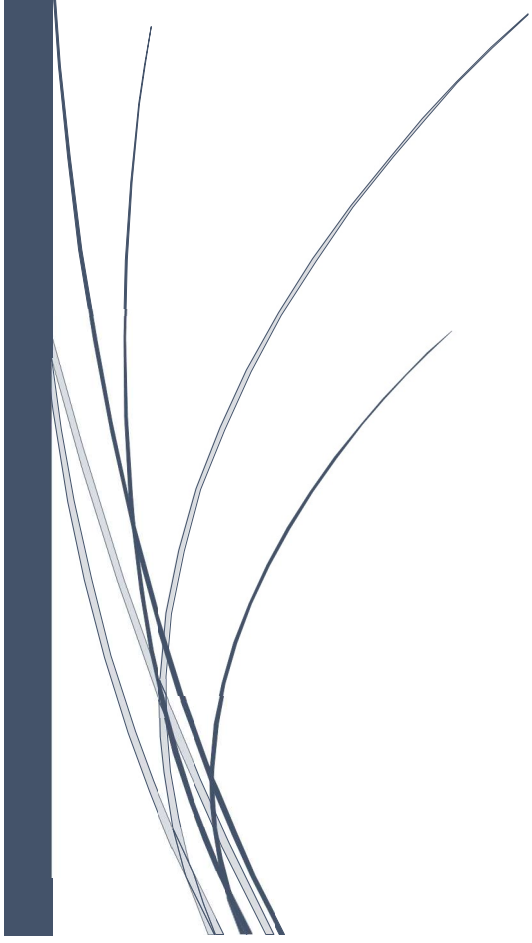
Topic Modelling - Tweets

AAYUSH NAGPAL



TOPIC MODELING

Introduction

Topic Modelling is an unsupervised machine learning technique that automatically analyses the text data in the given corpus of documents to determine cluster of words that can be used to describe corpus of documents. As part of this coursework, we will implement LDA models to perform topic modelling on tweets. LDA is a popular generative statistical algorithm used for topic modelling. LDA models generally expect large documents to perform its task effectively. But we know that the length of tweets in the dataset will vary between 18 words to 926 words only. Topic modelling for such documents is referred to as short topic modelling. In this coursework, we will compare performance of LDA models when using individual tweets and grouped tweets. The idea here is that LDA model should perform better when using grouped tweets.

Dataset Statistics

Before we discuss more about LDA models and its performance on the ungrouped dataset of tweet, we will try to understand the data. The image given below (Image-1) shows the distribution of tweets based on its length. It is clear from this graph that a large majority of the tweets contain less than 200 words. There are rare tweets with word count greater than 400.

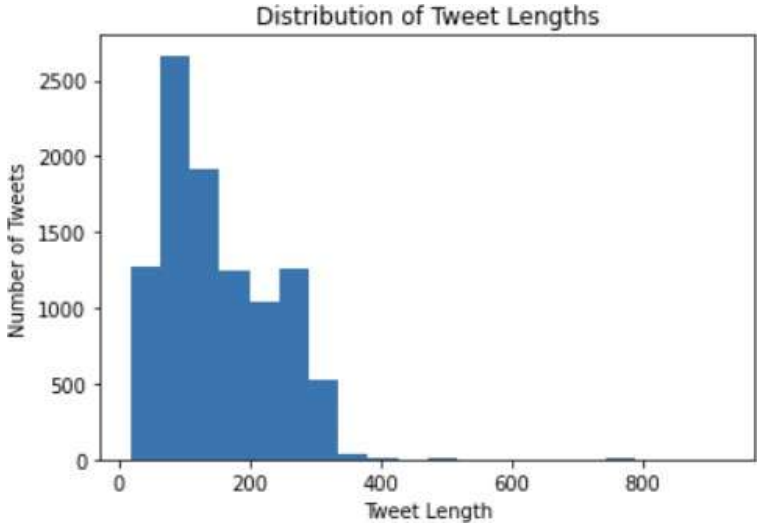


Image 1 - Tweet Distribution based on length

Looking at the tweets from the perspective of word count we can summarise the dataset as follows:

count	10001.0
mean	151.9
std	83.5
min	18.0
25%	84.0
50%	132.0
75%	215.0
max	926.0

Image 2 - Statistics on Tweets

In total there are 10001 tweets present in the dataset. The maximum length of tweet in terms of word count is 926 and minimum is 18. The average length of tweets in the dataset is 151.9. As mentioned above 75 percentiles of the data has a length of less than 215 words.

Step to perform Topic Modelling

As mentioned earlier topic modelling is machine learning technique that is used for natural language processing and text analysis. Let us now briefly outline the steps required to create a ML model for this task:

- **Data pre-processing** – In the earlier section we saw the structure and statistics for the given data. It is evident that any data collected from social media platforms like Twitter would be cluttered with marketing hyperlinks, emojis, emoticons, emails etc. In order to prepare this data for modelling we need to clean the data as part of data pre-processing stage. Here we use a regex to remove unwanted data from the tweets and convert the entire tweet into lower case. We can also use bigrams and trigrams to form links between words occurring together frequently.
- **Document term matrix creation –** Now that the data is cleansed, we will move on to the next stage of converting the text into its numerical representation. Generally, we use methods like TFIDF vectorizer or one hot encoding to convert text into a format that can be interpreted by machines. But for this exercise we will be using LDA models for which we do not need to perform this transformation.
- **Topic model algorithm selection –** Select an algorithm to be used for topic modelling. There are many popular choices like – Latent Dirichlet Allocation (LDA) models, non-negative matrix factorization, deep learning etc.
- **Model Training –** Next step is to train the model using vector representations or the compass text identify hidden topics within the corpus.
- **Topic Interpretation –** once the model is trained, we need to extract the topics and interpret the results. This involves identifying the top word associated with each topic and use other words to understand any underlying theme/topic.
- **Model Evaluation –** just like any other machine learning model it is important to evaluate the performance of the topic model to check its accuracy and meaningfulness. This is used to establish relationships between topics. Some of the methods for evaluation are as follows:
    - **Perplexity –** It is one of the most known methods. It tells how well the model predicts the held-out data. A lower score indicates good topic modelling.
    - **Coherence score –** It highlights the semantic similarity between words in the topic. It's a reflection on the underlying structure of the topics. The value for this ranges between 0 to 1. A higher coherence score is always preferred.
    - **Visualizations –** using the results from the model we can draw visualizations like bar-charts for document distribution between topics, heat maps, word clouds, etc.

Data Pre-processing Technique

As mentioned above, pre-processing is an important stage in topic modelling. During this process we will remove irrelevant data from the tweet as it can act as noise when fed to the LDA. Here, irrelevant data refers to hyperlinks, emojis, emoticons, etc which are usually part

of tweets. In this exercise, we will use the pre-processing stage to transform the data in the following ways:

- Remove emojis, flags and emoticons from the tweet as it does not contribute towards topic modelling.
- Remove emails addresses and '@'mentions. The email addresses and Twitter handles do not result in meaningful topics.
- Remove hyperlinks.
- Remove newline character and single quotation marks.
- Remove stop-words from the document(tweet) as these words are very high frequency words and can cause issues during topic modelling.
- Convert the documents into a list of tokens and transforms the text into lower case using simple_process function from genism library.

This completes the first stage of pre-processing where we have removed a lot of unnecessary information from our data. Let's look it the impact of prep-processing on the dataset. As we can observe from the graph below (Image 3), that a large majority of out documents have less than 40 words.

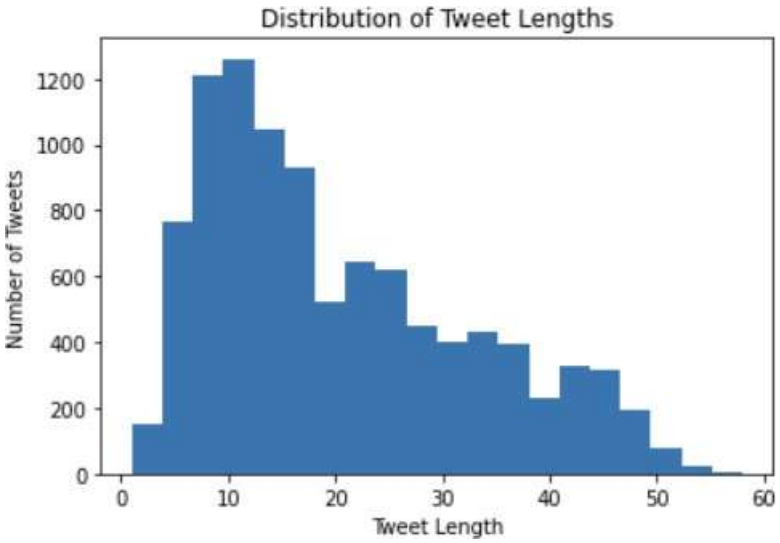


Image 3 – Distribution pre-processed Tweets

The statistics also show in Image 4 shows that on an average a tweet contains about 20.5 words and 75 percentiles of the documents in the dataset have less than 29 words. This proves that the document length is extremely short and we are essentially doing short topic modelling.

count	10001.000000
mean	20.417758
std	12.390628
min	1.000000
25%	10.000000
50%	17.000000
75%	29.000000
max	58.000000

Image 4 – Statistics for pre-processed Tweets

The next stage of pre-processing includes creation of bigrams, trigrams and lemmatization of words. Sometimes, individual words do not make sense without its context. This is where bigrams and trigrams are helpful as they capture the relationship between frequently co-occurring words.

Lemmatization is a method to normalize the text data by reducing it to their base form or lemma. Usually, data in text format contains same words in different forms based on their tense, position in a sentence and other grammatical structure. Overall, by creating bigrams and trigrams and performing lemmatization we improve the quality of the input data.

Model – 1

Selecting Optimal Number of Topics

To select the optimal number of topics for model I calculated the perplexity, coherence score and KL Divergence score of the model across a range of topics as shown in the image (Image-5).

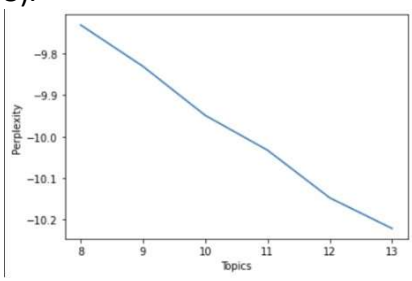


Image 5A – Perplexity

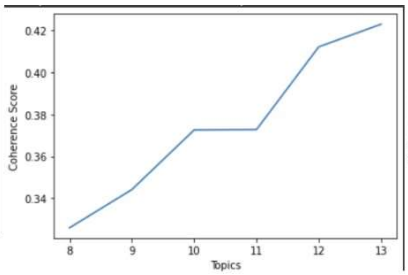


Image 5B - Coherence Score

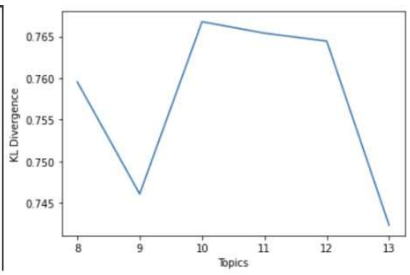


Image 5C - KL Divergence

From image 5A we can infer that as number of topics increase the log_perplexity of the model increases. One of the reasons that can be attributed to this is that as the number of topics increase it becomes more complex and difficult to fit the model. Because of this the model might overfit. This means that the optimal number of topics should lie between the range of 8 to 10 so that the perplexity of the model is not too high.

From image 5B we can infer that as the number of topics increase the coherence score keeps on increasing. As mentioned earlier, coherence measures the semantic similarity between topics and as we increase the topics coherence increases because similar words are grouped together. But after a certain point the coherence score becomes stagnant and does not change much. Coherence score is lowest for 8 therefore it's safe to also rule out values smaller

than 8. The remaining options are 9 and 10.
Image 5C shows the variations in KL divergence with increase in number of topics. As we know for an ideal model we would like to keep KL divergence as low as possible but at the same time the score cannot be extremely low as this can lead to overlapping and redundancy. As we can observe that the KL Divergence values are already very high so in order to make sure that the topics are coherent and have semantic relation, we will select optical number of topics as 9.

Evaluating the model

A – Quantitative Methods

Once we train the model using optimal number of topics, we can evaluate the model using different technique highlighted in the earlier section. The table below highlights the perplexity, coherence score, and KL Divergence of the model.

Perplexity	Log_Perplexity	Coherence_Score	KL Divergence
910.423	-9.83	0.344	0.746

Image 6 – Metrics for model with 9 topics

As we can see in the table above the perplexity value of the model is extremely high, the coherence score of the model is only 0.34 which extremely low for a model to generate good and coherent topics for the documents in the dataset. Also, the KL Divergence for the model is towards higher end which means that the there is a high level of dissimilarity between predicted topics and actual topics.

B – Visual Methods

Word Cloud

We can also use visualisation and human interpretation to evaluate the model. Image 7 shows a word cloud with the topics generated by the model. It is evident that topics generated by the model are not very coherent.

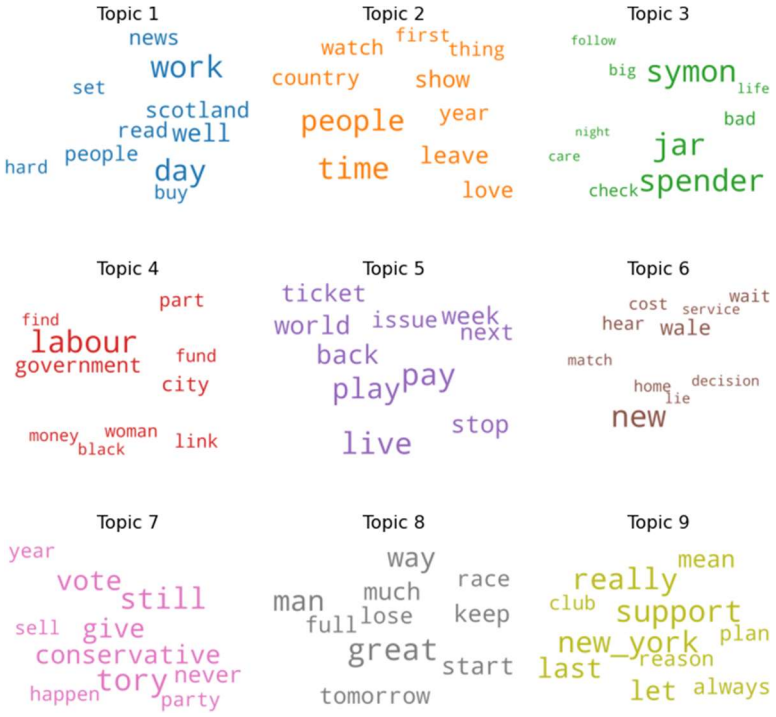


Image 7 – Word Cloud for 9 topics

With the exception of subject 7, where politics may be the projected theme, all of the topics in the word cloud are completely unrelated to one another.

Jaccard Distance

Based on the word distributions of two topics, the Jaccard distance is a distance metric that measures how distinct the two topics are. A high Jaccard distance score shows that the two topics are diverse in composition and share few words. The Jaccard matrix in relation to this model shows that the subjects produced by this model are inconsistent and do not have a common theme. This is further supported by a word cloud analysis.

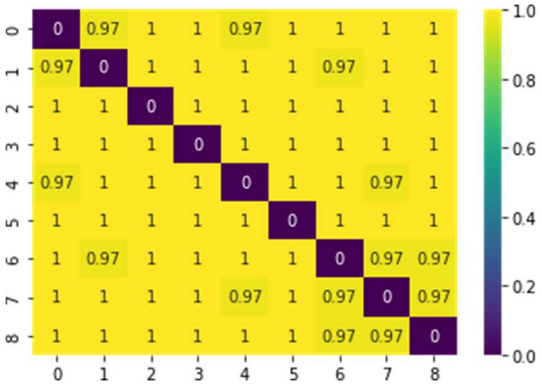


Image 8 – Jaccard Matrix

Word Count and its Importance

We can also use bar in bar charts to highlight the importance of keywords within each topic. Image 9 shows a bar in bar chart for each topic. Each chart shows top 10 keywords along with its weight and frequency within respective topic.

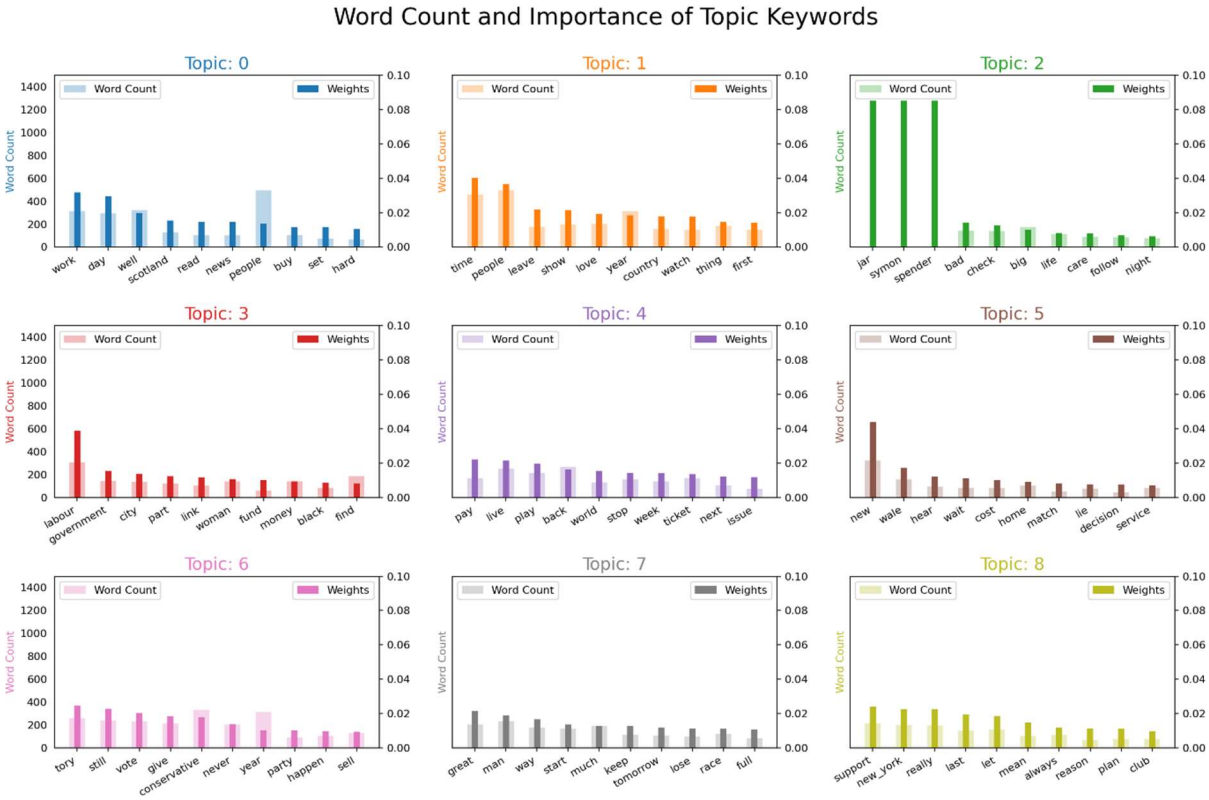


Image 9 – Bar in Bar Chart for LDA model using tweets

While working with this type of chart we need to be careful of words that are repeating across different topics. For example- the word “people” is under Topic 0 as well as Topic 1. We should also be careful of the words who is relative frequency is much higher than its weight.

Inter-topic Distance Mapping

A much interesting and useful visualisation technique is Intertopic Distance Map (via multidimensional scaling). An analytical tool for visualising the relationships between words in a topic model is the Intertopic Distance Map. In a 2D or 3D space, each point represents a topic, and the distance between points shows how similar the respective topics are to one another.

It can be used to quantify topic diversity and coherence, locate groups of related subjects, and gain understanding of how topics are distributed throughout the dataset. Image 10 shows the map for this model.

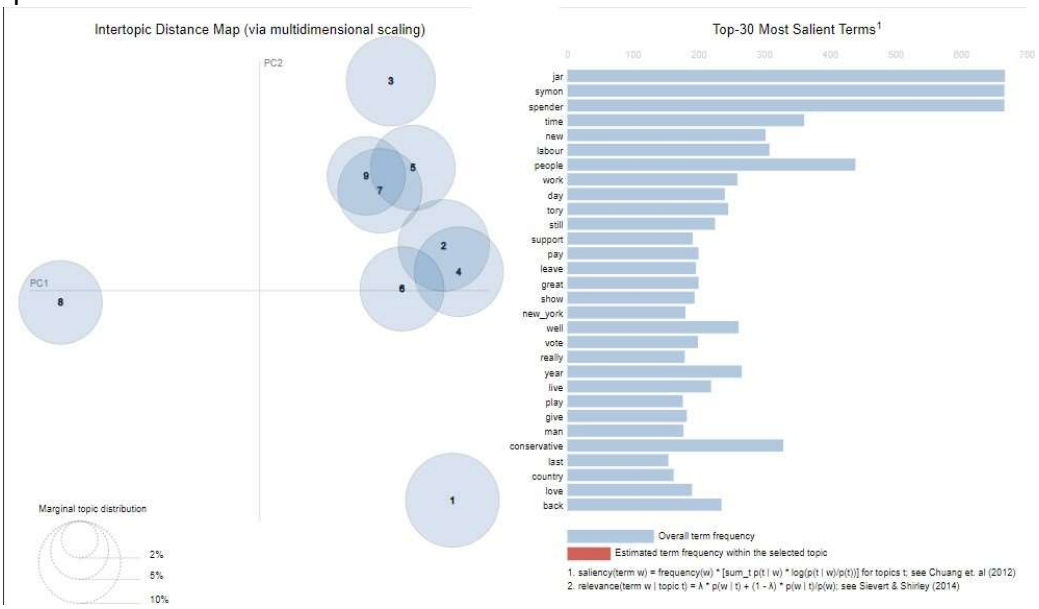


Image 10 – Intertopic Distance Map

We can infer from the image above that there is a lot of overlapping between topics which means that the topics generated by the model are not coherent and have a lot of similarity. This may indicate that the model is unable to meaningfully represent the underlying structure of the data, which can be viewed as an indication of poor model performance.

Thus, it is fair to conclude that the model has failed to deliver good, coherent topic with an underlying theme and structure. Due to its poor performance, the model may not be appropriate for the intended application and may need to be significantly improved before it can be used successfully. It may be important to address the underlying problems by expanding and diversifying the dataset, modifying the model's hyperparameters, and putting procedures in place to lessen topic overlap and duplication in order to enhance the model's performance.

Model - 2

Introduction

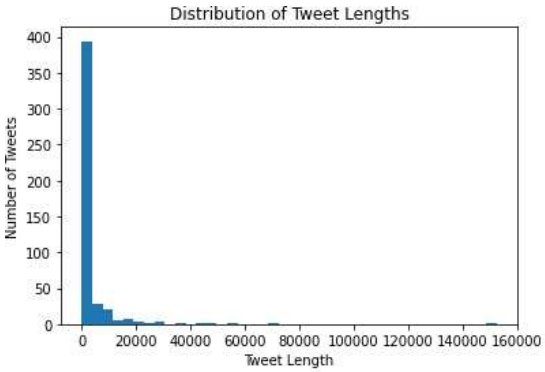
We will create another model using the dataset which contains the same set of tweets grouped with single-pass clustering. Each tweet in this dataset is mapped to a group number. We group these tweets together based on the group number which results in a dataframe that contains 471 records. Each record corresponds to a group number and a set of tweets concatenated together. Image 11 shows us the grouped content in dataframe.

	group	text	text_length
0	0	@pansexualflower Criterion certainly have US rights to the restoration but very unlikely they ha...	6563
1	1	CN US #China\n#Chinese #Foreign #Ministry said that this is a weather balloon that had deviated ...	3687
2	2	“As they began to understand the children’s operational schemes, they were more purposeful in th...	14152
3	3	Excess Deaths Skyrocket Again In England And Wales\n\n15,804 deaths, and 1,568 excess deaths wer...	54365
4	4	@altgirlalterego scotland, seychelles, syria, south africa, south korea @Nigel_Farage Instead of...	9953
5	5	Horse Racing History: Today in Racing 4th February - https://t.co/aXqt99W5Fv\nThis day 4th Febru...	18573
6	6	@dominos ridiculous service from your store. Remembered why I don’t order from you. Your manager...	2116
7	7	Check out Snake Eyes DVD (2021) https://t.co/Bo8Ubf6QFj #eBay via @eBay_UK Check out this item i...	19294
8	8	de Jouvenel's Sovereignty (1957) has a fable about 'Babylon', a city whose metaphysical foundati...	10401
9	9	They break the law with impunity, and have no shame.\n(The Hunting Act 2004)\nhttps://t.co/VKzcQ...	8580

Image-11 Grouped Tweets in Dataframe

Dataset Statistics

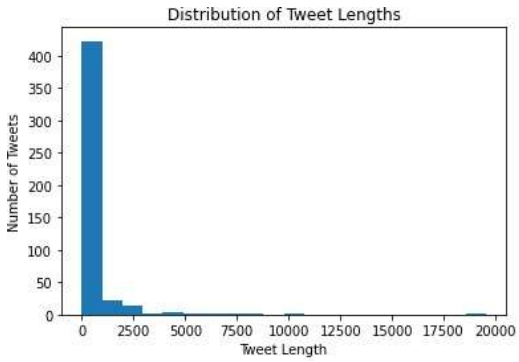
Before we begin with text pre-proessing, we need to understand the structure of the data. As shown in Image-12, the mean length of the grouped tweets dataset is 3235. This means that on an average the word count of grouped tweet is 3235 words. The minimum length of tweet(text) in the dataset is 31 and maximum is 152602 words. Image 12 shows did distribution of merge tweets based on text length.




```
count       471.00
mean       3235.56
std       10102.88
min          31.00
25%         189.50
50%         594.00
75%        1922.50
max      152602.00
Name: text length, dtype: float64
```

Image 12 – Distribution and statistics of grouped tweets

Now, we will perform the same set of data pre-processing steps as followed for the previous model. After pre-processing, a lot of the redundant and irrelevant data is removed from the tweets. This affects the distribution of tweets based on length as well as its internal structure.



count	471.00
mean	432.22
std	1370.48
min	1.00
25%	20.00
50%	68.00
75%	247.50
max	19532.00

Image 13 - Distribution and statistics of processed grouped tweets

Selecting Optimal Number of Topics

We will again use the LDA model to perform topic modelling. But before we can begin training our model, we need to select the optimal number of topics for this model. To select an optimal number of topic, we will follow the same process as earlier by calculating the perplexity, coherence score and KL divergence forecast for a range of values. The results for which are plotted using graphs and shown in image 12.

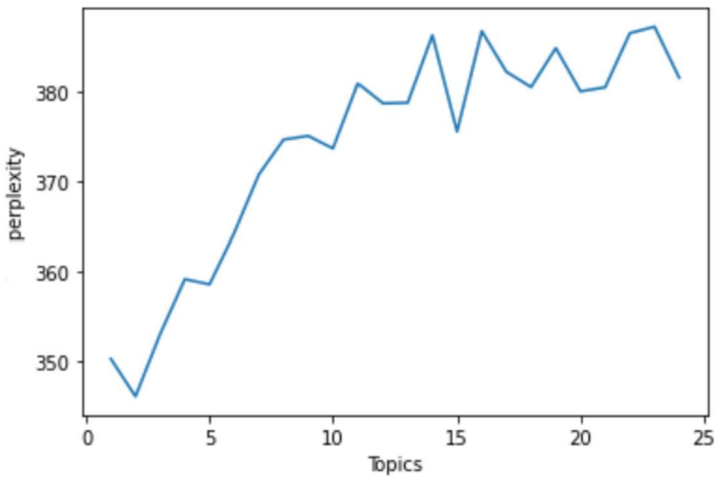


Image 12 – Perplexity for grouped tweets dataset

The first graph shows change in perplexity with increase in number of topics. The LD model gives us log perplexity and we can convert it to perplexity using the following formula:

Perplexity=2**(-log_perplexity)

This graph does not really help us selecting optimal topic as for wide range the perplexity values are acceptable.

Now, if we look at coherence graph (Image 13), we can observe that coherence value fluctuates a lot. Keeping in mind that we need keep perplexity score as small as possible, the only possible values for optimal topics is either 5 or 6.

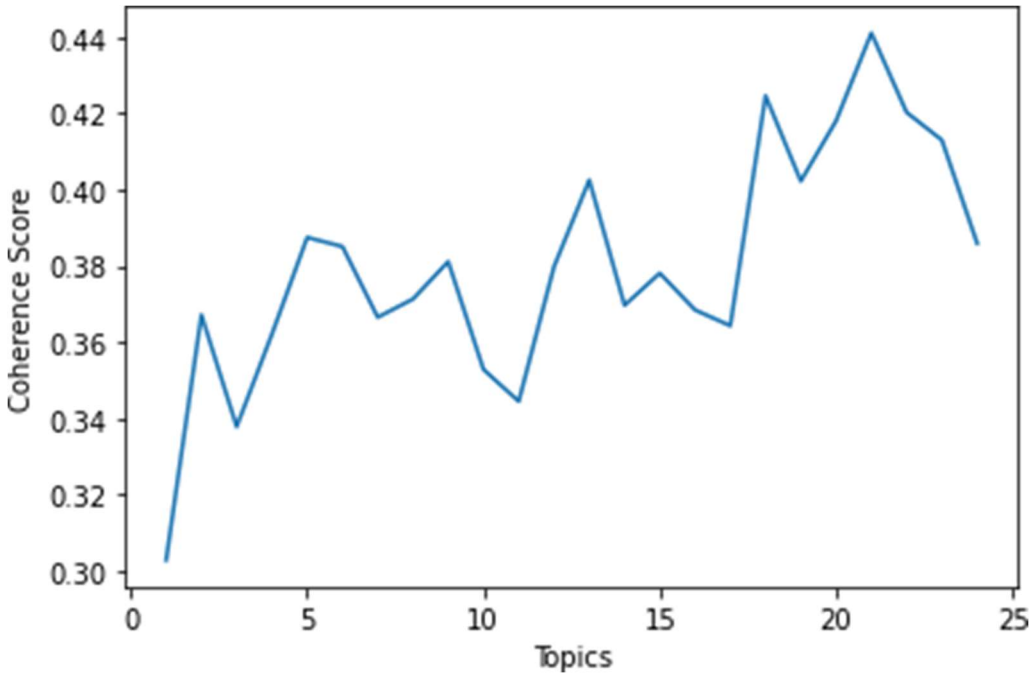


Image 13 -Coherence Score for grouped tweets dataset

If we look the KL divergence score from the graph below (Image 14), we observe that the KL divergence scores for 5 and 6 are too close to decide the optimal value.

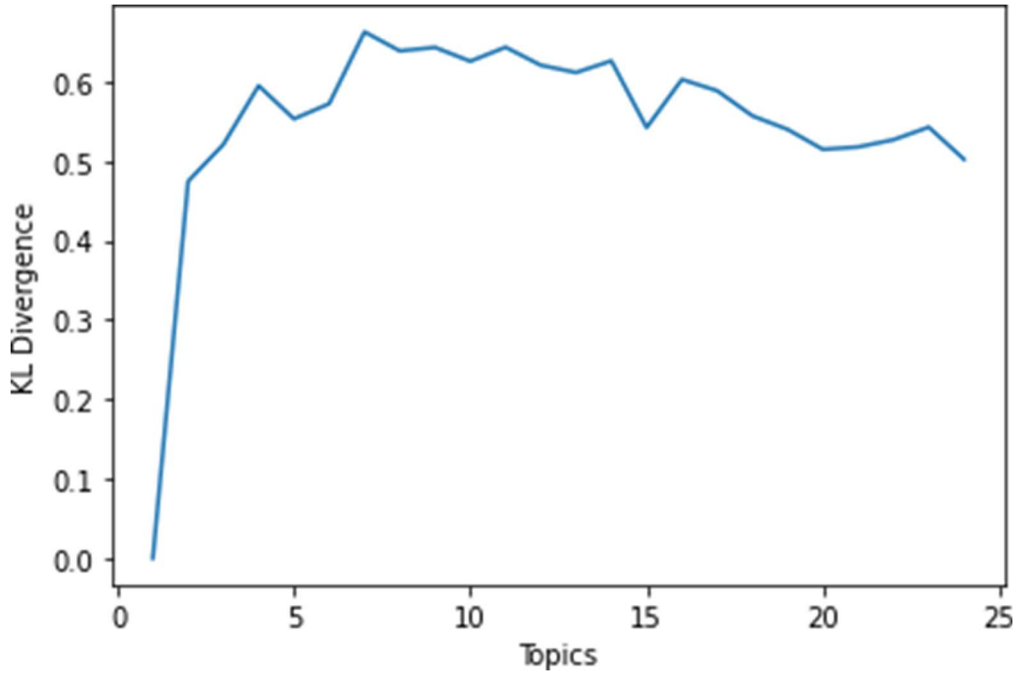


Image 14 – KL Divergence for grouped tweets dataset

Since, it was not possible to determine the optimal number of topics using coherence, perplexity and KL divergence values. Next step was to create word clouds by using 5 and 6 as optimal number of topics. As you can observe from the word cloud below (Image 15) that the 6 topics generated are much more coherent and have some underlying theme. Therefore, the

optimal number of topics is 6 (for grouped tweet dataset).

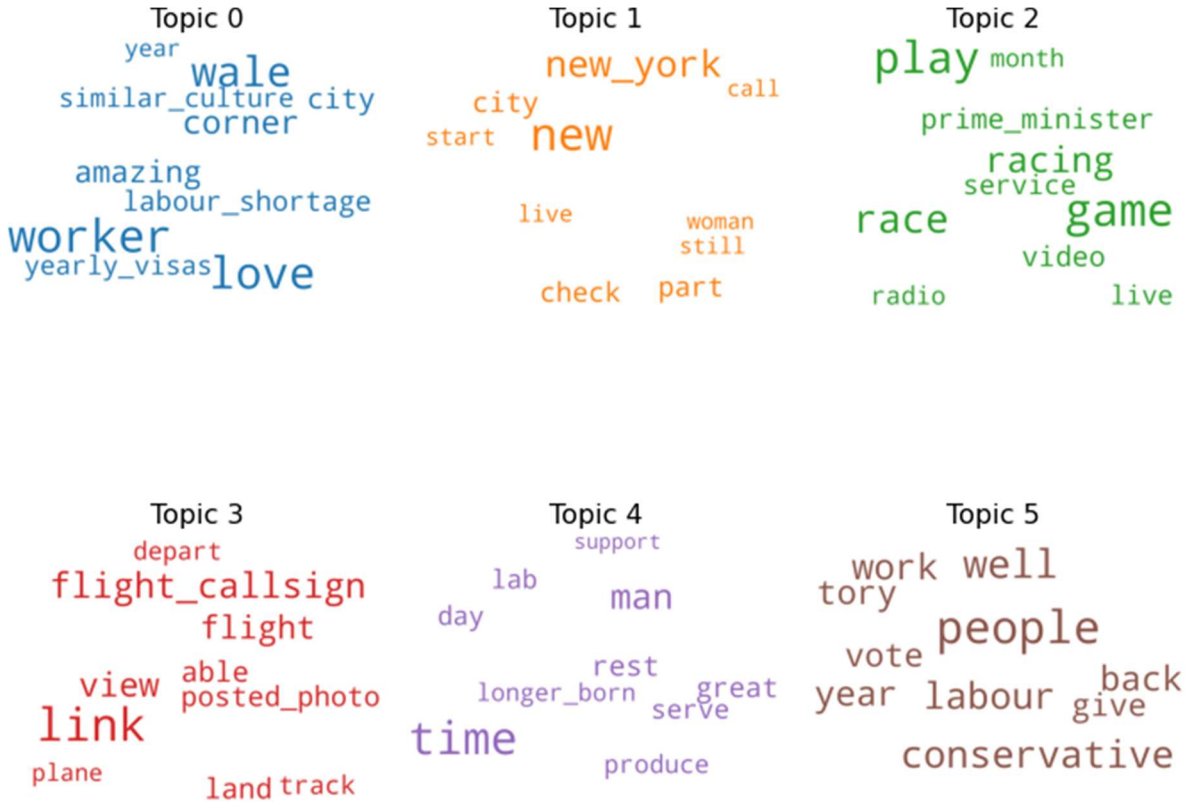


Image 15 – Word cloud for grouped tweets dataset

Evaluating the model

A – Quantitative Methods

Once the model is ready, we can use different evaluation techniques to determine the performance of the model. To start with we will calculate the coherence score, perplexity and KL Divergence of the model.

Coherence Score	Perplexity	KL Divergence
364.23	0.385	0.572

Image 16 – Model evaluation scores

- The coherence of score is 0.384 which indicates that the topic generated by the model will not very coherent.
- Although the perplexity is considerably better than the first model. It still very high for a model to perform effectively.
- Although KL Divergence alone is not a measure of performance of the model but a score of 0.572 indicates that the topics generated by the model will have some level of similarity. The topics will not be completely identical but there would be some words which will be repetitive across multiple topics.

B – Visual Methods

Word Cloud

Image 15 shows a word cloud of top 10 tokens from each topic. The size of the word represents the frequency of the word in that topic. Looking at the word clouds, one can interpret that certain topics have an underlying theme. At times a theme might fit the topic appropriately and on other occasions it might feel a bit far-fetched.

If you look at “Topic 3” the three most prominent words are ‘link’, ‘flight’, ‘flight_callsign’ and other words like ‘depart’, ‘plane’ etc indicates that the topic is related to ‘Air Travel’. Similarly, “Topic 5” contains words like ‘tory’, ‘labour’, ‘conservative’, ‘vote’ and ‘people’ which clearly tells us that the tweets grouped under this topic are related to **politics.** We can also say that “Topic 0” (i.e the first topic) is about **workers and cultures.**

But when we look at examples of Topic 1 the tweets may or may not be about New York city. Topic 4 looks like a collection of random as it is difficult interpret an underlying theme.

Jaccard Distance

If we take top 10 tokens from each topic and plot a heatmap using Jaccadian Distance between each topic then it would result in the following heat map for this model

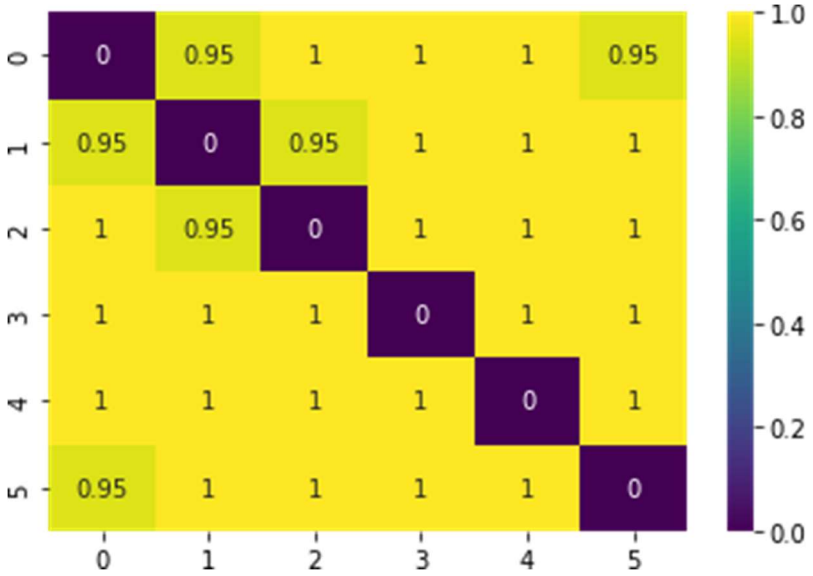


Image 16 – Heat Map for Grouped Tweets Model

We may deduce from the map above that most themes are distinct and do not contain overlapping terms based on the top 10 tokens. Yet, Jaccard distance alone cannot assess how well the topic is formed.

Word count and its importance

For a visual representation of the importance and frequency of words inside each topic, we may also create bars in bar charts. When frequency outweighs weight, this form of representation can help us determine whether popular terms are included in the topic and whether the same phrases are included in several topics.

Looking at the graphs in Picture 17, we can deduce that the topics do not contain many common terms since for the majority of the words in the topics, the weight of the word is greater than the frequency of the words.

The second subject has terms like "still," "call," and other similar words that have high frequency but low weights, indicating that either the word is a frequent one or the model is

missing the context that goes along with it. All of the top 10 words for the final topic share the traits of being low weight and high frequency. This indicates that the topic is poorly defined and that the model's hyper parameter has to be improved in order to provide better groups and topics.

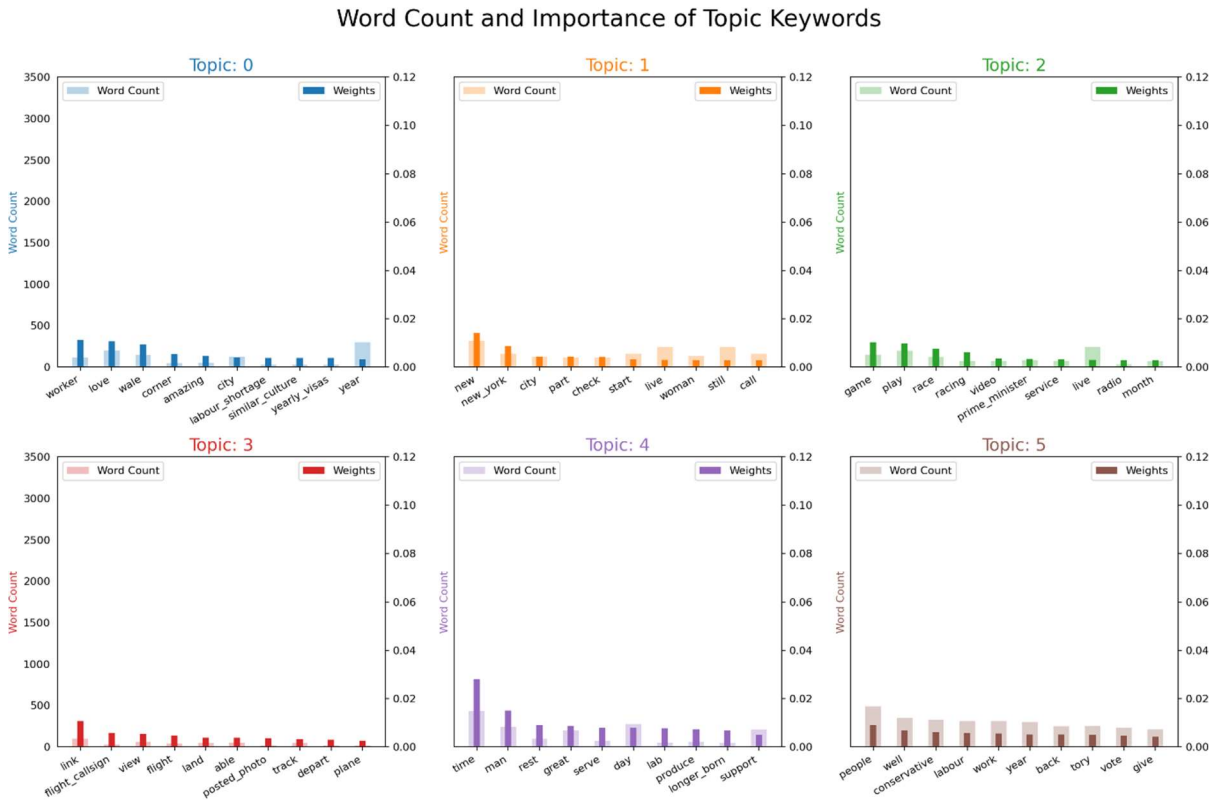


Image 17 – Word Count and its Importance for Model -2

Inter-topic Distance Mapping

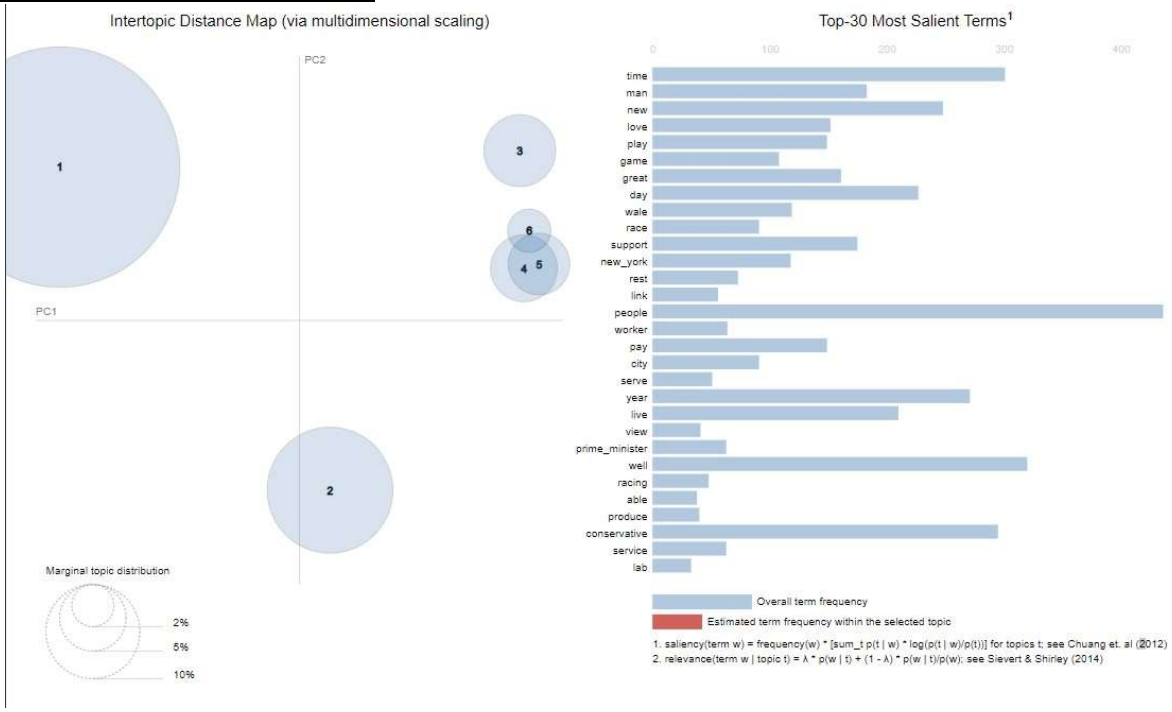


Image 18 – Inter topic Distance Mapping for Model-2

As discussed in the earlier section the inter topic distance mapping is powerful visualization. We can see from the graphic above that the topics are separated into their own quadrants. All connections between Topics 1, 2, and 3 are fully severed. While Topics 4 and 5 exhibit significant topic overlap, it can be deduced from the frequency chart on the right that the majority of these words are frequent ones whose frequency exceeds their weight in a specific topic.

Comparison of Task-1 & Task-2

Introduction

We will compare the two models' performances in this part. To determine which model is superior, we will use both quantitative and visual metrics.

A – Quantitative Methods

Input Data Statistics

We know that the in Task-1 we used a set of 10000 tweets to trains our model whereas in Task-2 we grouped together the same tweets on basis of group number assigned to the tweets. Let's have a look at the statistics of the both the datasets after data pre-processing stage.

count	10001.000000
mean	20.417758
std	12.390628
min	1.000000
25%	10.000000
50%	17.000000
75%	29.000000
max	58.000000

Task-1 Dataset

count	471.00
mean	3235.56
std	10102.88
min	31.00
25%	189.50
50%	594.00
75%	1922.50
max	152602.00

Task-2 Dataset

Image 19 – Dataset Statistics

From the image above, we can that the model in task 1 was trained using 10000 records with an average size of 20 words/record. It shows that 75percentiles of tweets in the dataset have less than 29 words with maximum size of 58 and minimum size 1. Because of the larger dataset and shorter records, the model may encounter more noise.
Whereas in task-2 although the number of records is just 471 but the average record size is approximately 3235 words record. These large documents should be able can provide model with the context to different words which can lead to creation more coherent and useful topics. As much as large documents are useful when using LDA model, there is a potential for overfitting and difficulty in interpreting topics easily.

Coherence Score, Perplexity and KL Divergence

When combined, the evaluation metrics stated in the headline can be really helpful in estimating a model's performance, even though alone it may not be able to determine which model is the best. In the table given below, we have summarised the evaluation metrics for both the models.

Model	Coherence Score	Perplexity	KL Divergence
Task- 1	0.34	910.424	0.746
Task-2	0.385	364.23	0.572

Image 20 – Evaluation Metrics*

Looking at the table, we can see that the model for task-2 has better coherence score which means that the topic generated in task 2 will be much more coherent and will have an underlying structure. The perplexity of task-2 model is only 1/3rd of its opponent which is significant improvement. But the KL Divergence score favours the former (Task-1).

Therefore, it can be suggested that Task-2 with a higher coherence score and a lower perplexity is preferable if the objective is to produce coherent and understandable topics. Lower perplexity suggests that the model is more accurate at predicting the subsequent word given the preceding word, which might lead to sentences that sound more natural.

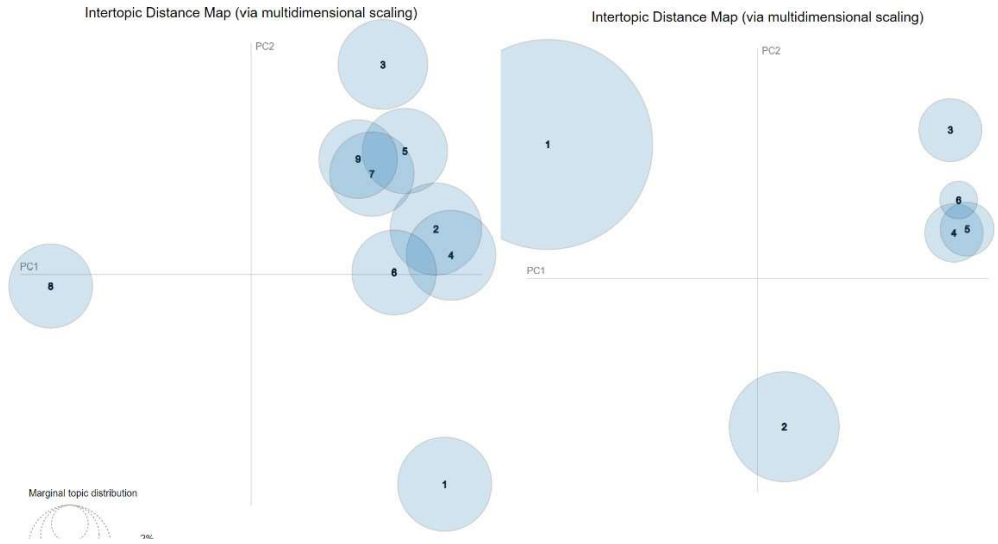
On the other hand, Task-1 with a smaller KL divergence score may be preferred if the objective is to compare the differences between two probability distributions (for instance, in some applications of natural language processing). Lower KL divergence suggests a closer similarity between the two distributions.

Based on the conclusions above we can say that Quantitative measures prove that the model for task-2 is superior.

B – Visual Methods

Intertopic Distance Map

In the earlier section, we have seen the intertopic distance map for both the models. Let us now compare both the maps and evaluate both the models with respect to each other.



IDM for Task-1 IDM for Task-2

Based on the image above, we can draw the following conclusions:

- Two topics are entirely separate in Model 2, demonstrating their uniqueness and the lack of any crossover with other issues. This may indicate that the model was successful in locating distinct and unambiguous themes in the data.
- Additionally, only one topic is slightly overlapping, indicating that the topics are relatively distinct from each other. The other two topics are highly overlapping, which suggests that there is some overlap between these topics, but they are still relatively distinct.
- In contrast, in Model-1, only three out of the ten topics are isolated, which suggests that there is significant overlap between the topics. This may indicate that the model is not effectively capturing the distinct themes within the data.

Overall, the fact that Model-2 has more isolated topics and fewer highly overlapping topics suggests that it has a more effective intertopic distance map, indicating better topic modelling performance.

Now let us select the isolated topics on each map and examine the word clouds corresponding to each topic to understand whether the models are creating good or bad topics.

The image below shows word clouds for isolated topics created in Task-1. We can conclude that these topics look distinct but are very generic and do not have a specific theme. These topics are a good example of poor topic modelling.

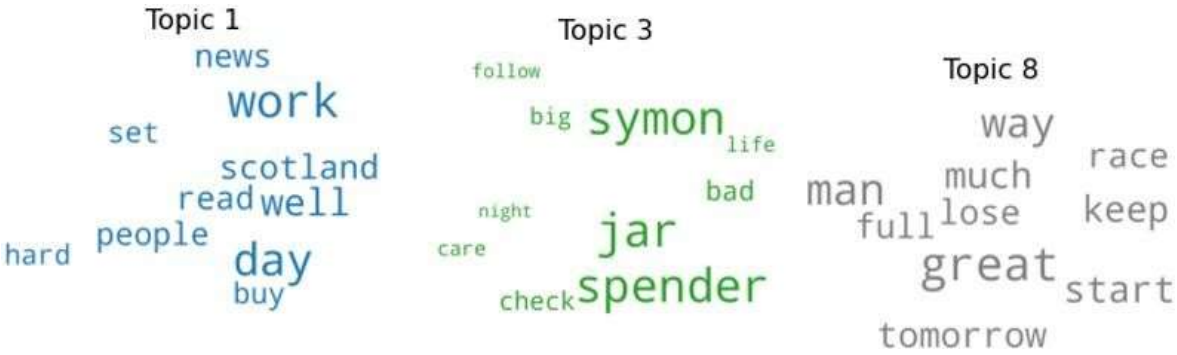


Image 22 – Word Clouds for Isolated Topics (Task-1)

Now let us have a look at the word clouds of isolated topics generated in task-2. By observing the image (23) below, it is safe to say that the first and last word clouds are examples of good topics. The first word cloud is related to work and culture, the last word cloud is about playing games (though prime_minister is an odd word with respect to other words present in the topic) and the orange word cloud in the middle is very generic and has repetitive words like 'live' which also exists in Topic 2.

Image 22 – Word Clouds for Isolated Topics (Task-1)

Word Clouds

Word Cloud is a useful tool to understand how good the topic formation is based on whether a user can interpret the underlying structure or theme of the topic. Image-7 and Image 15are the word clouds for Task-1 and Task-2 respectively.

For Task-1, it is very difficult identify a theme for the generated topics. For Topic-4 and Topic-7 the theme can be identified as political but it is very difficult to identify other topics. Topic 3 and Topic-8are examples of bad topic as words in the word cloud don't seem to relate to each other in a meaningful way.

For Task-2, it quite easy to identify that the words within each word cloud are coherent. For example, Topic 3 contains words that are coherent with flights and planes, Topic 4 contains words which have political context. These topics are examples of good topics.

Identify documents & topic distribution

The data frame shown in the image above highlights the dominating topic/assigned topic for a document as well as its contribution. For certain documents the contribution of the topic is below 0.5 which means that the topic distribution is poor. For example – Document3, Document-4, Document 8 & Document 9 show very poor topic distribution.

	Document_No	Dominant_Topic	Topic_Perc_Contrib	Keywords	Text
0	0	5	0.7819	new, wale, hear, wait, cost, home, match, lie, decision, service	[criterion, certainly, restoration, unlikely, claim, alone, cost, analysis, decision, bfi, else,...
1	1	3	0.6562	labour, government, city, part, link, woman, fund, money, black, find	[balloon, deviate, course, military, threat]
2	2	8	0.6858	support, new_york, really, last, let, mean, always, reason, plan, club	[understand, child, operational, scheme, purposeful, visit, collect, offer, child, love, child, ...
3	3	6	0.3451	tory, still, vote, give, conservative, never, year, party, happen, sell	[excess_death, wale, death, excess_death, register, wale, week, end, year, average, happen]
4	4	0	0.1111	work, day, well, scotland, read, news, people, buy, set, hard	[]
5	5	0	0.7038	work, day, well, scotland, read, news, people, buy, set, hard	[horse_race, history, today, racing, th_february, day, horse_race, news, history, extensive, hor...
6	6	5	0.6759	new, wale, hear, wait, cost, home, match, lie, decision, service	[ridiculous, service, store, remember, order, manager, treat, wait, refund, apparently, promise]
7	7	5	0.6790	new, wale, hear, wait, cost, home, match, lie, decision, service	[live, stream, premier_league, match, head, start, match, part]
8	8	2	0.3277	jar, symon, spender, bad, check, big, life, care, follow, night	[check, snake, eye, dvd_ebay]
9	9	5	0.3332	new, wale, hear, wait, cost, home, match, lie, decision, service	[fable, city, metaphysical, foundation, gradually, abandon, polity, become, ripe, takeover, icar...

Image 23 – Documents and Topic Contribution for Task-1

Now, let us look at the documents and corresponding topic distribution for Task-2. The image below shows a data frame where for most of the document's topic contribution is high. For example- Document-1 and Document-4 are prime examples of good topic distribution.

	Document_No	Dominant_Topic	Topic_Perc_Contrib	Keywords	Text
0	0	5	0.5745	people, well, conservative, labour, work, year, back, tory, vote, give	[criterion, certainly, restoration, unlikely, claim, alone, cost, analysis, decision, bfi, else,...
1	1	5	0.9884	people, well, conservative, labour, work, year, back, tory, vote, give	[balloon, deviate, course, military, threat, newyorkcity, toronto_iran, couple, set, sale, win, ...
2	2	5	0.7008	people, well, conservative, labour, work, year, back, tory, vote, give	[understand, child, operational, scheme, purposeful, visit, collect, offer, child, love, child, ...
3	3	1	0.5614	new, new_york, city, part, check, start, live, woman, still, call	[excess_death, wale, death, excess_death, register, wale, week, end, year, average, happen, prem...
4	4	0	0.9970	worker, love, wale, corner, amazing, city, labour_shortage, similar_culture, yearly_visas, year	[instead, worker, corner, worker, similar_culture, yearly_visas, labour_shortage, worker, corner...
5	5	5	0.5287	people, well, conservative, labour, work, year, back, tory, vote, give	[horse_race, history, today, racing, th_february, day, horse_race, news, history, extensive, hor...
6	6	5	0.7055	people, well, conservative, labour, work, year, back, tory, vote, give	[ridiculous, service, store, remember, order, manager, treat, wait, apparently, promise, experie...
7	7	1	0.7569	new, new_york, city, part, check, start, live, woman, still, call	[check, snake, eye, dvd_ebay, via_check, etsy_shop, often, wonder, explore, much, wale, great, t...
8	8	5	0.5361	people, well, conservative, labour, work, year, back, tory, vote, give	[fable, city, metaphysical, foundation, gradually, abandon, polity, become, ripe, takeover, icar...
9	9	5	0.7808	people, well, conservative, labour, work, year, back, tory, vote, give	[break, law, impunity, shame, hunt, act, unionist, care, check, loss, power, law, rest, include,...

Image 24 – Documents and Topic Contribution for Task-2

To sum up, we can state that the Task-2 model is more effective at topic modelling and will produce topics that are more coherent.

Topic formation issues due to the nature of tweets

This section will cover potential difficulties in identifying important subjects from brief messages like tweets. These are a few potential problems I noticed and ran with while working on this coursework.

- **Short Length of Tweets**
  Tweets, commonly known as Twitter messages, have a character count cap of 280 characters. As a result, tweets frequently lack the information needed to build a meaningful topic because they are too brief.
  One problem with short text topic modelling is the difficulty of correctly identifying and extracting significant themes due to the little amount of text that is available. Tweets and headlines are two instances of brief texts that may lack co-occurring words required for topic modelling. Subjects that are uninteresting or distracting may result from this, making it challenging to distinguish between different topics. For example, this tweet is extremely short and, in most likelihood, will not provide good topics – "Another benchmarker. Brilliant, guys."
  Another example is this tweet where user is using informal language -
  "@gracieslvrr omg yay I'm so happy for u I hope u have the best time!! praying I see someone reselling manchester tickets"

- **Use of slang and abbreviations**
  Another problem is that users frequently utilize slang and abbreviations in tweets because of how brief they are. These slang terms and acronyms are useless for identifying themes. For example, this tweet -
  "@gracieslvrr omg yay I'm so happy for u I hope u have the best time!! praying I see someone reselling manchester tickets"

- **Context Ambiguity**
  It is often difficult to understand the emotion and feeling behind a tweet. Tweets frequently contain contextual ambiguity, which means that the meaning of a tweet may vary depending on the discussion taking place nearby or the larger cultural context. Due to this, it may be challenging for topic modeling algorithms to precisely identify and group related subjects. For example, this tweet from @CosyWarmPlumber - "@LondonGas Shame cos i like your stuff" does not provide any context about what exactly does the user like about LondonGas.

- **Nosiy Data**
  Tweets frequently include pointless information in the form of hashtags, URLs, etc. This means that a short text about topic modeling includes a lot of pointless information. For instance, the data in the accompanying tweet will be removed during pre-processing, leaving only a small set of words.

  "@thewriterswife @PeterTFortune It looks like it, to me.
  https://t.co/svvWf2JdmM
  https://t.co/1Xq95Eg3PE"

- **Hashtag Overflow**
  Twitter is the hub of hashtag trends. Every tweet has a hashtag associated with it. But some of the tweets are overloaded with hashtags. This means that these tweets are not able to contribute towards the task of topic generation. An example of hashtag overflow is shown below :-
  "@David__Osland We Need a #GeneralElectionNow #EnoughIsEnough #NonDom #GreenCard @Infosys in #Russia #Moderna #michellemone #fraud #DominicRaab #LiarJohnson @ExcludedUK #PMQs Questions 4 #Sunak #ToriesUnfitToGovern
  #Sunakout
  Petition https://t.co/z7Aaaw5iQB
  https://t.co/1ihUmVItco"

- **Dynamic Nature**
  Twitter is an open public platform where people can discuss and debate about all the different types of topics which means that Tweets are a highly dynamic, and subjects and trends are constantly changing. This can make it challenging to develop stable and meaningful topic models that can be applied over extended periods. It can be observed in the examples shared above that tweets can be categorized into a wide range of categories like sports, politics, health, etc.