# SENTIMENT ANALYSIS OF FINANCIAL STOCK MARKET NEWS USING PRE-TRAINED LANGUAGE MODELS

Widana Arachchi Sachith Kaushalya

(209343D)

Thesis/Dissertation submitted in fulfillment of the requirements for the degree Master of Science in Computer Science

Department of Computer Science

University of Moratuwa

Sri Lanka

July 2022

## DECLARATION

I declare that this is my own work and this thesis/dissertation2 does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis/dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date:

The above candidate has carried out research for the Master's Dissertation under my supervision.

Name of the Supervisor: Dr. Surangika Ranathunga

Signature of the Supervisor: Date:

## ABSTRACT

Sentiment analysis helps data analysts to find public opinion, actual meaning of the given text (positive meaning, neutral meaning or negative meaning) conduct market research, monitor brand and product reputation, and understand customer experiences of newly introduced items or service.

Stock news sentiment analysis is a useful task in the financial domain. However, this is different from the customer feedback for a product or brand, movie review and customer support reviews. This huge difference is because of the domain specific language in stock markets and lack of labeled data. This research implements a stock news sentiment analysis system using the latest transformer-based pre-trained language models in NLP. I could get higher sentiment classification results for the transformer-based pre-trained language models than the traditional classifications models in this research. Also I could reduce classification result bias for the particular stock market specific words, because of the transfer learning method. And I could introduce correlation between stock news sentiment and stock price change percentage value. This proposed model can predict the percentage change value of the stock when received a news.



**Additional key words and phrases**: Sentiment analysis, Deep learning, Language transformer models, Transfer learning

# ACKNOWLEDGMENTS

Throughout the writing of this dissertation I have received a great deal of support and assistance.

I would first like to thank my supervisor, Dr Surangika Ranathunga, whose expertise was invaluable in formulating the research questions and methodology. Your insightful feedback pushed me to sharpen my thinking and brought my work to a higher level.

I would like to extend my thanks towards DirectFN for providing me with all-important data to work on my research. These results and insights would not be possible without the valuable data you provided free of charge.

**TABLE OF CONTENTS**

**LIST OF FIGURES**

## LIST OF TABLES

## LIST OF ABBREVIATIONS

NLP = Natural Language Processing

BERT = Bidirectional Encoder Representations from Transformers

FinBERT = BERT model pre-trained on financial communication text

BART = Denoising Sequence-to-Sequence Pre-training for Natural Language model

RoBERTa = Robustly optimized Bidirectional Encoder Representations from Transformers

ALEART = A Light BERT for Supervised Learning

RNN = Recurrent Neural Network

CNN = Convolutional Neural Networks

# 1. INTRODUCTION

## 1.1. Background

### 1.1.1. Stock Market

Stock market gathers all of the dealers and traders who want to buy specific stock shares or who want to sell previously bought specific stock shares. All of the transactions are monitored and done with high transparency. All the transactions (buy or sell) have to be made through the stock market registered brokerage. Those brokerages play the role as an intermediate layer between end clients and the stock market.

Few years back traditional stock markets used to share paper cards to make deals with buyers and sellers. But the modern day's stock markets operate digitally. Many mobile applications, web sites, and desktop applications expose this market to buyers and sellers in the world.

### 1.1.2. Stock Market News

Stock market news is information about specific stock or whole market behavior such as market trends, company financial power and information about share dividend, company ownership movements, current financial states, company losses, company raisings, up trend of the company. These news are most important to the investors. According to the stock news, they can decide to either buy more stocks of the indicated company or sell their already bought stocks of the indicated company to another trader. Below three highlighted news are sample of stock market news in Saudi stock market.

***"Saudi Vitrified Clay Pipe Co. announces to Invites its Shareholders to Attend the Fourteenth Ordinary General Assembly Meeting (First Meeting) through modern technology means"***

***"Leejam Sports Company (Fitness Time) Announces Latest Development Regarding the Change in Ownership of the Company's Capital"***

***"Addendum Announcement from BinDawood Holding Co. in regards to dismissal of patent infringement case brought against its subsidiary Danube Co. for Foodstuffs & Commodities"***

Figure1, Figure 2 and Figure 3 indicate how stock news can change crude oil prices. Also Figure 4 and Figure 5 indicate how twitter and tesla stock price change against stock news. In those figures most of the price turning points have at least one news item. According to these figures, most probably news can decide stock price trends.


Figure 1 Crude oil price change and news 1[13]


Figure 2 Crude oil price change and news 2 [13]


Figure 3 Crude oil price change and news 3 [13]


Figure 4 Twitter price change and stock news [13]


Figure 5 Tesla price change and stock news [13]

### 1.1.3. Stock Price

Stock price means face value or price limit for a single share which is issued by a private company or any bonds. All of them have to be listed under a specific stock market. This value can increase and decrease according to different indicators and other financial formulas. That will directly affect the investors profit and company profit.

### 1.1.4. Previous Day Close Price

Previous close is generated using the previous day's final stock close price when the stock market officially closes. This value is referred to by investors to plot a chart data gap which can show the difference between previous day close prices and today's market trade price. A stock's closing value represents stock gain or loss for the day until executing the next trade. Most stock market news highlight stock price changes based on the difference from a security's market open to the market close.

### 1.1.5. Change Percentage

Percentage change represents how the current stock trade price has changed than the previous day`s close price.

Change = (current stock price - previous day close price)

Change % = (Change / previous day close price) * 100

## 1.2. Research Problem

All past stock market news has a direct or indirect relationship to the next stock price. Therefore negative sentiment or positive sentiment or neutral sentiment of a stock news is highly important for stock market traders to make profitable decisions.

Existing research for sentiment analysis of stock market news considers the stock market news sentiment impact for the index level but not for the stock level [1]. As an example assume there are more than 100 symbols listed in the stock market under a single index. One of those symbols has received positive news. Then only the stock price of that single stock will be increased. If we measure index price change after that news received output result will not be accurate and easily measurable. Because if the index price change contribution of that stock is very low then it can be neglected. Therefore the impact of the news can be unimportant. But that news changed the symbol stock price considerably. Therefore one has to analyze the price change which belongs to the news at the symbol level.

Also, how to handle stock market specific words in sentiment calculation is another concern. For example, "Sukuk" is an Arabic bond name. Most of the Middle East stock market news has included that word. But other stock market news, such as Sri Lanka, India, New York or London do not include that word.

## 1.3.Research objectives

Main objective of this research is to build an efficient financial news sentiment analysis system using the latest transformer based pre-trained language models, predict the real-time stock market news sentiment using this model. Newly proposed model should not affect domain specific words and phrases to the final sentiment calculation.

## 1.4.Thesis Structure

Section 2 presents a review of stock market news sentiment calculation models used in earlier research. Section 3 presents a brief description of the proposed stock market news sentiment calculation model task. Also section 4 presents proposed model evaluation results and analysis and section 5 and section 6 present conclusion of the research and what are main contribution of this research.

## 2. RELATED WORK

### 2.1. Overview

Stock market news sentiment analysis has relatively little literature up to now. Sometimes that may be due to a lack of domain knowledge in the stock market finance industry. However, following methods have been tested on sentiment analysis for stock market news,

1. Dictionary based sentiment calculation models.
2. Traditional machine learning models.
3. Deep learning models.
4. Transformer-based pre-trained Language models.

Transformer-based pre-trained Language models are the latest techniques in the NLP sentiment analysis techniques out of the above mentioned models. Also they give better performance and better accuracy prediction than others.

### 2.2. Dictionary based methods

In this dictionary based method [2], a reliable source of news data has to be selected as the inputs. After that, they have applied those selected news data to text preprocessing methods. After the text preprocessing, n-grams was applied. Then those processed data are compared with the words dictionary as the next step, which contains domain adapted words and phrases and their target sentiment class. As the final step, the results of the comparison for specific news articles are stored with their relevant sentiment value. After that these calculated score values are monitored against relevant stock prices. Based on the sentiment value and decision is taken either to sell or hold or buy.

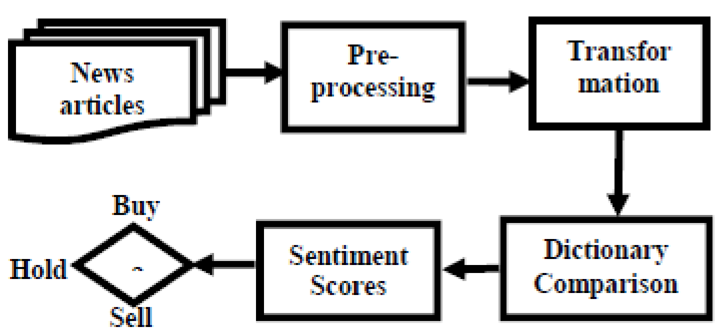


Figure 6 Dictionary-based news sentiment model

### 2.3. Deep Learning models

Deep Learning models make more accurate decisions without help from human interaction. In this task, Deep Learning models use a combination of multiple neural network layers. Each layer output result is an input for the next layer. Some of the deep learning methods are RNN, CNN and attention mechanisms with neural network. Word and sentence encoders are used to represent words and sentence vectors instead of original text as the input of the neural network.

Each layer of the deep learning model [4] extracts text patterns and relationships of the semantic information from the stock news. They have fed S&P500 companies stock news data and price data to the neural network. After the model has computed linear combination of the tf-idf weights with the word2vec representation of words in stock news. Those filtered output data are inputs of the next convolutional neural network.

### 2.4.Transformer-based pre-trained Language models.

The core idea behind these transformer models is training language models using very large corpora. Then downstream models are initialized using the weights which are learned from the language modeling task. As a result of that, the final result can achieve much better performance.

Most of the research has focused on using BERT based language models. They are BERT [5], RoBERTa [5], ALBERT [5], sXLNet [5], the FinBERT [5], [6]. In this proposed architecture they have applied NLP preprocessing methods. Then they have applied text encoding methods to that preprocessed data set. Then those encoded data fed to the BERT based language models. They have used the financial phrase-bank [8] data set which has manually annotated English stock news data. Also they have used SemEval-2017 data set [9]. Which contains financial news statements and headlines gathered from publically available dataset.

### 2.5.Zero shot text classification

Try to classify the given input text with given classification class labels. But all of the classification class and input text relation mappings does not clearly know about the previous data. First of all sentiment classification models have to learn a sentiment classifier using one set of class labels. Then have to evaluate using a different set of class labels where the sentiment classifier has not seen previously.

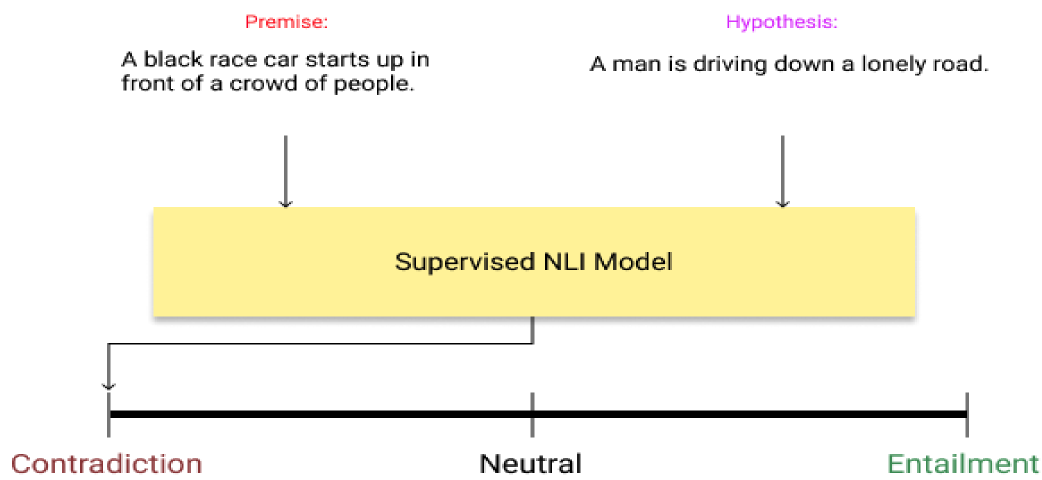


Figure 7 Supervised NLI System [14]

**2.6.Summary**

Pre-trained Language model implementations and deep learning models have shown better performance than other classifiers. Also they have self-learning mechanisms to understand features of the input data set. They consider the meaning of the whole context instead of single word meaning.

## 3. PROPOSED METHODOLOGY

### 3.1.Dataset

In this research we have selected the most active 20 symbols in the Saudi stock market during the past 10 years. This data set includes all stock market news. Also this data set includes symbol relevant stock price data which are tagged to the news. This proprietary dataset is given by DirectFN [10].

### 3.2.Proposed method to select correct polarity

Stock news and relevant stock price raw data are the inputs of the proposed model and correct news sentiment is the final result. According to the final result, the model can predict the price change percentage around the k value.

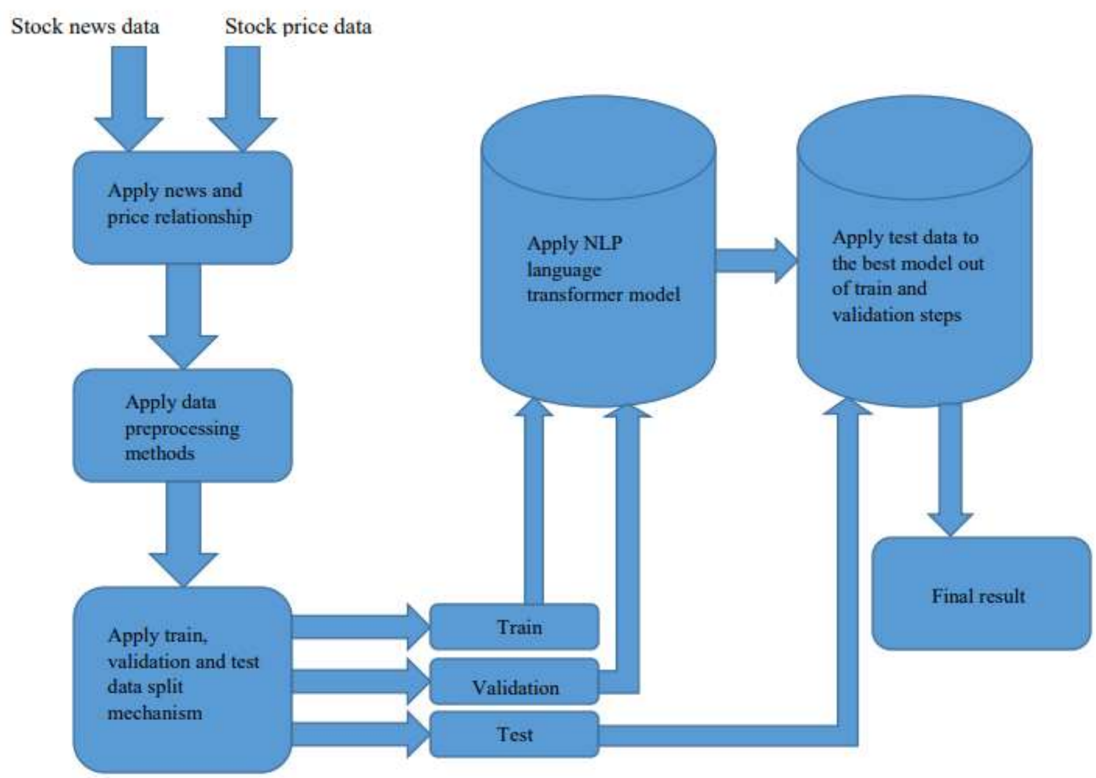


Figure 8 Complete proposed architecture

If the news sentiment is positive, the model can predict the stock price change percentage will increase more than the k value. If the news sentiment is negative, the model can predict stock price change percentage will increase more than k value to the negative side. If the news sentiment is neutral, the model can predict stock price change percentage will float between –k and k values.

In this research, we prepared Saudi stock market news dataset similar to the Financial Phrase-Bank dataset. To achieve that, this research has to follow the below steps.

1. Find the most active 20 symbols during the past 10 years from the database.
2. Then extract the stock market news from the database which are relevant to the given symbols with the news date. Now we have a dataset which has two data columns. They are news and the change percentage value of the symbol on a specific day which belongs to the news.
3. Then assign polarity for each news as shown below. k value (This will be described section 3.4.2) can vary from exchange to exchange. Because of the news content of the exchange.
    a. If ***change percentage* > *k***, that will be positive news.
    b. If ***change percentage* < -*k***, that will be negative news.
    c. If ***–k* < *change percentage* < *k*** that will be neutral news.

### 3.3.Evaluation models

Then we applied text encoding using different transformer models to extract valuable features from the texts. Following models are experienced with:

1. BART
2. BERT - Base
3. BERT - Large
4. XLNet - Base
5. XLNet - Large
6. RoBERT – Base
7. RoBERT - Large
8. ALBERT - Base
9. ALBERT - Large
10. finBERT

### 3.4. Implementation

#### 3.4.1. Zig Zag Indicator


Figure 9 Zig-Zag Indicator chart [16]

The zigzag indicator is a tool that analysts use to find out whether a security's trend is an uptrend or downtrend. It helps to identify significant changes in stock price while filtering out short term fluctuations. Also this calculation eliminates noise of the everyday other market conditions.

When implementing the zigzag indicator, minimum price movements' percentage have to be declared. This price movement value can vary from 5% to 9%. Also the default price percentage value for the zigzag indicator is 5%. These smaller price changes can represent the bigger informational picture to investors.

### 3.4.2. Find best k value and stock news sentiment.

This k value is the decision maker value in price change after predicting the value. Also this k value decides whether the given stock news is negative, neutral or positive during the model train and validation steps.

#### 3.4.2.1.K value and news sentiment relationship

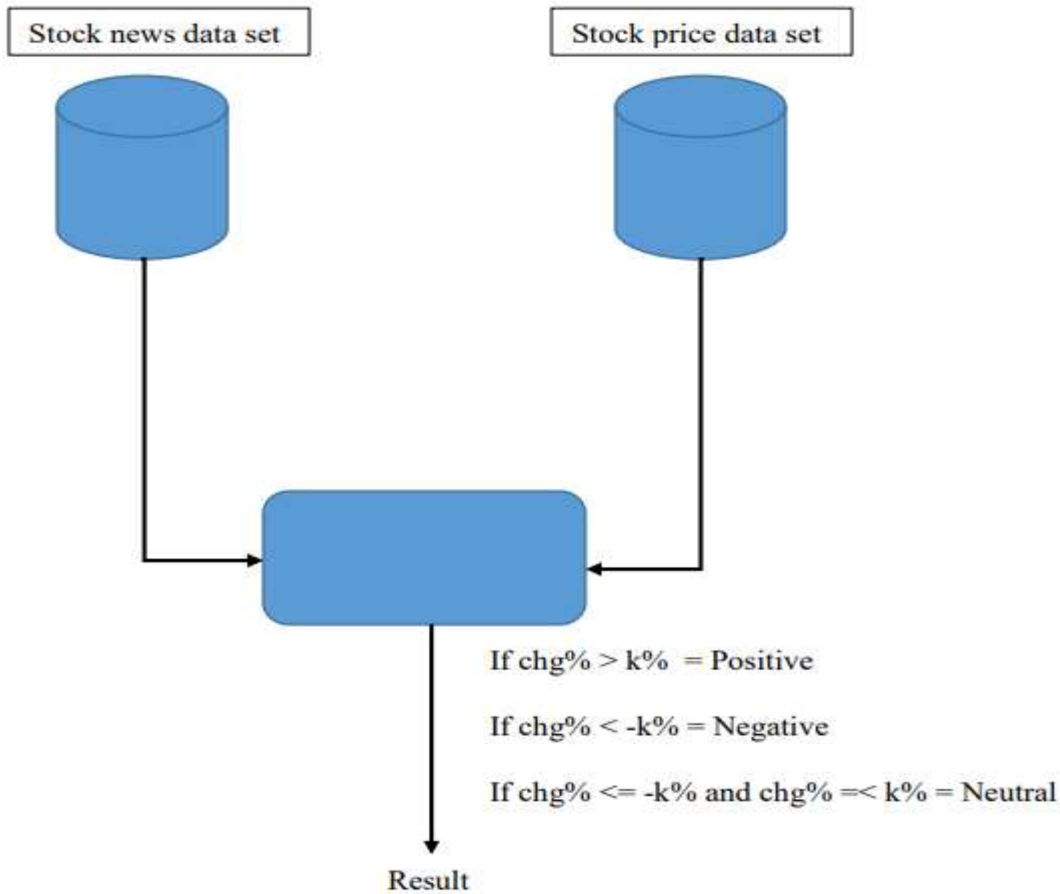


Figure 10 Mechanism of the stock sentiment calculation with stock price data

Number of data points for each class label (negative, neutral, positive) are changed against different k values. Below tables and charts indicate the number of data point's percentage over the k value. According to that logic, proprietary Saudi data set has to be divided into three parts. They are positive, negative and neutral.

Table 1 to Table 5 indicate a summary of negative, neutral and positive three classes against different k values.. The X axis represents the k value and Y axis represents the percentage of each class. Figure 8 to Figure 10 represent each class data points spread for train, validation and test data sets.

Table 1: Data points count and their percentage when K = 5 for final output classes

| Data set | Negative record count | Neutral record count | Positive record count | Total record count | Negative record count % | Neutral record count % | Positive record count % |
|---|---|---|---|---|---|---|---|
| Train | 303 | 1390 | 570 | 2263 | 13.389 | 61.422 | 25.187 |
| Validation | 62 | 674 | 97 | 833 | 7.442 | 80.912 | 11.644 |
| Test | 71 | 740 | 115 | 926 | 7.667 | 79.913 | 12.419 |
| | | | | | | | |
| Total | 436 | 2804 | 782 | 4022 | | | |

Table 2: Data points count and their percentage when K = 6 for final output classes

| Data set | Negative record count | Neutral record count | Positive record count | Total record count | Negative record count % | Neutral record count % | Positive record count % |
|---|---|---|---|---|---|---|---|
| Train | 278 | 1445 | 540 | 2263 | 12.284 | 63.853 | 23.862 |
| Validation | 46 | 710 | 77 | 833 | 5.522 | 85.234 | 9.243 |
| Test | 53 | 785 | 88 | 926 | 5.723 | 84.773 | 9.503 |
| | | | | | | | |
| Total | 377 | 2940 | 705 | 4022 | | | |

Table 3: Data points count and their percentage when K = 7 for final output classes

| Data set | Negative record count | Neutral record count | Positive record count | Total record count | Negative record count % | Neutral record count % | Positive record count % |
|---|---|---|---|---|---|---|---|
| Train | 205 | 1556 | 502 | 2263 | 9.058 | 68.758 | 22.182 |
| Validation | 34 | 739 | 60 | 833 | 4.081 | 88.715 | 7.202 |
| Test | 39 | 820 | 67 | 926 | 4.211 | 88.855 | 7.235 |
| | | | | | | | |
| Total | 278 | 3115 | 629 | 4022 | | | |

Table 4: Data points count and their percentage when K = 8 for final output classes

| Data set | Negative record count | Neutral record count | Positive record count | Total record count | Negative record count % | Neutral record count % | Positive record count % |
|---|---|---|---|---|---|---|---|
| Train | 133 | 1693 | 437 | 2263 | 5.877 | 74.812 | 19.310 |
| Validation | 22 | 770 | 41 | 833 | 2.614 | 92.436 | 4.921 |
| Test | 25 | 858 | 43 | 926 | 2.699 | 92.656 | 4.643 |
| | | | | | | | |
| Total | 180 | 3321 | 521 | 4022 | | | |

Table 5: Data points count and their percentage when K = 9 for final output classes

| Data set | Negative record count | Neutral record count | Positive record count | Total record count | Negative record count % | Neutral record count % | Positive record count % |
|---|---|---|---|---|---|---|---|
| Train | 86 | 1893 | 284 | 2263 | 3.800 | 83.650 | 12.549 |
| Validation | 14 | 793 | 26 | 833 | 1.680 | 95.198 | 3.121 |
| Test | 16 | 883 | 27 | 926 | 1.727 | 95.356 | 2.915 |
| | | | | | | | |
| Total | 116 | 3569 | 337 | 4022 | | | |

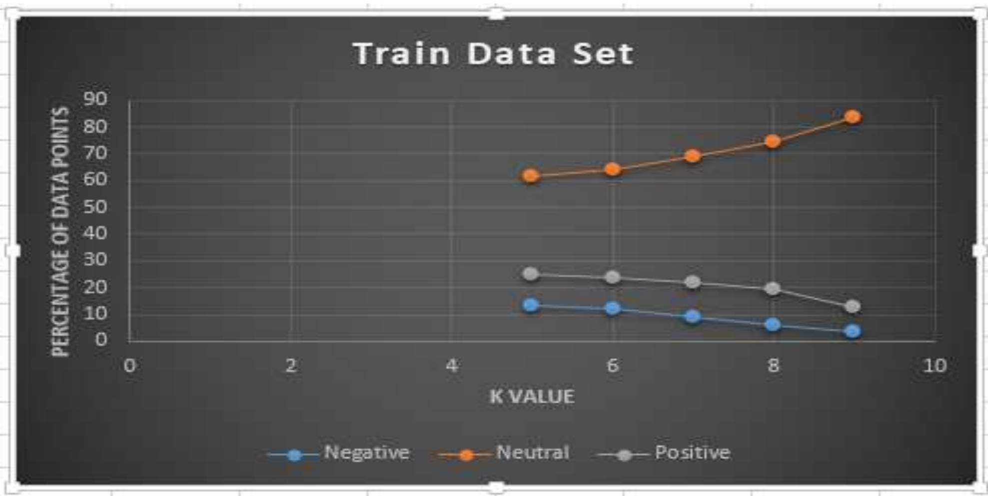


Figure 11 Number of data points in train data against different k values.

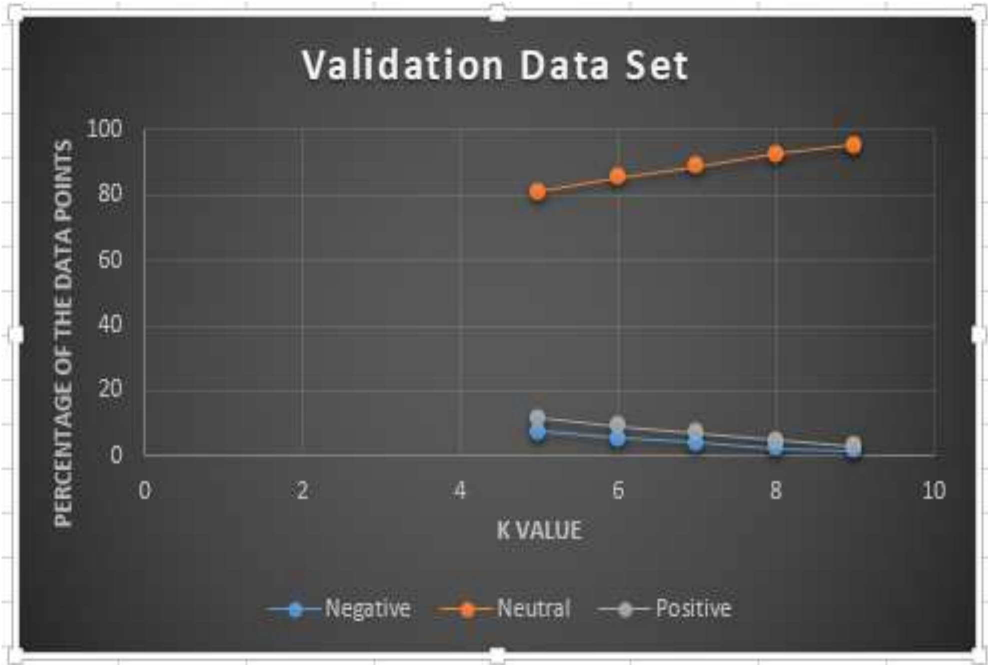


Figure 12 Number of data points in validation data against different k values.

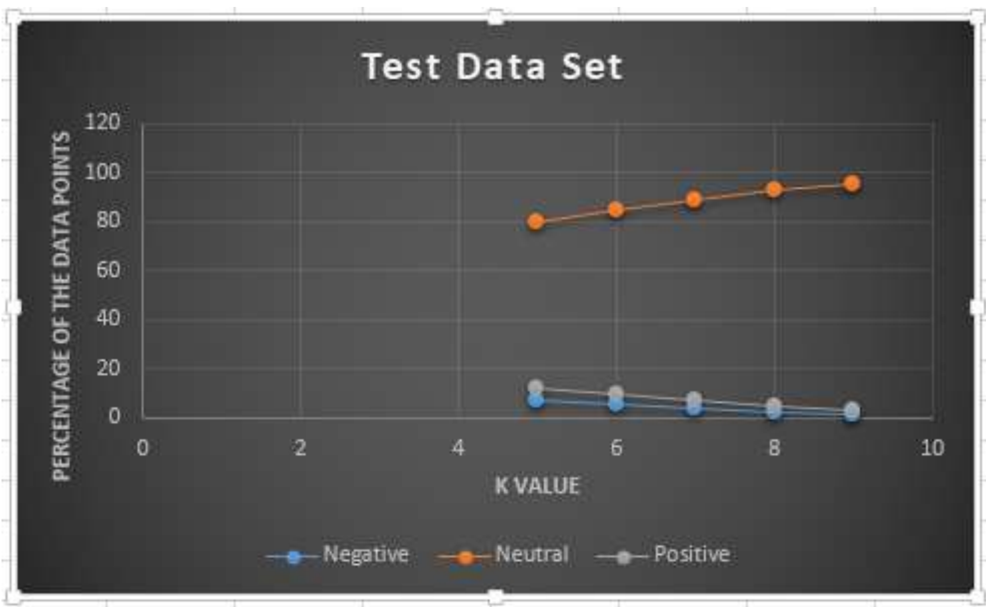


Figure 13 Number of data points in test data against different k values.

This research value of the change percentage as (k = 5%) for the sentiment calculation according to the Figure 8, Figure 9 and Figure 10 graphs data. This is because the final result got the maximum number of data points for positive news and negative news categories when k = 5%.

#### 3.4.2.2.K value and zero shot classification result relationship

Zero shot text classification result and sentiment calculated dataset accuracy values are compared with different k values. Figure 13 visualizes the relationship between zero shot classification results and proposed method. All results are in Table 6 and graphical representation is Figure 14.

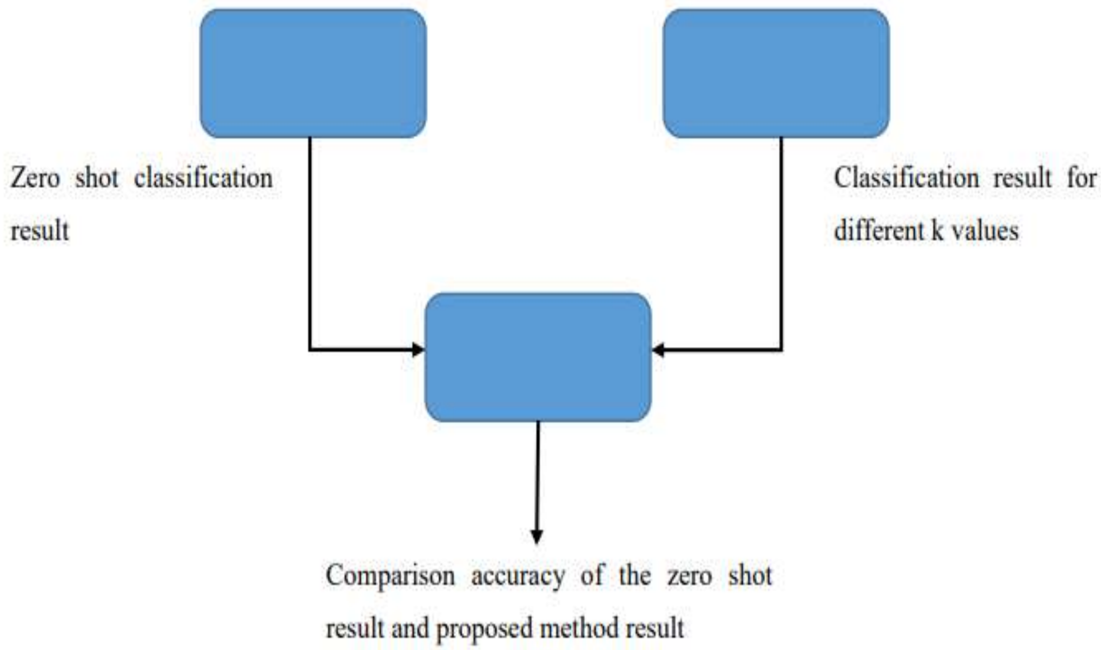


Figure 14 zero shot result and proposed method for result comparison.

According to the maximum data points for positive and negative classes, definition of the zigzag indicator and zero shot classification k = 5 is the most suitable margin for the news sentiment decision value.

Table 6: Result of zero shot classification for different k values

| K value | Accuracy | Recall | Precision | F1 Score |
|---|---|---|---|---|
| 5 | 0.815 | 0.815 | 0.8271 | 0.8210 |
| 6 | 0.7935 | 0.7935 | 0.799 | 0.7962 |
| 7 | 0.7858 | 0.7858 | 0.7738 | 0.7797 |
| 8 | 0.7525 | 0.7525 | 0.7554 | 0.7539 |
| 9 | 0.72 | 0.72 | 0.7345 | 0.7271 |

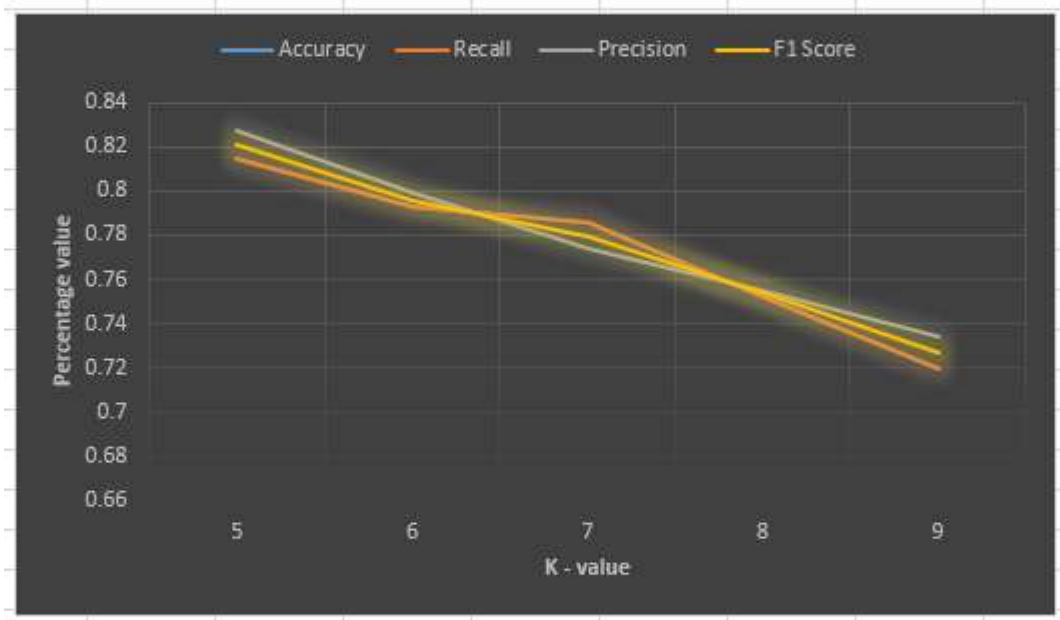


Figure 15 Zero shot classification result against different k values

# 4. RESULT AND ANALYSIS

Hyper parameter tuning and optimization is a main task for getting a better model. In this process when one model parameter is changed all other parameters were kept at a constant value and observed the behavior of the precision, recall and F1 score. Optimized parameters are learning rate, seed, evaluation steps and batch size.

## 4.1. ALBERT-base model optimization and evaluation

### 4.1.1. Training parameter optimization

Table 7 Accuracy values against learning rate for ALBERT-base

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.727737 | 0.813926 | 0.768422 |
| 0.0001 | 0.734636 | 0.693878 | 0.713676 |
| 0.001 | 0.736668 | 0.695078 | 0.715269 |

Table 8 Accuracy values against seed for ALBERT-base

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.785038 | 0.717887 | 0.749962 |
| 42 | 0.752828 | 0.666267 | 0.706908 |

Table 9 Accuracy values against evaluation steps for ALBERT-base

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.72773 | 0.8139 | 0.76840 |
| 1000 | 0.7346 | 0.69387 | 0.71365 |
| 1500 | 0.75282 | 0.66626 | 0.7069 |

Table 10 Accuracy values against batch size for ALBERT-base

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.73666 | 0.69507 | 0.71526 |
| 8 | 0.785 | 0.71788 | 0.749941 |
| 16 | 0.752828 | 0.666267 | 0.706906 |

### 4.1.2. Model evaluation

According to Table 7 to Table 10, we fine-tuned the ABERT-base model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.819
2. Recall : 0.819
3. Precision : 0.7345
4. F1 score : 0.7744

## 4.2. ALBERT-large model optimization and evaluation

### 4.2.1. Training parameter optimization

Table 11 Accuracy values against learning rate for ALBERT-large

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.6546 | 0.80912 | 0.72375 |
| 0.0001 | 0.6523 | 0.7953 | 0.7167 |
| 0.001 | 0.6244 | 0.7787 | 0.6930 |

Table 12 Accuracy values against seed for ALBERT-large

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.6546 | 0.80912 | 0.72375 |
| 42 | 0.6546 | 0.80912 | 0.72375 |

Table 13 Accuracy values against evaluation steps for ALBERT-large

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.6546 | 0.80912 | 0.72375 |
| 1000 | 0.6512 | 0.8051 | 0.7200 |
| 1500 | 0.6489 | 0.8037 | 0.7180 |

Table 14 Accuracy values against batch size for ALBERT-large

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.6498 | 0.80867 | 0.72058 |
| 8 | 0.6506 | 0.80912 | 0.72125 |
| 16 | 0.6485 | 0.80854 | 0.71973 |

### 4.2.2. Model evaluation

According to Table 11 to Table 14, we fine-tuned the ABERT-large model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8061
2. Recall : 0.8061
3. Precision : 0.6497
4. F1 score : 0.7195

## 4.3. BART-base model training, optimization and evaluation

### 4.3.1. Training parameter optimization

Table 15 Accuracy values against learning rate for BART-base

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.79836 | 0.69867 | 0.745196 |
| 0.0001 | 0.79721 | 0.68907 | 0.739206 |
| 0.001 | 0.79150 | 0.68187 | 0.732606 |

Table 16 Accuracy values against seed for BART-base

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.79836 | 0.69867 | 0.745196 |
| 42 | 0.79721 | 0.68907 | 0.739206 |

Table 17 Accuracy values against evaluation steps for BART-base

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.79150 | 0.68187 | 0.7326 |
| 1000 | 0.78894 | 0.66986 | 0.72454 |
| 1500 | 0.78093 | 0.66986 | 0.72114 |

Table 18 Accuracy values against batch size for BART-base

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.780940 | 0.669868 | 0.72115 |
| 8 | 0.788948 | 0.669868 | 0.72454 |
| 16 | 0.78093 | 0.66986 | 0.72114 |

### 4.3.2. Model evaluation

According to Table 15 to Table 18, we fine-tuned the BART-base model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8633
2. Recall : 0.8633
3. Precision : 0.8491
4. F1 score : 0.8561

## 4.4. BART-large model training, optimization and evaluation

### 4.4.1. Training parameter optimization

Table 19 Accuracy values against learning rate for BART-large

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.7714 | 0.76830 | 0.769847 |
| 0.0001 | 0.7652 | 0.7521 | 0.7585 |
| 0.001 | 0.7289 | 0.71699 | 0.72289 |

Table 20 Accuracy values against seed for BART-base

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.76885 | 0.74189 | 0.755129 |
| 42 | 0.75833 | 0.73572 | 0.74685 |

Table 21 Accuracy values against evaluation steps for BART-large

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.77912 | 0.73109 | 0.75434 |
| 1000 | 0.77687 | 0.72893 | 0.75213 |
| 1500 | 0.76888 | 0.71251 | 0.73962 |

Table 22 Accuracy values against batch size for BART-large

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.78231 | 0.72389 | 0.75196 |

| | | | |
|---|---|---|---|
| 8 | 0.79998 | 0.72897 | 0.76286 |
| 16 | 0.79023 | 0.7352 | 0.76172 |

#### 4.4.2. Model evaluation

According to Table 19 to Table 22, we fine-tuned the BART-large model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8061
2. Recall : 0.8061
3. Precision : 0.6497
4. F1 score : 0.7195

### 4.5. BERT-base model training, optimization and evaluation

#### 4.5.1. Training parameter optimization

Table 23 Accuracy values against learning rate for BERT-base

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.798214 | 0.732293 | 0.763834 |
| 0.0001 | 0.79221 | 0.731 | 0.760375 |
| 0.001 | 0.78904 | 0.72457 | 0.755432 |

Table 24 Accuracy values against seed for BERT-base

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.793498 | 0.720288 | 0.755123 |
| 42 | 0.8012 | 0.7347 | 0.76651 |

Table 25 Accuracy values against evaluation steps for BERT-base

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.800418 | 0.709484 | 0.752213 |
| 1000 | 0.81423 | 0.71247 | 0.759959 |
| 1500 | 0.7987 | 0.69874 | 0.745384 |

Table 26 Accuracy values against batch size for BERT-base

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.786503 | 0.703481 | 0.742679 |
| 8 | 0.78634 | 0.69842 | 0.739777 |
| 16 | 0.76247 | 0.67542 | 0.71631 |

### 4.5.2. Model evaluation

According to Table 23 to Table 26, we fine-tuned the BERT-base model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8358
2. Recall : 0.8358
3. Precision : 0.7738
4. F1 score : 0.8036

## 4.6. BERT-large model training, optimization and evaluation

### 4.6.1. Training parameter optimization

Table 27 Accuracy values against learning rate for BERT-large

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.6546 | 0.80912 | 0.723703922 |
| 0.0001 | 0.6523 | 0.7953 | 0.71673693 |
| 0.001 | 0.6244 | 0.7787 | 0.693065754 |

Table 28 Accuracy values against seed for BERT-large

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.6546 | 0.80912 | 0.723703922 |
| 42 | 0.6546 | 0.80912 | 0.723703922 |

Table 29 Accuracy values against evaluation steps for BERT-large

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.6546 | 0.80912 | 0.72375 |
| 1000 | 0.6512 | 0.8051 | 0.72 |
| 1500 | 0.6489 | 0.8037 | 0.718 |

Table 30 Accuracy values against batch size for BERT-large

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.6498 | 0.80867 | 0.72058 |
| 8 | 0.6506 | 0.80912 | 0.72125 |
| 16 | 0.6485 | 0.80854 | 0.71973 |

### 4.6.2. Model evaluation

According to Table 27 to Table 30, we fine-tuned the BERT-large model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8061
2. Recall : 0.8061
3. Precision : 0.6497
4. F1 score : 0.7195

## 4.7. FinBERT model training, optimization and evaluation

### 4.7.1. Training parameter optimization

Table 31 Accuracy values against learning rate for FinBERT

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.78969 | 0.704682 | 0.744768142 |
| 0.0001 | 0.7862 | 0.69840 | 0.739704 |
| 0.001 | 0.77984 | 0.68421 | 0.728902 |

Table 32 Accuracy values against seed for FinBERT

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.788225 | 0.690276 | 0.736006 |
| 42 | 0.77478 | 0.98470 | 0.867217 |

Table 33 Accuracy values against evaluation steps for FinBERT

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.792982 | 0.699880 | 0.743528 |
| 1000 | 0.78572 | 0.69410 | 0.737074 |
| 1500 | 0.76127 | 0.67452 | 0.715274 |

Table 34 Accuracy values against batch size for FinBERT

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.785039 | 0.703481 | 0.742026 |
| 8 | 0.77987 | 0.70230 | 0.739055 |
| 16 | 0.74689 | 0.98741 | 0.85042 |

#### 4.7.2. Model evaluation

According to Table 31 to Table 34, we fine-tuned the FinBERT model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8547
2. Recall : 0.8547
3. Precision : 0.8364
4. F1 score : 0.8454

### 4.8. RoBERTa-base model training, optimization and evaluation

#### 4.8.1. Training parameter optimization

Table 35 Accuracy values against learning rate for RoBERTa-base

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.72988 | 0.7719 | 0.7503 |
| 0.0001 | 0.72234 | 0.7507 | 0.7362 |
| 0.001 | 0.69875 | 0.7462 | 0.7216 |

Table 36 Accuracy values against seed for RoBERTa-base

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.75671 | 0.7012 | 0.7278 |
| 42 | 0.74049 | 0.6830 | 0.7105 |

Table 37 Accuracy values against evaluation steps for RoBERTa-base

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.7210 | 0.7442 | 0.7324 |
| 1000 | 0.7024 | 0.7261 | 0.714 |
| 1500 | 0.6924 | 0.7035 | 0.6979 |

Table 38 Accuracy values against batch size for RoBERTa-base

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.74049 | 0.68307 | 0.7106 |
| 8 | 0.75671 | 0.69457 | 0.7243 |
| 16 | 0.72678 | 0.67524 | 0.7 |

### 4.8.2. Model evaluation

According to Table 35 to Table 38, we fine-tuned the RoBERT-base model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8325
2. Recall : 0.8325
3. Precision : 0.7554
4. F1 score : 0.7920

## 4.9. RoBERTa-large model training, optimization and evaluation

### 4.9.1. Training parameter optimization

Table 39 Accuracy values against learning rate for RoBERTa-large

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.65468 | 0.80912 | 0.72375 |
| 0.0001 | 0.64375 | 0.78654 | 0.70801 |
| 0.001 | 0.62847 | 0.76254 | 0.68904 |

Table 40 Accuracy values against seed for RoBERTa-large

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.65468 | 0.80912 | 0.72375 |
| 42 | 0.64781 | 0.78531 | 0.70996 |

Table 41 Accuracy values against evaluation steps for RoBERTa-large

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.65468 | 0.80912 | 0.72375 |
| 1000 | 0.65468 | 0.80912 | 0.72375 |
| 1500 | 0.65468 | 0.80912 | 0.72375 |

Table 42 Accuracy values against batch size for RoBERTa-large

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.65468 | 0.80912 | 0.72375 |
| 8 | 0.65468 | 0.80912 | 0.72375 |
| 16 | 0.63789 | 0.78423 | 0.70353 |

### 4.9.2. Model evaluation

According to Table 39 to Table 42, we fine-tuned the RoBERTa-large model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8061
2. Recall : 0.8061
3. Precision : 0.6497
4. F1 score : 0.7195

## 4.10. XLNet-base model training, optimization and evaluation

### 4.10.1. Training parameter optimization

Table 43 Accuracy values against learning rate for XLNet-base

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.77739 | 0.69267 | 0.73258 |
| 0.0001 | 0.75078 | 0.68420 | 0.71594 |
| 0.001 | 0.71457 | 0.65412 | 0.68301 |

Table 44 Accuracy values against seed for XLNet-base

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.75641 | 0.64893 | 0.69856 |
| 42 | 0.77360 | 0.67947 | 0.72348 |

Table 45 Accuracy values against evaluation steps for XLNet-base

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.74460 | 0.72869 | 0.73655 |
| 1000 | 0.73894 | 0.71354 | 0.72601 |
| 1500 | 0.70352 | 0.69745 | 0.70047 |

Table 46 Accuracy values against batch size for XLNet-base

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.75652 | 0.67226 | 0.7119 |
| 8 | 0.74237 | 0.65438 | 0.6956 |
| 16 | 0.70354 | 0.62478 | 0.6618 |

### 4.10.2. Model evaluation

According to Table 43 to Table 43, we fine-tuned the XLNet-base model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8061
2. Recall : 0.8061
3. Precision : 0.6497
4. F1 score : 0.7195

## 4.11. XLNet-large model training, optimization and evaluation

### 4.11.1. Training parameter optimization

Table 47 Accuracy values against learning rate for XLNet-large

| Learning Rate | Precision | Recall | F1 |
|---|---|---|---|
| 0.00005 | 0.65468 | 0.80912 | 0.72375 |
| 0.0001 | 0.65468 | 0.80912 | 0.72375 |
| 0.001 | 0.65468 | 0.80912 | 0.72375 |

Table 48 Accuracy values against seed for XLNet-large

| Seed | Precision | Recall | F1 |
|---|---|---|---|
| 0 | 0.65468 | 0.80912 | 0.72375 |

| | | | |
|---|---|---|---|
| 42 | 0.65468 | 0.80912 | 0.72375 |

Table 49 Accuracy values against evaluation steps for XLNet-large

| Evaluation Steps | Precision | Recall | F1 |
|---|---|---|---|
| 500 | 0.65468 | 0.80912 | 0.72375 |
| 1000 | 0.65468 | 0.80912 | 0.72375 |
| 1500 | 0.65468 | 0.80912 | 0.72375 |

Table 50 Accuracy values against batch size for XLNet-large

| Batch size | Precision | Recall | F1 |
|---|---|---|---|
| 4 | 0.65468 | 0.80912 | 0.72375 |
| 8 | 0.65468 | 0.80912 | 0.72375 |
| 16 | 0.65468 | 0.80912 | 0.72375 |

### 4.11.2. Model evaluation

According to Table 47 to Table 50, we fine-tuned the XLNet-large model with high F1 values. Using those optimized parameters, the model is trained. Then we evaluate the model with test data. Final evaluation results are,

1. Accuracy : 0.8061
2. Recall : 0.8061
3. Precision : 0.6497
4. F1 score : 0.7195

When considering all the transformer model final evaluations, all of the base models performed better than the large models of them. XLNet - base and XLNet - large models gave the same accuracy metric values. Which were very close and equal to the large models of the other evaluated transformer models accuracy metrics values. All the final models test data evaluation results are in Table 51. All of them are sorted from high F1 Score to low F1 Score.

Table 51 Final model evaluation accuracy values

| Transformer Model | Accuracy | Recall | Precision | F1 Score |
|---|---|---|---|---|
| BART - base | 0.8633 | 0.8633 | 0.8491 | 0.85614 |
| FinBERT | 0.8547 | 0.8547 | 0.8364 | 0.84545 |
| BERT - base | 0.8358 | 0.8358 | 0.7738 | 0.80360 |

| | | | | |
|---|---|---|---|---|
| RoBERTa - base | 0.8325 | 0.8325 | 0.7554 | 0.79207 |
| ALBERT - base | 0.819 | 0.819 | 0.7345 | 0.77445 |
| BERT - large | 0.8061 | 0.8061 | 0.6497 | 0.71949 |
| ALBERT - large | 0.8061 | 0.8061 | 0.6497 | 0.71949 |
| RoBERTa - large | 0.8061 | 0.8061 | 0.6497 | 0.71949 |
| XLNet - base | 0.8061 | 0.8061 | 0.6497 | 0.71949 |
| XLNet - large | 0.8061 | 0.8061 | 0.6497 | 0.71949 |
| BART - large | 0.8061 | 0.8061 | 0.6497 | 0.71949 |

The large model has a lot of encoder and decoder layers than the base model. So, it was over-fitting when training the data set. But the base model has few encoder and decoder layers than the large model. The base model got only important features than the base models when input data classification.

Also in the same way I trained all of these models using financial phrase bank data set and evaluated using my Saudi data set. All the results are related for the parameter optimized models.

Table 52 Final evaluation results for financial phrase bank data trained data.

| Transformer Model | Accuracy | Recall | Precision | F1 Score |
|---|---|---|---|---|
| BART - base | 0.6857 | 0.6857 | 0.7899 | 0.7175 |
| FinBERT | 0.6706 | 0.6706 | 0.788 | 0.7075 |
| BERT - base | 0.7127 | 0.7127 | 0.8036 | 0.7465 |
| RoBERTa - base | 0.7678 | 0.7678 | 0.733 | 0.7484 |
| ALBERT - base | 0.8153 | 0.8153 | 0.7329 | 0.7667 |
| BERT - large | 0.7991 | 0.7991 | 0.6386 | 0.7099 |
| ALBERT - large | 0.7991 | 0.7991 | 0.6386 | 0.7099 |
| RoBERTa - large | 0.7991 | 0.7991 | 0.6386 | 0.7099 |
| XLNet - base | 0.6663 | 0.6663 | 0.7533 | 0.6862 |
| XLNet - large | 0.7991 | 0.7991 | 0.6386 | 0.7099 |
| BART - large | 0.7819 | 0.7819 | 0.789 | 0.7853 |

According to the Table 52 data highest accuracy and F1 score values belong to the ALBERT-base model and lowest values belong to the XLNet-base model. All the large models have nearly same accuracy and F1 score values. But all of them are lower than the values of the Table 51 data.

Above two evaluation scenarios we can prove this proposed model can extend to other financial data sets also.

BART is a sequence to sequence machine translation architecture. Well-developed BART models can works either generate new text using the comprehend text or text classification. But above both evaluated scenarios, BART models perform well other than the BERT and BERT derivative models.

First evaluation scenario used transfer learning architecture. But second evaluation scenario did not use transfer learning architecture. Finally, we can compare transfer learning applied evaluation result and non-applied evaluation result.

# 5. CONCLUSION

Using the final evaluation results this research could prove points which are mentioned in the research objective section.

First one is handle stock market specific words in sentiment calculation. That can indicate from Table 51 data has higher accuracy value than the Table 52 data. Because Table 51 data gather after the apply transfer learning architecture. This research trained and evaluate all the models using same stock market data. But in Table 52 output data gather using two data set. According to the final evaluation results we can prove first conclusion point.

The second conclusion point is introducing a new sentiment calculation formula, using stock time series data and news. All the data in Table 6 indicate zero shot classification result against to different k values. Most of the zero shot news sentiment classification results and newly proposed formula generated sentiment calculations are very close to each other for different k values. Then we can get a conclusion, there is a direct connection between stock price change percentage value and the news sentiment.

# 6. CONTRIBUTION

This research has introduced a new formula to find polarity of given stock news using the time series data of the given stock instead of manually annotation.

Also, this new formula focused to reduce the impact and bias of the stock market specific words for the polarity calculation.

## 7. FUTURE WORK

This model will work with other financial ratios and technical score values. Also, we will extend this model, which can assign sentiment score weight for the stock market news and announcements. We will extend this model for predict change percentage of a stock according to the stock news sentiment. And we will develop this model using twitter financial news feed and other financial news data feeds other than the direct stock exchange data feed.

## 8. REFERENCES


[1] Ieeexplore.ieee.org. 2022. Stock Market Prediction Analysis by Incorporating Social and News Opinion and Sentiment. [online] Available at: <https://ieeexplore.ieee.org/document/8637365/authors#authors> [Accessed 4 January 2022].

[2] Ieeexplore.ieee.org. 2022. Predicting the Effects of News Sentiments on the Stock Market. [online] Available at: <https://ieeexplore.ieee.org/document/8621884/authors#authors> [Accessed 6 January 2022].

[3] Ieeexplore.ieee.org. 2022. Forecasting Stock Market Movement Direction Using Sentiment Analysis and Support Vector Machine. [online] Available at: <https://ieeexplore.ieee.org/document/8326522> [Accessed 8 January 2022].

[4] Ieeexplore.ieee.org. 2022. Stock Price Prediction Using News Sentiment Analysis. [online] Available at: <https://ieeexplore.ieee.org/document/8848203> [Accessed 12 January 2022].

[5] Ieeexplore.ieee.org. 2022. Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers. [online] Available at: <https://ieeexplore.ieee.org/document/9142175> [Accessed 18 January 2022].

[6] Araci, D., 2022. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. [online] arXiv.org. Available at: <https://arxiv.org/abs/1908.10063> [Accessed 28 January 2022].

[7] Devlin, J., Chang, M., Lee, K. and Toutanova, K., 2022. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. [online] arXiv.org. Available at: <https://arxiv.org/abs/1810.04805> [Accessed 3 February 2022].

[8] Huggingface.co. 2022. financial_phrasebank · Datasets at Hugging Face. [online] Available at: <https://huggingface.co/datasets/financial_phrasebank> [Accessed 2 February 2022].

[9] Alt.qcri.org. 2022. Download the Full Training Data for SemEval-2017 Task 4 < SemEval-2017 Task 4. [online] Available at: <https://alt.qcri.org/semeval2017/task4/?id=download-the-full-training-data-for-semeval-2017-task-4> [Accessed 3 February 2022].

[10] Directfn.com. 2022. [online] Available at: <https://www.directfn.com/en/> [Accessed 1 January 2022].

[11] Medium. 2022. Zero-Shot Text Classification & Evaluation. [online] Available at: <https://towardsdatascience.com/zero-shot-text-classification-evaluation-c7ba0f56688e> [Accessed 10 February 2022].

[12] Arxiv.org. 2022. [online] Available at: <https://arxiv.org/pdf/2202.01924.pdf> [Accessed 16 February 2022].

[13] Investopedia. 2022. Investopedia. [online] Available at: <https://www.investopedia.com/> [Accessed 20 February 2022].