IT UNIVERSITY OF COPENHAGEN

First Year Project

Project #4: Natural Language Processing

Group 3:

Carl August Wismer (cwis@itu.dk)
Chrisanna Kate Cornish (ccor@itu.dk)
Danielle Marie Dequin (ddeq@itu.dk)
Gino Franco Fazzi (gifa@itu.dk)
Moneeca Abru Iftikhar Latif (abml@itu.dk)

2nd Semester, Spring 2021
Data Science

1 Introduction

Natural language processing ("NLP") concerns the interactions between computers and humans using language. The overall goal is for a computer to be able to process the contents of language in a human-like way using machine learning ("ML") to accurately extract information and insights as well as categorize and organize the data itself (see Liddy).

Present challenges in NLP involve speech recognition, natural language understanding and natural-language generation [1]. These challenges are increased when the source is noisy user-generated text from platforms such as Twitter, which often has a context that is difficult to derive on its own and uses very informal language.

This project focuses on assessing the potential for tweet evaluation in both binary classification for irony detection and multi-class classification for emoji prediction using ML techniques. For this purpose a model was constructed and tested for both datasets, and tasks were completed to evaluate the accuracy and limitations for such endeavors. The results of this study and the limitations realized can be used for future research, as prior knowledge of the limitations can aid in development of solutions. Therefore, NLP when applied to informal data such as tweets, can be more accurate and thereby useful.

2 Data and Processing

The used data was collected from the repository for the *TweetEval* benchmark (Findings of EMNLP 2020) (see Barbieri et al.). TweetEval consists of seven heterogeneous tasks in Twitter, all framed as multi-class tweet classification. All tasks have been unified into the same benchmark, with each data set presented in the same format and with fixed training, validation and test splits. From this collection, two datasets were selected according to the pursued classification research:

- Irony Detection - Binary Classification – 2 labels: irony, not irony (see Van Hee, Lefever, and Hoste).
- Emoji Prediction – 20 labels (see Barbieri et al.). The emoji labels in the dataset are shown in Figure 1.

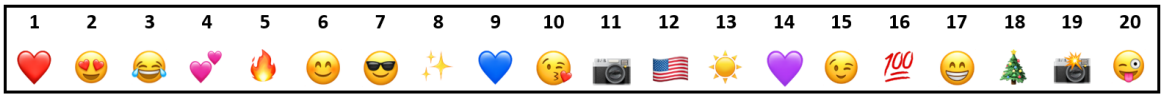


Figure 1: Emoji Labels

2.1 Pre-processing

The first task was to construct a tokenizer to process each tweet and decompose it into meaningful words or tokens. Using regular expression each tweet was tokenized according to discussed specifications. The training data was then loaded and tokenized. As a benchmark the output was compared to the output of *nltk library's TweetTokenizer* to verify validity and visualize any differences. These differences can be seen in Figure 2 with both datasets.

[1] Wikipedia: Natural Language Processing

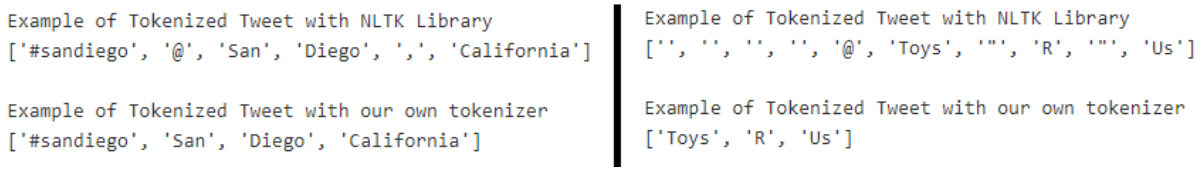


Figure 2: Example of tokenized tweet from Irony Dataset (left) and Emoji Dataset (right)

2.2 Characterizing the Data

The training data was then characterized to have a notion of its elementary corpus statistics.

- Irony Dataset:
  - Corpus size: 40,371
  - Vocabulary size: 10,300
  - Type/Token ratio: 0.25
- Emoji Dataset:
  - Corpus size: 511,305
  - Vocabulary size: 71,373
  - Type/Token ratio: 0.14

It can be seen that the emoji dataset has a much larger corpus, but a smaller type/token ratio, indicating a lot of repetitive language.

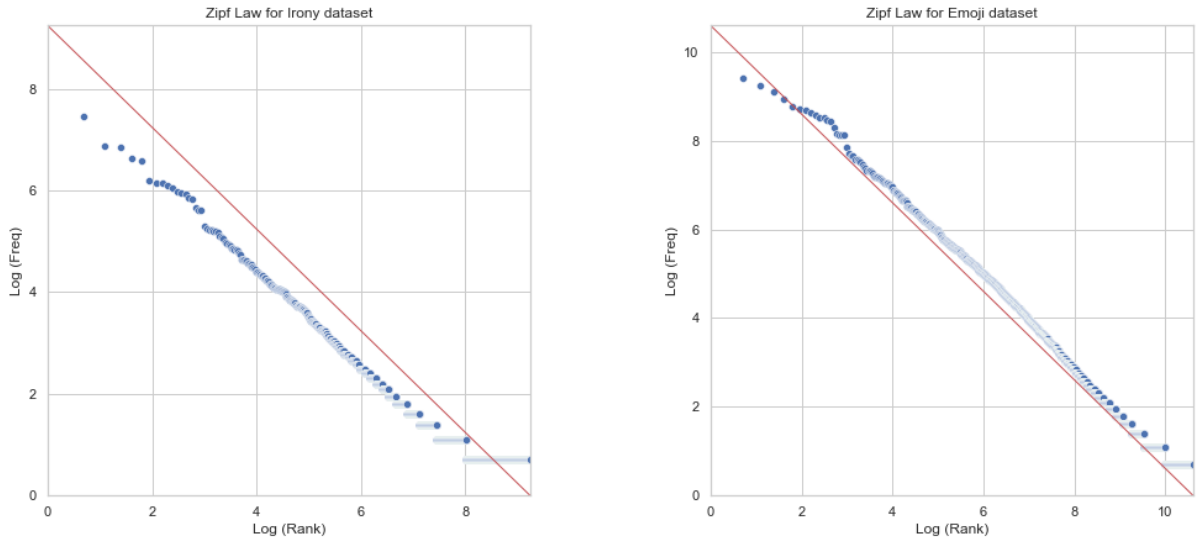
To follow privacy laws, all usernames were changed to "@user" before the data was acquired. The most common token was, as expected, the "@user" from each tweet, followed by very common articles, prepositions or subjects, like: "the", "to", "a", "I", as shown in Figure 3. Four out of the five most common words are shared across the datasets. On the other hand, least common tokens were usually "hashtags" or nouns, many of those with only one occurrence.

Rank	Dataset			
#	Irony		Emoji	
	Token	Freq	Token	Freq
1	'@user'	1,731	'@user'	12,209
2	'the'	964	'the'	10,346
3	'to'	934	'!'	9,045
4	'a'	757	'to'	7,645
5	'I'	724	'I'	6,535

Figure 3: Most frequent tokens.
(Green indicates the token is in the top 5 of both datasets)

As it can be seen in the graphs in Figure 4, the corpus of both datasets followed the empirical Zipf's Law, which states that given a large sample of words used, the frequency of any word is inversely proportional to its frequency rank. This is important to our model, since "under Zipf's law frequency falls off relatively slowly with rank. (...) Consequently, one should observe a fairly broad range of frequencies. (...) This is a remarkable property: you might initially expect to see rare words only rarely. However, while a particular rare word (e.g. "frequencies") is far less likely to occur than a particular common word (e.g. "a"), there are far more rare words than common words, and these factors balance almost exactly" (see Aitchison L).

Given that the datasets followed this law, this would indicate that there are certain tokens that occur with a large frequency that are empirically less informative than features that occur in a small fraction of the training corpus, and these very common words would not be helpful in training the models. This is taken into account further on in the study.



(a) Irony Training Data

(b) Emoji Training Data

Figure 4: Zipf's Law

3 Annotation

When working with irony detection, it is imperative that the research team shares an understanding of what constitutes irony. Each member of this research project independently annotated the same set of 100 tweets following the guidelines given to the original annotation team, labeling the tweet to be ironic or not. Afterwards, the labels were compared between members of the team as well as against the "true" labels assigned from the original research.

The results were, as anticipated, far from perfect. In only 40% of cases did this entire research team agree on one label. In addition, the highest individual coincidence where a team member annotated correctly was only 63%, while the team, on average, annotated 54% correctly. This reveals an important limitation when working with this type of data: subjectivity. Reading the exact same tweet was interpreted differently by each annotator, influenced by context, cultural background, beliefs, etc.

Apart from this previously stated subjective limitations, several other issues arose when trying to annotate the data, including:

- Lack of reference or context. For example, some tweets seem to refer to an unknown tweet, which would require the previous tweet to build context. Other tweets refer to attached photos, which were previously removed by *TweetEval* before the data were made available.
- Reference to pop/obscure culture, that requires prior knowledge to be able to correctly comprehend what the text is referring to in order to assess irony.
- Poor grammar, spelling and sentence structure due to informal nature of the text.

As a final observation, it is possible that the previous annotators misinterpreted the real intention of the tweet, meaning that the "true" label should not be viewed as an absolute truth.

4 Classification

The experiments were conducted using three different classifiers. The training data was run through a pipeline for vectorization and to train the models on the data. To handle the implications of Zipf's law a method from *scikitlearn* was used to reduce common word weighting. After experimenting with different parameters, the optimal parameters were chosen for the models.

Scikitlearn's SGDClassifier was ultimately chosen to build models for both the irony and emoji datasets. The models were then applied to the validation data which produced prediction results that were, given the challenges, unsurprisingly low. The irony prediction model had an overall accuracy of 63.8%. It is interesting to note that this is higher than the human-evaluated accuracy from the previous task.

The model used on the emoji validation data had an overall accuracy score of 31.9%. Specifically, the results show that emojis that have more ambiguous meaning are harder to predict. For instance, the 'winkingface' (#20) emoji was only being predicted correctly at 4.6%. On the other hand, certain emojis can be easier to predict, such as the 'christmastree' (#18) emoji that was predicted correctly at 81%, since the presence of words like 'christmas' are likely a strong indicator that the tweet contains this emoji. In addition, the model was only able to predict three emojis correctly with an accuracy of 50% or greater, and the accuracy of the majority of predictions fell between 0% and 30%, indicating the challenge associated with this task.

Finally, the models were used to predict the labels using the irony and emoji test data. The results of both models were similar to those of the validation results, with the irony model having an overall accuracy score of 64.8%, and the emoji model producing an accuracy score of 30.1%. The similarity to the validation data scores indicates that the model is not-overfitted to the data. The same emojis stood out as being the hardest and easiest to predict. The model accuracy score is the average of all predicted results. The true predictions per label can be seen for the irony test data (Figure 5), and for the emoji test data (Figure 6).

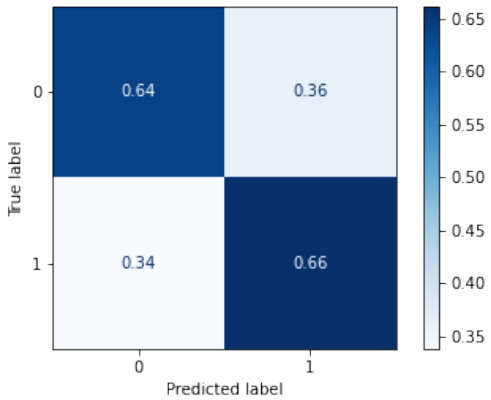


Figure 5: Irony Test Data Confusion Matrix

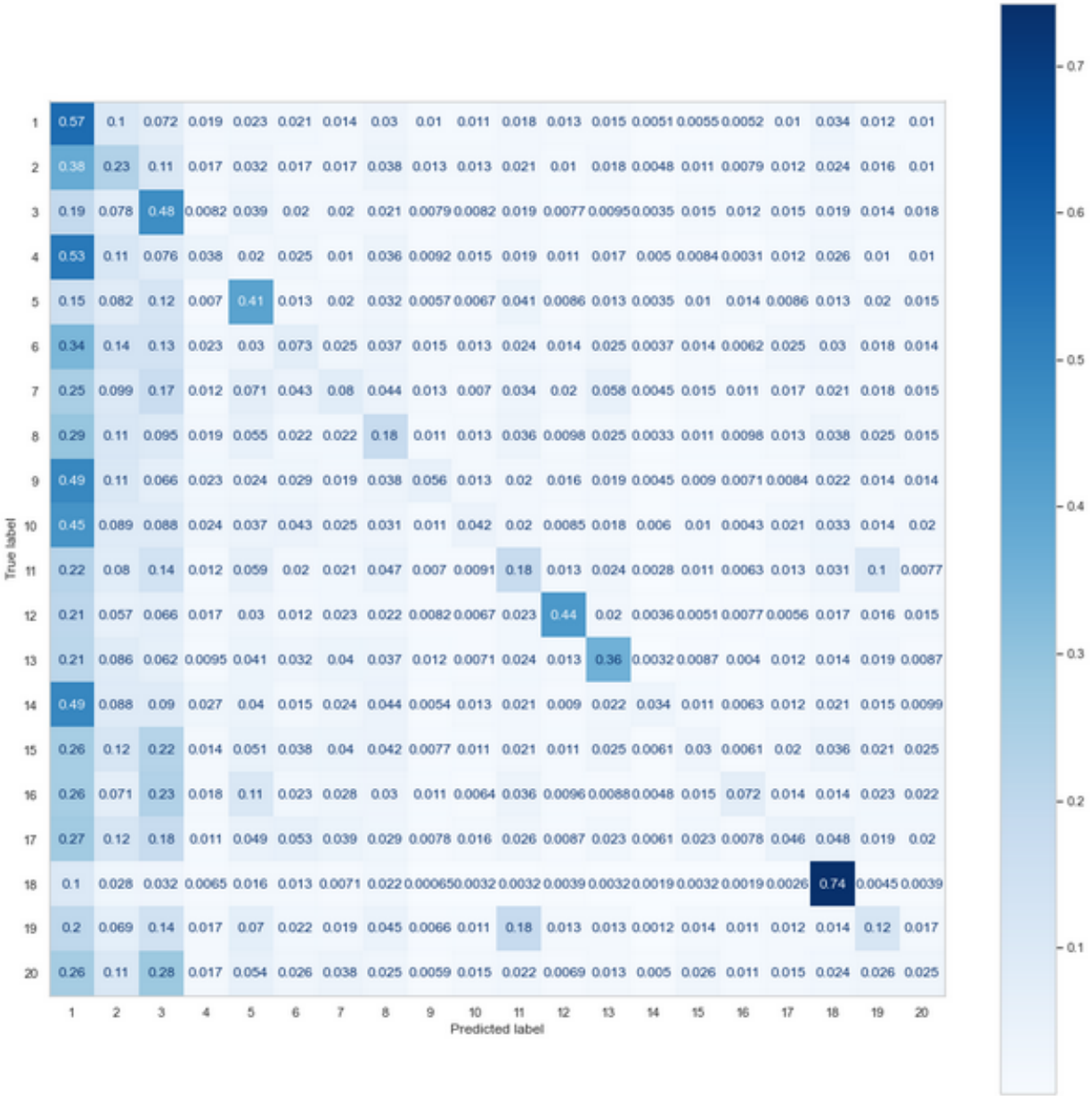


Figure 6: Emoji Test Data Confusion Matrix

5 Conclusion and Future Work

At first glance, one might doubt the importance of irony detection in written text; but since irony is defined as an "event in which what on the surface appears to be the case or to be expected differs radically from what is actually the case" [2], this becomes crucial for sentiment analysis (e.g. a feedback comment on a shop website could be interpreted as *positive* when it is in fact *negative*). In addition, digital assistants such as Siri or Alexa are growing in popularity. These rely on being able to correctly understand human language to carry out instructions. As the tasks that these assistants carry out become more sophisticated, their ability to understand increasingly complex human language must keep pace, requiring the ability to identify and respond appropriately when the literal meaning of the language does not match the intention.

Working with irony recognition is not a simple task. One might question if a program can reliably detect irony if researchers cannot even agree on what constitutes irony. This is especially true when dealing with text, since written language does not always clearly differentiate between irony and sincerity. Factors like tone, facial expressions and body language are key in recognizing irony, and since all of these factors are lost in written text this poses a limitation in predicting irony. However, the simple ML techniques used for this research showed better prediction results than the human counterparts. For future research, taking the limitations shown here into account, perhaps irony detection can become more accurate.

The emoji prediction task was chosen for two reasons. The simplest one is the application of correct *"emoji suggestion"* that some text editors use when a person is writing a message (mostly in social media and communication apps). While this can improve user experience, there is another motivation for this interest in emoji prediction: understanding the linguistics behind emojis and how the latter have changed, and continue to influence, how humans communicate. Some interesting questions arise: does this symbol language add richness to how people communicate? Is there a reliable translation between the written words and the sentiment expressed by emojis? As emojis are a part of human communication, it is important to be able to interpret their meaning and have the ability to make use of this data.

In contrary to irony identification, emoji prediction was a multi-class classification task with 20 labels. As such, it was expected that the accuracy of the prediction would be worse than the binary task, which was the case. Specifically, the results show that emojis that have more ambiguous meaning are harder to predict, as there might not be a strong correlation with any particular word. While this study used fairly simple ML techniques and pre-made libraries, the results were very close to the results from the *TweetEval* competition, which used state-of-the-art NLP techniques. This indicates that the sophistication of the techniques does not seem to be the biggest factor, and that future research may need to approach this problem in a different way.

An important limitation to note is that this work only trained and tested the model on the English language. It cannot be safely assumed that different languages will produce different patterns that would be more or less predictable. An interesting experiment would be to run similar tests with corpus from other languages, to see if the results are the same. This could be a positive contribution to linguistics.

There are many challenges still to be overcome with NLP. However, this and similar projects have shown that ML can be used to make sense of even ambiguous and context-less language that human readers have difficulty interpreting. The applications of this are wide and the accuracy can be improved if researchers take into account the known limitations and prior methods used.

[2] Wikipedia: Irony

References

Liddy, Elizabeth D. "Natural language processing" (2001).

Barbieri, Francesco, et al. "Semeval 2018 task 2: Multilingual emoji prediction". In *Proceedings of The 12th International Workshop on Semantic Evaluation*, 24–33. 2018.

Van Hee, Cynthia, Els Lefever, and Véronique Hoste. "Semeval-2018 task 3: Irony detection in english tweets". In *Proceedings of The 12th International Workshop on Semantic Evaluation*, 39–50. 2018.

Barbieri, Francesco, et al. "TweetEval:Unified Benchmark and Comparative Evaluation for Tweet Classification". In *Proceedings of Findings of EMNLP*. 2020.

Aitchison L, Latham PE, Corradi N. "Zipf's Law Arises Naturally When There Are Underlying, Unobserved Variables."