Sarcasm detector

Natural Language Processing Final Project

Islam-Bek Kurakbay

Haeun Lee

Jeyeong Park

Table of Contents

1. Abstract
2. Introduction
3. Development
4. Conclusion
5. Bibliography
6. Abstract

In this project, we will construct a sarcasm detector and detect sarcasm. From the sentence provided by the user, the sarcasm detector calculates the probability of the sentence being sarcasm in percentage and decides whether it is sarcasm or not. Naive Bayes, Logistic regression, LSTM/GRU, and Transformers models will be built and we choose the model which has the best performance of sarcasm detection. A GUI will be provided for users to use the sarcasm detector with their own sentences.

1. Introduction

What is Sarcasm? Sarcasm is the act of complaining, mocking. Often it is hostility disguised as humor. Sarcasm usage differs by region and cultures. It is used in daily life by some people when they feel inner anger or persistence. Sarcasm, some say, is the sign of high intellect. For some people, being sarcastic is similar to being toxic, but for some it is funny.

Here are some examples of the sarcasm “My favorite thing to do at 4am is to go to the airport. And yours??”. In this example, the author of the statement doesn’t feel very happy about going to the airport early in the morning, so she turned it into humor, sarcasm. Here is another example of sarcasm “Thank you for your feedback. It WaS ReAlLy InSIgGhTFul!”. In this instance, the user doesn’t like the fact that another person gives him/her feedback, but avoiding to seem impolite the user turns it into mockery, sarcasm.

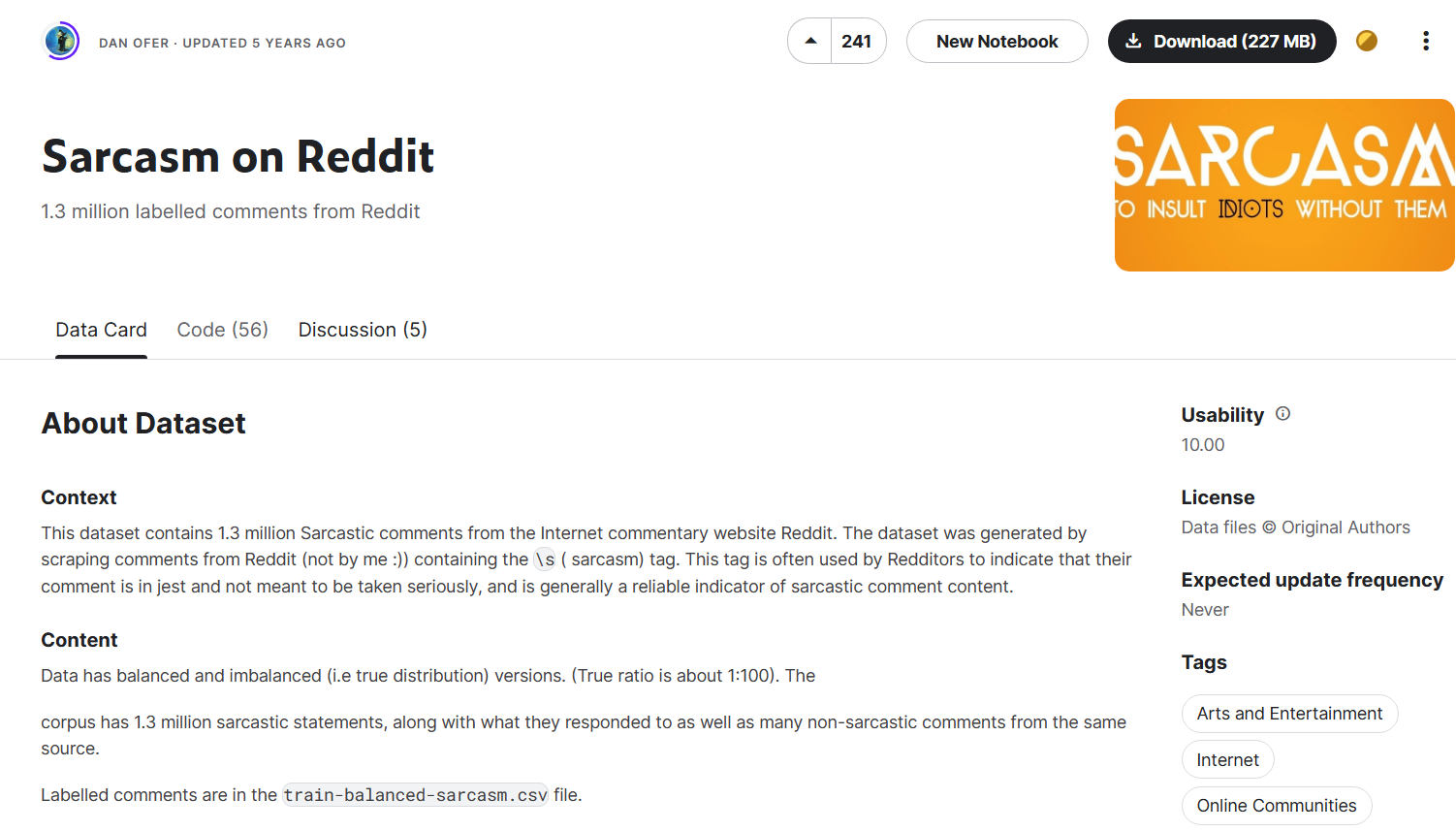
The term "sarkasmos" came from Greek meaning "to tear flesh, bite the lip in rage, sneer".

There are several types of sarcasm (Kaelyn Barron, 2023):

1. Self-deprecating
2. Brooding
3. Deadpan
4. Polite

Sarcasm is the narrow field in Natural Language Processing, since it focuses not on separate word sentiments, but overall on the context or whole spectrum of the message.

In our project we used the dataset from [Kaggle](https://www.kaggle.com/) called Sarcasm on Reddit. Reddit is the network of communities where people can divide into their interests, hobbies and passions. The dataset contains 1.3 million sarcastic statements and is balanced. This dataset was published by Dan Ofer in 2018.

Figure[1]. Sarcasm Dataset

By being able to classify sarcasm we can advance in understanding human behavior, which results in a huge boost in psychology, sociology, marketing and other human areas.

1. Development

The dataset (227mb) from Kaggle was downloaded and preprocessed using Python programming language. In this project, Jupyter Notebook was mainly used and Google Colab was used once to train LSTM model. The data was preprocessed, rushed through the machine learning models and the one with the best result was chosen for a GUI (Graphical User Interface) for better interaction.

The list of the libraries that were used:

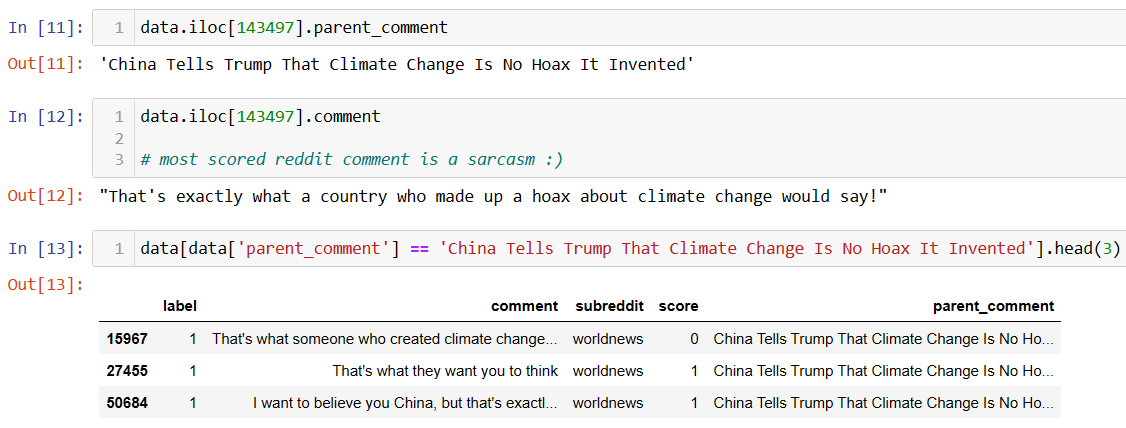
* Data Processing: Pandas, Numpy
* Data Visualisation: Matplotlib, Seaborn
* Machine Learning: Scikit-learn, Tensorflow, Transformers
* GUI: Tkinter

At first, the dataset was loaded into the Jupyter Notebook.

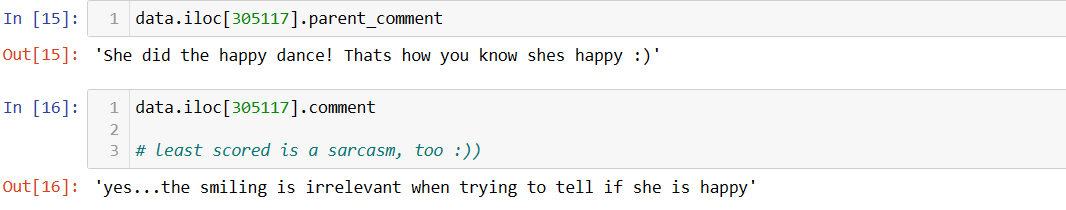
Figure[2]. Loading Data

Different columns are present in the dataset and the columns like author, ups, downs and dates are not our main targets, so they are dropped.

Dataset Analysis.

Most and least scored reddit comments are shown below. These are comments that Reddit users liked and did not like most.

Figure[3]. Most upvoted comment in Dataset

From the code above, it can be observed that without parent\_comment the actual comment makes less sense. Because, in this case the message needs some context. The most scored reddit post from the dataset is about politics and is sarcasm. It has a score of over 9000!!! In the 13th code section outcome it is shown that one parent\_comment can have more than one reply comments. 

Figure[4]. Most upvoted comment in Dataset

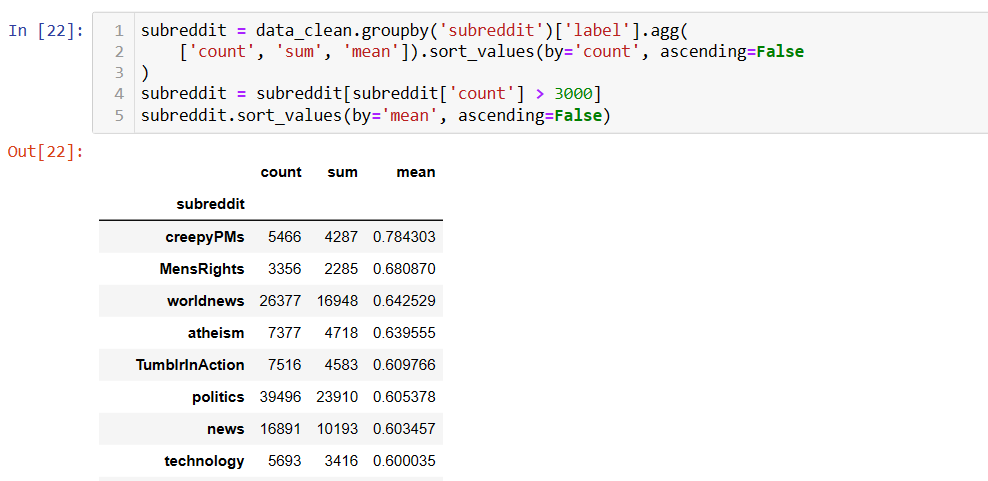
The least scored reddit comment shown above has a score of -507 and is labeled as sarcasm. Reddit users downscore comments when they find them inappropriate, unethical or deviant.

The dataset was checked for missing values and after none found the data visualization was performed (Yuri Kashnitsky, 2020).

Figure[5]. Most downvoted comment in dataset

In this histogram some conclusions can be made. At first, most of the comments have 6 to 150 word lengths. Secondly, there are more sarcastic comments that are 25 to 100 word lengths than regular comments. Thirdly, more regular comments are either very short or very long than sarcastic comments.

Most and least sarcastic subreddit topics.

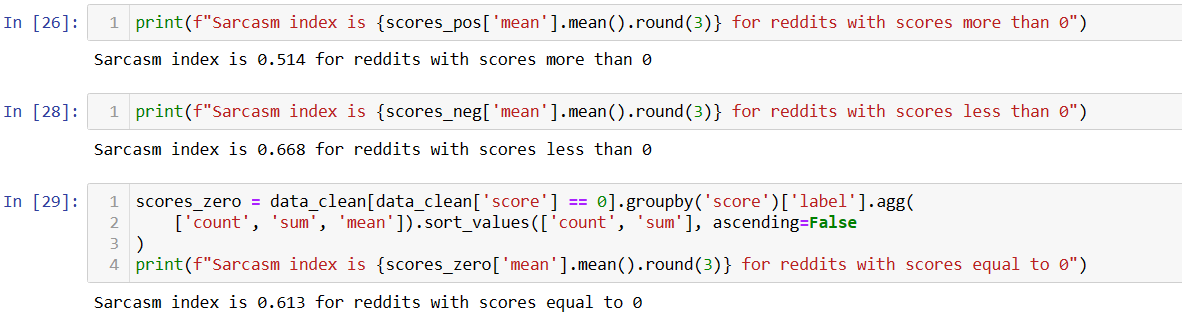
After the data visualization, analysis of the subreddits was performed and only the ones that had more than 3000 comments were filtered. Subreddit is the part of Reddit divided by definite topics. Based on the obtained data, a sarcasm percentage was calculated.

Figure[6]. Most sarcastic Reddit forums of dataset

From the output above it is clear that most sarcastic comments are in the subreddits called “creepyPMs” (PM - Private Messages), “MensRights”, “worldnews”, “atheism”, “politics”, “news” and others. This means that most sarcastic comments are related to the politics and religion forums, where aggressiveness and toxicity are more common.



The subreddits that have less sarcasm comments than regular comments are “AskReddit”, “gifs”, “funny” and others. It seems like more ordinary subreddit forums have less sarcasm and hence, less toxicity.

Figure[8]. Reddit comments analysis on their score ratings

In the section above, the comments were analyzed according to their score. Reddit comments that have positive scores, or received more approval, have less sarcastic comments ratio (0.514) than those comments with negative score (0.668). The neutral score (score=0) comments have a sarcasm ratio of 0.613.

Further, the data was transformed with the CountVectorizer class of Scikit-learn and splitted into train and test datasets. 

Naive Bayes.

The first model that will be used to predict sarcasm is Naive Bayes statistical-probabilistic model from Scikit-learn library. The algorithm assumes that all input features are independent of each other like in our case.



The Naive Bayes model performs well (67% average accuracy), but we will test out other models in order to achieve better results.

Logistic Regression.

The next model is the Logistic Regression model from Scikit-learn with TFIDF Vectorizer (Yuri Kashnitsky, 2020).

This model performs slightly better than Naive Bayes model with the average accuracy of 72%. Still, we test two more models to try getting a better sarcasm detector.

LSTM/GRU.

The third model is Long Short-Term Memory cell, which is mainly used for text classification. Gated Recurrent Model (GRU) is the same LSTM structure, but much simplified and faster in most cases. These two models perform very well, but need much more time to train. Without GPUs it will take around 40 hours for 10 epoch training for this dataset. That is why Google Colab with its free Nvidia Tesla T4 GPU was used to train GRU for a 100 times smaller dataset.



The Bidirectional GRU (BiGRU) was expected to have more than 80% test accuracy, however it barely hit 70% accuracy. The reason for that might be a very small dataset (total of 6500 comments) and a large number of unique tokens in comments. Link to the model: [Google Colab](https://colab.research.google.com/drive/1HghkpcdX0QPTNVsbfW_MmLG2-3NKUb3Z?usp=sharing)

LSTM and GRU neural network models perform well when the dataset is preprocessed and vast, however each token is processed one-at-time. This problem was resolved with the introduction of the Transformers in 2017 where many tokens were able to train simultaneously.

Transformers.

The Transformer model for sarcasm detector was used from [huggingface.co](https://huggingface.co/jkhan447/sarcasm-detection-RoBerta-base-CR?text=What+is+my+favorite+thing+to+do+at+4+am%3F+Of+course%2C+to+go+to+the+airport%21). The author of the pre-trained model is Jebran Khan and though its accuracy of 72.6% is similar to our results we decided to go on with the transformers model. 

We have tested our sarcasm test cases with the transformers models and 3 out 4 results were correctly predicted and labeled.

GUI.

In order to gain better sarcasm detection experience Graphical User Interface (GUI) was built using the tkinter library. 

1. Conclusion

Sarcasm is one of the challenging tasks in Natural Language Processing and solving this task rewards in better understanding human psychology. In this project we have provided an analysis and sarcasm classification on Reddit Comments Dataset using Naive Bayes, Logistic Regression, Bidirectional Gated Recurrent Units and Transformers. The most achieved results of 72% accuracy were obtained by using Logistic Regression and Transformers. Finally we built GUI for better user interaction with our sarcasm detector model.

In order to improve the performance of our models some changes can be implemented. First, parent comments play a significant role in determining whether an answer comment is sarcasm or not. Because the same comments were observed to be both sarcasm and regular. Second, text filtering would result in better model performance. For example, sarcasm messages often have unnecessary capital letters, expressive symbols, usage of hyperbolic language and relate to popular culture quotes or situations. Based on these suggestions, better sarcasm detecting models can be developed.

1. Bibliography

Dan Ofer (2018), Dataset on Kaggle, [Sarcasm on Reddit | Kaggle](https://www.kaggle.com/datasets/danofer/sarcasm)

Kaelyn Barron (February, 2023), TCK Publishing, [How to Identify and Use Sarcasm: Definition, Types, and Examples - TCK Publishing](https://www.tckpublishing.com/sarcasm/)

Vaswani, Ashish & Shazeer, Noam & Parmar, Niki & Uszkoreit, Jakob & Jones, Llion & Gomez, Aidan & Kaiser, Lukasz & Polosukhin, Illia (2017), [“Attention is all you need”](https://arxiv.org/pdf/1706.03762.pdf), [arxiv.org/pdf/1706.03762.pdf](https://arxiv.org/pdf/1706.03762.pdf)

Yuri Kashnitsky (2020), Sarcasm Detection with logit, [A4 (demo). Sarcasm detection with logit. Solution | Kaggle](https://www.kaggle.com/code/kashnitsky/a4-demo-sarcasm-detection-with-logit-solution)