[Clickbait Headlines Classification]

IT 469

Project proposal

Team members

|  |  |
| --- | --- |
| **Name** | **ID** |
| Ghadeer AlMaymoon | 441200441 |
| Dhai AlMurshed | 441200912 |
| Manar Bin Mutaieb | 441201106 |
| Randa Bakhashwin | 441203867 |
| Hadeel Bin Quayyid | 441200913 |

Supervide by:

Dr.Abeer AlDayel

Table of Contents

[Introduction 5](#_Toc135851471)

[Experiment setup 5](#_Toc135851472)

[a) Dataset to be used 5](#_Toc135851473)

[b) The methodology 8](#_Toc135851474)

[Evaluation and results 11](#_Toc135851475)

[Discussion 11](#_Toc135851476)

[Conclusion 12](#_Toc135851477)

[References 13](#_Toc135851478)

Table of Figures

[Figure 1 Dataset After Adding the emotions column 6](#_Toc135851447)

[Figure 2 Input After Word2vec Embedding 6](#_Toc135851448)

[Figure 3 Input After RoBERTa Tokenization 6](#_Toc135851449)

[Figure 4 Dataset Found on GitHub 7](#_Toc135851450)

[Figure 5 Dataset Found on Kaggle 7](#_Toc135851451)

[Figure 6 Top Five Emotions for Clickbait and non-Clickbait headlines 8](#_Toc135851452)

[Figure 7 Clickbait and Non-Clickbait results on RoBERTa model 12](https://studentksuedu-my.sharepoint.com/personal/441200441_student_ksu_edu_sa/Documents/project%20Template.docx#_Toc135851453)

[Figure 8 Clickbait and Non-Clickbait Headlines Examples 12](https://studentksuedu-my.sharepoint.com/personal/441200441_student_ksu_edu_sa/Documents/project%20Template.docx#_Toc135851454)

Table of Tables

[Table 1 case1: classification data distribution 7](#_Toc135851432)

[Table 2 Models Description 9](#_Toc135851433)

[Table 3 Comparing the Performance of Clickbait Classification without Adding Emotions 11](#_Toc135851434)

# Introduction

Nowadays, online news media consumption has spiked as a result of the widespread use of the internet. Acquiring information by reading the news and visiting websites that offer this content resulted in an overuse of clickbait headlines. **Clickbait headlines** are designed to attract your attention, so you click on a link to an article, image, or video (e.g., This Moment Between A Starbucks Barista And A Deaf Customer Is Going Viral). Using a headline that appeals to your curiosity and emotions rather than providing objective facts is key to making users click on that headline. Such practices not only hinder individuals from obtaining the information they need but also result in frustration and disappointment. This has led to a growing need for automated methods to detect clickbait and prevent its spread. Natural Language Processing is a wide discipline that offers text classification methods that can be employed to identify such practices. Upon digging into the subject of clickbait, we discovered that text classification has been utilized in various ways to combat this issue, as exemplified in the paper we came across‎[4]. As a result, in this project, we hypothesize that non-clickbait headlines use neutral language to attract readers' attention while providing accurate and informative information, while clickbait headlines use exaggerated emotional language to capture the reader's attention.

In this document, we will begin with the experiment setup part, which includes the following subsections: dataset to be used and methodology. Then we'll go over our evaluation results and the discussion. Finally, we will provide our conclusion along with the resources.

# Experiment setup

In this section, we will provide general information about the data to be used along with a brief summary of the methodology that we are going to apply.

## Dataset to be used

**Why was the dataset created?**

Through our research, we discovered that the dataset creators designed the data to prevent readers from being misled, as clickbait can deceive readers into clicking on an article but ultimately fail to deliver on their promises, leading to disappointment since clickbait typically refers to the practice of writing sensationalized or misleading headlines in order to attract clicks on a piece of content.‎[3]

**Who funded the creation of the dataset (if available)?**

This dataset is open source and was taken from Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. "Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media". In Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), San Fransisco, US, August 2016.‎[4]

**What preprocessing/cleaning was done? (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances)**

First, we added a column to the dataset called "emotions" that would identify the emotions in the headlines using a RoBERTa model trained on the go\_emotion dataset‎[5], The notebook provided at the given [link](https://colab.research.google.com/drive/1mKViLsvNunG23CuZD7KQd9OaA6lVGIop?usp=sharing) outlines the approach we followed to incorporate emotions into the clickbait dataset. The notebook also presents a breakdown of the prevalent emotions associated with both clickbait and non-clickbait headlines. This preprocessing step will enable us to train the models on both versions of the clickbait dataset (with and without the emotions column) and determine which one yields the most optimal outcomes. Then, we used word2vec embedding for the headlines and emotions columns for the Naïve-Bayes, SVM, LR, and LSTM models. In terms of the RoBERTa model, a RoBERTa tokenizer was implemented on the headlines and emotions columns.

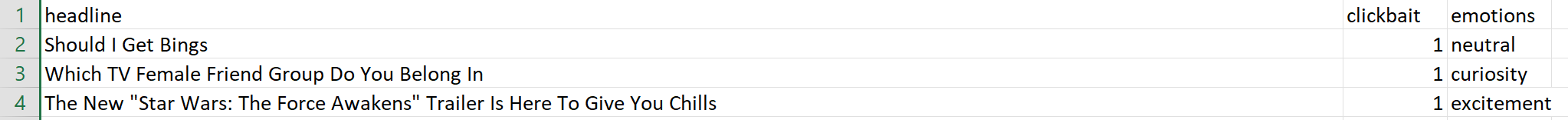


Figure Dataset After Adding the emotions column

|  |  |
| --- | --- |
| Figure Input After Word2vec Embedding | Figure Input After RoBERTa Tokenization |

**If it relates to people, were they told what the dataset would be used for and did they consent? If so, how? Were they provided with any mechanism to revoke their consent in the future or for certain uses?**

The dataset is not related to people, as the dataset creators scraped the web for online news headlines.

**Will the dataset be updated? How often, by whom?**

No, it has not been updated since it was published in 2017.

The dataset, originally found on Github‎[1], contains two raw text files for:

* + Clickbait headlines
  + Non-Clickbait headlines

However, the dataset was also available on Kaggle‎[2] as a single CSV file with two columns: the headlines and the label (clickbait and non-clickbait).

These headlines are from a range of online news sources like 'WikiNews,' 'The New York Times,' 'The Guardian,' 'The Hindu,' 'BuzzFeed,' 'Upworthy,' 'ViralNova,' 'Thatscoop,' 'Scoopwhoop,' and 'ViralStories’.

|  |  |
| --- | --- |
| Figure Dataset Found on GitHub | Figure Dataset Found on Kaggle |

Table case1: classification data distribution

|  |  |  |
| --- | --- | --- |
| Topic | Label1 | Distribution |
| Clickbait | 1 | 16,000 |
| Non-Clickbait | 0 | 16,000 |

## The methodology

In this section, we will navigate our approach to testing our hypothesis, which suggests that non-clickbait headlines utilize words that are not overly emotional while still attracting readers' attention through neutrality (e.g., Nintendo Wii sales decrease in Japan); they tend to prioritize providing accurate and informative information to the reader, whereas clickbait headlines aim to grab attention by using exaggerated emotional language (e.g., This Moment Between A Starbucks Barista And A Deaf Customer Is Going Viral). Emotions are powerful motivators, and headlines that use emotional language can be more effective in capturing the reader's interest than those that do not. Figure 6 displays a comparison between the top five emotions observed in clickbait headlines and those of non-clickbait headlines.

A picture containing text, screenshot, font, number

Description automatically generated

Figure Top Five Emotions for Clickbait and non-Clickbait headlines

In order to test the hypothesis of this experiment, we will follow these steps:

First, we will preprocess the headlines and emotion text using NLP techniques such as word2vec embedding for our baseline model SVM, which was chosen based on the paper that discussed clickbait classification in ‎[4]. Then, we will feed our preprocessed input before and after adding the emotions column to the SVM classification model described in detail in Table 2. After that, to compare the models (SVM with and without emotion), we attempted to determine the p-value for each model by measuring the statistical significance test. The resulted p-value will help us determine which variation will be better tested on different classifiers other than SVM, such as Naïve-Bayes, LR, LSTM, and RoBERTa models, which will be evaluated using f1 score, precision, recall, and accuracy.

In the evaluation, two things will be compared:

* evaluating the performance of the baseline model (SVM) before adding the emotion column and after using statistical significance testing (p-value).
* evaluating the performance of the classifiers against the best version of the dataset.

Table Models Description

|  |  |  |
| --- | --- | --- |
| Model Type | Basic Information about the model | Additional Information |
| Logistic Regression by sklearn‎[6] | The logistic regression model estimates the probability of the dependent variable taking on a certain value based on the values of the independent variables. The model uses a logistic function to transform the linear equation into a probability value between 0 and 1. | We use the Logistic Regression algorithm for our two datasets. The first dataset includes the headlines and emotions as features. We create separate embeddings for each feature using word2vec as we assume that they have distinct meanings. Clickbait is the class label for this dataset. The second dataset includes only the headlines and clickbait as the class label. |
| Naïve-Bayes by sklearn‎[6] | The algorithm works by first estimating the prior probabilities of each class label based on the training data, and then calculating the likelihood of each feature given each class label. These probabilities are then combined using Bayes' theorem to calculate the posterior probability of each class label given the features of a new data point. The class label with the highest posterior probability is then assigned to the data point. | We use the Multinomial Naïve-Bayes algorithm for our two datasets. The first dataset includes the headlines and emotions as features. We create separate embeddings for each feature using word2vec as we assume that they have distinct meanings. Clickbait is the class label for this dataset. The second dataset includes only the headlines and clickbait as the class label. |
| Support Vector Machine by sklearn‎[6] | It works by finding the hyperplane that best separates the data points into different classes. The hyperplane is chosen such that the margin between the hyperplane and the nearest data points of each class is maximized. | We use the Support Vector Machine algorithm for our two datasets. The first dataset includes the headlines and emotions as features. We create separate embeddings for each feature using word2vec as we assume that they have distinct meanings. Clickbait is the class label for this dataset. The second dataset includes only the headlines and clickbait as the class label. |

|  |  |  |
| --- | --- | --- |
| Model Type | Basic Information about the model | Additional Information |
| LSTM by keras‎[7] | Type of recurrent neural network (RNN) architecture used for processing sequential data. | We use LSTM for our two datasets. The first dataset includes the headlines and emotions as features. We create separate embeddings for each feature using word2vec as we assume that they have distinct meanings. Clickbait is the class label for this dataset. The second dataset includes only the headlines and clickbait as the class label. The LSTM model architecture for each dataset includes a Dropout layer with a rate of 0.2 to prevent overfitting, an LSTM layer that has 50 unit to capture the temporal dependencies in the data, a GlobalMaxPooling1D layer to reduce the dimensionality of the output from the LSTM layer, another Dropout layer, and a Dense layer with a sigmoid activation function to perform binary classification. |
| RoBERTa(Xml- RoBERTa)‎[8] | RoBERTa is a language model. It is based on the popular BERT (Bidirectional Encoder Representations from Transformers) model and is trained on a large corpus of text data using a masked language modeling objective. | This model was created by Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov on 26th July 201913.  The dataset was first split into 25% for testing and 25% for training, then 25% from the training set to the validation set. Then, the label class was mapped to a string instead of an integer, and after that, the input was tokenized using the XML-RoBERTa tokenizer. And finally, the model was trained on one epoch with 16 batches and a learning rate of 2e-5. |

# Evaluation and results

In this section, we will indicate the SVM classifier performance with and without the emotion column and the resulted p-value from measuring the statistical significance test, along with the performance of the different classifiers mentioned in Table 2 over the best version of the data (either with or without the emotion column).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Performance of clickbait classification | | | | | |
|  | F1-Score | Precision | Recall | Accuracy | P-value |
| Support Vector Machine without emotion column | 0.98 | 0.98 | 0.98 | 0.98 | 0.647 |
| Support Vector Machine with Emotion column | 0.978 | 0.978 | 0.978 | 0.978 |
| Logistic Regression | 0.893 | 0.912 | 0.895 | 0.895 | - |
| Naïve-Bayes | 0.966 | 0.966 | 0.966 | 0.966 | - |
| LSTM | 0.965 | 0.963 | 0.968 | 0.965 | - |
| RoBERTa | 0.994 | 0.994 | 0.994 | 0.994 | - |

Table Comparing the Performance of Clickbait Classification without Adding Emotions

# Discussion

After conducting a hypothesis test using a significance level (alpha) of 0.01‎[9] to compare the performance of the SVM model with the inclusion of the emotion feature and without it, the SVM model failed to reject the null hypothesis, which stated that adding this feature would not improve the performance of the model, indicating that there is no statistically significant difference between the two models. This finding proves that emotions may not play a significant role in distinguishing between clickbait and non-clickbait headlines, even though emotional language is commonly considered a critical aspect of clickbait headlines. However, this might be due to the fact that the dataset consists of well-structured headlines that contain more linguistic or syntactic features; these features may provide enough information for the classification models to distinguish between clickbait and non-clickbait headlines without the need for additional emotional features. In this case, the emotional features may not have contributed significantly to the classification performance, as the linguistic and syntactic features alone may have been sufficient. Based on that, we evaluated the other classification models mentioned previously using the clickbait dataset without adding the emotion feature. We found that there was no substantial difference in the F1 score values. However, the RoBERTa classifier appears to have the best performance among the others, with a 0.99 F1 score. We tested the RoBERTa classifier on nine examples, and it correctly classified seven out of nine examples, as shown in Figure 8 and Figure 7 below.

A picture containing text, font, document, screenshot

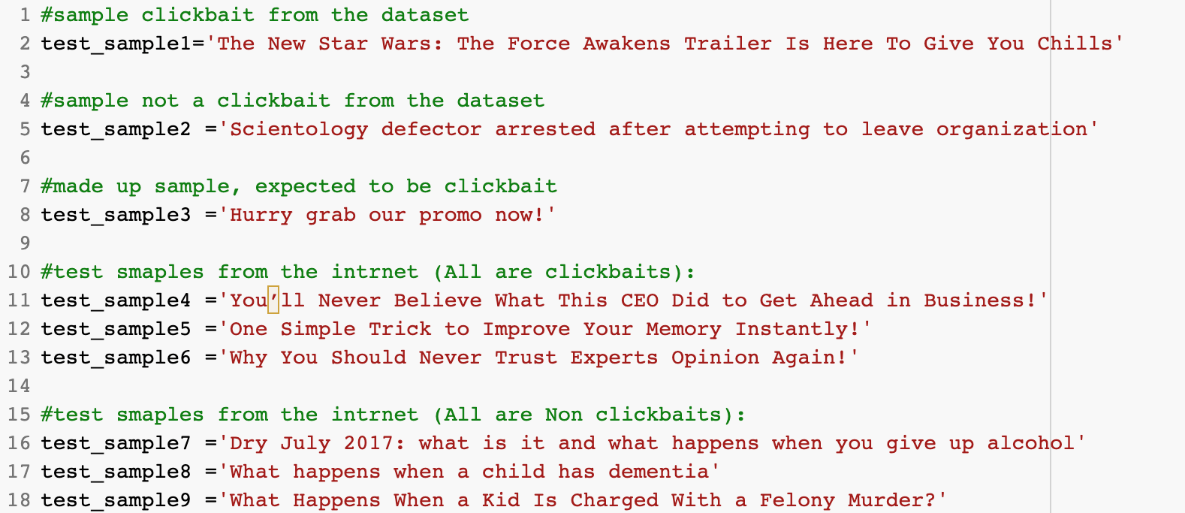
Description automatically generated

Figure 7 Clickbait and Non-Clickbait results on RoBERTa model

Figure 8 Clickbait and Non-Clickbait Headlines Examples

# Conclusion

This project aimed to investigate the relationship between emotions and clickbait headlines, hypothesizing that clickbait headlines use exaggerated emotional language to grab attention while non-clickbait headlines do not. We used various NLP and ML techniques to create a variation of the clickbait dataset that includes an emotion column, preprocess the text input to be suitable for entering the classification models using word2vec embedding and RoBERTa tokenization, classify the headline with and without the emotion feature using the SVM classification model, and evaluate different models on the best dataset, which was the one without the "emotion" feature. This might suggest that the existing features in the dataset are already informative enough for classification. However, there may be other types of features or representations that could be explored in future work to further enhance the accuracy of the model. One promising approach could be to incorporate a different model for adding emotional features to the dataset. By exploring different models and approaches, it may be possible to improve the effectiveness of clickbait classification and gain further insights into the role of emotions in headline language.

# References

1. “SVM clickbait classifier” github.com <https://github.com/bhargaviparanjape/clickbait> [Accessed 27 April]
2. “Clickbait Dataset” Kaggle.com <https://www.kaggle.com/datasets/amananandrai/clickbait-dataset> [Accessed 27 April]
3. “Clickbait” coschedule.com [https://coschedule.com/marketing-terms-definitions/what-is-clickbait#](https://coschedule.com/marketing-terms-definitions/what-is-clickbait) [Accessed 27 April]
4. A. Chakraborty, B. Paranjape, S. Kakarla and N. Ganguly, "Stop Clickbait: Detecting and preventing clickbaits in online news media," *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM),* San Francisco, CA, USA, 2016, pp. 9-16, doi: 10.1109/ASONAM.2016.7752207.
5. “roberta-base-go\_emotions” huggingface.co <https://huggingface.co/SamLowe/roberta-base-go_emotions> [Accessed 19 May]
6. “Supervised learning” supervised\_learning.html <https://scikit-learn.org/stable/supervised_learning.html> [Accessed 21 May]
7. “LSTM” keras.io <https://keras.io/api/layers/recurrent_layers/lstm/> [Accessed 21 May]
8. Liu, Ott, Goyal, Du, Joshi, Chen, Levy, Lewis, Zettlemoyer, & Stoyanov. (2019, July 26). RoBERTa: A Robustly Optimized BERT Pretraining Approach. In arXiv (1907.11692). Retrieved May 21, 2023, from <https://doi.org/10.48550/arXiv.1907.11692>
9. “Here is How to Interpret a P-Value of 0.000” statology.org <https://www.statology.org/here-is-how-to-interpret-a-p-value-of-0-000/#:~:text=A%20p-value%20simply%20tells%20you%20the%20strength%20of,Common%20significance%20levels%20include%200.1%2C%200.05%2C%20and%200.01> [Accessed 22 May]