MAPÚA MALAYAN COLLEGES MINDANAO

ClickSaverPH: Clickbait Detection using MobileBERT and BiGRU for Philippine News Headlines

Matthew Szenel S. Ang

James Paul N. Abid

Andre Deuce B. Tan

Bachelor of Science in Computer Science

Thesis Adviser

Jeanie R. Delos Arcos, DIT

College of Computer and Information Science

ClickSaverPH: Clickbait Detection using MobileBERT and BiGRU for Philippine News Headlines

By

Matthew Szenel S. Ang

James Paul N. Abid

Andre Deuce B. Tan

A Thesis Report Submitted to the College of Computer and Information Science
in Partial Fulfillment of the Requirements for the Degree

Bachelor of Science in Computer Science

Mapúa Malayan Colleges Mindanao
Davao City

March 2024

APPROVAL SHEET

The thesis, entitled **“Click Saver PH: Clickbait Detection using MobileBERT and BiGRU for Philippine News Headlines”** prepared and submitted by Group CS-003 consisted of **Matthew Szenel S. Ang, James Paul N. Abid,** and **Andre Deuce B. Tan** in partial fulfillment of the requirements for the degree of **Bachelor of Science in Computer Science** is hereby accepted.

Jeanie R. Delos Arcos, DIT
(Signature over printed name/date)
THESIS ADVISER

Accepted as partial fulfillment of the requirements for the degree of **BACHELOR OF SCIENCE IN COMPUTER SCIENCE**

Rhodessa J. Cascaro, DIT
(Signature over printed name/date)
DEAN

ACKNOWLEDGEMENT

The proponents of this research would like to extend their deepest gratitude to several people who were pivotal to the success of this study. To the institution, Mapua Malayan Colleges Mindanao, for providing us the opportunity to conduct this research.

To our research advisor, Dr. Jeanie R. Delos Arcos, for her guidance, support, and patience throughout the development of this project. Despite inconveniences, she continually gave us proper support and advice to achieve our objectives.

To our course facilitator, Sir Neil P. Magloyuan, for his guidance, support, and patience throughout the course. Despite the delays, he continually encouraged us to finish our study and arranged the schedule to make it possible to go through with our defense.

To our panelists, panel chair Dr. Patrick D. Cerna, and panel members Ma'am Genevieve A. Pilongo, and Sir Martzel P. Baste, for their feedback during the defenses of the project. Their insight and feedback allowed us to improve the quality of our study.

To our Dean of College of Computers and Information Sciences, Dr. Rhodessa J. Cascaro, for giving us her support as well as the opportunity to go continue with our project despite delays.

To the journalist experts Mr. Ian Ray Garcia, Editor-in-Chief of Mindanao Times, and Ms. Cristina E. Alivio, Editor-in-Chief of Sunstar Davao, and the rest of our respondents for taking time out of their busy schedules in assisting us in the testing of our web application and for their feedback. To our friend, Lance Endaya, for helping us understand the web application development process better.

Lastly, to our family and friends, who were our greatest support throughout this entire study.

TABLE OF CONTENTS

TITLE PAGE	i
APPROVAL SHEET	ii
ACKNOWLEDGEMENT	iii
TABLE OF CONTENTS	iv
LIST OF TABLES	v
LIST OF FIGURES	vi
LIST OF ABBREVIATIONS	viii
ARTICLE 1: Click Saver PH: Clickbait Detection using MobileBERT and BiGRU for Philippine News Headlines	1
Abstract	2
1. Introduction	3
2. Related Works	8
3. Material and Methods	16
4. Results and Discussion	38
5. Conclusions and Recommendations	53
References	55
Appendix A: Gantt Chart	62
Appendix B: User Acceptability Test	63
Appendix C: Letter of Request (Survey)	64
Appendix D: Letter of Request (Journalist Expert #1)	65
Appendix E: Letter of Request (Journalist Expert #2)	66

LIST OF TABLES

Table 1. List of Attributes 18

Table 2. ISO 9126 Quality Model (Chua & Dyson, 2004; Wang et al., 2019) 35

Table 3. Modified ISO 9126 Quality Model used in the study. 36

Table 4. Likert-Range Conversion Table 37

Table 5. Number of headlines breakdown per source, classification, and language 39

Table 6. Number of headlines used in training and testing breakdown per source, classification, and language. 40

Table 7. Training and validation results of the MobileBERT-BiGRU model. 43

Table 8. Test results of the MobileBERT-BiGRU model. 43

Table 9. Usability Testing Results (by Sub-characteristics) 49

Table 10. Usability Testing Results (by Characteristics) 50

Table 11. Journalist Experts Ratings (by Sub-characteristics) 50

Table 12. Journalist Experts Ratings (by Characteristics) 51

LIST OF FIGURES

Figure 1. Agile Methodology 16

Figure 2. Click Saver PH Conceptual Framework 18

Figure 3. MobileBERT Structure and Transformer architecture. Modified from Devlin et al. (2018) and Sun et al. (2020). 22

Figure 4. A visual comparison between the transformer block architecture of (a) BERT, (b) IB-BERT teacher, and (c) MobileBERT student. Adapted from Sun et al. (2020). 23

Figure 5. BERT/MobileBERT input representation. Modified from Devlin et al. (2018). 23

Figure 6. Gated Recurrent Unit (GRU) Model Structure Adapted from Cho et al. (2014). 25

Figure 7. Bidirectional Gated Recurrent Unit (BiGRU) Model. Modified from Schuster & Paliwal (1997). 26

Figure 8. Structure Diagram of MobileBERT-BiGRU Model. Modified from Li et al. (2023). 27

Figure 9. Structure Diagram of MobileBERT-BiGRU Model with sample input. 28

Figure 10. Index page of Click Saver PH (formerly Click2Save) web application. 31

Figure 11. Result when inputting a headline to be classified as clickbait or not. 31

Figure 12. Click Saver PH Flowchart 32

Figure 13. System Architecture Diagram	33
Figure 14. Training accuracy	41
Figure 15. Training loss	41
Figure 16. Validation accuracy	42
Figure 17. Validation loss	42
Figure 18. Test Results Confusion Matrix	44
Figure 19. Index page	45
Figure 20. Input Form (No Input)	46
Figure 21. Input Form (Non-clickbait Output)	46
Figure 22. Input Form (Clickbait Output)	47
Figure 23. History Section	48
Figure 24. About Section	49

LIST OF ABBREVIATIONS

API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
BiGRU	Bidirectional Gated Recurrent Unit
BiLSTM	Bidirectional Long Short-Term Memory
CNN	Convolutional Neural Network
COVID-19	Coronavirus Disease
CPU	Central Processing Unit
DNN	Deep Neural Networks
FNN	Feedforward Neural Network
GloVe	Global Vectors
GRU	Gated Recurrent Unit
GPT	Generative Pre-trained Transformer
GPU	Graphics Processing Unit
HTML	HyperText Markup Language
ISO	International Organization for Standardization
IT	Information Technology
LSTM	Long Short-Term Memory
LR	Logistic Regression
mBERT	Multilingual Bidirectional Encoder Representations from Transformers
MBFC	Media Bias/Fact Check

MNB	Multinomial Naïve Bayes
MobileBERT	Mobile Bidirectional Encoder Representations from Transformers
MLM	Masked Language Model
MLP	Multi-Layer Perceptron
NBC	Naïve Bayes Classifier
NLP	Natural Language Processing
NUJP	National Union of Journalists in the Philippines
PNFC	Philippine Fake News Corpus
PNN	Parallel Convolutional Neural Network
RAM	Random Access Memory
ReLU	Rectified Linear Unit
RFE	Recursive Feature Elimination
RNN	Recurrent Neural Network
SVM	Support Vector Machine
TL	Transfer Learning
XML	Extensible Markup Language

ARTICLE 1

Click Saver PH: Clickbait Detection using MobileBERT and BiGRU for Philippine News Headlines

Matthew Szenel S. Ang[a*], James Paul N. Abid[a], Andre Deuce B. Tan[a]

[a]College of Computer and Information Science, Mapúa Malayan Colleges Mindanao, Davao City, Philippines 8000

Emails: msAng@mcm.edu.ph, jpAbid@mcm.edu.ph, adTan@mcm.edu.ph

*Corresponding author

College of Computer and Information Science

Mapúa Malayan Colleges Mindanao

Gen. Douglas MacArthur Hwy, Talomo

Davao City, Philippines

Email: msAng@mcm.edu.ph

Abstract

Following the digitization of journalism and its spread over social media platforms, the number of sensationalized titles meant to grab viewers' attention for increased traffic of their website (or "clicks") have increased dramatically. This trend of journalism known as "clickbait" has drastically affected the credibility of online journalism, often exaggerating or misrepresenting the content of their articles or videos, with little in the way of actual substance. To combat this problem, the paper aimed to determine the viability of using a pre-trained MobileBERT model and the BiGRU model in the training of a model capable of accurately detecting clickbait. To achieve this, a new dataset of clickbait headlines as well as a web application was developed. The research found that the model can detect clickbait headlines with an accuracy of 97.88% which affirms the validity of MobileBERT and BiGRU in the use of future models. Recommendations for further study include gathering a larger dataset to be used in training, the use of other BERT models for use in clickbait detection, and use of better hardware in training the model.

Keywords: clickbait detection, Philippines, Filipino news, headlines, bilingual, English, MobileBERT, Bidirectional Gated Recurrent Unit (BiGRU), web application, clickbait dataset

SDGs: Goal 17 (Partnerships for the Goals)

1. Introduction

1.1 Background of the Study

The internet has become an integral part of the daily lives of people around the globe, having provided infrastructure for convenient mass communication and remote forms of employment, among other things. One industry which has adapted to this new landscape of global interactivity is journalism, with online or digital journalism having rapidly caught up to its cable and printed counterparts. From 2010 to 2020 there was a massive 33.6% decrease in newspaper revenue in the United States alone as media began to shift from print to digital (Grundy, 2022).

Within the increasing digitalization of journalism, however, came a noticeable trend of tabloidization in news articles; often characterized by exaggerated or misleading headlines with little to no substance within the article proper. This subset of sensationalized news articles, known as clickbait, has been used with increasing frequency by less credible news sources as a means of enticing readers to view the article, thereby increasing online traffic on the hosted website which in turn generates additional advertisement revenue (Ahmad et al., 2020; Chen et al., 2015).

The proliferation of clickbait articles, especially on social media networks such as Facebook and X (formerly known as Twitter), has been attributed with the increase of misinformation found online, such that in an international digital news report conducted in 2022, around 54% of the markets studied are concerned with the veracity of news on the internet, with people who primarily use social media as a source demonstrating a higher level of concern (Newman, 2022).

The consequences of this growing wave of sensationalized and misleading articles are not isolated from the online landscape, however. One common topic frequently touched on by clickbait articles due to its global relevancy was the COVID-19 pandemic and materials relating to it. In 2021, an article published by Forbes had a misleading headline that claimed that the COVID-19 vaccine altered a person's DNA, which inaccurately described the process that the article itself was referring to. The headline was changed after criticism; however, the damage had been done and the original headline had been widely spread around social media, igniting more conspiracy theories and further misinformation regarding the topic of the vaccine (Teoh, 2021). Other countries would report misleading articles more related to politics, such as the Philippines (Newman, 2022), one such news claiming that martial law had been declared after a Chinese water cannon attack at the Philippine Coast Guard. The clips highlighted by the video would then later clarify that one of the senators had merely suggested the imposition of martial law to which no other senator explicitly supported (VERA Files, 2023a). Individual users would also be put further at risk by advertisements which clickbait sites often promote. These advertisements will most often endorse scams and malware, which the hosting website will often overlook due to potential revenue and an overall higher clickthrough rate than traditional display ads (Chen et al., 2015).

In the case of the Philippines, the effects of allowing such misinformation to be easily accessible should not be understated. It is reported in 2023 according to Google's advertising statistics that an equivalent of 72.5% of the total population in the Philippines uses social media. Furthermore, the report adds that an equivalent of 49.5% of the total population, roughly half of the total population in the Philippines, is reported to use the video sharing social media platform known as YouTube (Kemp, 2023).

It should be noted, however, that YouTube does not have clear policies against misinformation, allowing many clickbait and provocative titles to continually be posted without being fact-checked through any official sources. Many channels which post these clickbait videos often have high upload and subscriber counts, which may reinforce a false image of credibility to viewers. Moreover, the videos uploaded by these channels often contain titles which may be classified as clickbait, often containing provocative and sensationalized titles or thumbnails which may present a false narrative (Tuquero, 2021). One such example, the video titled “CHINA KINUHA NA PATI ILOCOS! PBBM BINALAAN NA CHINA! GYERA NA TALAGA! BIGLAANG ANUNSIYO NG PANGULO” which has been posted by a popular YouTube channel and has earned over 39,000 views. This video falsely claims that China has invaded the Ilocos Region of the Philippines but does not substantiate its claims. Furthermore, the video plays military footage which has been confirmed by fact checking organization VeraFiles to be clips of the Armed Forces of the Philippines’ military exercises and is unrelated to any military operations in the Ilocos Region involving the Chinese (VERA Files, 2023b).

Thus, there is a need for the development of tools which may be used in allowing internet users to better identify clickbait on the internet; in which this study will be verifying the viability of the MobileBERT and BiGRU algorithms in the development of such tools.

1.2 Statement of the Problem

Serial tabloidization and sensationalized headlines have pervaded online journalism which has caused cases of misinformation and rumors of false news to be spread around the internet, notably around social media networks. Not only can such misinformation have real world consequences, especially regarding topics such as the COVID-19 pandemic or national politics, but these articles may also harm individual users via promoting untrustworthy advertisements promoting scams or malware.

This study aimed to address this problem through the development of a new lightweight model for clickbait detection for both the Filipino and English languages using a pre-trained MobileBERT language model with a Bi-directional Gated Recurrent Unit Recurrent Neural Network model to allow users to identify clickbait articles through their headlines and promote the identification of clickbait headline trends.

1.3 Research Objectives

The main objective of this study was to develop a bilingual clickbait detection system for Filipino news headlines and a web application to interact with the model. The deep learning model was trained using datasets obtained from online sources to learn to classify whether a news headline is clickbait or not. The following are the specific objectives of the study:

1. To generate an updated dataset consisting of clickbait and non-clickbait Philippine news headlines in both English and Tagalog for use in training a clickbait detection model.

2. To develop a bilingual clickbait detection system for Philippine news headlines using a pre-trained MobileBERT language model and a Bi-directional Gated Recurrent Unit (BiGRU) Recurrent Neural Network (RNN) model.
3. To evaluate the performance of the trained clickbait detection model using accuracy, precision, recall, and F1 score.
4. To develop a web application to interact with the clickbait detection model.

1.4 Significance of the Study

Bilingual language models trained in clickbait detection have previously developed, however, few have been developed specifically for use in the Filipino language. In these previous studies, techniques such as deep learning with recurrent neural networks (RNN), long short-term memory (LSTM) networks, bidirectional encoder representations (BERT) have been used. This study tested the efficacy of using lighter weight deep learning techniques using MobileBERT and BiGRU for better access and use by the general Filipino public. This is intended to reduce misinformation caused by reading headlines only rather than reading the entire article or watching the entire video as the headlines tend to sensationalize and greatly exaggerate the intended message.

2. Related Works

2.1 Clickbait Detection

Daoud and El-Seoud (2019) developed a linear Support Vector Machine (SVM) model for clickbait detection trained on social media posts. The researchers emphasized keeping the number of features to a minimum, as having a huge number of features affects the model's performance and ability to be used in a real-time application. Thus, they chose to train their model on only 24 features. Some of these features are: Similarity, Formality, Readability, Bag of words, and Noun extraction. The researchers also trained a Logistic Regression model to compare with the linear SVM model. The models were trained on 2,495 posts which consisted of 762 clickbait posts and 1,697 not clickbait posts. For validation, they used 19,487 posts which consisted of 14,774 clickbait posts and 4,713 non-clickbait posts. Both models achieved 79% accuracy. The researchers chose the SVM model over the Logistic Regression model because it can handle noise in the data better, and by using the Grid Search algorithm, makes the former faster to train. The researchers highlight the relevance of the social media post elements, and the importance of several features extracted from social media posts elements, the article title, and the article body in detecting clickbait. Furthermore, the study demonstrates that it is possible to create an accurate clickbait detection model with a low number of features, proving that it is possible to create a real-time classifier.

Klairith and Tanachutiwat (2018) developed a Thai clickbait detection model using deep learning neural networks. The study tested out three (3) deep learning neural networks that have been used in Natural Language Processing (NLP) tasks. Namely, a Feed-Forward Neural Network (FNN) and two Bidirectional Recurrent Neural Networks

(RNN), one using the Long-Short Term Memory (LSTM) architecture and the other using the Gated Recurrent Unit (GRU) architecture. The researchers further tested these architectures by using two (2) different types of embeddings, Character Level embedding and Word Level embedding. Through the use of Scrapy, a total of 30,000 titles had been collected from various content publishers since February 2017. The non-clickbait and clickbait titles are manually chosen and separated to prevent false negatives. With an f1-score of 0.98 and an accuracy rate of 0.98, the BiLSTM with word level embedding had the best results.

Pandey and Kaur (2018) developed a clickbait identification system using a Bi-directional Long Short-Term Memory (BiLSTM) neural network, and an evolutionary algorithm, namely the Genetic Algorithm, for hyperparameter tuning. The researchers undertook this study as the intentional usage of eye-catching and misleading content to attract users has been prevalent. The study compared the LSTM and Bi-LSTM model with other algorithms such as Support Vector Machines (SVM), Logistic Regression, Random Forest, and Multi-Layer Perceptron (MLP). In addition, the study compared using GloVe or Word2Vec for generating word embeddings, and different activation functions for the LSTM and BiLSTM model. Their dataset has a total of 80,000 headlines that were gathered from popular news websites such as CNN, Buzzfeed, TheHindu, The Guardian, ScoopWhoop, Clickhole, and Wikinews. The researchers manually labeled each title with non-clickbait title and clickbait title. The non-clickbait titles have a total of 40,600 and 39,400 clickbait titles. Results show that BiLSTM with GloVe embeddings and ReLU activation performs the best with an accuracy score of 98.78%.

2.2 Clickbait Detection in the Philippines

Dimpas et al. (2017) developed a bilingual clickbait detection model that can classify if a headline is clickbait or non-clickbait for both Filipino and English languages. The proposed model used a Bidirectional Long-Short Term Memory (BiLSTM) architecture and Word2Vec to develop the model. For training, the researchers collected data from Philippine-based websites. The dataset consisted of full Filipino headlines, full English headlines, and headlines containing both languages. The criteria were based on their reputation as a news source for non-clickbait data sources, and if they contained sizable amounts of clickbait articles for clickbait data sources. The BiLSTM model achieved a 91.5% validation accuracy.

Cruz et al. (2019) constructed the first expertly curated benchmark dataset called "Fake News Filipino" for Filipino fake news detection, and benchmarked several transfer learning (TL) methods for fake news detection. Fake News Filipino is composed of 3,206 news articles, each labelled either *real* or *fake*, with an even split between these two types. The fake news articles were sourced from website tagged as *fake news sites* by the National Union of Journalists in the Philippines (NUJP), and Vera Files, a non-profit fact-checking organization. The real news articles were sourced from established Philippine news websites like The Philippine Star, Abante, and Bandera. The methods benchmarked by the researchers include a Siamese neural network augmented with Long-Short Term Memory (LSTM) and Rectified Linear Unit (ReLU) activations which served as a baseline, and the TL methods ULMFiT, BERT, and GPT-2. Results show that all the TL methods tested are significantly more accurate than the baseline Siamese neural network model by at least 10%, with a finetuned GPT-2 model being 17% more accurate at 96.28% accuracy.

Fernandez and Devaraj (2019) identified seventy-six (76) linguistic cues or features in distinguishing credible news and not credible news. The identified linguistic cues were broadly categorized into eight (8) categories: *Readability Scores*, *Linguistic Dimensions*, *Summative Cues*, *Affective Cues*, *Informality Cues*, *Cognitive Cues*, and *Time Orientation Cues*. These features were experimented on three (3) learning algorithms, namely: Gaussian Naïve Bayes, Logistic Regression, and Support Vector Machines (SVM). A recursive feature elimination (RFE) algorithm was used to select the best performing features for each learning algorithm from the total seventy-six (76) features. The RFE algorithm was augmented with Grid Search (GS) in finding the optimal hyperparameters for Logistic Regression and SVM. Two (2) datasets consisting of Philippine news in English containing the news headline, content, author, date, URL, news source, and a label was created for training the models. The first dataset, referred as "Philippine Fake News Corpus – PNFC", contained news articles published from January 1, 2016 to October 30, 2018 and was used in training and testing the models. The second dataset, referred to as "Jan2019_data" contained news articles published within January 2019 was used to validate and test the models to see if they are able to classify news that were not part of the PNFC dataset. Results showed that the models were able to attain an accuracy and a precision of 94% when classifying news articles based on their headline and content. The models that classify news articles based on just its headline had accuracies and precisions between 84-86%. The models that classify news articles based on just its content had an accuracy and precision of 88%. Differences between credible and not credible news were also discovered. Not credible news tended to have longer headlines compared to credible news, credible news had longer contents, and credible news tended to be reported in the past tense while not credible news were written more in present tense.

2.3 BERT as a Language Model in Clickbait Detection

Ahmad et al. (2020) made an Experimental Evaluation of Clickbait Detection with use of multiple machine learning algorithms such as Support Vector Machine (SVM), Logistic Regression (LR), Naïve Bayes Classifier (NBC), Long Short-Term Memory (LSTM), Parallel Convolutional Network (PNN), and Bidirectional Encoder Representations from Transformers (BERT) for clickbait detection. The dataset has a total of 32,000 of titles of clickbait and non-clickbait titles that was collected from Buzzfeed, Viral Nova, Up worthy, Wiki News, New York Times, The Hindu, and The Guardian. The scoring method that was used for this study to evaluate each algorithm were accuracy, precision, recall and F1-score. BERT has scored higher in the accuracy of 0.977, BERT and SVM has the same Precision score of 0.979, BERT scored is the second lowest in recall with 0.97, but only a 1% percent difference with the highest, which is LR with the score of 0.98%. BERT scored the highest in the rest with F1 score with 0.983%.

Fakhruzzaman et al. (2021) proposed a study by using a neural network with pre-trained language model M-Bert that will detect clickbait headlines that has been for click counts or pay-per-click (PPC). With a total of 6,632 headlines as a training dataset, the used neural network had performed well with a score 0.914 on both accuracy test and f1 score, and 0.916 in precision core, and a ROC-AUC score of 0.92.

Gothankar et al. (2021) made an experiment using logistic regression, random forests, and Multilayer perceptron to be used for detecting clickbait YouTube videos. The dataset has been taken using Google YouTube API consists of for the list of video IDs fetched from a GitHub source. The dataset has a total of 8,219 videos, of which 4,300 are non-clickbait and 3,919 are clickbait videos. The result with the BERT gives an accuracy of 94.5%, higher than the other techniques used for this experiment, which are Logistic regression, Random Forest classifier, Multi-layer perceptron.

Mahtab et al. (2022) tested various methods in clickbait detection for Bengali news articles. These methods include Statistical Classifier Models (Logistic Regression, Random Forest, Decision Tree, Support Vector Machines, etc.), Deep Neural Networks (CNN, Bi-LSTM, Bi-GRU), and pre-trained fine-tuning language representation approaches (mBERT, BanglaBERT, Indic-BN-BERT, RoBERTa, DistilBERT, etc.). For data collection, they used scrapping and already existing dataset. The researchers scraped most popular Bengali news portal websites, and collected over the total of 11,826 unique news headlines along with content in our dataset. They used an existing dataset which is called Banfake dataset with a total of 48,000 authentic news taken from popular and reliable Bangla online news portals. Another 2,000 titles that are fake and misleading clickbait titles and satire news are added to the dataset. Despite the labels on the articles they gathered, there were overlaps between satire, clickbait, and deceptive labels. As a result, they manually classified these articles as clickbait or non-clickbait to keep the same labeling style across all articles. Their dataset now includes all the clickbait news from that collection and increased with 13,460 total unique news. Results show that generally, Deep Neural Networks perform better than Statistical Classifier Models.

However, Transformer models outperform both classifier models and deep neural networks. Furthermore, most of the Transformer models that were trained only on titles outperformed their title-content combined counterparts, emphasizing the importance of the stylistic cues of titles to improve clickbait detection performance.

2.4 Usage of RNN in Text Classification

Harjule et al. (2020) compared several algorithms for text classification, specifically, sentiment analysis. The algorithms used were the lexicon-based SentiWordNet and Word Sense Disambiguation, and the machine learning based Multinomial Naïve Bayes (MNB), Logisitic Regression (LR), Support Vector Machines (SVM), and Recurrent Neural Network with Long-Short Term Memory (RNN-LSTM). Two (2) datasets were used separately in training and testing the model, namely the "Sentiment140" dataset from Standford University which contained 1.6 million tweets, and "Crowdflower's Data for Everyone library" which contained 13,870 entries. Results showed that the machine learning based models were far more accurate than the lexicon-based models, with RNN having the highest accuracy.

Zulqarnain et al. (2019) proposed the usage of Gated Recurrent Units (GRU) over other existing Recurrent Neural Network (RNN) models for usage in text classification. Namely, GRU was compared to RNN, MV-RNN, and LSTM. The study benchmarked the models on two (2) datasets: Google snippets and TREC. Results show that GRU achieved the best performance in terms of accuracy and error rates.

2.5 BERT and BiGRU for Text Classification

Li et al. (2023) takes the Chinese test data from Weibo platform as their research object, with 1,060,000 original Weibo data from January 1st to May 31, 2020. They used the BERT-FGM-BIGRU model and traditional model and compared the models. The conclusion reveals that BERT-FGM-BIGRU model has shown better results in classification effect and has the highest accuracy compared to the other models.

Liu et al. (2020) used BERT-BIGRU-Softmax with hybrid masking, review extraction and attention mechanism as their deep learning model for Sentiment Analysis. With over 500 thousand product reviews, the results showed that the Proposed model of BERT-BIGRU-Softmax has outperformed the other models including RNN, BiGRU, and Bert-BiLSTM with over 95.5% accuracy and still retained a lower loss for the e-commerce reviews.

Yu et al. (2021) proposed the BERT-BiGRU model as a means to fix the problems and difficulties in NLP (natural language processing) like the use of metaphors, semantic diversity, and grammatical specificity. Using f1 score, the results show that the BERT-BIGRU model has good performance with 0.9, proving that the BERT-BIGRU model has great performance when it comes to Chinese text classification task.

Zhang et al. (2022) constructed a deep learning model, BERT-BIGRU-ATT, to extract disease-medication relationships. The researchers used a Chinese pre trained BERT model to generate word embeddings for the question-and-answer data from online health communities in China. The model of BERT-BiGRU-ATT showed better effects than the other two models: LSTM and Gru. BERT-BiGRU-ATT proved that it has greater effects with knowledge extraction, and completing the extraction task of DDEs (diseases, drugs, and drug effect) with an f1 score of 87.26%.

3. Material and Methods

3.1 Research Design

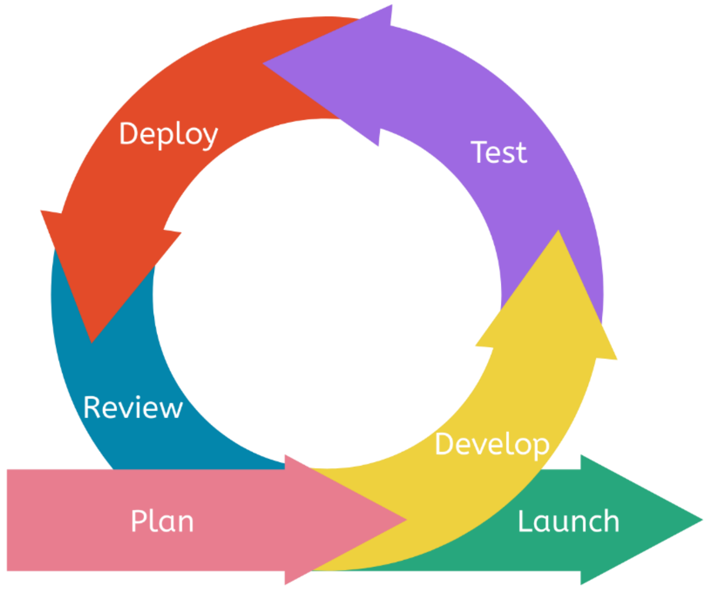


Figure 1. Agile Methodology

This study adopted the Agile software development methodology. Agile takes an iterative approach and divides a project into phases. The planning phase is where the project's requirements were evaluated to determine the feasibility of the project, as well as serve as an outline of the project. The development phase is where the development process of the project occurs. Here, developers utilized the necessary tools in order to build the requirements established during the planning phase. After developing the functions and features of the project, the next phase was the testing phase. Here, testing was done to ensure that the project meets the requirements, and to identify any possible bugs or issues in the code that need to be addressed. Once testing and bug fixing is done, the project was then deployed in the deployment phase where it was used by the public. In this stage, feedback from users of the system was gathered in order to identify further improvements and fixes that need to be made to the system. The phases of the project were then repeated for every new iteration until the final iteration of the project.

3.2 Data Collection

A dataset was generated for use in training and testing the model in this study. The data gathered consists of non-clickbait and clickbait Philippine news headlines with English and Filipino samples for each classification. These news headlines were obtained through scraping various sources available online. The clickbait news headline samples were obtained from online sources that were tagged by Vera Files, a non-profit fact-checking organization in the Philippines, and the Digital Public Pulse 2022 Philippine General Elections Research Report (Bunquin & Gaw, 2022). The non-clickbait news headline samples were sourced from established and trusted news outlets like Philippine Daily Inquirer, and Abante. These trusted news brands are available in the National Library of the Philippines and have been confirmed to be factual according to Media Bias/Fact Check (MBFC), an American website known for rating the political bias and factuality of news outlets globally with a methodology that has been openly shared. Furthermore, these trusted sources have also been used in the previous studies on Philippine clickbait and fake news detection.

3.3 Instruments

Various tools were used for data collection. First, Beautiful Soup, an open-source Python library used in extracting data from HTML and XML files, was used to parse the HTML content of the selected sites of their article headlines. Second, Selenium WebDriver, a web browser automation framework, was used to automate the actions of the web browser while scraping. Lastly, sites with public APIs, mainly YouTube, were used to extract the relevant data from the websites through API calls or requests. Table 1 lists the data attributes to be extracted and used for the study's dataset.

Table 1. List of Attributes

Attribute	Description
Label (Required)	Classification type of the article headline (clickbait or not clickbait).
Headline (Required)	The headline or title of an article.
Brand/Channel (Optional)	The source the headline was obtained from.
Category (Optional)	The specific category the news headline belongs to.
Date (Optional)	The date the news was published.
Link (Optional)	The hyperlink to the specific news source.

3.4 Conceptual Framework

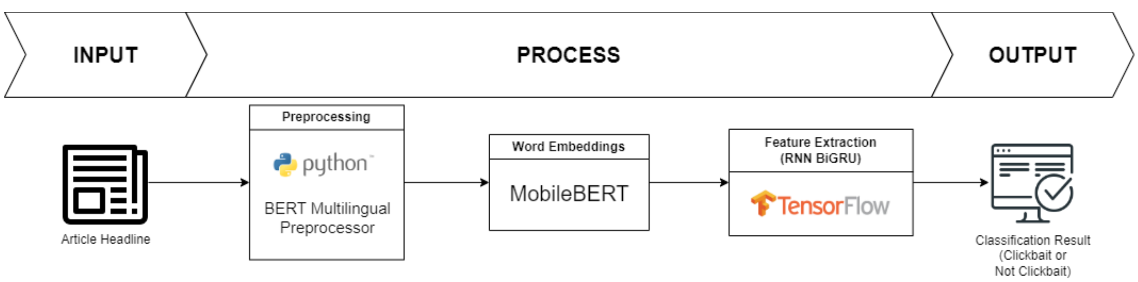


Figure 2. Click Saver PH Conceptual Framework

Figure 2 illustrates the conceptual framework of the Click Saver PH project. The user inputs an article headline text to be evaluated by the system. Before the text is inputted into the model, it will first be preprocessed. Click Saver PH will be using the

BERT Multilingual Preprocessor to process text inputs into the format that BERT expects. Afterwards, the preprocessed text will then go through the pre-trained MobileBERT model which converts the preprocessed text into tensors. These tensors will then be passed to the RNN BiGRU model for feature extraction. The weights resulting from the feature extraction will then be passed to the activation layer, which then outputs the classification result on whether the article headline is a clickbait headline or not.

3.5 MobileBERT Language Model

BERT (Bidirectional Encoder Representations from Transformers) is an open-source language model introduced by Google in 2018. It is designed to be pre-trained from unlabeled text bidirectionally, taking both the left and right context, in use for natural language tasks. The pre-trained model can then be fine-tuned for different tasks. The model achieves state-of-the-art results when benchmarked on eleven (11) of the most common NLP tasks, and obtains better results compared to other language models (Devlin et al., 2018). It is based on the Transformer model architecture (Vaswani et al., 2017), specifically, a Transformer encoder-only structure (Devlin et al., 2018). The Transformer architecture is built on the “attention” mechanism (Vaswani et al., 2017) which works similarly to human attention that selectively chooses the information to remember, with trivial events being less likely to be remembered, and significant events are more likely to be remembered (Mnih et al., 2014). Other language models, including OpenAI’s GPT model, are unidirectional, which only captures context and information from one side. The unidirectional nature of these models is said to limit the potential of pre-trained representations. To overcome this limitation, BERT uses the “masked language model” (MLM) pre-training method. This method enables BERT to learn bidirectionally, in that it captures the context and information from both the left and right side. The BERT model has gained wide usage after its introduction, especially in research studies. The model has been utilized in over 150 studies (Rogers et al., 2020). One of the most notable applications of the BERT model is in Google's very own search engine, Google Search (Nayak, 2019). The BERT model is one of the largest NLP models. However, while it achieves state-of-the-art results, the size of the model makes it suffer from some drawbacks. It has a heavy model size and high latency, requiring more resources to run. This makes it difficult to run the BERT model on resource-limited devices. To solve this issue, the MobileBERT model was proposed (Sun et al., 2020).

MobileBERT is a more compact and faster version of the BERT model. Specifically, it is a thin version of the $BERT_{LARGE}$ model modified with bottleneck structures and a meticulous balance of self-attentions and feed-forward networks. The MobileBERT model is trained by first training an inverted-bottleneck $BERT_{LARGE}$ (IB-BERT) model which will act as a teacher model. Afterwards, knowledge transfer or distillation is conducted to transfer the knowledge of the IB-BERT model to the MobileBERT model. Compared to $BERT_{BASE}$, it obtains similar performance while being 4.3x smaller and 5.5x faster, allowing it to be easily deployed on mobile or more resource-limited devices (Sun et al., 2020).

MobileBERT is not the only attempt at making a more compact and faster approach to the BERT model. The DistilBERT model also utilizes the knowledge distillation process, allowing it to be 40% smaller and 60% faster than $BERT_{BASE}$ while retaining 97% of its performance (Sanh et al., 2019) Another model is the TinyBERT model that also utilizes the knowledge distillation process using a $BERT_{BASE}$ model as its teacher, and further compacts the BERT model. The TinyBERT model is built with only 4 or 6 layers unlike the usual 24 layers. Results show that TinyBERT_4 with 4 layers is 7.5x smaller and 9.4x faster than $\text{BERT}_{\text{BASE}}$ and retains 96.8% of its performance, and TinyBERT_6 with 6 layers is able to perform on-part with $\text{BERT}_{\text{BASE}}$ (Jiao et al., 2019).

While these (3) models were created for the same purpose, in that to be more compact and faster than the original BERT model, the approach for each model differs. The first difference is in the knowledge distillation process. Other methods use the distillation process in both the pre-training and fine-tuning stage, while MobileBERT only uses it for the pre-training stage. Another difference is the approach to compressing BERT. Other models reduce BERT's depth, while MobileBERT reduces its width (Sun et al., 2020).

The performance of each model also differs. MobileBERT is 2.64x smaller and 2.45x faster than DistilBERT, with the former significantly outperforming the latter (Sun et al. 2019). TinyBERT_4 and $\text{MobileBERT}_{\text{TINY}}$ both achieve the same average score, while TinyBERT_6 is able to outperform $\text{MobileBERT}_{\text{TINY}}$ (Jiao et al., 2019). The base MobileBERT model performs similarly to the TinyBERT_6 model. However, it may be unfair to compare MobileBERT and TinyBERT since the former has 24 layers and is task-agnostically distilled from $\text{IB-BERT}_{\text{LARGE}}$, while the latter has 4 layers and is task-specifically distilled from $\text{BERT}_{\text{BASE}}$ (Jiao et al., 2019; Sun et al., 2020). With these differences in mind, MobileBERT will be used as the language model of the study.

The structure of MobileBERT is shown in Figure 3. The general architecture and the input/output representations of MobileBERT remains the same with BERT. *Tok* is the token or the words, *E* is the input embedding or encoder, *Trm* is the Transformer block, *T* is the token output, [CLS] is the special classification token used in classification tasks and is the first token of every sequence, and [SEP] is the separator token that terminates the sentence. In sentence pair tasks, another [SEP] token is inserted between the sentence pair to separate them (Devlin et al., 2018). The changes made to MobileBERT is in its Transformer layer, which is also shown in Figure 3. A comparison of the architecture between the original BERT, IB-BERT, and MobileBERT is shown in Figure 4 (Sun et al., 2020). The input representation of MobileBERT is shown in Figure 5. The input embedding *E* is the sum of three (3) parts: the Token Embeddings, Segmentation Embeddings, and Position Embeddings (Devlin et al., 2018).

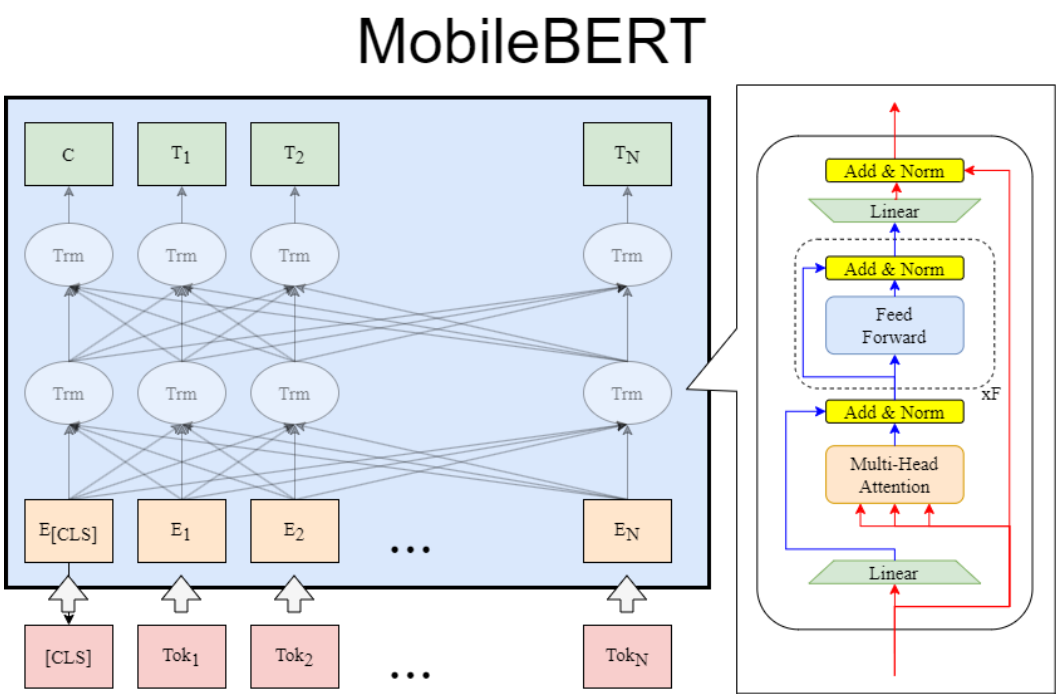


Figure 3. MobileBERT Structure and Transformer architecture. Modified from Devlin et al. (2018) and Sun et al. (2020).

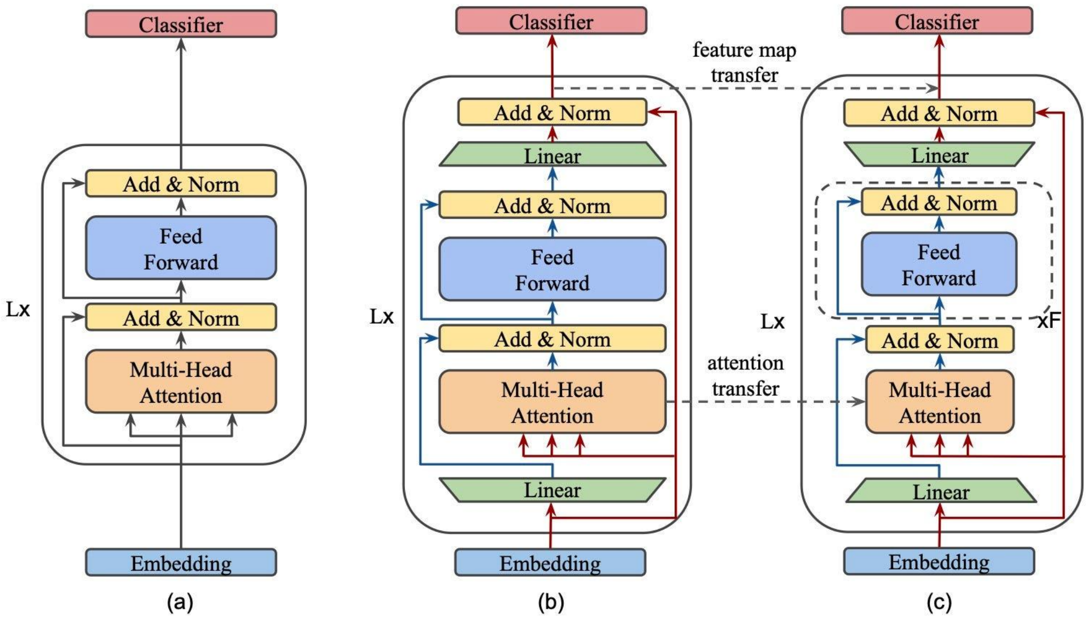


Figure 4. A visual comparison between the transformer block architecture of (a) BERT, (b) IB-BERT teacher, and (c) MobileBERT student. Adapted from Sun et al. (2020).

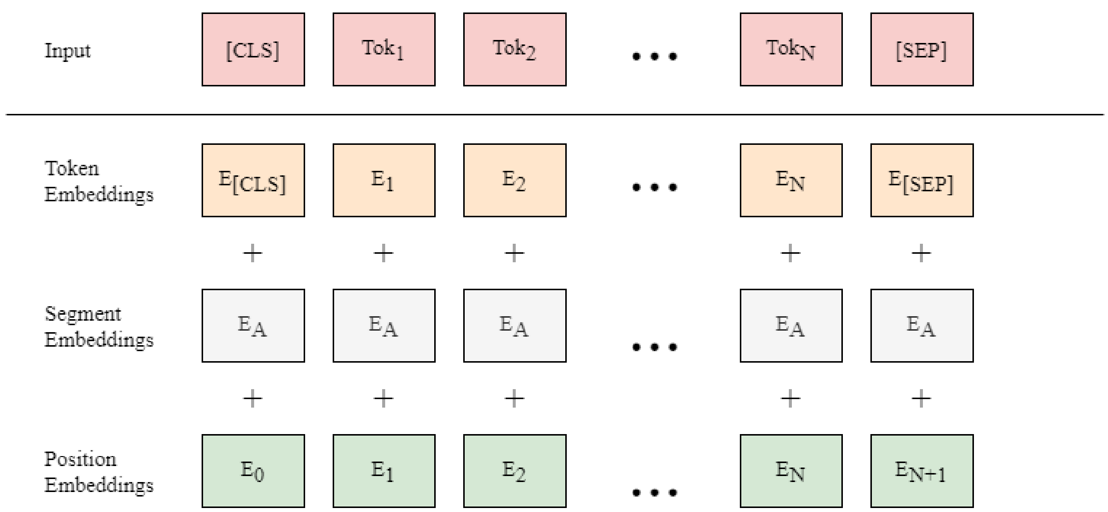


Figure 5. BERT/MobileBERT input representation. Modified from Devlin et al. (2018).

This study will be using a pre-trained multilingual MobileBERT model found in TensorFlow Hub as its language model. This model was trained on the Multilingual Wikipedia Dataset.

3.6 Gated Recurrent Unit (GRU)

Recurrent Neural Networks (RNNs) are frequently used in natural language applications. Some examples of its applications include virtual assistants such as Siri, Alexa, and google Assistant; speech recognition software, language translation, and predictive text (Kostadinov, 2018). However, RNN suffers from a serious problem during backpropagation, where in some cases, the gradient can vanish causing a severe impediment on learning of the neural network. In order to solve the vanishing gradient problem, the Long Short-Term Memory (LSTM) architecture was proposed (Hochreiter & Schmidhuber, 1997). In an LSTM architecture, each hidden unit contains a memory cell and three (3) gates: the *input gate*, *output gate*, and *forget gate*. The memory cell stores information, the input gate regulates the information to be stored in the memory cell, the output gate regulates the information to be accessed from the memory cell (Hochreiter & Schmidhuber, 1997), and the forget gate manages the information to be discarded from the memory cell (Gers et al., 2000).

Another type of RNN that solves the problem of vanishing gradients is the Gated Recurrent Unit (GRU). GRU is an architecture similar to LSTM, however, it has a simpler structure and is less computationally expensive to implement (Cho et al., 2014). Compared to LSTM, GRU merges the memory cell and hidden state into just the hidden state, and a candidate hidden state is added. Furthermore, while LSTM has three (3) gates, GRU only has two (2) gates, the *update gate* and the *reset gate* (Chung et al., 2014). The update gate controls the information to be added into the current state, and the reset gate manages the information to be discarded from the previous state (Cho et al., 2014). Despite the reduced number of gates, GRU is able to achieve similar performance to LSTM while having a shorter training time (Chung et al., 2014). Figure 6 illustrates the structure of a GRU model.

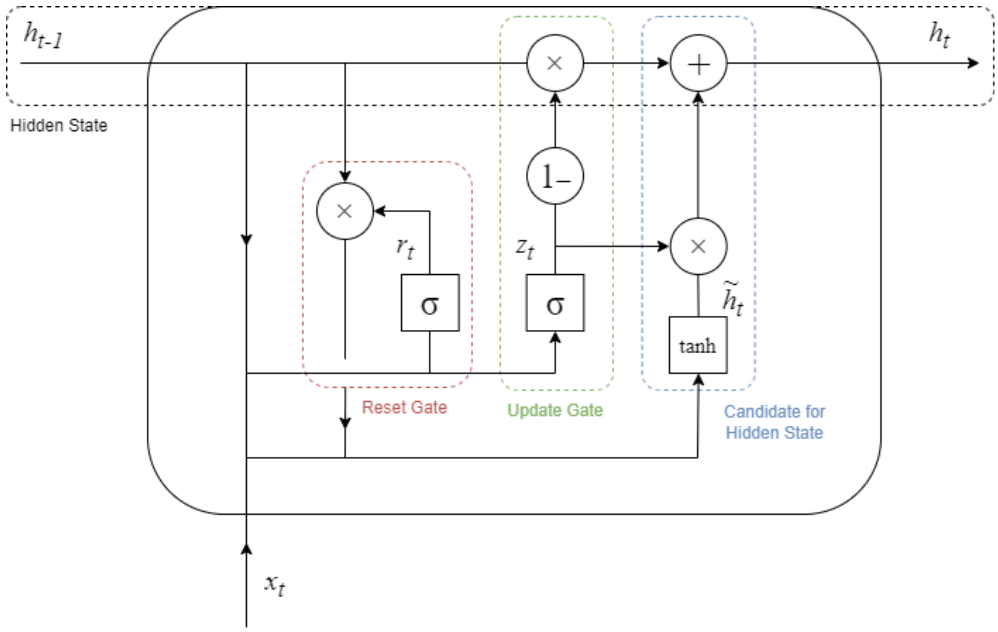


Figure 6. Gated Recurrent Unit (GRU) Model Structure Adapted from Cho et al. (2014).

x_t is the input at time *t*. r_t is the reset gate at time *t*. z_t is the update gate at time t. h_t is the state of the hidden layer at time *t*. $\tilde{h}_t$ is the candidate hidden state at time *t*. The reset gate, update gate, candidate for hidden state, and hidden state at time *t* are calculated as follows:

$$r_t = \sigma(W_r x_t + U_r h_{t-1})$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1})$$

$$\tilde{h}_t = tanh(W_h x_t + U_h(r_t \odot h_{t-1})$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

where *x* is the input, *W* and *U* are the weights (Chung et al., 2014).

While the RNN model (RNN, LSTM, or GRU) is frequently used in solving problems involving time sequential data, the model is only able to capture input information from one direction (Schuster & Paliwal, 1997). In the case of using a unidirectional RNN model in natural language application, when analyzing the possible meaning of a word, only the words preceding the word are captured, and in this case, having the information of the succeeding words contributes to understanding the meaning of a word. Thus, a Bidirectional Gated Recurrent Unit (BiGRU) model will be used for this study to overcome this limitation. A bidirectional recurrent neural network (BRNN) splits the neurons of an RNN into two (2) layers that goes in opposite directions, one does a forward pass or in the positive time direction, and the other one does a backward pass or in the negative time direction. Both layers get their input from the input layer, and are connected to the output layer, but they are not connected to each other. This allows the model to capture all input information from both sides and utilize their outputs in creating the final output (Schuster & Paliwal, 1997). Figure 7 illustrates the structure of a BiGRU.

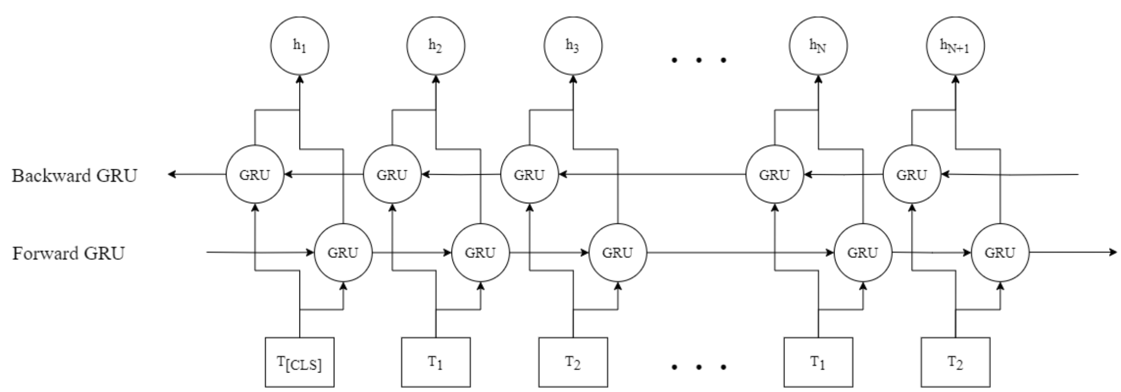


Figure 7. Bidirectional Gated Recurrent Unit (BiGRU) Model. Modified from Schuster & Paliwal (1997).

3.7 MobileBERT-BiGRU Model

To solve the prevalent problem of clickbait news article headlines, especially with the lack of clickbait detection models in the Philippines, a bilingual clickbait news article headlines detection model will be proposed to identify whether a news article headline is clickbait or not clickbait. This model will be using a pre-trained multilingual MobileBERT language model, and a BiGRU model for feature extraction. Figure 8 illustrates the structure of the MobileBERT-BiGRU model. Figure 9 illustrates the structure of the model with a sample input.

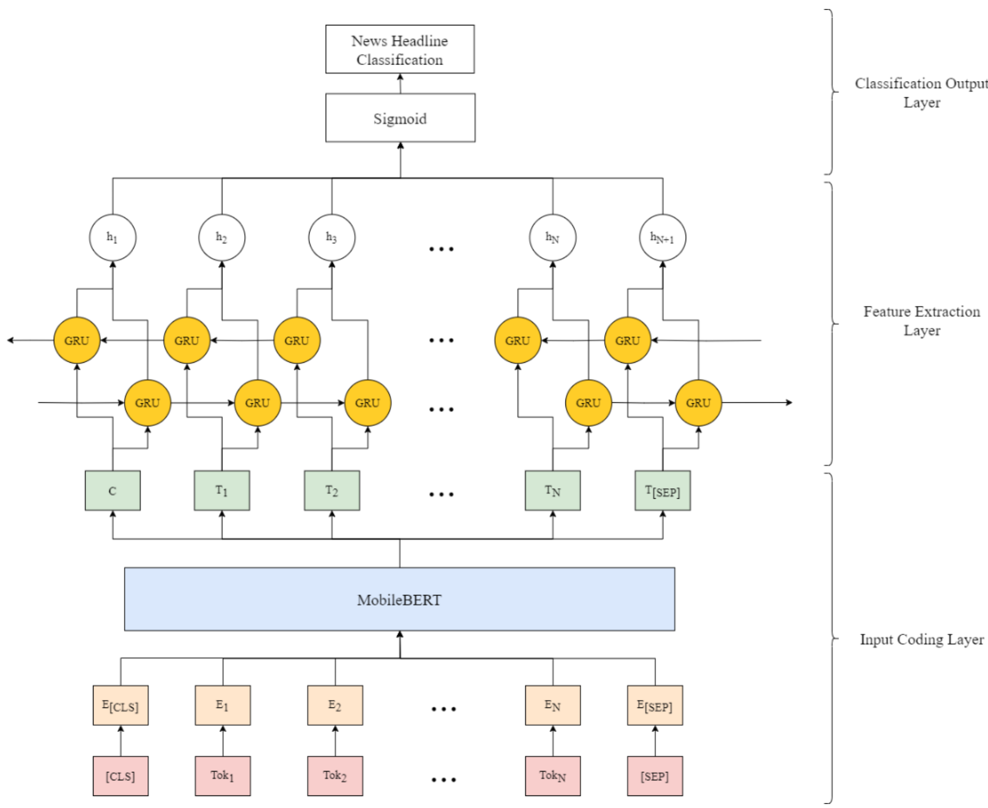


Figure 8. Structure Diagram of MobileBERT-BiGRU Model. Modified from Li et al. (2023).

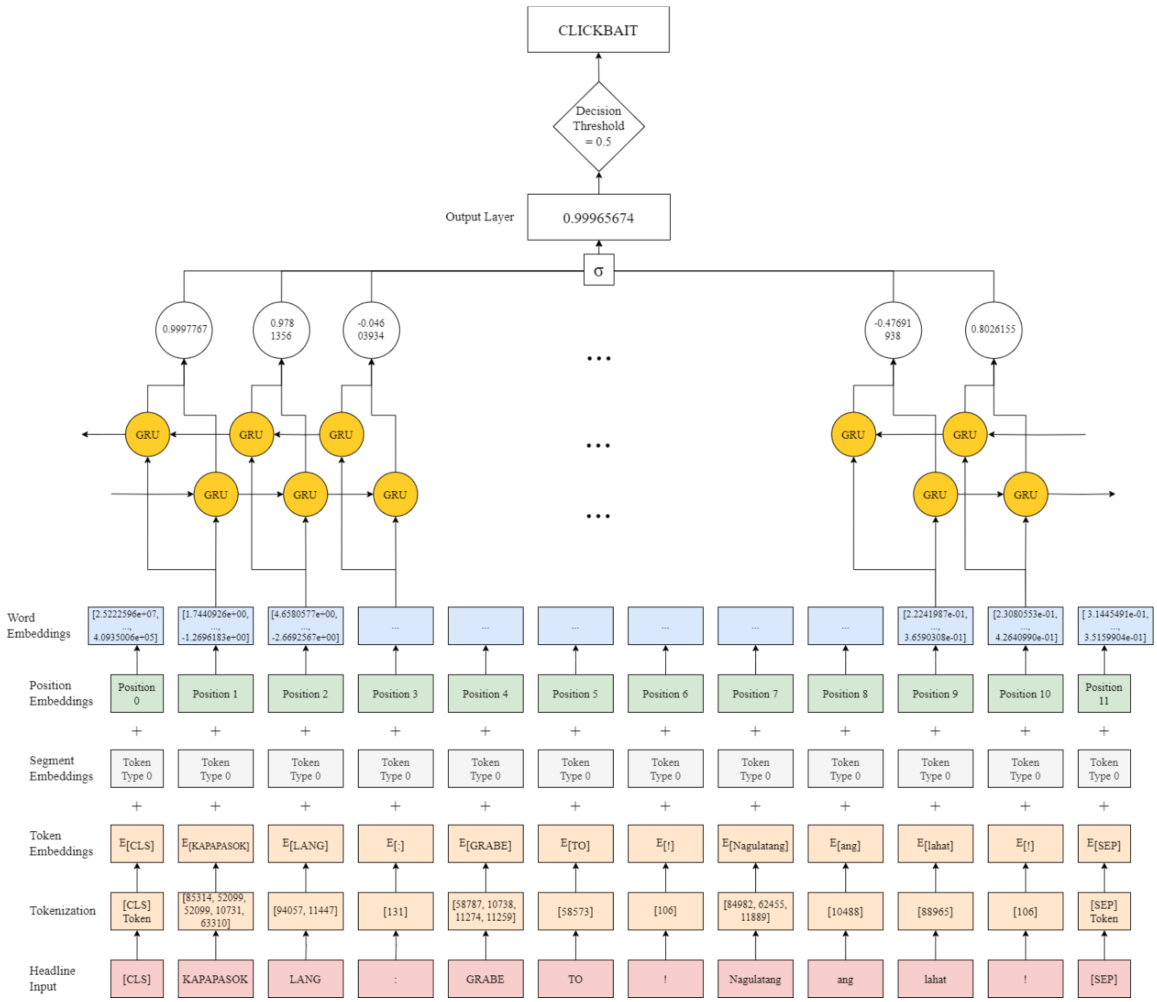


Figure 9. Structure Diagram of MobileBERT-BiGRU Model with sample input.

3.8 Evaluation Metrics

$$\Pr e\,cision = \frac{TP}{TP+FP} \tag{1}$$

Precision is the proportion of correctly positive prediction made for all positive cases. It measures how many of the positive predictions are actually correct. Precision is calculated by dividing the total number of positive predictions or the number of true positives plus the number of false positives by the number of true positives.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Recall, also called as True Positive Rate, is the percentage of true positive predictions out of all actual positive cases, indicating how many positive cases the classifier correctly identified. Recall is determined by dividing the total number of true positives by the sum of true positives and false negatives.

$$F1score = 2 \cdot \frac{\Pr ecision \cdot Recall}{\Pr ecision + Recall} \tag{3}$$

$$F1score = \frac{2}{\frac{1}{Recall} + \frac{1}{\Pr ecision}} \tag{4}$$

The formula for the f1 scoring is given above, will be used by the researchers as their evaluation metric. The formulas that are shown in formula 1 and 2 are the same and have the same result and each can be used alternatively. The F1 Score is a measure that combines recall and precision of a model. TP stands for True Positive when the prediction is positive and correct. FP stands for False Positive when the prediction is positive, but it is negative. False negative when the prediction is negative while it is correct.

3.9 Web Application Wireframe

A web application was made for users to use the Click Saver PH (formerly Click2Save) news headline clickbait detection model. Figure 10 shows the index page of the Click Saver PH web application. Figure 11 shows the results of inputting a news headline to be classified as clickbait or not clickbait. A text below the input textbox will appear stating whether the input headline is clickbait or not clickbait. Figure 12 illustrates the process of the Click Saver PH web application.

Figure 10. *Index page of Click Saver PH (formerly Click2Save) web application.*

Figure 11. *Result when inputting a headline to be classified as clickbait or not.*

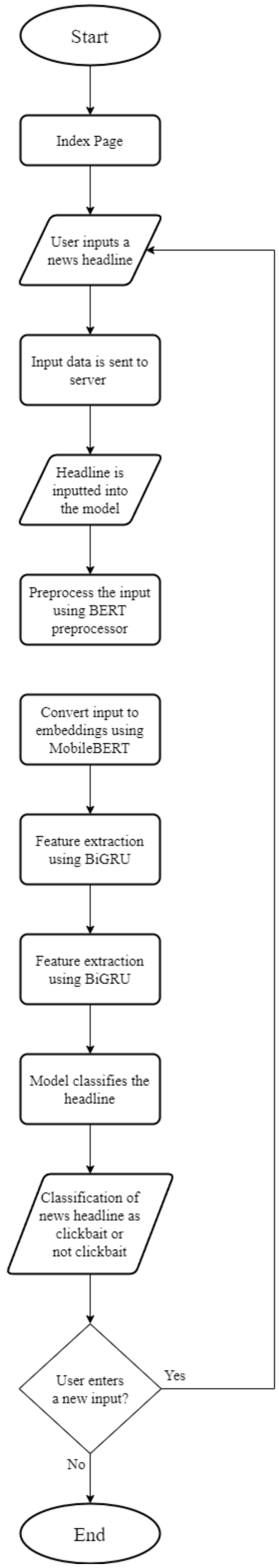


Figure 12. Click Saver PH Flowchart

3.10 System Architecture Diagram

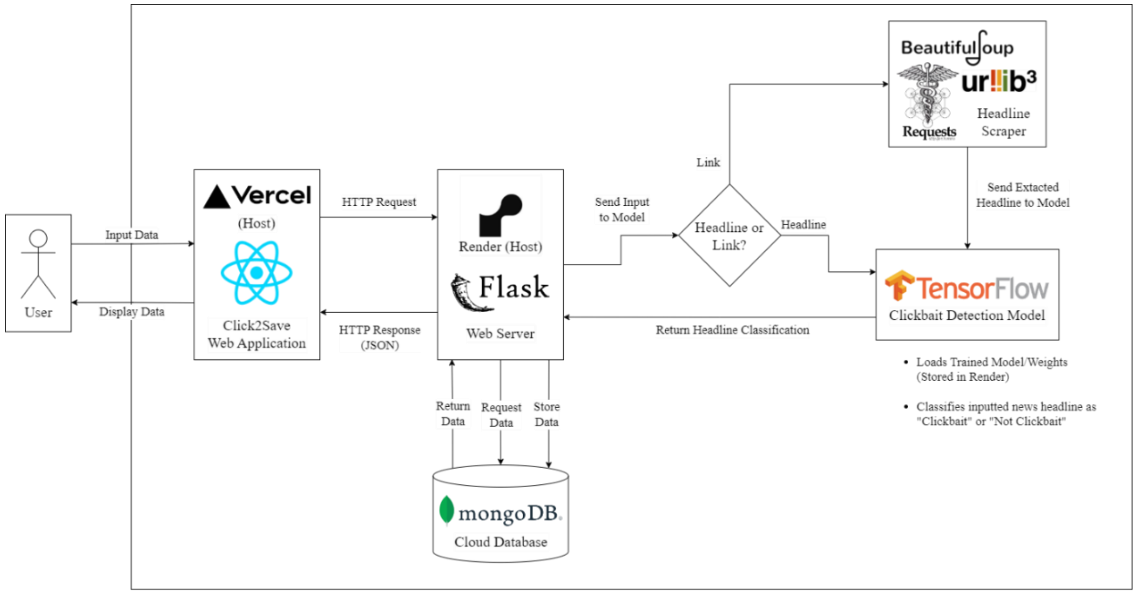


Figure 13. System Architecture Diagram

Figure 13 illustrates the study's system architecture. React, a JavaScript library used in frontend development that utilizes modular components, was used in building the user interfaces of a website. The React web application was hosted on the Vercel cloud hosting platform. The web application then communicates with a REST application programming interface (API) made using the Flask micro web framework which handles HTTP requests and responses. If the input is a headline, it is passed directly towards the clickbait detection model to be processed. If the input is a link, the link is scraped for its headline, then the scraped headline is passed to the clickbait detection model. Once the clickbait detection model, developed using the Tensorflow library, receives the headline, it processes the headline then outputs the classification which the API passes back to the user interface to display to the user. The backend API and the model are hosted on Render cloud hosting platform. Finally, data such as the headline, classification, and source link (if applicable) are stored in the database hosted on MongoDB Atlas.

3.11 Hardware & Deployment

Google Colab, Google's hosted service of Jupyter Notebook, was used in developing and training the model for the study. Specifically, the NVIDIA Tesla T4 GPU was used in training the model and was availed through the Colab Pro subscription. The frontend of the web application was deployed on the Vercel cloud hosting platform with the free Hobby tier. The backend API service of the web application which includes hosting and running the classification model was deployed on Render, a cloud hosting platform, utilizing the paid Pro tier which provides 4GB RAM and 2 CPU cores.

3.12 Sampling and Usability Survey

In gathering respondents to evaluate the performance of the model, a selected sampling of people from the Davao City region were chosen to evaluate the usability of the application. In doing so, the researchers were able to ascertain the intuitiveness and usability of the user interface among people who use technology with differing levels of frequency in their day-to-day lives. In determining the accuracy of the model, professionals with expertise in journalism and editorial experience, one from the Mindanao Times and one from SunStar Davao, were contacted and had agreed to assist the researchers in verifying the reliability of the model's results.

In evaluating the performance of the application, the ISO 9126 model was used as the basis for the metrics utilized. In lieu of specific quality requirements, ISO 9126 instead outlines a general framework of software quality evaluation as a more adaptable metric that can be used across a number of different systems (Chua & Dyson, 2004). These metrics provided six characteristics, each with a number of sub-characteristics as criteria, used to measure the external quality of internal software characteristics during the operation of the application (Wang et al., 2019). Table 2 lists the characteristics and sub-characteristics of the ISO 9126 quality model with descriptions.

For this study, several characteristics and sub-characteristics of the ISO 9126 quality model were omitted because these characteristics were either difficult to assess, requiring a trained IT professional to evaluate, or not applicable (Chua & Dyson, 2004; Valenti et al., 2002). In addition, the 5-point Likert rating scale was used to scale the respondent's answers. Table 3 lists the characteristics and sub-characteristics used in the study with descriptions. Table 4 lists the Likert-Range Conversion Table.

Table 2. ISO 9126 Quality Model (Chua & Dyson, 2004; Wang et al., 2019)

Characteristic	**Sub-Characteristic**	**Description**
Functionality	Suitability (F1)	Is the software capable of performing the required task?
	Accuracy (F2)	Are the results provided by the software within expectations?
	Interoperability (F3)	Can the software interact with other systems?
	Security (F4)	Does the software prevent unauthorized access?
Reliability	Maturity (R1)	Have the faults in the software been eliminated over time?
	Fault Tolerance (R2)	Is the software capable of handling errors?
	Recoverability (R3)	Can the software resume working and recover affected data in case of a failure?
	Reliability Compliance (R4)	Does the software adhere to the existing reliability standards?
Usability	Understandability (U1)	Does the user recognize how to use the system easily?
	Learnability (U2)	Can the software be learnt easily?

	Operability (U3)	Can the software work with a minimal effort?
	Attractiveness (U4)	Does the interface look appealing?
	Usability Compliance (U5)	Does the software meet the existing usability standards?
Efficiency	Time Behavior (E1)	How quickly does the software respond?
	Resource Utilization (E2)	Does the software utilize resources efficiently?
	Efficiency Compliance (E3)	Does the software adhere to the existing efficiently standards?
Maintainability	Analyzability (M1)	Can faults be easily diagnosed?
	Changeability (M2)	Can the software be easily modified?
	Stability (M3)	Can the software continue functioning if changes are made?
	Testability (M4)	Can the software be tested easily?
Portability	Adaptability (P1)	Can the software be moved to other environments?
	Installability (P2)	Can the software be installed easily?
	Portability Compliance (P3)	Does the software comply with portability standards?

Table 3. Modified ISO 9126 Quality Model used in the study.

Characteristic	**Sub-Characteristic**	**Description**
Functionality	Suitability (F1)	Can the web app effectively perform its designated functions?
	Accuracy (F2)	Are the results of the app as anticipated?
Usability	Understandability (U1)	Does the user comprehend how to use the app easily?
	Learnability (U2)	Can the app be learnt easily?
	Operability (U3)	Can the app be used with minimal effort?
	Attractiveness (U4)	Does the app interface look good?

Efficiency	Time Behaviour (E1)	Does the app respond quickly?

Table 4. Likert-Range Conversion Table

Likert-Point	Range	Descriptive Rating	Description
5	4.20 – 5.00	Strongly Agree (SA)	The respondent completely agrees.
4	3.40 – 4.19	Agree (A)	The respondent moderately agrees.
3	2.60 – 3.39	Neutral (N)	The respondent neither agrees nor disagrees.
2	1.80 – 2.59	Disagree (D)	The respondent moderately disagrees.
1	1.00 – 1.79	Strongly Disagree (SD)	The respondent completely disagrees.

3.11 Ethical Considerations

The researchers have considered ethical approaches in evaluating and validating the performance of the model developed. In reaching out to the respondents chosen to assist in testing, a letter of consent was sent out. All respondents were assured that their data and responses would be kept strictly confidential and would not be used outside of this research.

4. Results and Discussion

4.1 Dataset Generation

A total of 27,505 English non-clickbait, 6,321 English clickbait, 18,430 Tagalog non-clickbait, and 8,813 Tagalog clickbait news headlines were collected. While only the Label and Headline fields were required, other fields such as Brand/Channel, Category, Date Published, and Source Link were also collected if present for auditing purposes, and for future uses.

As it is required to have equal amounts of data from each classification to prevent bias when training the model, the size of the dataset was further reduced. The dataset for use with the model contains a total of 24,000 rows, containing 6,000 rows each for English non-clickbait, English clickbait, Tagalog non-clickbait, and Tagalog clickbait news headlines. For sources whose data were not entirely used, a random sample was taken from them. Furthermore, for sources with a Date field, equal amounts of samples were obtained from each month. Specifically, for the Philippine Daily Inquirer dataset whose data ranges from February 2019 to January 2024, one hundred (100) samples were taken from each month per year.

The dataset for use in model development was split into three (3) sets for training, validation, and testing with a split ratio of 80:10:10 respectively. Each set contains equal amounts of English non-clickbait, English clickbait, Tagalog non-clickbait, and Tagalog clickbait news headlines.

The breakdown of the datasets are detailed in Table 5 and Table 6.

Table 5. Number of headlines breakdown per source, classification, and language

	Non-Clickbait Headlines		**Clickbait Headlines**	
English	Philippine Daily Inquirer	27,505	Esquire PH (News)	2442
			Cosmopolitan PH (Entertainment)	2008
			GetRealPhilippines	1200
			Cosmopolitan PH (News)	671
English (Total)		27,505		6,321
Tagalog	Abante	18,430	Philippine Government News	1,239
			Pinas News Insider	1,207
			Terong Explained	1,076
			NEWSFILES	1,032
			PH TV	937
			Balitang Pinas	914
			Boss Balita TV	636
			BALITAAN NG BAYAN	414
			PHILIPPINES TRENDING NEWS	360
			WHISTLE-BLOWER PH 2.0	332
			BALITA NI JUAN	332
			BANAT NEWS PH	167
			BANAT NEWS UPDATES	95
			PINAS VIRAL NEWS	72
Tagalog (Total)		18,430		8,813
Total		45,935		15,134

Table 6. Number of headlines used in training and testing breakdown per source, classification, and language.

	Non-Clickbait Headlines		**Clickbait Headlines**	
English	Philippine Daily Inquirer	6,000	Esquire PH (News)	2,442
			Cosmopolitan PH (Entertainment)	1,687
			GetRealPhilippines	1,200
			Cosmopolitan PH (News)	671
English (Total)		6,000		6,000
Tagalog	Abante	6,000	Philippine Government News	1,239
			Pinas News Insider	718
			NEWSFILES	1,032
			PH TV	937
			Boss Balita TV	636
			BALITAAN NG BAYAN	414
			PHILIPPINES TRENDING NEWS	360
			WHISTLE-BLOWER PH 2.0	332
			BALITA NI JUAN	332
Tagalog (Total)		6,000		6,000
Total		12,000		12,000

4.2 Training Results

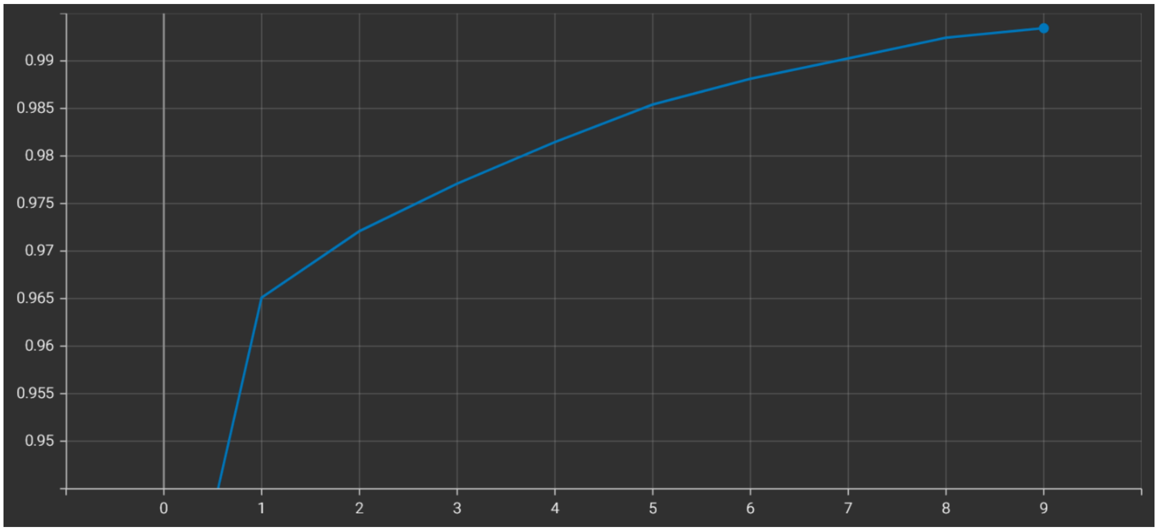


Figure 14. Training accuracy

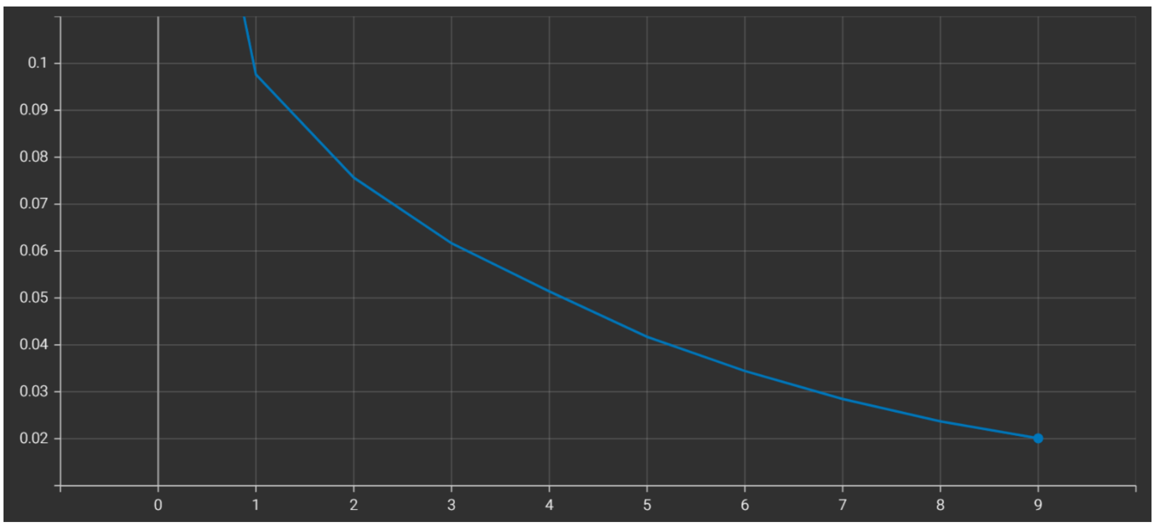


Figure 15. Training loss

Figure 14 and 15 above illustrate the training accuracy and training loss respectively. The model begins with a training accuracy of about 91.99%, jumps to 96.51% on the second epoch, and continues to improve every subsequent epoch. The same can be observed for the model's training loss as it decreases every epoch.

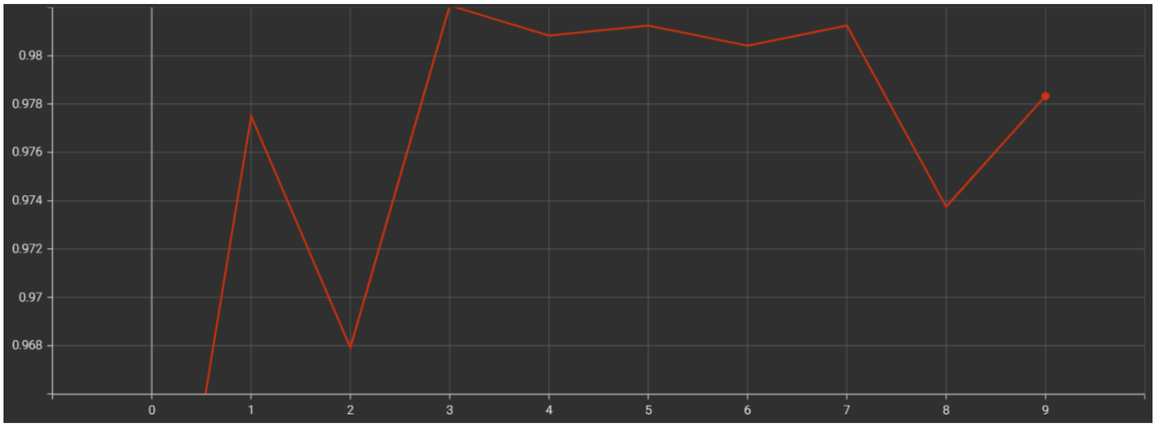


Figure 16. Validation accuracy

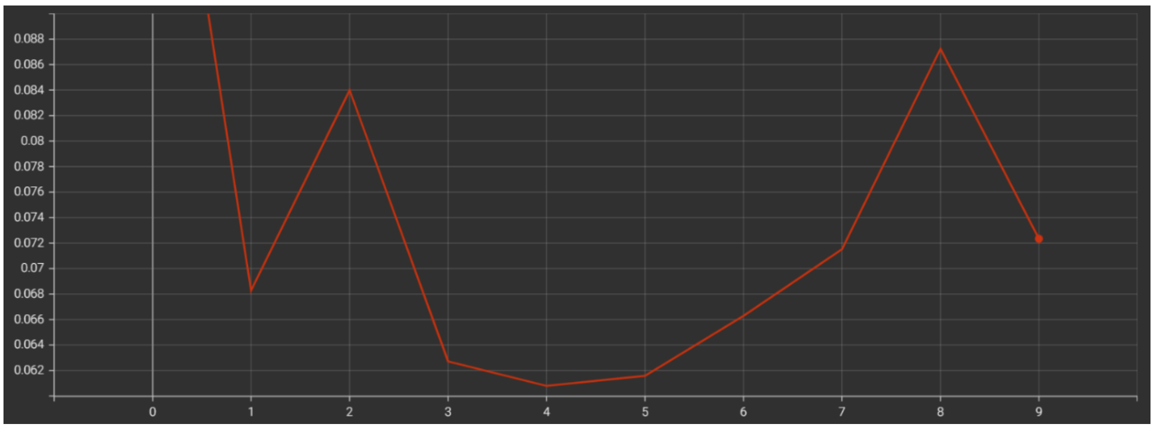


Figure 17. Validation loss

Figure 16 and 17 above illustrate the validation accuracy and validation loss respectively. Validation accuracy stays mostly above 97.8% per epoch. However, when observing validation loss, it stops improving after the 5th epoch and slowly starts increasing. Because validation loss continued to not improve after five (5) more epochs, training was stopped at the 10th epoch, and the state of the model during the 5th epoch is what the study will utilize. Table 7 summarizes the best training and validation performance of the MobileBERT-BiGRU model.

Table 7. Training and validation results of the MobileBERT-BiGRU model.

Accuracy	Val. Accuracy	Loss	Val. Loss
98.15%	98.08%	0.05139	0.06079

4.3 Testing Results

Table 8 below shows the accuracy, precision, recall, and F1 score of the model after testing it. The model was tested on the remaining 2,400 rows of data that was not used in training nor validation. The model manages to attain an accuracy of 97.88%, and precision, recall, and F1 scores above 0.97. These numbers show that our model has good classification ability for non-clickbait and clickbait headlines in both English and Tagalog and has achieved a strong balance between precision and recall; efficiently reducing false negatives as well as false positives.

Table 8. Test results of the MobileBERT-BiGRU model.

Accuracy	Precision	Recall	F1 Score
97.88%	0.9848	0.9725	0.9786

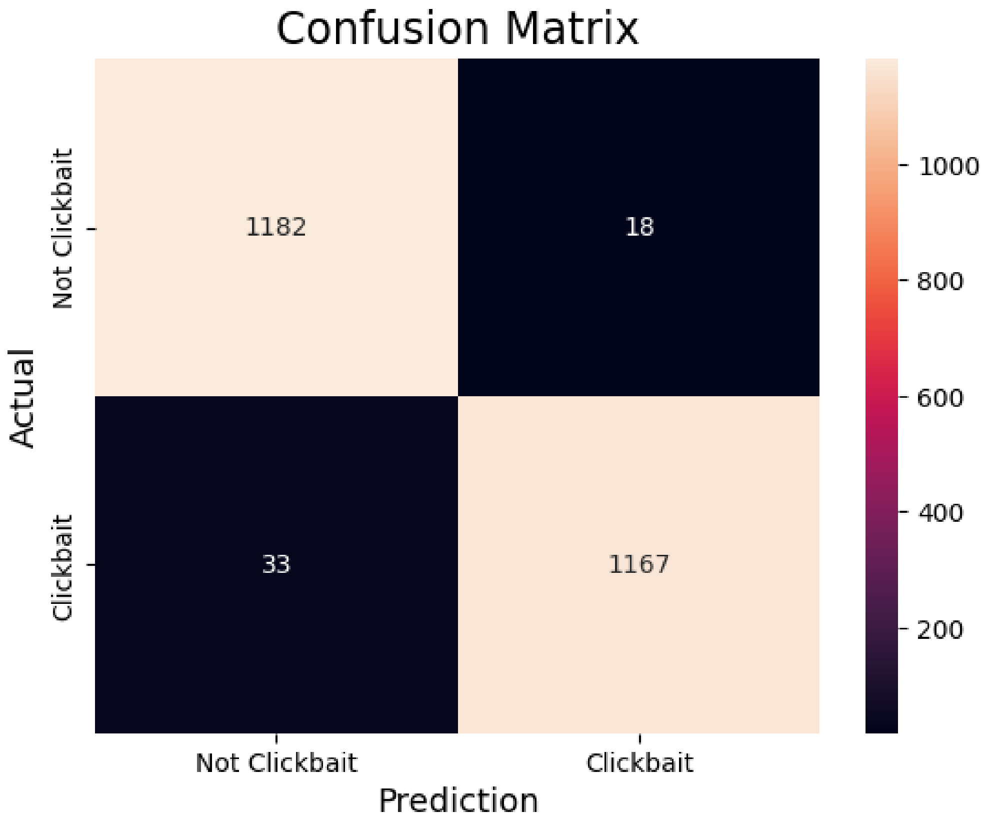


Figure 18. Test Results Confusion Matrix

Figure 18 illustrates the confusion matrix of the test results of the MobileBERT-BiGRU model. Results show that the model correctly identifies 1,182 headlines as “Not Clickbait” and 1,167 headlines as “Clickbait”. However, it falsely identified 18 non-clickbait headlines as clickbait, and 33 clickbait headlines as non-clickbait.

4.4 Web Application

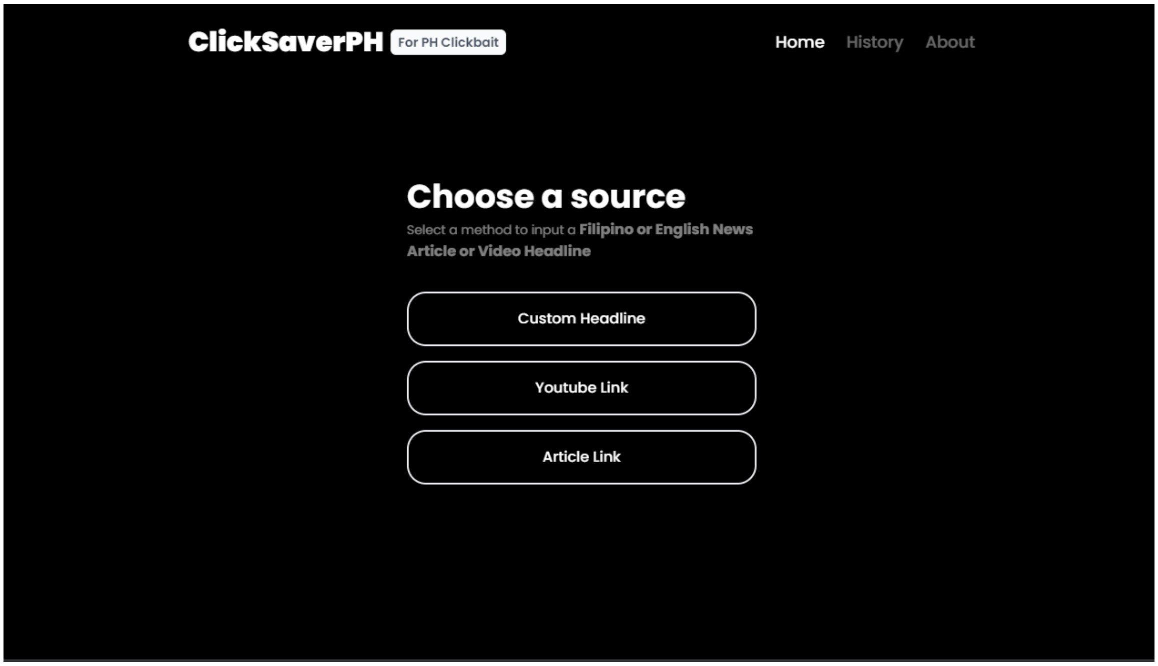


Figure 19. Index page

Figure 19 shows the index page of the proposed web application. The layout of the web application is simple. The user is first given three (3) options whether they want to input the headline themselves, input a YouTube link, or input a link to an article. There is also a navbar which redirects the user to the other pages.

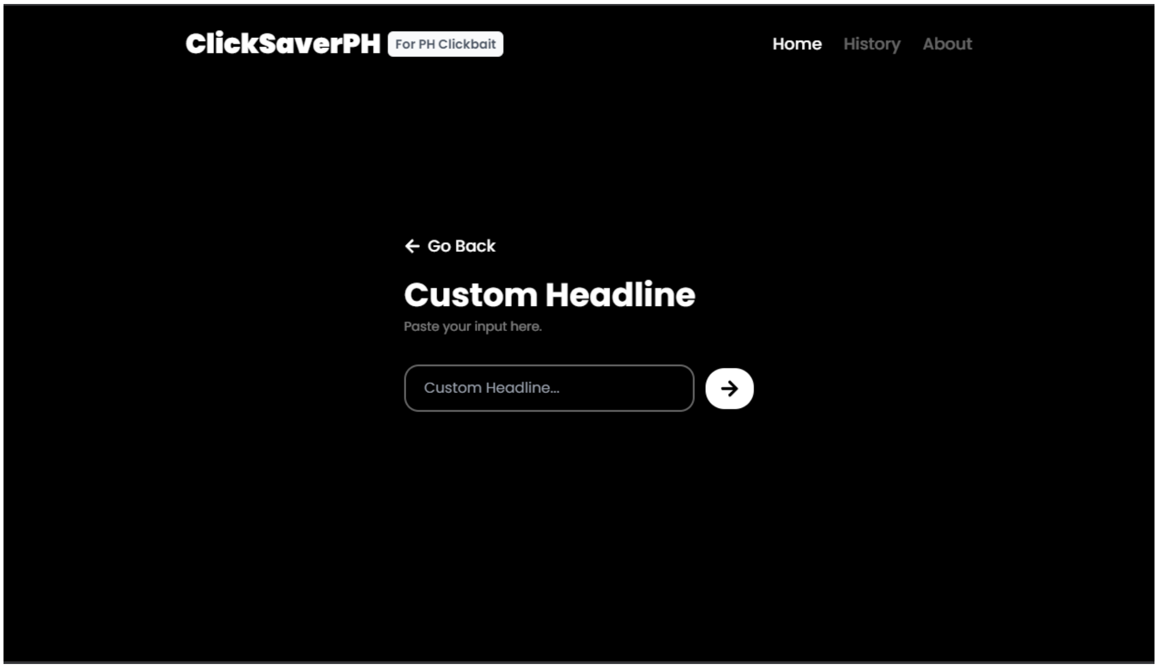


Figure 20. Input Form (No Input)

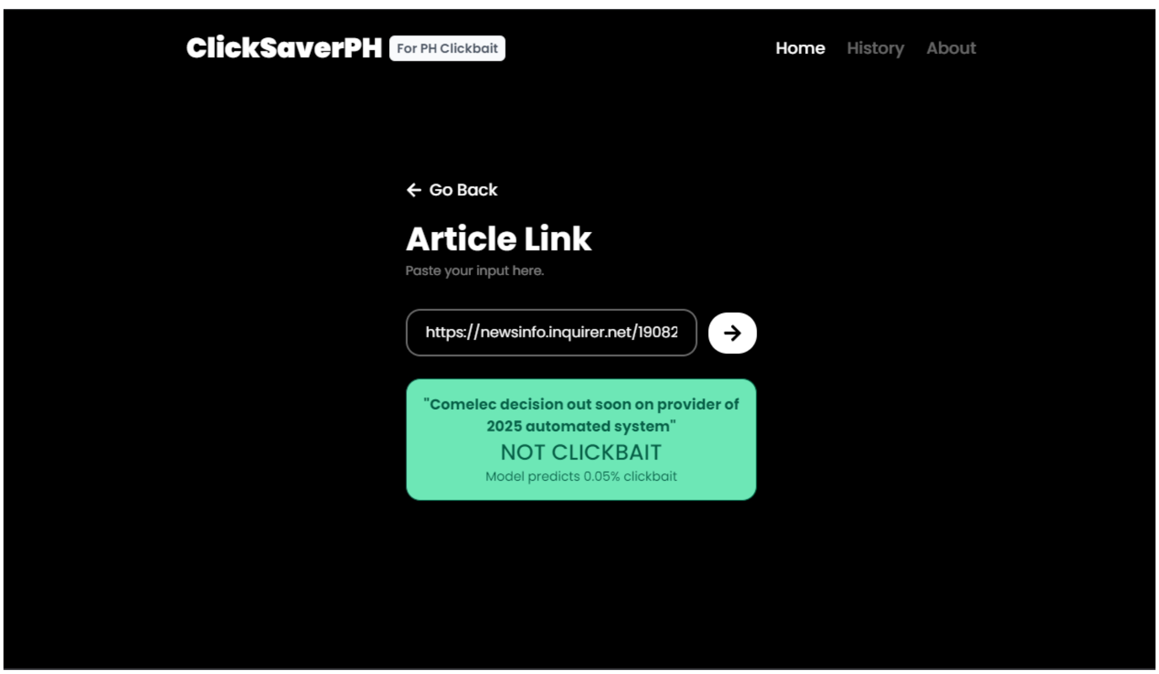


Figure 21. Input Form (Non-clickbait Output)

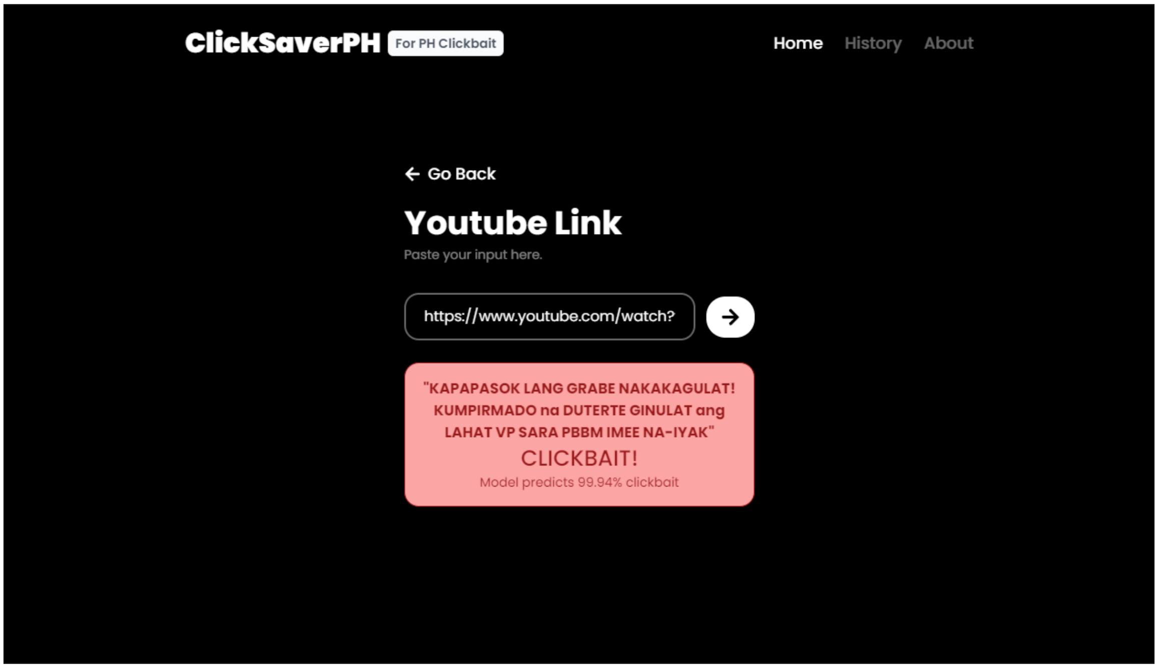


Figure 22. Input Form (Clickbait Output)

Figure 20 shows the form for users to input after selecting one of the options in the Index page, with this figure specifically showcasing the form for inputting a headline. Users can input a news headline of their choosing and then click on the arrow button to check the classification of the headline. The “YouTube Link” and “Article Link” options both have the same input form design but replaces the header with the name of the corresponding option, and should only have links as input. Figure 21 shows the output when the headline is not clickbait, displaying a green box signaling that the output is “Not Clickbait”, the title input, the classification, and the percentage of how much the model thinks that the title is clickbait. A lower percentage means that the headline is likely to be not clickbait, while a higher percentage means that the headline is likely to be clickbait. Figure 22 shows the output when the headline is clickbait, displaying a red box signaling that the output is ”Clickbait”, and the same kinds of information displayed in the previous figure.

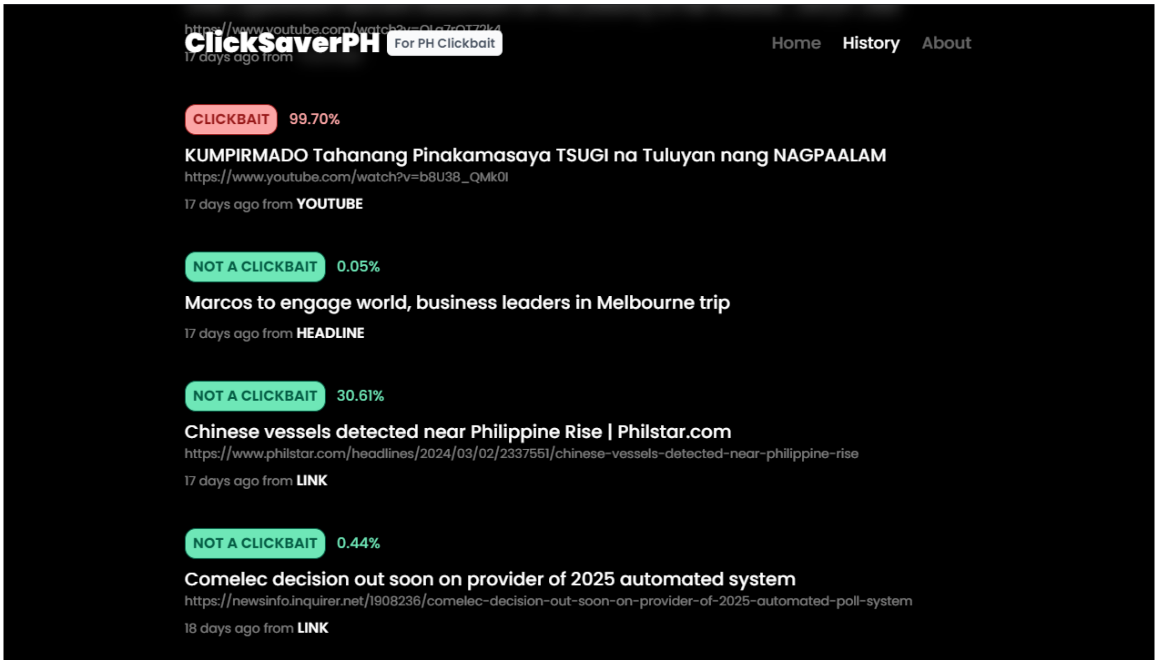


Figure 23. History Section

Figure 23 shows the "History" page of the web application. Here, users can view headlines that were previously evaluated by the model. The information displayed includes the headline, the classification, the percentage of how much the model thinks the headline is clickbait, the duration since the classification, the source method, and optionally, the article link if a link was entered.

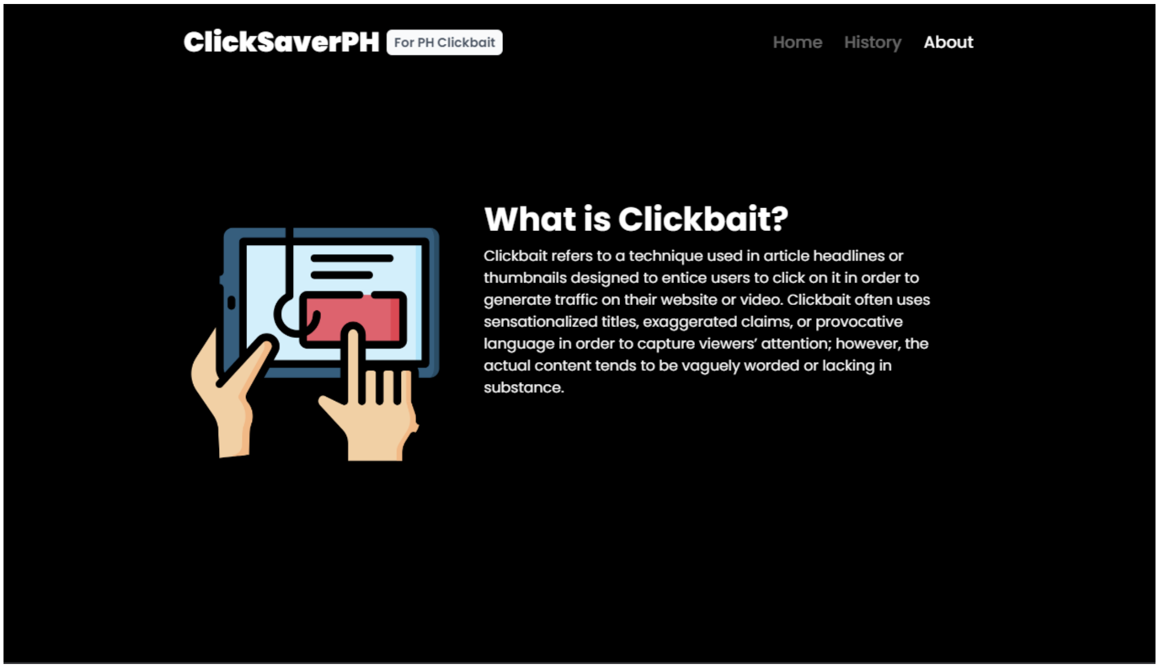


Figure 24. About Section

Figure 24 shows the About section of the web application. Here, the user is given a short explanation on what clickbait is.

4.5 Usability and Reliability Survey Results

Table 9. Usability Testing Results (by Sub-characteristics)

Characteristic	Sub-characteristic	Average Rating	Interpretation
Functionality	Suitability (F1)	4.57	SA
	Accuracy (F2)	4.04	A
Usability	Understandability (U1)	4.83	SA
	Learnability (U2)	4.78	SA
	Operability (U3)	4.78	SA
	Attractiveness (U4)	4.52	SA
Efficiency	Time Behaviour (E1)	4.83	SA

Table 10. Usability Testing Results (by Characteristics)

Characteristic	Average Rating	Interpretation
Functionality	4.31	SA
Usability	4.73	SA
Efficiency	4.83	SA

Table 11. Journalist Experts Ratings (by Sub-characteristics)

Characteristic	Sub-characteristic	Expert #1 Rating	Expert #2 Rating	Average Expert Rating	Interpre-tation
Functionality	Suitability (F1)	5	5	5	SA
	Accuracy (F2)	4	2	3	N
Usability	Understandability (U1)	4	5	4.5	SA
	Learnability (U2)	4	5	4.5	SA
	Operability (U3)	4	5	4.5	SA
	Attractiveness (U4)	3	5	4	A
Efficiency	Time Behaviour (E1)	5	5	4	A

Table 12. Journalist Experts Ratings (by Characteristics)

Characteristic	Expert #1 Rating	Expert #2 Rating	Average Rating	Interpretation
Functionality	4.5	3.5	4	A
Usability	3.75	5	4.38	SA
Efficiency	5	5	5	SA

In selecting the respondents for testing, several criteria were used. This demographic included the respondents, at least eighteen (18) years old and living in the Davao City region. Of thirty (30) respondents, eight (8) participants were students from various universities around Davao City, two (2) were faculty staff of Mapua Malayan Colleges Mindanao, with the remaining respondents having been career professionals, two (2) of which are professional journalists and editors-in-chief of their respective publication for their professional opinion on the reliability of the model.

Table 9 and 10 show the results of the survey conducted on the usability of the web application with a total of thirty (30) respondents. Survey results show that, overall, the web application was met with positive response by the respondents. Respondents found that the web application could effectively perform its designated functions, and that it was easy to learn and understand how to operate the application with minimal effort. Furthermore, the design was well received, and the web application was found to have responded quickly. However, while the model was generally accurate, some inputted headlines were incorrectly classified during the testing. These were generally headlines relating to international news and were not within the initial scope of the application. Furthermore, certain words were found to be incorrectly weighted towards a clickbait result such as “Russia” due to a bias caused by the political climate and general trends in clickbait news in the Philippines. Lastly, the model was found to be prone to misclassifying inputs unrelated to news content, which are outside of the scope of the model.

Table 11 and 12 display the results of the survey conducted on the reliability of the web application with the assistance of the professional opinion of two (2) journalists, both of whom hold the position of editor-in-chief within their respective publications. The survey indicates mixed results regarding the accuracy of the model and consistently higher scores regarding its efficiency and usability. The first journalist found the performance of the model to be satisfactory in identifying common clickbait trends. Further comments indicate concerns regarding the longevity of the app, a request to add additional fact checking functionality, and a more appealing design for the web application. The second journalist, however, found the model's reliability to be mixed, citing false positives found in inputting headlines originating from reliable news sources and government agencies. Upon further analysis, this had indicated the model had bias for longer article headlines toward a clickbait result. The relationship between character count in article headlines and clickbait trends was previously unaccounted for and may be used in further studies with larger datasets.

5. Conclusions and Recommendations

This study aimed to develop a clickbait detection model that would be able to classify if a Philippines news headline is clickbait or not clickbait in both English and Tagalog using a pre-trained MobileBERT model and a BiGRU model. Because existing datasets were last updated in 2019, a new updated dataset consisting of clickbait and non-clickbait news headlines in both English and Tagalog was generated for use in training, validating, and testing the clickbait detection model. In addition, a web application was developed and deployed for users to interact with the model. A total of 27,505 English non-clickbait, 6,321 English clickbait, 18,430 Tagalog non-clickbait, and 8,813 Tagalog clickbait news headlines were collected. To prevent bias when training the model, the overall number of titles was reduced from the total to 24,000 rows consisting of equal amounts of clickbait and non-clickbait news headlines in English and Tagalog. Test results show that the model has good classification ability with an accuracy of 97.88%. Survey results show that the web application was well received.

Recommendations for future studies include gathering a larger dataset for use in training the model. A larger dataset may be used to address certain clickbait trends in misclassified cases not accounted for in the model's initial training. Moreover, a more varied dataset may dissuade bias caused by politics-related clickbait news specific to the Philippines. Other BERT models may also be used such as Tagalog-BERT and RoBERTa Tagalog, or the training of an original English-Tagalog BERT model for better Tagalog clickbait detection. Furthermore, the BERT models may be fine-tuned for text classification as an alternative to using BiGRU. The development of a browser extension or plugin rather than a web application may also be considered. Lastly, better hardware may also allow for larger datasets and more expensive algorithms to be used in training the model.

References

Ahmad, I., Alqarni, M. A., Almazroi, A. A., & Tariq, A. (2020). Experimental evaluation of clickbait detection using machine learning models. *Intelligent Automation and Soft Computing*, *26*(6), 1335–1344. https://doi.org/10.32604/iasc.2020.013861

Bunquin, J. B., & Gaw, F. (2022). Digital Public Pulse: Mapping the digital landscape of Philippine politics. *Philippine Communication Society Review*, 181–194.

Chen, Y., Conroy, N. J., & Rubin, V. L. (2015). Misleading online content: Recognizing clickbait as "false news." *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection*, 15–19. https://doi.org/10.1145/2823465.2823467

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. http://arxiv.org/abs/1406.1078

Chua, B. B., & Dyson, L. E. (2004). Applying the ISO 9126 model to the evaluation of an e-learning system. In R. Atkinson, C. McBeath, D. Jonas-Dwyer, & R. Phillips (Eds.), *Beyond the comfort zone: Proceedings of the 21st ASCILITE Conference* (pp. 184–190). https://ascilite.org/conferences/perth04/procs/pdf/chua.pdf

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. http://arxiv.org/abs/1412.3555

Cruz, J. C. B., Tan, J. A., & Cheng, C. (2019). *Localization of Fake News Detection via Multitask Transfer Learning*. https://doi.org/10.13140/RG.2.2.23028.40322

Daoud, D. M., & El-Seoud, M. S. A. (2019). An effective approach for clickbait detection based on supervised machine learning technique. *International Journal of Online & Biomedical Engineering*, *15*(3), 21–32. https://doi.org/10.3991/ijoe.v15i03.9843

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.

Dimpas, P. K., Po, R. V., & Sabellano, M. J. (2017). Filipino and english clickbait detection using a long short term memory recurrent neural network. In *2017 International Conference on Asian Language Processing (IALP)* (pp. 276–280). IEEE. https://doi.org/10.1109/IALP.2017.8300597

Fakhruzzaman, M. N., Jannah, S. Z., Ningrum, R. A., & Fahmiyah, I. (2021). *Clickbait Headline Detection in Indonesian News Sites using Multilingual Bidirectional Encoder Representations from Transformers (M-BERT)*. http://arxiv.org/abs/2102.01497

Fernandez, A. C. T., & Devaraj, M. (2019). Computing the Linguistic-Based Cues of Fake News in the Philippines Towards its Detection. In *Proceedings of the 9th International Conference on Web Intelligence, Mining and Semantics* (pp. 1–9). ACM. https://doi.org/10.1145/3326467.3326490

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471. https://doi.org/10.1162/089976600300015015

Gothankar, R., Di Troia, F., & Stamp, M. (2021). Clickbait Detection in YouTube Videos. In *Artificial Intelligence for Cybersecurity* (pp. 261–284). Cham: Springer International Publishing. http://arxiv.org/abs/2107.12791

Grundy, A. (2022, June 7). *Service Annual Survey Shows Continuing Decline in Print Publishing Revenue*. United States Census Bureau.

https://www.census.gov/library/stories/2022/06/internet-crushes-traditional-media.html

Harjule, P., Gurjar, A., Seth, H., & Thakur, P. (2020). Text Classification on Twitter Data. *2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE)*, 160–164. https://doi.org/10.1109/ICETCE48199.2020.9091774

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., & Liu, Q. (2019). *TinyBERT: Distilling BERT for Natural Language Understanding*. http://arxiv.org/abs/1909.10351

Kemp, S. (2023, February 8). *Digital 2023: The Philippines - datareportal – global digital insights*. DataReportal. https://datareportal.com/reports/digital-2023-philippines

Klairith, P., & Tanachutiwat, S. (2018). Thai Clickbait Detection Algorithms Using Natural Language Processing with Machine Learning Techniques. *2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST)*, 1–4. https://doi.org/10.1109/ICEAST.2018.8434447

Kostadinov, Simeon. (2018). *Recurrent Neural Networks with Python Quick Start Guide : Sequential Learning and Language Modeling with TensorFlow.* Packt Publishing Ltd.

Li, Z., Zhou, L., Yang, X., Jia, H., Li, W., & Zhang, J. (2023). User Sentiment Analysis of COVID-19 via Adversarial Training Based on the BERT-FGM-BiGRU Model. *Systems*, *11*(3), 129. https://doi.org/10.3390/systems11030129

Liu, Y., Lu, J., Yang, J., & Mao, F. (2020). Sentiment analysis for e-commerce product reviews by deep learning model of Bert-BiGRU-Softmax. *Mathematical Biosciences and Engineering*, *17*(6), 7819–7837. https://doi.org/10.3934/MBE.2020398

Mahtab, M., Haque, M., & Hasan, M. (2022). *BanglaBait: Using Transformers, Neural Networks & Statistical Classifiers to detect clickbaits in New Bangla Clickbait Dataset* [Brac University]. http://hdl.handle.net/10361/17128

Mnih, V., Heess, N., Graves, A., & Kavukcuoglu, K. (2014). Recurrent Models of Visual Attention. *Advances in Neural Information Processing Systems*, *27*. http://arxiv.org/abs/1406.6247

Nayak, P. (2019, October 25). *Understanding searches better than ever before*. The Keyword. https://blog.google/products/search/search-language-understanding-bert/

Newman, N. (2022, June 15). *Overview and key findings of the 2022 Digital News Report*. Reuters Institute for the Study of Journalism. https://reutersinstitute.politics.ox.ac.uk/digital-news-report/2022/dnr-executive-summary

Pandey, S., & Kaur, G. (2018). Curious to Click It?-Identifying Clickbait using Deep Learning and Evolutionary Algorithm. *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 1481–1487. https://doi.org/10.1109/ICACCI.2018.8554873

Rogers, A., Kovaleva, O., & Rumshisky, A. (2020). A Primer in BERTology: What We Know About How BERT Works. *Transactions of the Association for Computational Linguistics*, *8*, 842–866. https://doi.org/10.1162/tacl_a_00349

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. http://arxiv.org/abs/1910.01108

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681. https://doi.org/10.1109/78.650093

Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y., & Zhou, D. [ICLR 2020 Conference Paper498 Authors]. (2023, May 5). *Response to all the reviewers* [Comment on the online forum post *MobileBERT: Task-Agnostic Compression of BERT by Progressive Knowledge Transfer*]. OpenReview. https://openreview.net/forum?id=SJxjVaNKwB&noteId=S1gxqk_7jH

Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y., & Zhou, D. (2020). *MobileBERT: a Compact Task-Agnostic BERT for Resource-Limited Devices*. http://arxiv.org/abs/2004.02984

Teoh, F. (2021, December 2). *Inaccurate clickbait headline in Forbes article used to promote the false claim that COVID-19 vaccines change our DNA*. Health Feedback. https://healthfeedback.org/claimreview/inaccurate-clickbait-headline-forbes-article-used-to-promote-false-claim-that-covid-19-vaccines-change-our-dna/

Tuquero, L. (2021, September 22). *Red flag for 2022: Political lies go unchecked on YouTube showbiz channels*. Rappler.

https://www.rappler.com/nation/elections/political-lies-unchecked-youtube-showbiz-channels-red-flag-candidates-2022/

Valenti, S., Cucchiarelli, A., & Panti, M. (2002). Computer Based Assessment Systems Evaluation via the ISO9126 Quality Model. *Journal of Information Technology Education: Research*, *1*, 157–175. https://doi.org/10.28945/353

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need. *Advances in Neural Information Processing Systems*, *30*.

VERA Files. (2023a, August 15). *VERA FILES FACT CHECK: Martial law NOT declared after Chinese water cannon attack*. VERA Files. https://verafiles.org/articles/vera-files-fact-check-martial-law-not-declared-after-chinese-water-cannon-attack

VERA Files. (2023b, November 24). *VERA FILES FACT CHECK: China did NOT invade Ilocos*. VERA Files. https://verafiles.org/articles/vera-files-fact-check-china-did-not-invade-ilocos

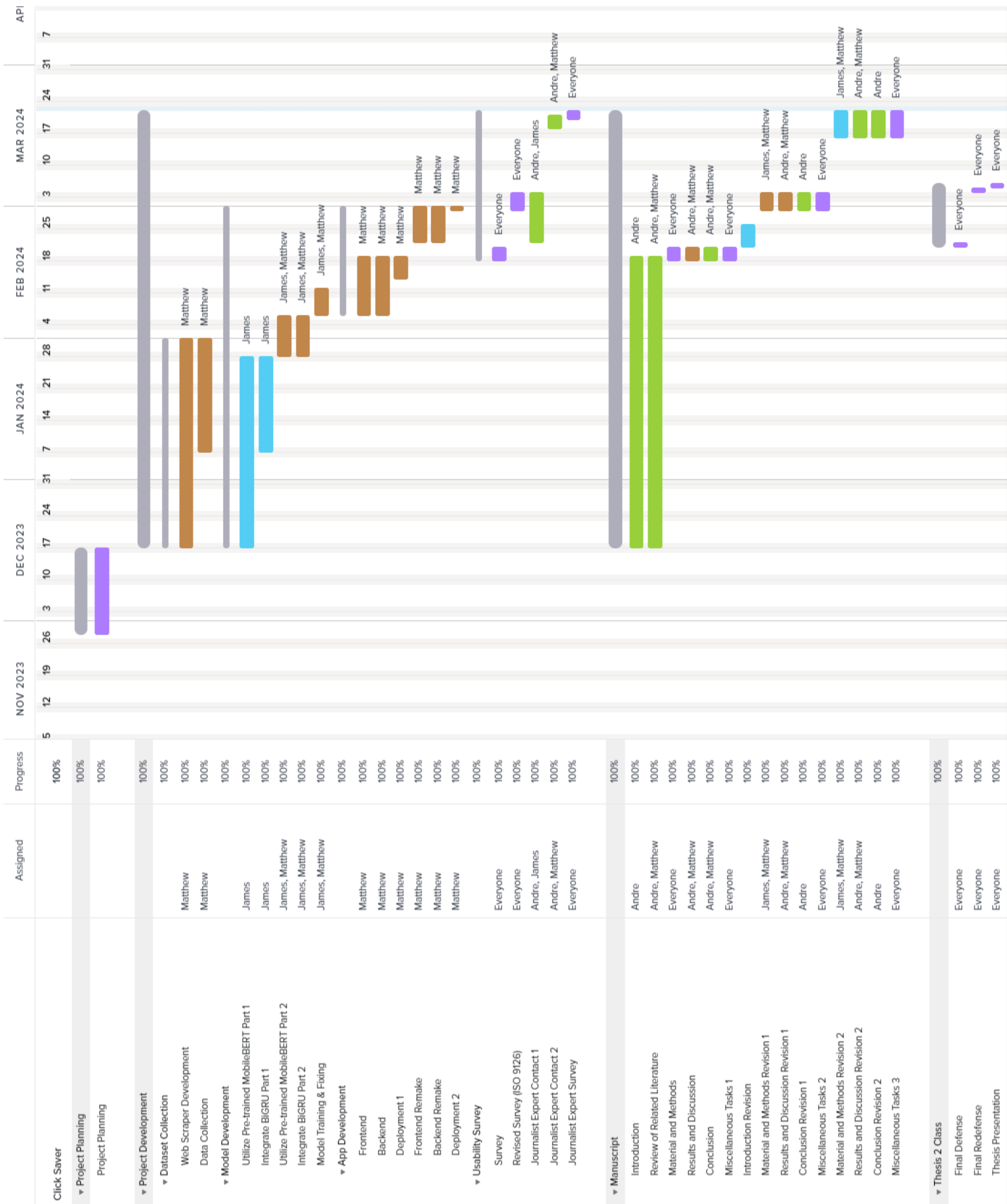
Wang, G., Bernanda, D. Y., Andry, J. F., Nurul Fajar, A., & Sfenrianto. (2019). Application Development and Testing Based on ISO 9126 Framework. *Journal of Physics: Conference Series*, *1235*(1), 012011. https://doi.org/10.1088/1742-6596/1235/1/012011

Yu, Q., Wang, Z., & Jiang, K. (2021). Research on Text Classification Based on BERT-BiGRU Model. In *Journal of Physics: Conference Series* (Vol. 1746, Issue 1, p. 012019). IOP Publishing Ltd. https://doi.org/10.1088/1742-6596/1746/1/012019

Zhang, Y., Li, X., Yang, Y., & Wang, T. (2022). Disease- and Drug-Related Knowledge Extraction for Health Management from Online Health Communities Based on BERT-BiGRU-ATT. *International Journal of Environmental Research and Public Health*, *19*(24), 16590. https://doi.org/10.3390/ijerph192416590

Zulqarnain, M., Ghazali, R., Ghouse, M. G., & Mushtaq, M. F. (2019). Efficient processing of GRU based on word embedding for text classification. *JOIV : International Journal on Informatics Visualization*, *3*(4), 377–383. https://doi.org/10.30630/joiv.3.4.289

Appendix A: Gantt Chart



Appendix B: User Acceptability Test

Rating	Description
5	Strongly Agree
4	Agree
3	Neutral
2	Disagree
1	Strongly Disagree

Criteria	Description	5	4	3	2	1
Functionality						
Suitability	The web app can effectively perform its designated functions.					
Accuracy	The results of the app are as anticipated.					
Usability						
Understandability	It is easy to comprehend how to use the app.					
Learnability	It is easy to learn how to use the app.					
Operability	The app can be used with minimal effort.					
Attractiveness	The layout and design of the app looks good and is visually appealing.					
Efficiency						
Time Behaviour	The app responds quickly.					

Appendix C: Letter of Request (Survey)

College of Computing and Information Science (CCIS)
Malayan Colleges Mindanao

March 1, 2024

Good day!

We are 4th-year students in Mapua Malayan Colleges Mindanao under the Bachelor of Science in Computer Science program.

In accordance with our capstone project, we have proposed and are developing Click Saver PH, a system aiming to combat misinformation by allowing users to detect clickbait in titles and headlines for Filipino and English news in the Philippines.

We are asking if you could take some time out of your day to assist us in testing the application and rating its performance in the provided form. Your input will ensure the accuracy and practical applicability of our project findings.

All responses gathered will be strictly confidential and your participation is greatly appreciated.

Respectfully Yours,

JAMES PAUL N. ABID

MATTHEW SZENEL S. ANG

ANDRE DEUCE B. TAN

Noted by:

JEANIE R. DELOS ARCOS
Thesis Adviser

Appendix D: Letter of Request (Journalist Expert #1)

College of Computing and Information Science (CCIS)
Malayan Colleges Mindanao

March 2, 2024

Dear Mr. Garcia,

We hope this message finds you well. We are BS Computer Science students from Mapúa Malayan Colleges Mindanao to seek your expertise in validating our new clickbait detection web application. Our application, Click Saver PH, aims to combat misinformation by helping users identify clickbait headlines accurately. Before the application is finalized, we're keen to ensure its effectiveness and reliability through validation tests.

We believe your insights would be invaluable in this process. Your expertise as a journalist could greatly enhance the refinement of our application.

Your contribution would not only improve our application but also advance our shared goal of promoting digital literacy. Should you have any questions or require further information, please do not hesitate to contact us at +63 995 172 3678 or at msAng@mcm.edu.ph.

We thank you for your consideration with our request.

Warm regards,

JAMES PAUL N. ABID

MATTHEW SZENEL S. ANG

ANDRE DEUCE B. TAN

Noted by:

JEANIE R. DELOS ARCOS
Thesis Adviser

Appendix E: Letter of Request (Journalist Expert #2)

College of Computing and Information Science (CCIS)
Malayan Colleges Mindanao

March 19, 2024

Dear Ms. Alivio,

We hope this message finds you well. We are BS Computer Science students from Mapúa Malayan Colleges Mindanao to seek your expertise in validating our new clickbait detection web application. Our application aims to combat misinformation by helping users identify clickbait headlines accurately. Before the application is finalized, we're keen to ensure its effectiveness and reliability through validation tests.

We believe your insights would be invaluable in this process. Your expertise as a journalist could greatly enhance the refinement of our application.

Your contribution would not only improve our application but also advance our shared goal of promoting digital literacy. Should you have any questions or require further information, please do not hesitate to contact us at +63 995 172 3678 or at msAng@mcm.edu.ph.

We thank you for your consideration with our request.

Warm Regards,

JAMES PAUL N. ABID

MATTHEW SZENEL S. ANG

ANDRE DEUCE B. TAN

Noted by:

JEANIE R. DELOS ARCOS
Thesis Adviser