Group 15 - Project Final Report

Authors: Chua Sim Nee | Lee Yu Xian Grant | Liew Jun Wei | Low Ee Ter (NUS School of Computing)
Email: chuasimnee@u.nus.edu | grant@u.nus.edu | e0309430@u.nus.edu| e0310486@u.nus.edu

Abstract

Every second, an average of 6,000 tweets are posted on Twitter, with many indicating some form of emotion. With these tweets, we hope to accurately determine the emotions embedded within them using natural language processing (NLP) techniques and be able to **generalise** the emotional sentiments attached to different topics. Using this model, we aim to design a tool that generates sentiment reports which could prove to be useful in the work of researchers. With that in mind, we conceptualised and implemented our own preprocessing methods and compared between **Support Vector Machine**, **Multinomial Naive Bayes**, **Random Forest** and **k-Nearest Neighbours** classifiers to build, train and tune our model.

1 Introduction

There has been significant research in recent years regarding sentiment analysis on various text formats. However, most of these studies tend to be conducting sentiment analysis on longer and more formally written text formats, such as news articles, reviews and forum posts.

Being avid social media users, we were inclined to focus our research on tweets instead. We hope to be able to derive several insights from the textual data embedded within tweets. As thousands of tweets are generated daily, we believe there is more than sufficient tweet data available to support our study.

After conducting a literature review (discussed in section 2), we found that most current sentiment analysis research that has been performed on tweets have been limited to classifying the tweets with only a positive or negative sentiment (Mohammad & Bravo-Marquez, 2017 and Gaind & Padgalwar, 2019). It is undeniable that emotional sentiment analysis on text is harder to achieve as compared to a positive or negative sentiment analysis on text (Ullah et al, 2017).

Emotions are subjective and complex (Lindquist & Barrett, 2008). There are pitfalls when it comes to emotional sentiment analysis on text, especially for short-length text like tweets. Sarcasm, double negation, multi-polarities or the lack of context can lead to very different interpretations of a single tweet.

Therefore, we hope that given a context, we will be able to accurately determine the composition of emotions embedded within any particular tweet. In particular, we would like to find answers to the following questions:

1. When analysing a tweet, what are the best features that describe the emotional sentiment?
2. For a particular emotion, what are the most common expressions used to describe it in tweets?
3. Given a particular topic, what are the most common emotions expressed in tweets?

The distinctive aspect of our project is that we intend to use our generalised model (trained on a generalised set of tweets) on tweets about a specific topic and study the effectiveness of our model in deriving the emotional sentiments surrounding that topic.

The goal of our project is to seek answers to the above mentioned research questions. To do so, we plan to implement various features and compare model performance to identify the most effective features. We will also implement algorithms to extract from tweets the words with the highest frequency count for each of the emotions. Lastly, we will run our trained model on tweets about various topics and derive the distribution of emotions about the topics. We will be discussing these in more detail in section 7.

2 Related work

In order to gain a clearer understanding of where current research stands, we conducted a literature review focusing on two main themes:

1. Sentiment analysis on online text sources
2. Methods to perform emotional sentiment analysis

Turney (2002) used a lexicon-based approach in order to identify the polarity of reviews. Fei et al. (2012) suggested a dictionary-based technique that uses only adjectives to perform a similar analysis. Also working with movie reviews, Pang & Lee (2002, 2004) extracted unigrams, bigrams, a combination of both, parts-of-speech (POS) tagging and adjectives as features. They found that a Naive Bayes (NB) classifier tends to work better than a support vector classifier (SVC) for smaller feature spaces, whereas SVC tends to outperform the NB classifier as the feature space increases in dimensions. Other features used by researchers include word embeddings as well as term frequency-inverse document frequency (TF-IDF) (Yu, 2008). Decision trees are also popular models used by sentiment analysis researchers, namely Barros et al. (2013), Kim et al. (2017) and Henny-Krahmer (2018).

Rout et al. (2018) performed their sentiment analysis on tweets and discovered that what worked best for shorter tweets was a Multinomial NB (MNB) classifier using unigrams as a feature. They also suggest a number of data preprocessing steps. Firstly, they suggest the removal of non-English tweets. In addition, they also suggest the removal of URLs, question words, stop words, special characters, hash symbols and retweets (prepended with "RT") as well as the augmentation of informal words with repeated characters (e.g. 'coooool'). In addition, Go et al. (2009) and Mohammad et al. (2013) both found that emoticons tend to be redundant, since the rest of the tweet would convey the sentiment anyway.

Rout et al. (2018) also proposed two approaches to perform sentiment analysis. The first is an unsupervised approach, making use of lexicons and a scoring function to determine the polarity of the tweet. The second approach is supervised, in which a machine learning classifier model is trained on a training dataset via extracted and engineered features. This approach identified the most effective features being unigrams, POS tags and adjectives.

Note that the most research studies thus far have been limited to sentiment classification with a positive-negative label. While sentiment analysis has become a rather saturated research field, we intend to go a step further and perform classification based on emotional sentiment instead.

To do so, we can make use of emotion models which define how emotions are represented. Dr. Paul Ekman (1999) provides us with a theoretical discrete emotion model, which asserts that there exist 6 fundamental, independent emotions that originate from separate neural systems as a result of how an experiencer perceives a situation: *Happiness*, *Anger*, *Sadness*, *Fear*, *Surprise* and *Disgust*. He posits that the combination of these emotions (to various degrees) produces other complex emotions. (Gu et al., 2019)

Acheampong et. al (2020) proposed the use of certain annotated datasets to train the emotion model on, including the SemEval dataset which is labelled based on Dr. Ekman's emotion model.

Mohammad & Bravo-Marquez (2017) studied the most effective features on emotion intensities in the context of tweets, and found that they were word n-grams, word embeddings as well as emotion lexicons. In particular, they recommended a number of emotion lexicons, including the NRC Word-Emotion Association Lexicon (NRC EmoLex) which associates English words with 8 emotions (anger, fear, anticipation, trust, surprise, sadness, joy, disgust) as well as 2 sentiments (positive, negative).

3 Data collection and preprocessing

3.1 Data collection

Our training dataset was retrieved from the readily accessible database Kaggle. It is a well-labelled dataset, with 40,000 general tweets and 13 types of emotions annotated. The 13 types of emotions represented in our dataset are: *Empty*, *Sadness*, *Enthusiasm*, *Neutral*, *Worry*, *Surprise*, *Love*, *Fun*, *Hate*, *Happiness*, *Boredom*, *Relief* and *Anger*.

In order to perform a more meaningful analysis of the emotions embedded in tweets, we decided to augment our dataset to carry only the 6 basic emotions: *Happiness*, *Anger*, *Sadness*, *Worry/Fear*, *Surprise* and *Disgust*. This is in accordance with Dr. Ekman's emotion model (Ekman, 2019), as discussed in our literature review.

We proceed to augment the training dataset by grouping the 13 types of emotions in our dataset into 6 groups, each representing one of the abovementioned basic emotions:

- **Happiness**: *Love*, *Fun*, *Relief*, *Enthusiasm* and *Happiness*
- **Anger**: *Hate* and *Anger*
- **Worry**: *Worry*
- **Sadness**: *Sadness*
- **Surprise**: *Surprise*
- **Disgust**: Not applicable

Note that tweets annotated with *Empty*, *Neutral* and *Boredom* as sentiments were not grouped as we felt that these three sentiments were more similar to a "null" emotion (i.e., a lack of emotion). As such, they did not clearly convey an emotion that could be appropriately categorised as any one of the 6 emotions in Dr. Ekman's framework. Take note that our training dataset did not contain any tweets annotated with sentiments that corresponded to the basic emotion **Disgust**.

After relabelling the dataset, it now consists of tweets with the following 5 labelled emotional sentiments:

- **Happiness**: 13,112 tweets
- **Anger**: 1,433 tweets
- **Worry**: 8,459 tweets
- **Sadness**: 2,187 tweets
- **Surprise**: 2,187 tweets

3.2 Training data

Prior to running each classifier model on the dataset, we implemented two types of train-test splits in order to derive a training set and a validation set:

1. A fixed split
2. A randomised split

The fixed train-test split is used to compare performance between the classifier models. This was done so that the models were trained on the same training set and tested on the same validation. This ensures that the comparison analysis is done as fairly as possible.

The randomised train-test split is used to tune model parameters after identifying the best performing model.

3.3 Data preprocessing

The training dataset is preprocessed before being used to train the model. This is done in the following order:

1. Removal of short tweets and non-English tweets from dataset
We set a threshold length of 4 words. This meant that each tweet should be at least 4 words long in order to be deemed as meaningful data for the models to be trained on.

Tweet	**Status**
This damn guy...	Removed
Di mana dia anak kambing saya?	Removed
@michaeljackson Lol I love your 1990s musiic it's SO COOOOL #thriller 😀 http://google.com	Retained

2. Removal of mentions, URLs, repeated characters, hashtags and non-alphanumeric characters
These removals are done as per advised by our literature review. Removal of repeated characters is done for words with 3 or more subsequent letters. In addition, we also account for repeated "haha"s and "lol"s (as in "hahahaha" and "lolololol"). Take note that besides removing repeated characters, we also take note of the number of words with repeated characters (this is used as an engineered feature, which will be discussed in the next section).

Removal of...	**Augmented tweet**
(original)	@michaeljackson Lol I love your 1990s musiic it's SO COOOOL #thriller 😀 http://google.com
Mentions	Lol I love your 1990s musiic it's SO COOOOL #thriller 😀 http://google.com
URLs	Lol I love your 1990s musiic it's SO COOOOL #thriller 😀
Repeated characters	Lol I love your 1990s musiic it's SO COOL #thriller 😀
Hashtags	Lol I love your 1990s musiic it's SO COOL thriller 😀
Non-alphanumeric characters	Lol I love your 1990s musiic it's SO COOL thriller

3. Lowercasing & replacing Internet abbreviations with full versions
In order to perform the replacement of Internet abbreviations, we manually compiled a dictionary of 250 common Internet abbreviations as well as their full versions.

Preprocessing step	Augmented tweet
Lowercasing	lol i love your 1990s musiic it's so cool thriller
Replacing Internet abbreviations	laugh out loud i love your 1990s musiic it's so cool thriller

4. Removal of stopwords & spelling correction
Spelling collection was performed using the *pyspellchecker* library, with an edit distance of 1.

Preprocessing step	Augmented tweet
Removal of stopwords	laugh out loud love 1990s musiic cool thriller
Spelling correction	laugh out loud love 1990s music cool thriller

4 Approach

As for our approach, we decided to take on a supervised approach, as suggested by Rout et al. (2018).

4.1 Feature engineering

The first step was to engineer features into a vector representation to be learnt by the chosen models. Based on our literature review, we decided on 4 word-based features:

1. Count (unigram and bigram models)
Given the short nature of tweets (280 characters, or 140 characters prior to November 2018), we expect tweets to be generally succinct. As such, emotions tend to be more likely expressed via unigrams and bigrams. This is corroborated by Rout et al. (2018).

2. TF-IDF (unigram and bigram models)
We were interested in exploring the possibility of normalising unigram and bigram counts using their overall weightages among all tweets in the dataset.

3. Word embeddings (skip-gram model)
Following our literature review, we learnt that word embeddings were a popular feature included by sentiment analysis researchers. It was likely that word embeddings would be able to represent not just positive-negative sentiments but also emotional sentiments as well. We used the *fastText* library to implement the word embeddings.

4. Emotion lexicons
Lexicons were used by Turney (2002) to determine polarity of movie reviews. Lexicons were also used by Rout et al. (2018) in their unsupervised approach for sentiment analysis. We decided to apply the use of NRC EmoLex. Given that NRC EmoLex associates English words with 8 emotions (anger, fear, anticipation, trust, surprise, sadness, joy, disgust) as well as 2 sentiments (positive, negative), we decided to represent the lexicons as a 10-dimensional feature in the resulting feature matrix.

In addition, we wanted to explore the use of other features that were specific to social media content such as tweets (as opposed to content found on other online platforms such as movie reviews). We hence implemented algorithms to extract the following features:

5. Number of fully-capitalised words
Fully-capitalised words (i.e., words with all characters in uppercase, e.g., "GREAT") are often used as an informal way of emphasising the word.

6. Number of exclamation marks
The presence of exclamation marks could possibly be indicative of the intensity of emotional sentiment embedded in the tweet.

7. Number of words with repeated characters
Similar to the fully-capitalised words, words with repeated characters may be used informally to emphasise certain descriptive words, usually used with adjectives.

Note that these features were extracted from the original tweets, alongside the preprocessing steps described in section 3.

4.2 Model selection

Based on our literature review, the 3 most commonly researched machine learning models used for sentiment analysis are:
1. Support Vector Classifier (SVC)
2. Random Forest (RF) classifier
3. Multinomial Naive Bayes (MNB) classifier

In addition, we wanted to include another classification method to use as a baseline for comparing the performance of the above 3 machine learning models. We decided on the K-Nearest Neighbours (K-NN) classifier for this purpose. The K-NN classifier acts as a good baseline due to its simplicity and quick calculation time.

We proceeded to train these 4 models on the training dataset derived from the fixed train-test split (as mentioned in section 3.2). In order to facilitate a more comprehensive comparison between the models, we trained each model with only one of the 7 engineered features (described in section 4.1) at one time. The trained models were then tested against the validation set and F1 scores were measured to ascertain performance.

After collecting the results above, we could then identify the best performing model as well as the most effective features. This essentially gives us an empirical answer for our first research question, as described in section 1.

4.3 Model tuning

Each model was tuned to derive the following parameters which gave the best performance:

Model	Parameter(s)
SVC	`kernel='sigmoid'`
RF	`n_estimators=100`
MNB	-
K-NN	`n_neighbours=1`

4.4 Exploratory research

Using our best performing model, trained on the entire training dataset using a feature matrix representing all features, we seeked to find the answers to our second and third research questions.

In order to identify the most common expressions used to describe a particular emotion about a particular tweet, we ran the model on an unseen dataset of topical tweets. In this case, we did this on multiple unseen datasets, including one about Covid-19 and one about Elon Musk. The frequency counts of the tokens are then calculated and output to the program log.

In order to derive the proportion of emotions about a particular topic, we conducted statistical analysis on the emotion predictions and displayed the results via a pie chart. These results are further discussed in section 7.

5 Performance evaluation

We ran each individual model with each feature on the training dataset, and evaluated it on the testing data using the F1 scores.

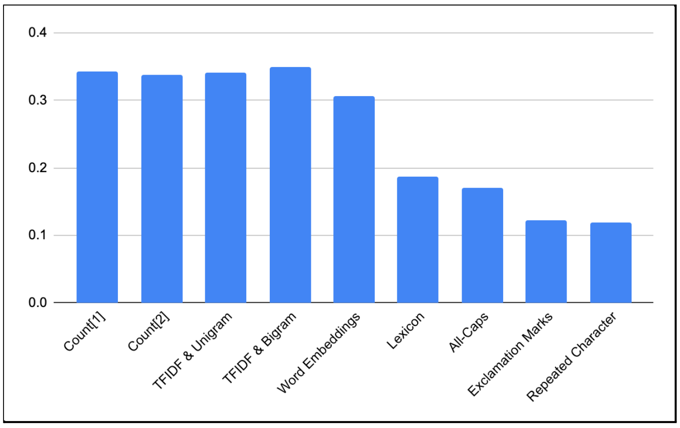

Figure 1: F1 scores for SVC model

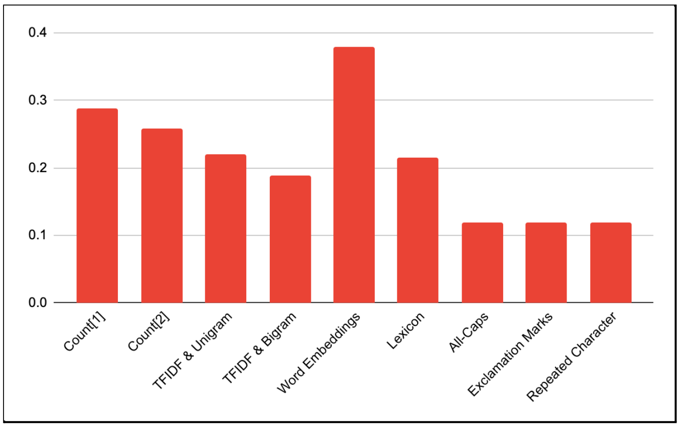

Figure 2: F1 scores for MNB model

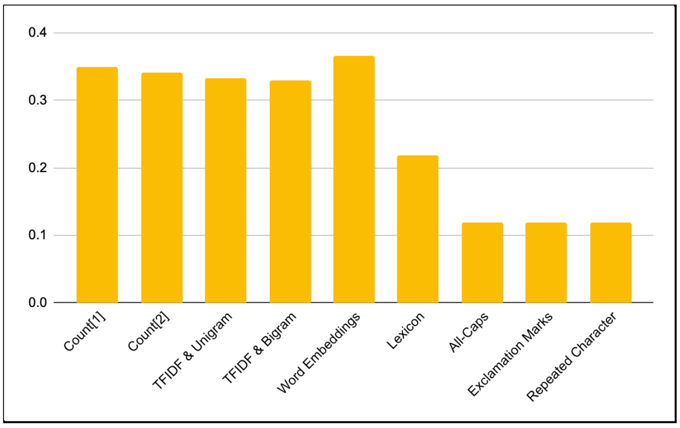

Figure 3: F1 scores for RF model

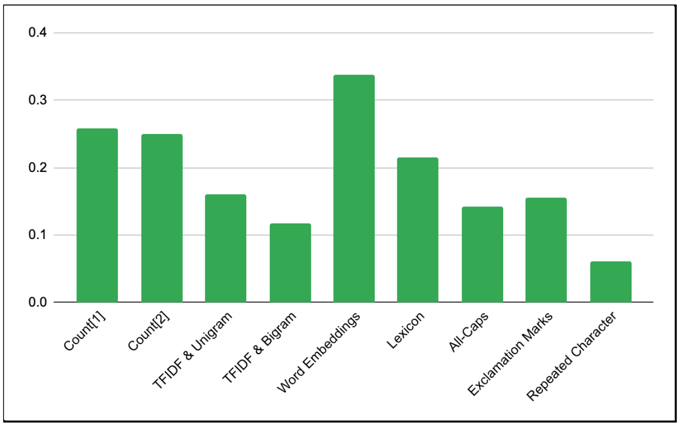

Figure 4: F1 scores for K-NN model

From the graphs above, we can identify that the use of word embeddings is the best performing feature, as it gave the highest F1 score across all models, ranging from from 0.306 to 0.380, whereas using repeated characters gives the worst performance. Take note that we have used the F1 macro score, which is a metric which combines both precision and recall. (The F1 macro score treats more frequent classes and less frequent classes as equally important.)

The lower performance of the worst 3 features (all-caps, exclamation marks, repeated characters) is not unexpected, as these features each capture only a small part of the tweet and are meant to be supplementary features to be used in combination with the other features.

The following graph presents a closer look at the 3 best performing features, which we shall use to compare between the models:

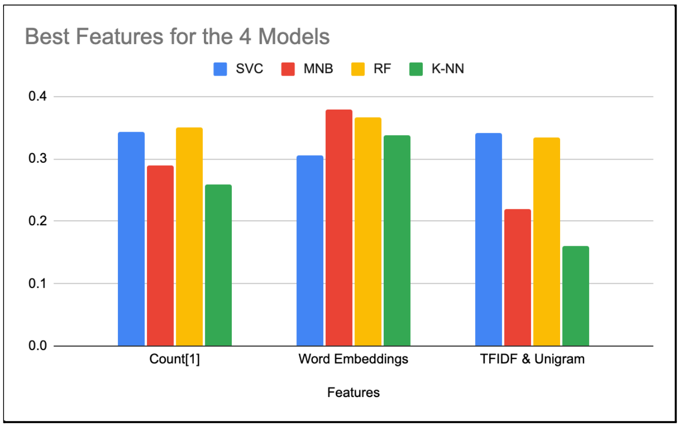

Figure 5: F1 scores for all models with top 3 features

From the column chart above, it is starkly apparent that RF is consistently the best performing model, followed by SVC and MNB. K-NN tends to be the weakest model and this may be due to the simplicity of this model. We have previously stated that K-NN is used as a baseline for comparing against the other more complex models. It is possible that RF performs the best as it is very robust against overfitting. In addition, it performs well with high-dimensional data, which does occur in our training dataset, due to the vectorization of the abovementioned features.

We then chose the best performing model, Random Forest, and ran it on all the features combined. This gave an F1 score of 0.385.

6 Ablation Study

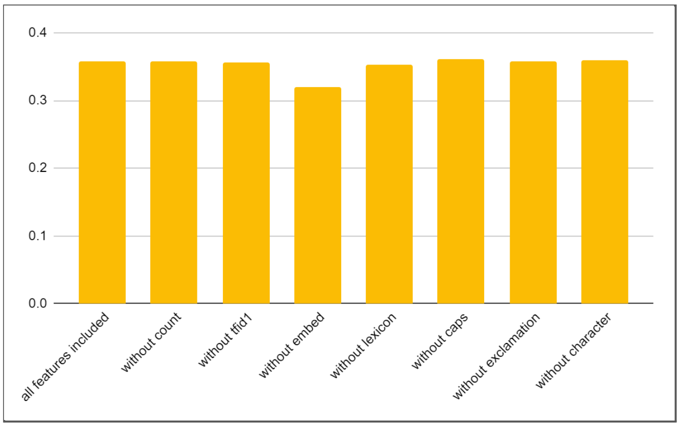

Figure 6: F1 scores for RF model with different combinations of features

After running the RF model with all the features, we removed one feature at a time and re-ran the model with the rest of the features. We noted that upon the removal of most features, the performance of the model was not significantly affected. The main exception to this was the model without the word embedding feature. This corresponds to the shortest bar in the chart above, which shows the combination with the lowest F1 score. This suggests that the use of word embeddings is the most effective feature in the model. The effectiveness of these features are further analysed in the Discussion section below.

7 Discussion

Based on our empirical results, we attempted to formulate answers to our research questions specified earlier.

Q1: When analysing a tweet, what are the best features that describe the emotional sentiment?

Taking reference from the comparison of F1 scores in Figure 3, the three most effective features are:

1. Word embeddings
Word embeddings are used to represent each word as a vector, where words with similar meanings are represented similarly in the vector space. We believe this is a better feature as it is able to capture context of the words in the tweet, irrespective of its relative position and associate common synonyms (and similar-context words) such as "happy" and "joy" with the emotion **Happiness**.

2. Word count
Word count is seen to have outperformed TF-IDF. This is likely because in emotion analysis, the occurrence of words that indicate emotions matter more, as compared to the weighted frequency of these words in the tweet. For instance, even if the word "happy" occurs several times in a tweet, the overall emotion of the sentence is still likely to be classified as the emotion **Happiness**.

3. TF-IDF with unigrams
This was surprising to us as we had expected the model's performance to improve further by using bigrams, due to its ability to capture more context. However, using unigrams gave the best performance. This could be attributed to the fact that our dataset is based on tweets, which are often short and limited by number of characters. As such, Twitter users would likely prefer to express emotions using singular words which would be better captured through the use of unigrams.

Analysis of the effectiveness of other features
Other features such as all-caps, exclamation marks and repeated characters did not perform as well as they are unable to capture the full emotions expressed in tweets. As for the lexicon feature, we believe the poor performance is attributed to the fact that certain context-specific emotional words are not captured by the NRC EmoLex model that we had used.

Q2: For a particular emotion, what are the most common expressions used to describe it in tweets?

To do this, we first train our dataset using our best-performing model, in this case, Random Forest and the full feature matrix. After doing so, testing was performed on an unseen tweet dataset of another topic, in this case, on COVID-19. For each of the basic emotions, we retrieve the frequency count of the tokens and output the top 20 most common words used to describe those emotions.

One assumption made here is that emotional sentiment analysis performs similarly when given a domain-specific dataset, even though our model is trained on a generalised dataset.

Using our model, the 20 most common unigrams for **Anger** are:

Word	Frequency	Word	Frequency
covid19	39	stop	5
people	16	trump	4
sick	13	air	4
coronavirus	8	like	4
couid	7	get	4
work	6	virus	4
health	5	masks	3
public	5	jesus	3
wear	5	access	3
age	5	dead	3

In our findings, we discovered that words about the topic occur the most frequently for all emotions. This is also reflected in other topical datasets that we tested the model on. (*e.g. "Tesla" is the most common unigram for Elon Musk's tweet dataset*). This is likely due to the fact that users often direct an emotion towards a particular topic, which in this case is likely the main topic of the tweet dataset.

Another interesting finding that we had encountered was that the word 'trump' only appears for the emotion **Anger**. Based on this, it could be an indication that Donald Trump (as an entity) is likely to be associated with angry sentiments, in the context of COVID-19. Additionally, we also realised that several tweets misspelt the word 'covid' as 'couid', however this was not captured by our spelling correction algorithm during data preprocessing.

Q3: Given a particular topic, what are the most common emotions expressed in tweets?

We chose a topic about COVID-19 vaccination since it is currently a hot topic and there have been several mixed emotions regarding it. We tested our trained RF model on an unseen dataset based on COVID-19 vaccination-related tweets. We then wrote an algorithm to calculate the distribution of emotions that our model outputs for these tweets. The results are as shown below:

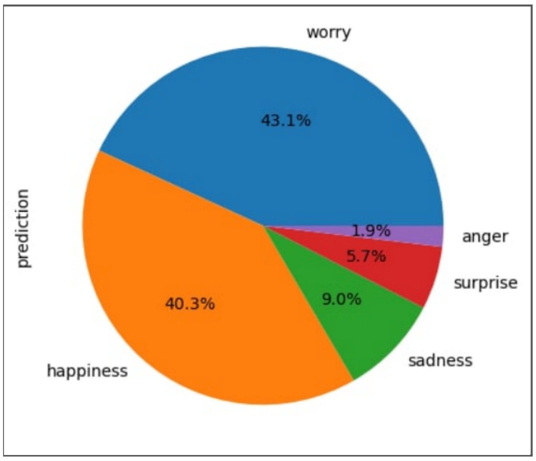

Figure 7: Distribution of emotions expressed in unseen COVID-19 dataset

From the diagram, we can see that **Worry** and **Happiness** are the most common emotions relating to COVID-19 vaccinations. This seems to be reasonable as most individuals tend to be either happy about receiving the vaccine or worried about the possible side-effects of taking the vaccine. This was also a useful sanity check for us to ensure that our model was performing meaningfully.

8 Limitations

Insufficient data preprocessing
Our current preprocessing steps may not be able to handle the various intricate deviations from the formal English language. For example, in our limited time, we may not have been able to address all common Internet abbreviations and variations of them. It is also possible that our spelling correction implementation is not robust enough to account for informal spellings of certain words.

Over-preprocessing of data
It is also possible that the preprocessing steps that we have taken could have reduced the accuracy of our classifier model. For example, the replacement of Internet abbreviations with full phrases may have altered the emotional meaning or intensity of the tweet.

Lack of linguistic-based features
Our best performing features tend to be machine learning-based features such as count and TF-IDF. We had avoided the use of POS tagging as we had been working under the assumption that language used in tweets (and other short-length social media texts) may not be as grammatically similar to regular English. However, this assumption may not be true. Besides, there exist techniques that allow us to achieve reasonable accuracy of POS tagging for social media texts (Neunerdt, 2013).

Assumption: Correctly labelled data
Our research assumes that the data in our training dataset have been accurately labelled by its crowdsourced audience. It also assumes that the training data is free of bias. However, upon closer inspection, we realised this assumption may not hold true. For instance, the tweet "*gotta work tmrw*" is labelled as sadness but this is highly subjective as some individuals may instead find joy in their work.

Assumption: Emotional sentiment analysis works well for domain-specific dataset, when trained with generalised dataset
This is the assumption raised previously in Q2 of section 7. In most cases, this assumption holds if emotions are expressed using generic emotion-related words (e.g. "sad", "crying") within the datasets. However, certain domains tend to have domain-specific emotional words and this cannot be captured when trained with a generalised dataset. For instance, in the domain of the stock market, the usage of the word "bull" likely captures the emotion of happiness whereas the word "bear" likely captures the emotions of sadness or anger. However, given that we have trained our model on a generalised model, if we were to test our model on a tweet dataset about the stock market, it is unlikely that our model will successfully evaluate the emotions encoded in these words.

9 Future scope

We would like to address some of the limitations raised previously. Further preprocessing steps, such as prepending "not_" to every word after a negation, can be taken to better represent our data. This will allow us to more accurately capture the sentiment of negated adjectives in tweets.

More effort and time should also be spent towards minimizing the demographic bias represented within our data. This can be done via more careful control over the demographic of our training dataset, such as using a language detector which is aware of varieties of English used by minority communities. We could also expand the emotional classification to tweets of other languages.

Additionally, our current project focuses solely on classifying tweets into the six basic emotions. In the future, we would like to expand the project and look into training a model that is robust in classifying more complex emotions. In addition, we would also like to investigate ways in which we can extract the intensities of the emotions embedded in tweets as well.

Lastly, we would also like to explore using other feature engineering techniques, such as the parts-of-speech based weighting scheme (PSW), in which different weights are assigned to the emotional words in tweets based on its POS tags (Wang et al., 2018). Recent studies have shown that this scheme could potentially outperform the current commonly used schemes in emotional analysis.

10 Conclusion

In our study, we have experimented with using different machine learning models to classify emotions in tweets. In addition, we have engineered 3 novel features that can be considered for use in the sentiment/emotion analysis of informal texts such as tweets.

In addition, we have found that word unigrams and word embeddings tend to be effective features. This corroborates well with the research that has been done by Mohammad & Bravo-Marquez (2017). We have also found that the Random Forest classifier as a model has given the best performance in this emotion classification task. This view is supported by previous sentiment analysis research done by Barros et al. (2013), Kim et al. (2017) and Henny-Krahmer (2018).

Furthermore, we also discovered that emotional sentiment analysis generalises rather well, as its performance remains relatively high, even when tested on unseen datasets of another domain. This research has inspired us to not only further explore ways to improve the accuracy of our emotion classifier model, but also to explore use cases in which these analytical tools could prove to be useful.

11 Source code

Our source code can be accessed on our GitHub repository via the following link: https://github.com/grrrrnt/notion-emotion-twitter. We encourage all interested individuals to build upon our work and findings, should they find it meaningful to do so.

12 Acknowledgements

We would like to extend our gratitude to our module's professor, Dr. Kan Min-Yen, for his guidance and passionate teaching on Natural Language Processings. With valuable guidance and advice, it has inspired us greatly to work on this project.

We would also like to thank our teacher assistant, Mr Anab Maulana Barik, for his mentoring and guidance regarding our project and providing us with some directions for us to work towards.

13 References

Acheampong, F. A., Wenyu, C., & Nunoo-Mensah, H. (2020). Text-based emotion detection: Advances, challenges, and opportunities. *Engineering Reports, 2*(7), e12189.

Barros, L., Rodriguez, P., & Ortigosa, A. (2013, September). Automatic Classification of Literature Pieces by Emotion Detection: A Study on Quevedo's Poetry. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction (pp. 141-146)*. IEEE.

Ekman, P. (1999). Basic emotions. *Handbook of cognition and emotion, 98*(45-60), 16.

Fei, G., Liu, B., Hsu, M., Castellanos, M., & Ghosh, R. (2012). A dictionary-based approach to identifying aspects implied by adjectives for opinion mining. In *Proceedings of 24th international conference on computational linguistics (p. 309)*.

Gaind, B., Syal, V., & Padgalwar, S. (2019). Emotion detection and analysis on social media. *arXiv preprint arXiv:1901.08458*.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N project report, Stanford, 1*(12), 2009.

Gu, S., Wang, F., Patel, N. P., Bourgeois, J. A., & Huang, J. H. (2019). A model for basic emotions using observations of behavior in Drosophila. *Frontiers in psychology, 10*, 781.

Henny-Krahmer, U. (2018). Exploration of Sentiments and Genre in Spanish American Novels. In *DH (pp. 399-402)*.

Kim, E., Padó, S., & Klinger, R. (2017, August). Investigating the relationship between literary genres and emotional plot development. In *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (pp. 17-26)*.

Lindquist, K. A., & Barrett, L. F. (2008). Emotional complexity. In *M. Lewis, J. M. Haviland-Jones, & L. F. Barrett (Eds.), Handbook of emotions (p. 513–530)*. The Guilford Press.

Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242.*

Mohammad, S. M., & Bravo-Marquez, F. (2017). Emotion intensities in tweets. *arXiv preprint arXiv:1708.03696*.

Neunerdt, M., Trevisan, B., Reyer, M., & Mathar, R. (2013). Part-of-speech tagging for social media texts. In *Language Processing and Knowledge in the Web (pp. 139-150)*. Springer, Berlin, Heidelberg.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on empirical methods in natural language processing (Vol. 10, pp. 79–86)*. doi:10.3115/1118693.1118704.

Pang, B., & Lee, L. (2004) A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on association for computational linguistics (p. 271)*. doi:10.3115/1218955.1218990.

Rout, J. K., Choo, K. K. R., Dash, A. K., Bakshi, S., Jena, S. K., & Williams, K. L. (2018). A model for sentiment and emotion analysis of unstructured social media text. *Electronic Commerce Research, 18*(1), 181-199.

Turney, P.D. (2002). Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics (pp. 417–424)*. doi:10.3115/1073083.1073153.

Ullah, M. A., Islam, M. M., Azman, N. B., & Zaki, Z. M. (2017, February). An overview of multimodal sentiment analysis research: Opportunities and difficulties. In *2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR) (pp. 1-6)*. IEEE.

Wang, Y., Kim, K., Lee, B., & Youn, H. Y. (2018). Word clustering based on POS feature for efficient twitter sentiment analysis. *Human-centric Computing and Information Sciences, 8*(1), 1-25.