Colab Link:

https://colab.research.google.com/drive/1eKFi6GHZGm3IQO5-h9t-Au-TbyUW6m-1?usp=sharing

Group Members: John Chen, Jing Yu

Breakdown: All the work was done jointly.

Part 1:

D. If we are analyzing tweets or other text that includes hashtags or mentions (e.g. "@user"), we might want to keep those characters so that we can analyze trends around specific topics or users. Also, if we are analyzing sentiment in social media data where certain forms of punctuation such as exclamation marks or question marks may be indicative of strong sentiment or intent, then removing all punctuation would remove important information that could affect the accuracy of the analysis.

I. No, fitting a ROC curve does not make sense since the number of classes is not binary. Anyhow, we can provide a ROC curve for a new model that only looks at two classes of the three.

J. The accuracy rate is smaller with TF-IDF vector transformation.

K. The accuracy rate is the same for both lemmatization and stemming.

Bonus: NBM is a generative model because it calculates the joint probability of input and output. NBM assumes the input and output are independent of each other.

Part 2:

Topic Analysis

Introduction:

With the overwhelming amount of tweets on Twitter, nowadays, it is hard to focus on the topics that have been talked about on social media. It might even be hard for you to judge which topics were important to notice. How can you distinguish which tweets are irrelevant? The only way to do this correctly is to read more than 10,000 tweets and summarize them into topics, which seems impractical. Unless you want to read and memorize all tweets in the past months, topic modeling might be the front-runner to solve this problem. The data set used in this report is tweets extracted from Twitter. It includes the text body of each tweet. It will be analyzed through topic modeling to find the topics that are prevalent in the corpus. In “Latent Dirichlet Allocation”, the authors described a modeling method used to spot prevalent topics in the text.

Data description:

The data are tweets extracted from Twitter. There are 10,000 observations with 7 attributes in the table: Tweet text, favorite count, creation time, retweet count, user statuses count, user screen name, and user followers count. Only one of those features is relevant to this report: Tweet text. For the cursor used to extract tweets, the parameters are api.search_tweets(), q = “#cdnpoli + -filter:retweets”, and lang=”en” to search for tweets within the past 6 to 9 days with the hashtag “cdnpoli” in the English language. Retweets were excluded to prevent duplicates of the same tweet. After surfing the web, there have been similar projects in the past that used Latent Dirichlet Allocation to extract topics from tweets. Since LDA is a generative unsupervised model and our data sets contain no similar tweets, it was expected that the results of this report would be different from the projects on the web. There seem to be no limitations to data as all values are present and the data has a sufficient amount. From the EDA stage, each tweet length has about more than 20 words. We will remove tweets with less than 6 words since they don't carry any meaning.

Exploratory data analysis:

In this stage, we checked for missing values and data that should be filtered in the preprocessing stage of this analysis. We plan to use LDA, so we must remove non-meaningful words. One of the most significant graphs we produced was the histogram of tweet length. Since the graph is skewed to the left, most of the tweets had significant numbers of words with the lowest word count being greater than 20 words. We can assume that all tweets have meaningful context in this case and no tweets need to be removed in the preprocessing stage. We also generated a word cloud of the most common words found in the tweet text. HTTPS is the most common word along with some other URL substrings. We should remove URLs in the preprocessing stage since URLs provide no meaningful context. The other words on the word count are mostly words that relate to politics. This is expected since only tweets with #cdnpoli were queried. The word cloud also indicated some stop words that needed to be removed.

Machine learning model description:

The machine learning model we used is called Latent Dirichlet Allocation (LDA), which is a generative probabilistic model for topic modeling. It is based on the assumption that each document is a mixture of various topics, and each topic is a probability distribution over a fixed vocabulary. In other words, each document can be represented as a mixture of topics, and each topic is characterized by a set of words that are likely to occur together. The LDA algorithm takes as input a set of documents and outputs a set of topics, along with the words that are most likely to occur in each topic. It does this by iteratively updating the probability distributions of the topics and the words in each document until it converges to a stable solution.

Since our goal is to cluster the tweets into topics without prior knowledge of what the topics are, LDA is an appropriate choice for our research problem, which is specifically designed for topic modeling. As an unsupervised model, one of its strengths is its ability to identify latent topics in a document collection without prior knowledge of the topics, which means that we do not need to label the data beforehand. However, one of its weaknesses is that the number of topics must be specified beforehand, which can be a challenge if the optimal number of topics is not known. Additionally, LDA assumes that the documents are generated by a fixed set of topics, which may not always be true in practice.

To evaluate the performance of our LDA model, we use a combination of perplexity and coherence. Perplexity measures how well the model predicts new data and a lower perplexity score indicates a better model. Coherence measures the interpretability of topics generated by the model and a higher coherence score indicates a better model. To ensure our model performs well, we choose the number of topics that gives the best balance between low perplexity and high coherence, and then we decide the interpretability of each topic manually. As an unsupervised learning algorithm, LDA does not have a direct baseline model. However, it is possible to compare the performance of our LDA model with other topic modeling algorithms.

Results and Conclusions:

Our analysis extracts 9 topics from a total of 50 topics generated by LDA, which are prevalent and interpretable in the corpus. These topics are “Liberal MP Han Dong resigns”, “David Johnston”, “Shambhavi Anand broke Indian abortion law twice”, “Twitter blocks access to accounts of prominent Canadians”, “Foreign interference”, “Fake scandals roil Ottawa politics”, “U.S., Canada kept migrant crossing deals a secret to avoid the rush at the border”, “Pierre Poilievre is wrong about inflation”, “The World Leaders Attending The Queen’s Funeral”, “U.S. President Joe Biden visited Ottawa”. One insight is that most of these topics are extracted by names of politicians or terms in politics. Therefore, domain knowledge would be helpful in extracting some domain-specific topics in the corpus.

Our LDA model performed well, with a coherence score of 0.525 and a perplexity score of -17. However, before gaining these results, we ran into many issues. First, we built the LDA model using the scikit-learn library, but the perplexity score unexpectedly increases as the number of topics increases. By surfing the web, we found lots of people had the same issue, so it may be a bug in that library. Instead, we used the Gensim library to build the model and calculate both scores. The perplexity score is negative since the library uses a negative log-likelihood. Although the score decreases in this model, it is difficult to choose an appropriate number of topics since it continues to decrease significantly even for a large value. This may be due to overfitting of the model, where the model is capturing noise in the data rather than the underlying themes. Therefore, we evaluated the coherence and interpretability of the topics generated. The coherence score stops significantly increasing at 50, so we chose the parameter of the number of topics for our LDA model to be 50, which reflects the number of distinct themes that emerged from the tweets.

After that, we used the idea of word intrusion to evaluate the interpretability of each topic manually and extracted 9 meaningful topics from them. Our results provide insights into the most prevalent topics with the cdnpoli hashtag on Twitter in the last few days. They reflect the political hot spots that Twitter users follow.

Our LDA model worked better than some approaches for identifying the most frequent topics discussed on Twitter. As a model method of topic modeling, the unsupervised model requires less manual input than supervised algorithms, such as topic classification. That’s because they don't need to be trained by humans with manually tagged data. This ability makes it ideal for analyzing social media data, which is often unstructured and challenging to categorize. However, they do need high-quality data and need it in bucket loads. Otherwise, the model may not be as accurate as the classification algorithm.

If we had more time or could collect more data, we would explore different techniques to improve the performance of our model. One approach could be to explore other parameters of the LDA model to improve its performance. Another could be to use a more advanced text preprocessing technique that can handle emojis, slang, and abbreviations commonly used on social media. Furthermore, we could analyze the sentiment of the tweets associated with each topic to gain further insights into the attitudes and opinions of Twitter users.