Project Report

SDS 3386 | Fall 2022

Professor: Tanya Schmah

Francophones vs Anglophones: a Tweet sentiment study in the Ottawa-Gatineau region

By Team Light:

Yephihy Akissi Paule Noura Offia [300201661]
Mershab Issadien [300027272]
Lendl Lapointe [300187898]

Content

Introduction

Our original idea was to study the difference of sentiment between francophone and anglophones Twitter users in the Ottawa-Gatineau region during the pandemic.

But we did not get access to historic tweets older than a week, so we choose instead to analyse tweets that we would pull progressively, each week.

From November 14 to December 16, we were able to pull 4000+ tweets.

Motivating question

- **Are there any differences in sentiments between tweets in French and those in English?**

It also led us to ask ourselves these questions:

- What is the general mood of the tweets?
- Are users from one particular area more likely than the others to post negative tweets ?
- Are users tweeting in one particular language more likely than the others to post negative tweets?

Prior studies

Twitter Sentiment Analysis on Coronavirus: Machine Learning Approach

Published in February 2021 by Cristian R. Machuca, Cristian Gallardo and Renato M. Toasa

This study is the most similar to our original objective that we found.

It involves a sentiment analysis of English tweets during the COVID-19 pandemic from January 2020 to July 2020. Each month, the 50000 top tweets with the #coronavirus hashtag were downloaded. Using a Logistic Regression Algorithm to classify the tweets as positive or negative, they were able to attain a classification accuracy of 78.5%

They found out that people mostly remained positive during the pandemic, with 54% of the users showing positive feelings

Methodology

Data collection

The tweets were collected using a combination of the document database MongoDB, and the Python tweepy library. Supplying the API access token and secret credentials, we were able to pull in tweets to our machines locally, and save them onto the database. Access to our public github repo here, you see the Python codebase with the data input service.

We created a Datastore class in Python that not only allowed us to easily retrieve tweets in a json readable manner to convert into DataFrames, but also to retrieve tweets by field, and checking if they exist. All tooling built on top of the pymongo library to facilitate data retrieval for visualization.

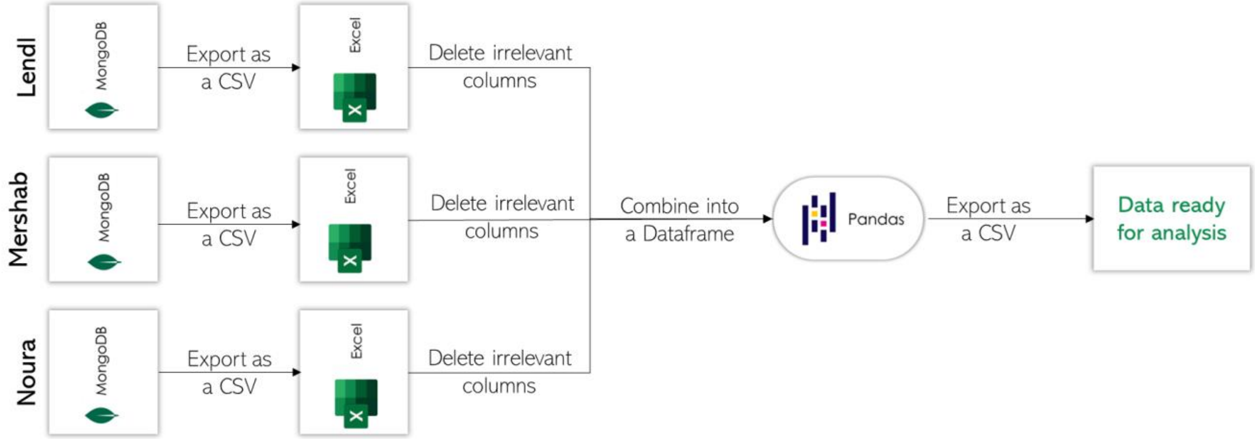
Data wrangling

Each one of us had different tweets stored in their local machine using MongoDB.

To merge the three databases (one from each member), we had to :

- Export our MongoDB documents as a CSV
- Clean the CSVs using Excel (Delete irrelevant columns)
- Use Pandas to combine all the CSVs into a DataFrame
- Export the DataFrame as a CSV

The final CSV exported is the one used to run the analysis using Jupyter notebook.



After running the classifier algorithms on the tweets, we save the results as new columns of the input DataFrame and export it as a csv.

Visualization

- **Word Clouds**
  Using the word cloud package , we were able to generate different word clouds (global, English and French) containing the most tweeted words.

  To eliminate irrelevant words (such as 'the', 'la') from the graph, we had to define stop words (words to ignore). We used a combination of built-in french and English stop words from the nltk.corpus package, as well as a custom list of 500+ words.

  We designed a custom package named 'word.py', to make the process smoother.

- **Time Charts**
  While some time charts were made using the plotly.express package, the majority of the time charts were made with Altair

  We had sometimes to preprocess the data using

- **Geomaps**
  Using the geopandas library to convert the places.csv dataframe into a GeoDataFrame was quite simple, and and all the same DataFrame techniques are applicable.

- **Pie Charts**
  While we intended to use Altair to generate pie charts, pie charts are not supported by our version of Altair.

- **Others**
  The remaining charts were made using Altair. PowerPoint was sometimes used to annotate the graphs.

Modeling

- We mainly used the **bert-base-multilingual-uncased-sentiment** made by nlptown.

This model was pretrained on product reviews and gives as output the positivity level of a text , on a scale of [1,5] , 1 being the most negative and 5 the most positive.

To make things smoother, we will refer to this model as the 'BERT classifier'. Its accuracy is as following:

Language	Accuracy (exact)	Accuracy (1 level off)
English	67%	95%
French	59%	94%

[i]

- To have another point of reference, we used another algorithm: **distilbert-base-uncased-finetuned-sst-2-english** by Hugging Face. We will call it the 'DILBERT classifier to make things short'.

We have to note that since this classifier was only designed for English text, we used the English tweets results to compare the two classifiers.

- We then modeled our results using **Logistic Regression** to predict them given the language of a tweet or the location of its author.
  We chose the logistic regression since the output we tried to predict is a binary variable, negative or positive

All the statistical tests were done using the BERT classifier results

Statistical tests

We analyzed multiple ideas regarding what type of statistical analysis we should use for this project. In the end, we chose to run a **chi-square test**.

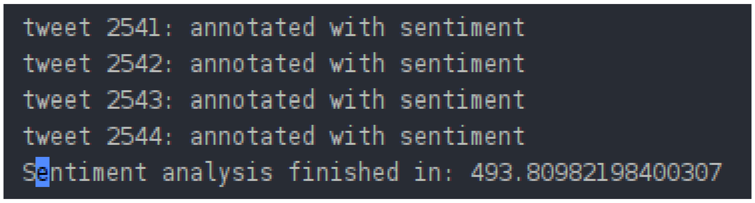
We applied the chi-square test because we tried to find an existing relationship between two categorical and qualitative variables: language, and sentiments.

Technical difficulties

As each of our machines had a different dataset of tweets, we had many problems at merge time. There were difficulties performing a mongodump and mongorestore procedure (standard database backup and restore), so we settled on using the built in MongoDB Compass GUI tool's CSV exporter. However, this caused many inconsistencies between our datasets and upon merge, amassed into a Collection (documents are grouped by Collections) which had multiple different data formatting types. Making the DataFrame conversion impossible to implement with the merged data through this method.

As a solution, first we got rid of the problematic tweets, then we wrote a custom ETL (Extract Translate Load) layer to clean and convert raw tweets into JSON (considering our desired fields) and creating a unified csv load and merge protocol.

Another technical problem we encountered is that the sentiment classifier models were too heavy to run on some of our team member's laptops. This was solved by running the classification of tweets on a powerful desktop machine, while speeds increased slightly, classification of tweets still took 493.8 seconds for 2544 tweets.



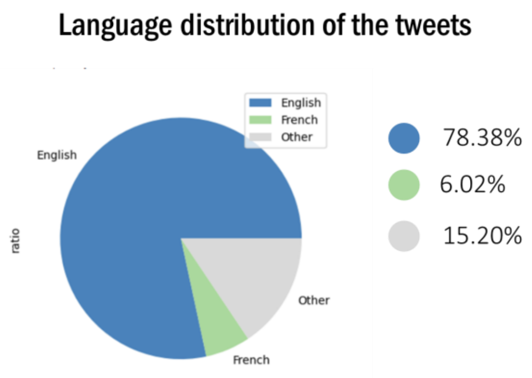
```
tweet 2541: annotated with sentiment
tweet 2542: annotated with sentiment
tweet 2543: annotated with sentiment
tweet 2544: annotated with sentiment
Sentiment analysis finished in: 493.80982198400307
```

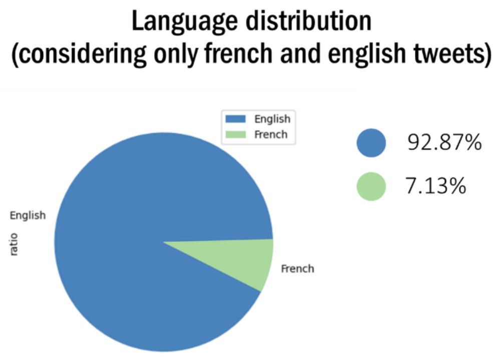
This lack of performance is due to the use of CPU to perform inference on the classifier models. We moved to a GPU based model inference and saw speed improvements of up to <GPU_TIME>. This allowed us to iterate easily over fresh tweets that come in and generate new master_tweets.csv dataframes for analysis consumption.

Results

Overview

Language Distribution



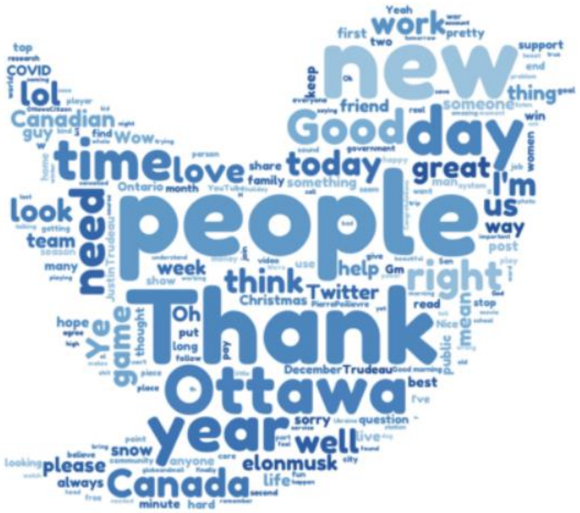


There are significantly more English tweets in our database.

While “other” languages make 15.20% of the total tweets, its is because “other” includes tweets with an undetermined language.

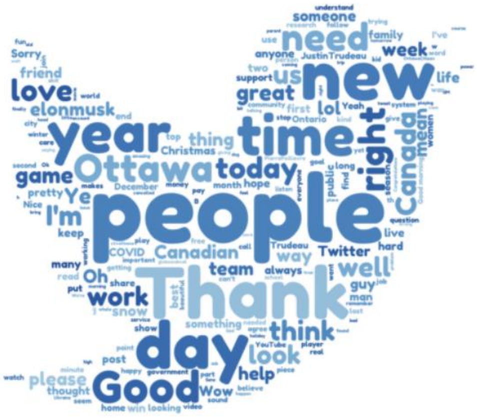
Most common words

Global

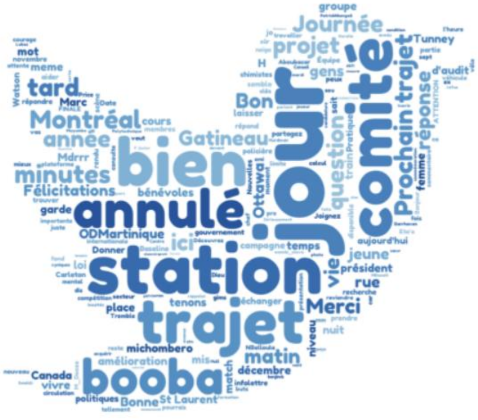


- It’s not really a surprise that the most common words of the database are in english, given that more than 90% of the tweets are in that language.
- When looking at the expressions usually used in conversations (thank,sorry, please, ye, yeah, lol, wow), we can see that most of the interactions between users are on the friendly side.
- “Ottawa” might indicate that users tend to tweets about news or events related to this city. But it could also be from the tweets made by official services, news or shops accounts
- Fun fact: Elon Musk’s handle “elonmusk” made it to this chart. Seems like many users interacted with this account !

English vs French



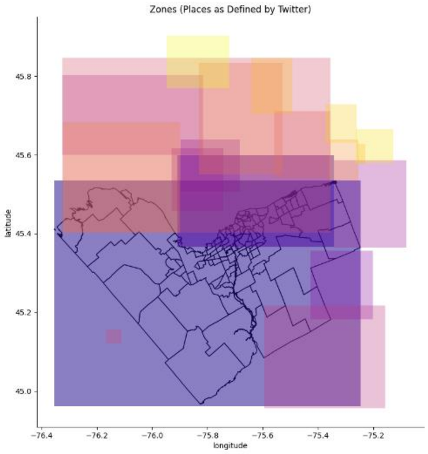
English

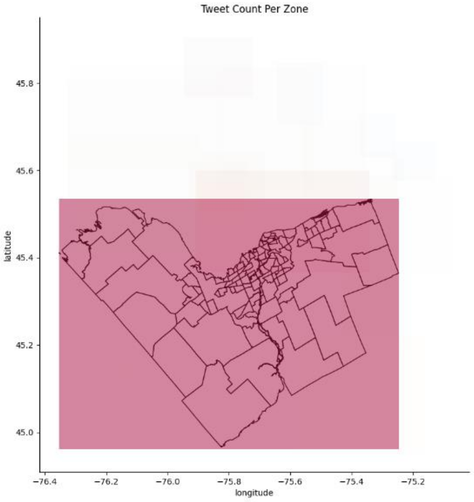


French

- Ottawa is more mentioned in the English tweets
- In the French tweets, the most mentioned places are instead Montréal and Gatineau. Ottawa is still mentioned, but on a smaller scale
- “booba” is mentioned that much because of a reaction thread to one of the artist’s tweet
- With the words “Mdrrr”, “Félicitations” and “Merci”, we can see that the interactions are friendly too between francophones users

GeoMaps



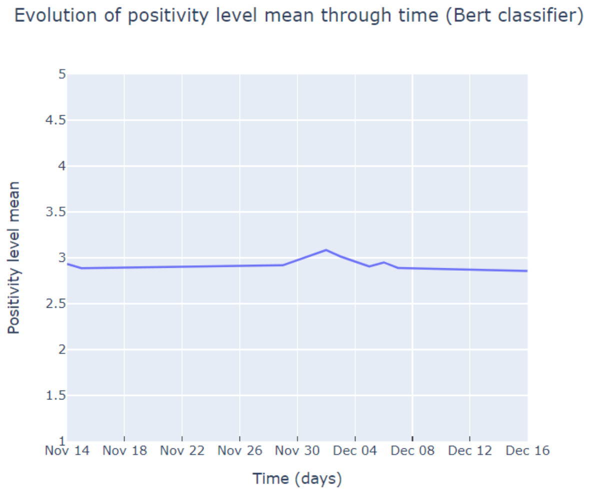


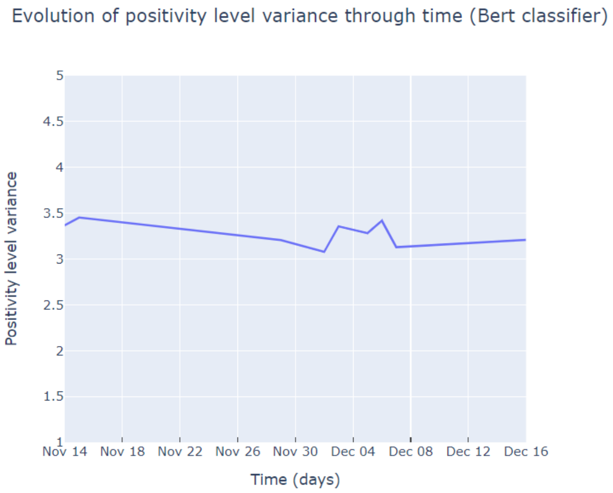
Shown above are the Twitter Place API bounding boxes for each zone. Unfortunately, as shown in the right figure, majority of the tweets are labelled as coming from the greater Ottawa Area, and not as fine grained as we would have hoped. This is likely because many of the tweets were not geotagged, and therefore we were reliant on the user’s self-reported profile “location”. In addition, public geomap data for Quebec is limited, and we were therefore unable to recreate Quebec lines.

BERT vs DILBERT

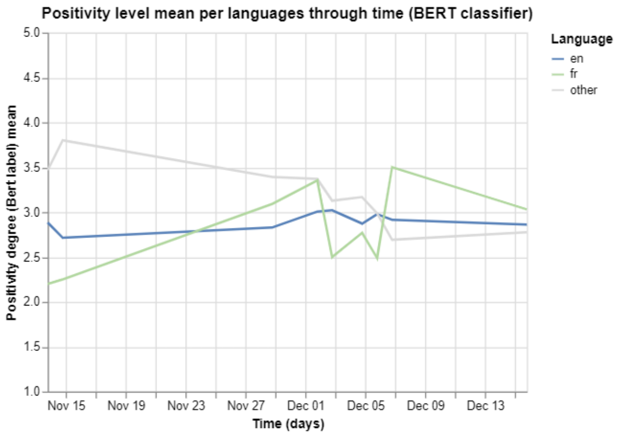
Evolution of positivity over time

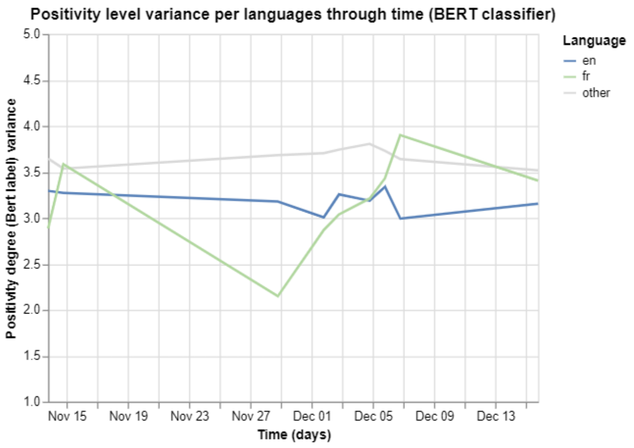
BERT Classifier


Evolution of positivity level mean through time (Bert classifier)


Evolution of positivity level variance through time (Bert classifier)

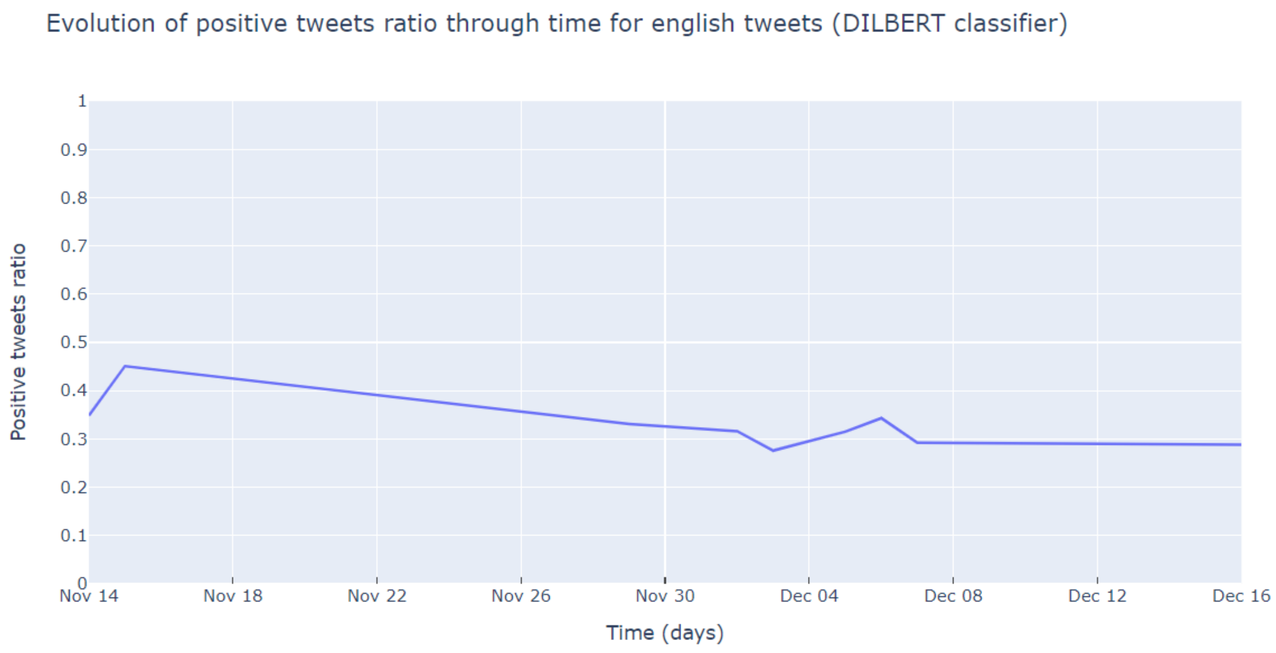
Seeing the evolution of its mean and variance, the positivity level seems to be stable through time. The mean is always around 3, while the variance remains close to 3.5. That indicates that the tweets (according to the BERT classifier), are neutral (but are more inclined to be positive) over time.


Positivity level mean per languages through time (BERT classifier)


Positivity level variance per languages through time (BERT classifier)

Perhaps because of the smaller count of tweets, the positivity level of the French tweets is irregular.

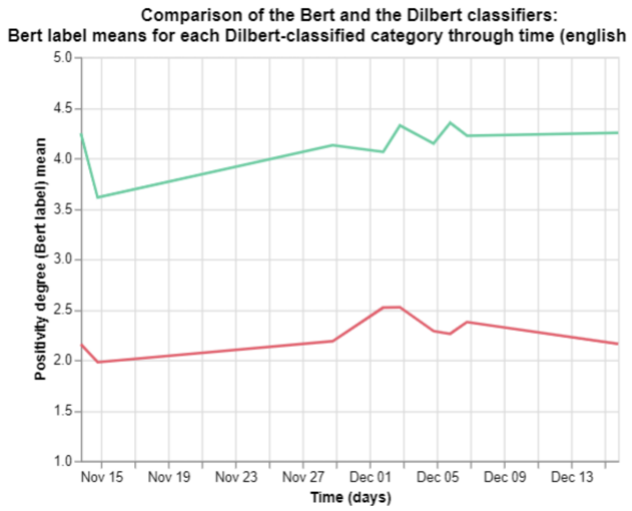
DILBERT Classifier (English tweets only)



$$Positive\ tweets\ ratio = \frac{Number\ of\ tweets\ classified\ as\ positive}{Total\ number\ of\ tweets}$$

Since this ratio stays lower than 0.5, according to the DILBERT classifier, the tweets are more negative through time. It could be explained by the fact that the DILBERT classifier has a stricter labeling: there is no “neutral” tag.

Coherence between the classifier’s outputs



Overall, it seems that the algorithms results are coherent:

- Tweets classified as positive by the DILBERT algorithm are on the positive-neutral spectrum of the BERT algorithm output
- Tweets considered negative by the DILBERT algorithm are negative-neutral according to the BERT classifier

We can conclude that **the BERT classifier is reliable enough** since its output is consistent with the DILBERT algorithm results.

Statistical inference:

Chi-square

For the chi-square tests, the hypothesis are as follows:

H0: there is no relation between the language behind the tweet and the language behind them.
H1: there is a relation between the language of the tweet and the sentiments behind them.

Chi square table

Score(sentiment)\language	English	French
1	793	60
2	50	7
3	215	17
4	167	23
5	609	43

With a significance level of 0.05, we found that there was no relationship between the language of the tweets and the sentiments behind them.

Since the p-value was 0.07, we couldn't reject the null hypothesis; therefore, there is no relationship between people's tweets and the language they use to express themselves. On another hand, running a post hoc for the pairwise comparison test confirms that there is no difference between tweets in English and the ones in French.

Logistic regression

First, the data was split into a train set and a training set, test size is 0.3 with a specific seed. Negative and positive responses are changed respectfully to 0 and 1, Ottawa is equal to 1, and Gatineau to 0, which gives us this model:

$$Sentiment\ (negative\ or\ positive)\ =\ intercept +\ B1\ (language)\ +\ B2\ (location)$$

Once the logistic regression model is applied to the data, we find that:

$$Sentiment =\ -0.23 + 0.28(language) - 0.35(location)$$

When applying the odds ratio (exponential of the coefficient), we obtained those interpretations:

- People who tweet from Gatineau in French have **0.71** times the odds for their tweets to be negative compared to people who tweet in Ottawa in English

- People who tweet in English have **1.33** times the odds to tweet a negative tweet compared to people who tweet in French. In other words, higher chance for their tweets to be negative.

- People who tweet in Ottawa **0.71** have times the odds to tweet a negative tweet compared to people who tweet from Gatineau. In other words, a lower chance for their tweets to be negative.

Confusion matrix

172	2
135	2

True positive predictions: 172
True negative predictions:2
False positive predictions:135
False negative predictions:2

Which gives us an accuracy score of 0.56

Conclusion

- There is no significative difference between the tweets in French and those in English.
- People tweeting in Ottawa are less likely than those in Gatineau to post negative tweets
- The mood of the tweets is neutral in general. That said, it tends to be positive, considering the variance of the positivity level , and the friendliness of most of the interactions between users
- People tweeting in English are more likely than those tweeting in French to post negative tweets.

Our logistic regression model is only 56% accurate. It means there is room for improvement.

The BERT classifier was determined to be reliable enough, seeing the coherence of its results and the DILBERT classifier output.

Ethical concerns

Because we used the Twitter API, we were able to collect tweets even if they came from private accounts. That means we likely used tweets that some users were not comfortable with them visible to everyone.

Also, because of the smaller number of French tweets, when doing the French word cloud, we obtain a figure full of user handles. We had to include these handles to the custom word cloud list because we judged them not relevant. But that means that anyone that checks the word.py script can see these handles, causing an anonymity issue. Even though the handles are not easily recognizable among the other stop words, because some of the stop words were not real word, but rather random combinations of letters and/or digits (example: '9f95piqvir').

Contributions

While each member had to deal with all the steps of the project, these are the areas in which they contributed the most

Noura: Visualization, Data wrangling, Classifiers comparison, word.py

Mershab: Data collection, Twitter API ETL, Classification Pipeline, Heatmaps

Lendl: Statistical test, Modeling, Conclusions

References

Geopandas docs. GeoPandas 0.12.2 - GeoPandas 0.12.2+0.gefcb367.dirty documentation. (n.d.). Retrieved December 17, 2022, from https://geopandas.org/en/stable/index.html

Machuca1, C. R., Gallardo1, C., & Toasa2, R. M. (2021, February 1). *IOPscience*. Journal of Physics: Conference Series. Retrieved December 17, 2022, from https://iopscience.iop.org/article/10.1088/1742-6596/1828/1/012104

Mongodb Compass. MongoDB. (n.d.). Retrieved December 17, 2022, from https://www.mongodb.com/products/compass

Tweepy Docs. API - tweepy 4.12.1 documentation. (n.d.). Retrieved December 17, 2022, from https://docs.tweepy.org/en/stable/api.html

Wordcloud. PyPI. (n.d.). Retrieved December 17, 2022, from https://pypi.org/project/wordcloud/

Our Repo:
https://github.com/Mershab99/SDS3386Final

[i] bert-base-multilingual-uncased-sentiment documentation on Hugging Face