CSE 440 : Data Analysis

Project Title: Clickbait Detection

Group Members:

Mohammad Aman Ullah Khan : 19301139
Nirjhar Gope: 19301140
Mohammad Siam : 23141065
Mohammad Ariful Islam : 20101192

Chapter 1

Introduction:

The purpose of this project report is to examine the phenomenon of clickbait, a type of content that employs sensational headlines or images to entice readers and increase website traffic or social media engagement. This is usually done to get people to visit a website or interact more on social media. The main goal of clickbait is to get more people to click on a link, watch a video, or interact with a social media account. The content behind the clickbait headline might not be as interesting, informative, or important as the title makes it sound. In the past few years, clickbait has become more common on social media sites like Twitter, where users are constantly flooded with content meant to get their attention. In this study paper, we show how to use a Naive Bayes model for clickbait spoiling, which is a way to change clickbait headlines so that they don't work as well but still say what they mean.

It's important to exercise caution and critically evaluate the information before clicking on any link. These headlines offer a surprising answer to get people interested, but they rarely give any useful information in the tweet itself. But the hints given for each of the four cases show that the answers to the questions in the headlines are often simple or obvious. In the first example, the related page doesn't tell you much more, and in the second example, the spoiler is something most readers would already know. The third and fourth spoilers give more information. The third one gives text from the linked page, and the fourth one gives a list of things.

This makes us want to talk about the job of "clickbait spoiling," which is to find or make a spoiler for a clickbait post.

Data Analysis:

It is essential, before undertaking any kind of study, to identify whether or not all of the columns are meaningful. In this undertaking, one of the columns must hold text data and another must hold label data. We must decide which column will hold each of these types of data. You are free to delete any extra columns that aren't necessary. The DataFrame component of Pandas enables one to accomplish this.

Before commencing the analysis, you should make sure that the dataset has an even distribution of values. The dataset for this project is divided into three categories, and it is essential that each category have the same number of samples to prevent the model from having a predisposition to favor any one

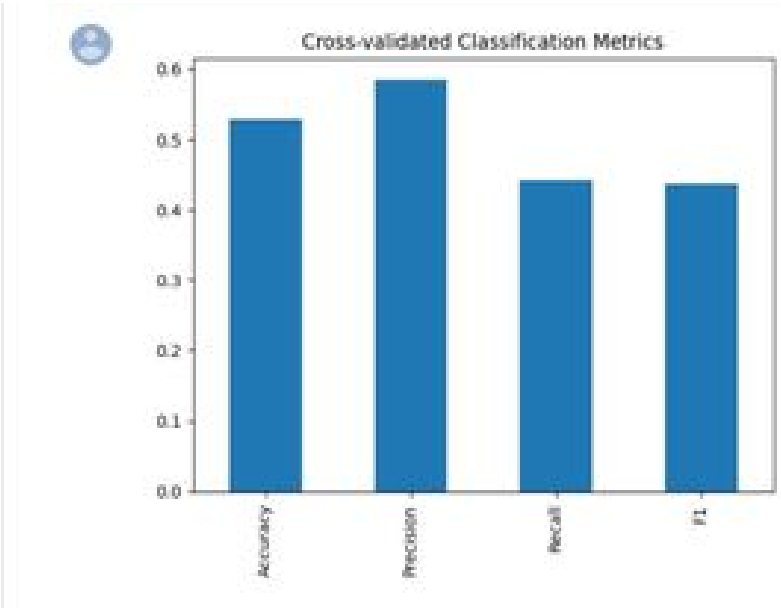
of the categories. The number of pupils in each class can be plotted on a bar or pie chart, which can then be used to determine the class balance. It was established that there was no class that had an overwhelming majority in this undertaking; therefore, the class balance was considered to be evenly balanced.Because the majority of machine learning methods require numerical input, it is essential for natural language processing to convert text labels into their corresponding numerical values. Before we can train our model, the text values that make up the labels for this project will first need to be transformed into their corresponding numerical values. Creating a dictionary that converts each unique text label into a unique number value is one way to get this job done. This project consists of three classes, and we may decide to create a dictionary with three key-value pairs. In this dictionary, each key will be a distinct text label, and each value will be a distinctive number.

Before being used in machine learning models, text data must first be converted into numerical vectors using natural language processing (NLP). One method for achieving this goal is to make use of word embeddings that have already been pre-trained. Word embeddings that have already been pre-trained are included in the Spacy package, which may be used to convert text data to 300-D vectors. The "$en_core_web_lg$" $model is responsible for the generation of word vectors for English text. We can l$
$and then we can make use of the loaded model to convert each sentence into a 300-dimensional vector.$
$In general, these procedures are necessary for the proper preparation of the dataset prior to the training of a machin$

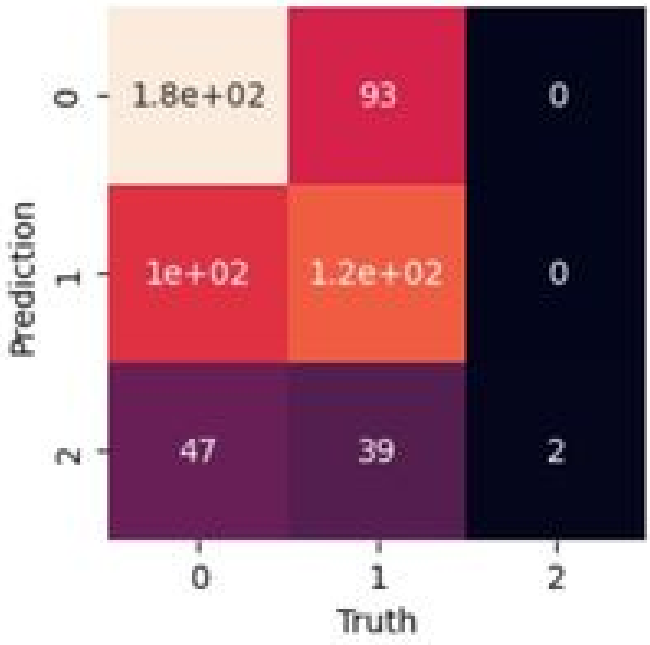
Result:

We have chosen the TfidfVectorizer for text representation in vectors and two models: Logistic Regression and Multinomial Naive Bayes.

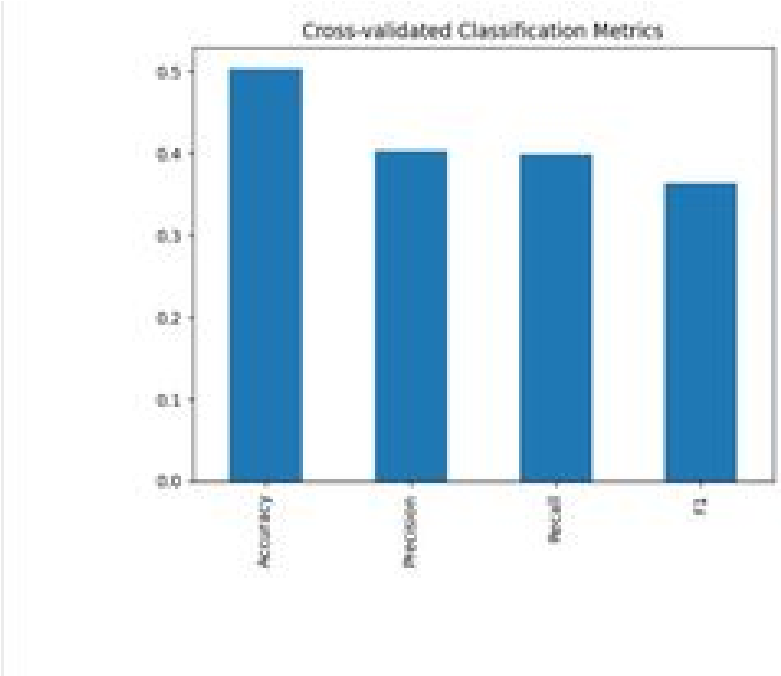
For the Logistic Regression model, the accuracy, precision, recall and f1 scores were 0.53, 0.59, 0.44 and 0.44 respectively.



Confusion Matrix:



For the Naive Bayes model, the accuracy, precision, recall and f1 scores were 0.50, 0.40, 0.40 and 0.36 respectively.



The Logistic Regression model has a higher accuracy (0.53) and precision (0.59) and recall (0.44) than the Naive Bayes model (0.50 and 0.40 and 0.40 respectively), which indicates that the Logistic Regression model made more correct positive predictions and fewer false positives and identified more true positives compared to the Naive Bayes model.

In this comparison, Logistic Regression performs slightly better than the Naive Bayes model. However, both the f1 scores are low even if the f1 score for the first model is better. There is much improvement to be done to get better scores for successfully detecting clickbait

References

References:

1. Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media. In 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2016, San Francisco, CA, USA, August 18-21, 2016, pages 9–16.

2. Zihang Dai, Guokun Lai, Yiming Yang, and Quoc Le. 2020. Funnel-Transformer: Filtering out Sequential Redundancy for Efficient Language Processing. In Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6- 12, 2020, virtual.

3.Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. CoRR, abs/1901.04085.

4.Martin Potthast, Tim Gollub, Matthias Hagen, and Benno Stein. 2018. The Clickbait Challenge 2017: Towards a Regression Model for Clickbait Strength. CoRR, abs/1812.10847.

5.Stephen E. Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. Found. Trends Inf. Retr., 3(4):333–389.