Sentiment Analysis of Amazon Electronic Product Reviews using Deep Learning

by

Jemimah Ojima Abah

A dissertation submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Data Analytics

Dublin Business School

Supervisor: Dr. Shazia Afzal
Abstract

Due to increased usage of technology (social media, online marketing, internet services) over the years, Sentiment Analysis which is analysis done to understand the opinion expressed in a piece of text, has become a hot and trending topic in the world today. Sentiment Analysis opens the door to a plethora of intriguing applications in almost every possible domain, from politics to social media rant on trending topics, education, movies, product and service reviews. A thriving aspect of Sentiment Analysis is the sentiment polarity in customer reviews; companies receive criticism and commendation from customers to understand their views on different products and determine which products are more favourable than others. This project attempts to build two deep learning models: Convolutional Neural Network (CNN) & Long Short-Term Memory (LSTM) to automatically detect and classify sentiment polarity in Amazon Electronic review dataset. The raw text is processed into their respective word vector representation using GloVe Embeddings. Accuracy, Precision, Recall, and F1 Score are used to assess the selected models. Both Baseline models obtain 93% Accuracy. The results demonstrate that the models able to accurately classify the reviews.

Index Terms—sentiment analysis, sentiment polarity detection, deep learning, convolutional neural networks, long short-term memory, glove embeddings
Declaration

I, Jemimah Ojima Abah, declare that this research is completed by myself for the award of Master of Science in Data Analytics. This report is an original by me, no part of this report has been previously presented to any institution or published in any journal.

In addition, I would like to declare that the work upholds ethical academic standards. There is no information present that can harm anyone. To conclude, this work follows the academic honesty policy provided by Dublin Business School.

Signed: Jemimah Ojima Abah

Date: 24th May, 2021
Acknowledgements

First of all, I am grateful to my Family and friends for their love and support (financially, emotionally, spiritually, educationally) during the period of this research.

I would like to appreciate my project supervisor Dr. Shazia Afzal of Dublin Business School for her unwavering support throughout my dissertation. It was a privilege to work under her guidance, I could never ask for a better supervisor.

Finally, I am grateful to Dublin Business School for the provision of resources to carry out this project.
## Contents

### 1 Introduction
1.1 Background & Motivation ........................................... 1
1.2 Problem Statement .................................................. 2
1.3 Research Objectives ................................................ 2
1.4 Research Question .................................................... 3
1.5 Project Report Structure .......................................... 3

### 2 Literature Review
2.1 Introduction .......................................................... 4
2.2 Sentiment Analysis ................................................... 4
2.3 Application of Sentiment Analysis ................................. 5
2.4 History and Evolution of Sentiment Analysis ...................... 5
2.5 Techniques for Sentiment Analysis ................................. 6
   2.5.1 Lexicon Based Approach .................................... 7
   2.5.2 Machine Learning Approach ................................ 7
2.6 Sentiment Analysis using Machine Learning & Deep Learning .................................................. 9
   2.6.1 Traditional Machine Learning ............................... 9
   2.6.2 Artificial Neural Networks ................................ 10
   2.6.3 Convolutional Neural Networks ............................. 11
   2.6.4 Recurrent Neural Network ................................ 12
   2.6.5 Recursive Neural Networks ................................. 15
2.7 Challenges of Sentiment Analysis ................................. 16
   2.7.1 Domain/Language Specific ................................ 16
   2.7.2 Quality Data .................................................. 16
   2.7.3 Bias and Fairness ............................................. 16
2.8 Conclusion ........................................................... 17

### 3 Data Mining Methodology
3.1 Data Selection ....................................................... 19
3.2 Data Preparation & Preprocessing ................................. 19
   3.2.1 Exploratory Data Analysis ................................. 19
## 3.2.2 Data Cleaning & Feature Selection

---

## 3.3 Data Transformation

---

## 3.4 Data Mining

---

## 3.5 Design Specification

---

#### 3.5.1 Overview of Project Architecture

---

#### 3.5.2 Environment Settings

---

## 4 Experimentation

### 4.1 Data Preparation & Preprocessing

---

### 4.2 Data Transformation

---

### 4.3 Data Mining

---

### 4.4 Hyperparameter Tuning

---

## 5 Evaluations and Results

### 5.1 Experiment 1 - Baseline Models

---

### 5.2 Experiment 2 - Models with UpSampled Data

---

### 5.3 Experiment 3 - Hyperparameter Tuning of Baseline Models

---

### 5.4 Discussion

---

## 6 Conclusions and Future Work

### 6.1 Conclusion

---

### 6.2 Future Work

---

## References

---

## Appendix A: Data Preparation, Preprocessing & Transformation

### A.1 Import Libraries

---

### A.2 Reading Data

---

### A.3 Word Cloud

---

### A.4 Data Transformation

---

### A.5 UpSampling

---

### A.6 Text Processing

---

### A.7 PostgreSQL

---

## Appendix B: Experimentation - Models and Evaluation

### B.1 Baseline CNN Model

---

#### B.1.1 Model Definition

---

#### B.1.2 Training and Validation Accuracy & Loss

---

#### B.1.3 CNN Model Classification Report

---

### B.2 Baseline LSTM Model

---

#### B.2.1 Model Definition

---
B.2.2 Training and Validation Accuracy & Loss ........................................ 45
B.2.3 LSTM Model Classification Report ............................................. 46
List of Tables

3.1 System Configuration .................................................. 24
5.1 Baseline Model Results .................................................. 28
5.2 UpSampled Model Results .............................................. 29
5.3 Hyperparameter Tuning Results ...................................... 31
5.4 CNN Hyperparameter Tuning Settings ............................. 33
5.5 LSTM Hyperparameter Tuning Settings ............................ 33
# List of Figures

1. Overview of the Approaches to Sentiment Analysis
2. Knowledge Discovery in Databases Process Flow Diagram
3. Ratings Count Plot - The x-axis reveals the five classes of Ratings whilst the y-axis shows the count for each Rating class
4. Brands Pie Chart
5. Review Word Cloud
6. Target Variable
7. Project Architecture
8. Baseline CNN Model Confusion Matrix
9. Baseline LSTM Model Confusion Matrix
10. UpSample CNN Model Confusion Matrix
11. UpSample LSTM Model Confusion Matrix
12. CNN Model Confusion Matrix - (Hyperparameter Tuning Experiment)
13. LSTM Model Confusion Matrix - (Hyperparameter Tuning Experiment)

A. Import Libraries
B. Reading Dataset
C. Word Cloud Code Snippet
D. Rating Transformation
E. UpSample Code Snippet
F. Text Cleaning Code Snippet
G. PostgreSQL 1 Code Snippet
H. PostgreSQL 2 Code Snippet
I. CNN Model Definition - Code Snippet
J. CNN Training and Validation Graphs: Accuracy/Loss are on the y-axis, the number of Epochs are on the x-axis
K. CNN Model Classification - Code Snippet
L. LSTM Model Definition - Code Snippet
M. LSTM Training and Validation Graphs: Accuracy/Loss are on the y-axis, the number of Epochs are on the x-axis
Abbreviations

**ABSA** Aspect-Based Sentiment Analysis.

**ACSA** Aspect-Category Sentiment Analysis.

**ANN** Artificial Neural Network.

**ATSA** Aspect-Term Sentiment Analysis.

**CNN** Convolutional Neural Network.

**DL** Deep Learning.

**ELU** Exponential Linear Unit.

**GloVe** Global Vectors for Word Representation.

**GRU** Gated Recurrent Unit.

**JSON** JavaScript Object Notation.

**KDD** Knowledge Discovery in Databases.

**LSTM** Long Short-Term Memory.

**MAE** Mean Absolute Error.

**MAPE** Mean Absolute Percentage Error.

**ML** Machine Learning.

**NLP** Natural Language Processing.

**PReLU** Parametric Rectified Linear Unit.

**ReLU** Rectified Linear Unit.
RMSE  Root Mean Square Error.

RNN  Recurrent Neural Network.

SA  Sentiment Analysis.

SVM  Support Vector Machines.

Tanh  Hyperbolic Tangent Activation Function.

TF-IDF  Term Frequency–Inverse Document Frequency.
Chapter 1

Introduction

1.1 Background & Motivation

Sentiment Analysis (SA) is the process of identifying and classifying a text, extracting opinions and information that a writer is trying to pass across. Opinion and sentiment have been a significant influence on human behaviour, beliefs, and perception of reality; to an extent, humans are conditioned to see the world in light of others they look up to.

Businesses and organisations continuously focus on customer satisfaction as it is a crucial determinant of company and organisational success. While some organisations are keen on increasing revenue and reducing cost, other organisations such as government and nonprofit organisations are interested in political and taxpayer support.

The following are ways sentiment analysis improve a business:

i Brand monitoring and Crisis Management: Sentiment analysis allow business owners to monitor customers sentiment about their brands and products; this helps business owners counter fake news and banter about products. It allows business owners to discover unfavourable content and opinion of customers and prevent such content from going viral.

ii Customer Services: Understanding customer sentiment helps highlight the weak and strong areas in the customer support system of a business based on their reviews. This analysis helps businesses determine how to satisfy customer better.

iii Marketing and Business Strategies: Understanding customer opinions from social media and other sources helps business owners provide an appropriate campaign that relates specifically to their target audience.
Increased Sale Revenue: When all the above are satisfied, customers feel satisfied and heard, increasing their desire to patronise the company which are interested in their satisfaction and wellbeing while introducing others (friends, family, colleagues).

1.2 Problem Statement

With the rapid switch of marketing from traditional to digital, which has led to a change in the way people shop, the importance of word of mouth cannot be turned down or ignored. Companies such as Amazon, Yelp provide environments for customers to provide constructive criticism on products and services they offer. Social media platforms such as Twitter, Facebook, Instagram also play a vital role in allowing customers to air their views about different products.

When analysed and utilised effectively, these reviews allow businesses to build customer trust and loyalty, boost sales, attract new customers, and drive revenue. The desire for businesses to grow and become better has triggered an interest in Sentiment Analysis, and many researchers and analysts have proposed different techniques in the quest of finding optimum solutions. Some of this researches have been discussed in Chapter 2 of this paper.

This paper focuses on understanding various Deep Learning Algorithms and how they can accurately predict customer sentiments and opinions. The literature review explains previous works done by other researchers in similar areas, giving a broader understanding of the research.

1.3 Research Objectives

The objectives of this research are to:

i Explore the history, evolution and techniques of sentiment analysis.

ii To develop various Deep Learning models to analyse the sentiment of customers using Techniques.

iii To analyse the accuracy of various Deep Learning models on the customer review sentiment analysis and effectively utilise Deep Learning models to grow the business.

iv To demonstrate the difficulties that arise when attempting to analyse sentiment using Deep Learning Techniques and how they can be addressed.
Apply Deep Learning techniques on Amazon Electronic Product Reviews and evaluate the models.

1.4 Research Question

This research aims to develop a Deep Learning model that best predicts customers' sentiment based on their reviews using the Amazon Review Dataset Amazon, 2015. This research answers the following questions as related to the objectives listed above.

i What are the suitable Deep Learning Algorithms for sentiment analysis of customer reviews?

ii Which Deep Learning Algorithm can most effectively be used to improve the business while satisfying customers?

iii What are the difficulties in Sentiment Analysis of Customer Reviews, and How can they be resolved?

1.5 Project Report Structure

The remainder of this research project is organized as follows:

- Chapter 2 (Literature Review) presents the state-of-the-art in the Sentiment Analysis research area.

- Chapter 3 (Data Mining Methodology) describes the data mining methodological approach utilized to achieve the research objectives and the Design Specification which gives the reader a brief overview of the project architecture and experiment settings.

- Chapter 4 (Experimentation) discusses the steps used to execute the project.

- Chapter 5 (Evaluations and Results) details the results and evaluation metrics used in the experiments

- Chapter 6 (Conclusion) concludes the project and highlights ways this project can be extended.
2.1 Introduction

This chapter provides a narrative review of the research. It is divided into Eight (8) sections, the sections are; Sentiment Analysis, Application of Sentiment Analysis, History and Evolution of Sentiment Analysis, Sentiment Analysis Techniques, Sentiment Analysis using Deep Learning, Challenges of Sentiment Analysis, and finally a Conclusion.

2.2 Sentiment Analysis

The sentiment is a term that describes “a view or opinion that is held or expressed” by someone. Opinions are central to almost all human activities and have a significant impact on our behaviour. Our beliefs and perceptions of reality and the decisions we make are heavily influenced by how others see and evaluate the world. As a result, when we need to make a decision, we frequently seek others’ advice. (Farhadloo and Rolland, 2016)

“Sentiment analysis is a series of methods, techniques, and tools about detecting and extracting subjective information, such as opinion and attitudes, from language” (B. Lui, 2009). It is commonly used to extract emotions, sentiments and summarisation from big data, and analysts can use this information to make some predictions (Gupta, R. Tiwari and Robert, 2016). The most critical aspect of sentiment analysis is analysing the body of text to find and understand the opinion expressed based on various factors such as mood, modality.

The sentiment that appears in texts comes in two flavours: explicit, where the subjective sentence directly expresses an opinion (“It is a beautiful day”), and implicit, where the text implies an opinion
(“The earphone broke in two days”) (Liu, 2012). Sentiment analysis is most effective when applied to subjective rather than objective context. When text is based on objective context, it is mainly made up of typical sentences that do not express any mood (Jaspreet, 2018).

2.3 Application of Sentiment Analysis

Sentiment analysis can be used in various areas ranging from individuals to large organisations and governments. Sentiment analysis is instrumental as it provides a comprehensive overview of the public opinion behind various topics, including product reviews, politics, movie reviews, and other aspects of everyday life. In education, sentiment analysis is used to identify students’ learning curve, foresee their performances, and understand students’ needs for lecturers to provide effective teaching techniques (Archana Rao and Baglodi, 2017), (Mite-Baidal et al., 2018). In the business field, sentiment analysis helps monitor trends on customers’ overall opinion about a product or brand. Sentiment analysis on movie reviews provides a true reflection of the emotions being conveyed. The government uses sentiment analysis to analyse trending government-related topics, such as policies and politics.

2.4 History and Evolution of Sentiment Analysis

A study conducted by Mäntylä, Graziotin and Kuutila (2018) was to bits of help in understanding the volume of work in sentiment analysis, the impact of sentiment analysis, popular venues for publishing sentiment analysis studies. This research was accomplished by observing previous annual trends that were enlightening on the history of this topic. The research attempted to explain opinion analysis from the beginning of the 20th century from subjective analysis to computational analysis.

(Ahlgren, 2016) carried out a bibliometric study on sentiment analysis to create a statistical background on sentiment analysis as a whole and its progress over the years. This paper evaluated the evolution of sentiment analysis based on The increase in the topic’s popularity over the years, which topics were being discussed, leading researchers in the field.

From their research, the following pieces of information were obtained.

- The first set of papers that matched the research was published post World War II (1940), for example, titled “The Cross-Out Technique as a Method in Public Opinion”. These analyses
were primarily done to investigate public opinion in countries that had suffered from the war.

• In the mid-’90s, The computing revolution began to have little impact as researchers used computer systems for expert opinion and sentiment analysis.

• Still, the topic was subtle until the mid-2000s, when there was an increased wave of modern sentiment analysis due to online reviews’ need and growth.

• The sentiment analysis field experienced exponential growth within recent years; there was an about 50-fold increase on this topic between 2005 and 2015. The number of researches on this topic increased from 101 to 5699 within these ten years.

• The paper showed that the research topics between 2014 to 2016 focused more on Twitter and Facebook, different from 2013 that focused more on products and political reviews.

• The result showed that methods used in sentiment analysis had been classified into: Machine Learning, Natural Language Processing, Sentiment Analysis Specific Method.

• Most researchers’ target issue was product and service reviews, movies, health, argumentation, interaction with the audience, election, expertise, sarcasm, dialects, spam review.

2.5 Techniques for Sentiment Analysis

Kaur and Sidhu (2018) delve into the Sentiment Analysis research area. The paper gives a clear description of the most recent development in the field in a literature survey.

The sentiment analysis process is explained in detail first, highlighting the process’s practical applications and its importance to business nowadays; this is followed by reporting on the different Sentiment Analysis levels, approaches, and methodologies researchers have used to explore the research area. The authors finish by discussing the application of several deep learning algorithms for sentiment analysis.

Sentiment Analysis is mainly divided into three major categories, i.e. lexicon Based, Machine Learning Approach, and Hybrid Approaches.
2.5.1 Lexicon Based Approach

The lexicon-based approach involves calculating orientation for a document based on the document’s words or phrases’ semantic orientation; it scans through the documents for words that express positive or negative feelings to humans. By searching their existing lexicons, lexicon-based approaches determine the sentiment orientation of an opinionated phrase of interest. The lexicon generation process is typically initiated with a seed list of opinion words with which their opinion orientations are associated (Mohsen et al., 2016). This strategy relies entirely on lexical features, which can be a dictionary, vocabulary, or a bag of words. All of the words in the dictionary are assigned scores, and various strategies, such as word positioning, phrases, are also assigned scores that can be used to compute sentiment for the document. This lexicon-based approach can be divided into two (2) main strategies: dictionary-based and corpus-based approaches.

Dictionary-Based Approach

In this method, a list of seed opinion words must be prepared ahead of time, and then a dictionary is then used to help grow the list to generate more opinion words. Dictionary-based methods expand the initial seed set by utilising online dictionaries such as WordNet and some additional information within them (such as their glosses). These approaches use synonym and antonym relationships between words and some machine learning techniques to create a more significant list.

Corpus-Based Approach

Unlike dictionary-based opinion word generation, corpus-based opinion word generation is based on syntactic or co-occurrence patterns in large text corpora. One of the primary advantages of corpus-based opinion word generation over a dictionary-based generation is the possibility of obtaining domain-dependent and/or context-dependent orientations. In this approach, rules are designed for connective terms such as And, Or, But, Neither-or, Neither-nor, and the seed set is expanded by leveraging them. Using the particular corpus of interest, domain-specific lexicons can be generated.

2.5.2 Machine Learning Approach

The traditional Machine Learning approach is still widely used in sentiment analysis tasks such as sentiment classification (Ligthart, Catal and Tekinerdogan, 2021); it uses different Machine Learn-
ing algorithms and some linguistic features (Jaspreet et al., 2018). Consequently, researchers have proposed a large number of approaches to satisfy the need for sentiment analysis.

**Unsupervised Learning**

These type of machine learning algorithms use unlabeled input data without a target variable. Using algorithms, hidden structure/pattern is discovered (Ahmad et al., 2017). The working principle of this algorithm is to identify the hidden associations in unlabeled data. The unsupervised learning methods are based on calculating similarity differences between data. Examples of unsupervised learning include Apriori and K-Mean.

**Semi-Supervised Learning**

Semi-supervised learning makes use of both labelled and unlabelled data to train the model. A set of unlabelled data is complemented with some examples of labelled data (often limited) included building a classifier. Semi-Supervised Learning-based methods appear to be a promising option for dealing with tweet sentiment analysis because there are many unlabelled tweets available instead of a small number of annotated tweets. Labelled tweets’ possession is frequently a costly process requiring skilled experts, whereas unlabeled tweets’ possession is relatively affordable. Simple semi-supervised classification models, such as Self-training, can be extremely useful in this context (da Silva et al., 2016).

**Supervised Learning**

Regression analysis is a type of probabilistic modelling technique that investigates the relationship between a dependent (target) variable and an independent variable (s) (predictor). Examples of regression sentiment analysis include used in; Naive Bayes and maximum entropy. A classification problem is a descriptive statistics problem that predicts a class label for a given example of input data. It includes a decision tree classifier, rule-based classifier and linear approach. The linear approach can be further divided into two (2); Support Vector Machine (SVM) and Artificial Neural Networks.
2.6 Sentiment Analysis using Machine Learning & Deep Learning

2.6.1 Traditional Machine Learning

(Haque, Saber and Shah, 2018) used supervised learning models for sentiment analysis using a large-scale dataset of Amazon Reviews. In the research, unlabelled Amazon Product Review datasets were gathered from three (3) different JavaScript Object Notation (JSON) formats and labelled. Pre-processing of the dataset involved tokenization, where the sequence of strings was separated into individual words and unnecessary characters such as punctuations were taken out, stop words were removed, Part of Speech tagging was assigned, assigning part of speech to given words. Feature extraction involved Bag of Words, Term Frequency–Inverse Document Frequency (TF-IDF), and Chi-Square.

The study uses traditional Machine Learning models such as Support Vector Machine, Stochastic Gradient Descent, Linear Regression, Random Forest, and Decision Tree. In this experiment, Support Vector Machine provided the best results of the 3 data sets as Support Vector Machine works
better with large datasets having an accuracy of 94.02%.

(Karthika, Murugeswari and Manoranjithem, 2019) perform a Sentiment Analysis on FlipKart: India’s biggest online store dataset using traditional machine learning techniques: Support Vector Machines and Random Forest. The dataset is prepared for analysis by removing the data inconsistencies and the unneeded attributes in the dataset. The ratings in the dataset are grouped into three, 1 and 2 star ratings are regarded as Negative, 3 as Neutral and 4 and 5 star ratings are considered to be Positive. The analysis is done using SPYDER. The selected models are evaluated using Accuracy, Precision, Recall and F-Measure. The authors report the Random Forest classifier surpasses the Support Vector Machines classifier across all evaluation metrics.

(Han, Chiu and Chien, 2019) examines the impact of Fisher Kernel function classification features on Support Vector Machines. The authors argue that conventional sentiment analysis methods do not take into proper consideration the latent semantic information in the textual information. Their analyses are done with undisclosed internet post data, however the authors report significant results of their proposed methods of the mainstream feature extraction methods.

### 2.6.2 Artificial Neural Networks

According to He et al. (2010), Von Neumann architecture is still used in today’s computer hardware and software systems. They can only solve actual problems mechanically by using predetermined programs, and their flexibility is less than that of humans when it comes to tackling specific problems such as sentiment analysis. Furthermore, they lack the mechanism and potential for developing methods from the environment and active adaptation to it.

Artificial Neural Network imitates the human nervous system as the human brain is an information-processing network system formed by a complex and mutual connection; it is essentially a biological neuron engineering approach (Mishra and Srivastava, 2014). The human brain, amongst other functions, is conditioned to memorise, compute, reason and think logically, precept and learn from the environment, and evolve with the environment. Like the human brains, Artificial Neural Networks can be used to solve problems that are difficult to solve using traditional methods, develop new information denotation, storage, methodological approaches, and an information system that is closer to human intelligence.
2.6.3 Convolutional Neural Networks

In a paper by (Jian Yang and Juan Yang, 2020), a new model was proposed to solve Aspect Based Sentiment Analysis (ABSA) using the fusion of the Convolutional Neural Networks model and self-attention mechanism. The authors narrate that the structure of a sentence can influence classification results in ABSA related tasks. The authors note that related research reports the self-attention algorithm’s excellent performance and gating mechanism to solve this.

Experiments are conducted on two types of Aspect Based Sentiment Analysis: Aspect-Category Sentiment Analysis (ACSA) and Aspect-Term Sentiment Analysis (ATSA). The datasets used for the experiments are SemEval 2016 and SemEval 2014, respectively.

In the ACSA tasks, the proposed model contains an embedding layer, the self-attention algorithm, 3 CNN filters, the gating mechanism, the entire connection layer and the softmax layer. In the ATSA, the proposed model consists of an extra CNN filter in the convolutional neural network area. The dot-product algorithm inspires the self-attention algorithm; its sole aim is to increase the learning of the correlation between the words in the text. The authors use accuracy as their primary evaluation metric and report decent results.

(Liao et al., 2017) investigate Convolutional Neural Networks’ application in real-world situations by analysing Twitter data’s sentiment. The paper introduces this approach due to the popularity of CNNs in image classification and recognition tasks. Also, the authors juxtapose image and textual data, stating textual data can be transferred into a matrix, which is identical to an image pixels’ matrix; therefore, the same operation can be applied to both kinds of data.

Experiments are conducted on the Movie Review Data and STS Gold Dataset. The methodology comprises the data collection stage, where the datasets are sourced; the feature extraction stage immediately follows this. The textual data, i.e. sentences, are processed into a word vector matrix/representations; next, the data is split into training and testing sets. Subsequently, the training set is used as input for the CNN algorithm. The author reports that the highest accuracy model has windows filters set to 4, 5, 6 with a hundred feature maps each, a dropout rate of 0.5, and L2 regularisation is set to 0.001. Further research recommendations include feature extraction with word2vec, multilayer convolutional neural network algorithms.

(Ouyang et al., 2015) tackle the automatic classification of sentiment polarity into five classes
(Negative, Positive, Neutral, Somewhat Negative and Somewhat Positive) using Convolutional Neural Network. Inspiration is drawn from the Deep Learning (DL) algorithm’s favourable results and the model being computationally easier to train.

The proposed technique uses word2vec to process the vector representations of the text, and finally, the CNN model to identify the sentiment in the text. The analyses are performed using Caffe: an open-source DL framework. The dataset is sourced from movie reviews from RottenTomatoes.

The processed vector representations serve as inputs to the CNN model, which comprises two convolutional layers with Parametric Rectified Linear Unit (PReLU) activation, each followed by dropout layers, max-pooling layers and normalisation layers. There is a convolutional layer with a max-pooling layer after the previous layers, and finally, a fully connected layer for output. Accuracy is used to measure the proposed framework, with the authors presenting the best results in the state-of-the-art.

(Tammina and Annareddy, 2020) employ a deep learning technique, the deep Convolutional Neural Network, to capture sentiment using Amazon Product Reviews and IMDB movie reviews. In the paper, the authors compare the deep learning model with traditional machine learning models, including Random Forest and Naïve Bayes. Results showed that Convolutional Neural Network had the highest accuracy of 74% on the Amazon Product Review dataset and 68% on the IMDB dataset.

2.6.4 Recurrent Neural Network

Recurrent Neural Network is a feed-forward Neural Network introduced by Ethan (1990). It is connected between neurons and forms a directed cycle, which generates feedback loops within the RNN. Because RNN incorporates memory cells intended to capture information about long sequences, prediction is sequential and hidden layers from one prediction give the next (Patel and A. K. Tiwari, 2020, Habimana et al., 2020, Dang, Moreno-Garcia and De la Prieta, 2020).

(Sarkar, 2019) explored sentiment polarity detection in Bengali tweets using Long Short-Term Recurrent Neural Network. The research paper focuses on Bengali tweets because of the availability of the Indian language on social media and due to most of the work done in sentiment polarity detection being carried out in the English Language.

The author used a Bengali tweet dataset initially used in Sentiment Analysis in Indian Languages
at the 3rd Mining Intelligence and Knowledge Exploration.

The methodology follows two different stages; in the first stage, the tweets are pre-processed. Characters irrelevant to the task are removed at this stage; this is immediately followed by employing SentiWordNet to label the tweets according to their polarity. The second and final stage involves building and training the model to predict the polarity of the tweets.

There are three experiments described in this research paper. All three experiments are conducted to perform a comparative analysis of all Machine Learning algorithms used in the research. The first experiment explores training a Multinomial Naïve Bayes model using pre-processed Unigram and Bigram tweet features. The second experiment involves training a Support Vector Machine model with only Unigram tweet features. The last experiment encompasses classifying the tweets using the Short-Term Long Memory (LSTM) model.

The author reports the proposed method LSTM model performs best with an accuracy of 55.27%, the SVM with 53.73% and the Multinomial Naïve Bayes 53.07%. The author further highlights that the size and mislabelling are the drawbacks of using the dataset. With the results, the author concludes that with a few modifications, the proposed method could automatically classify the polarity of other Indian languages.

(Uddin, Bapery and Arif, 2019)’s research looks at the effectiveness of Machine Learning in analysing Depression on Social Media. The authors utilise Long Short-Term Memory (LSTM) Deep Recurrent Network to investigate Bangla social media data. The authors acknowledge that their paper is among the original papers that explore this research area. Their methodology encompasses two steps: The dataset is prepared for training in the first step. The data is sourced, it is cleaned, pre-processed, and labelled manually by a sociology student. The final dataset contained 984 non-depressive and depressive tweets, respectively.

In the second step, the machine learning model is trained using the processed dataset from the previous stage; the dataset is split into 80% for training, 10% for validation, and 10% for test. In this stage, the model hyperparameters are also tuned to achieve optimal values that yield the best model performance.

The authors report the best accuracy obtained is 86.3% after hyperparameter tuning. The best LSTM model has five layers with 128 neurons, 25 batch size and a learning rate of 0.0001 for over
20 epochs. In conclusion, the authors state that results establish that Machine Learning can be utilised for complex psychological tasks like depression analysis.

The paper by (Monika, Deivalakshmi and Janet, 2019) aimed to classify Twitter posts about six US airlines into three polarity categories: Positive, Negative, and Neutral; this is done using a combination of Recurrent Neural Network and Long Short-Term Memory models.

The dataset is sourced from Twitter. The experiments involve two steps; in the first stage, the tweets are pre-processed using GloVe vector word embeddings. The training of the model stage follows this. The model architecture includes an input layer, two LSTM layers, each followed by Dropout layers, a fully connected Dense Layer and an output layer with Softmax activation function to produce the predictions.

The authors report favourable results that allow them to conclude the proposed method being a reliable option for predicting US airline-related tweets’ polarity. However, the authors recommend that in order to improve performance, further research studies should be carried out on the Bidirectional LSTM Model (Bi-LSTM).

(Archary and Coetzee, 2020), in a paper, attempts to forecast Apple, Microsoft, and Amazon’s stock volatility using Social Media posts and technical features, which was gathered over a decade and Deep Learning.

The methodology used involves different stages. The first stage, data for the experiments are collected. OCHL data: Opening Price (O) The company’s stock price at the beginning of a trading day, Closing Price (C) The company’s stock price at the close of a trading day, Highest Price (H) The company’s stock price at its highest value during the trading day and the Lowest Price (L) The company’s stock price at its lowest value during the trading day are gathered over 2516 days of trading. Then the researchers’ source posts about the selected companies from Twitter using their respective hashtags “$AAPL, $AMZN, $MSFT.”

The social media posts are cleaned using the following steps: first off, the tweets with more than one cash tag are removed. Subsequently, the tweets that were empty and or were not in the English Language were removed too. Python library sci-kit learn is used to tokenise the tweets. Finally, two datasets are organised by month and are combined into one dataset.

A four-layer Recurrent Neural Network is utilised to make the stock market price prediction, The first three layers contain Gated Recurrent Units (GRU), and the last layer makes the prediction. Vari-
uous experiments are carried out, single technical features, multiple technical features and technical features and social media content to perform a comparative analysis. The authors use Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) to evaluate the performance of their proposed model. The proposed model performs poorly compared with the state-of-the-art, to which the authors attribute this to the method in which the data is pre-processed; however they report the research objective: to predict the stock prices of Microsoft, Amazon and Apple were successfully achieved.

Mukherjee et al. (2019), adopted five (5) variants of Recurrent Neural Networks (RNN). The variants include a simple RNN, Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM) and Bidirectional LSTM. Using the Amazon Product Review Dataset, these variants were used to see which model performs the best in multi-class sentiment classification. The output of a simple RNN is found to be the worst in all situations, while the bidirectional LSTM model is the best performing. To prevent bias, another experiment was carried out by over sampling the datasets. The result indicate that the models perform better after oversampling process is done.

2.6.5 Recursive Neural Networks

(Aydin and Güngör, 2020) examines the efficacy of applying an ensemble model in the aspect-based sentiment analysis domain. The ensemble model consists of Recursive and Recurrent Neural Networks. The authors draw attention to these models being widely adopted in the literature; furthermore, detailing their proposed hierarchical combination method elevates both models’ advantage.

The experiments are performed on the SemEval-2014 Task 4 competition dataset. To train the ensemble model, the data is first pre-processed. The text data undergoes tokenisation with a spaCy library in Python. Next, aspect term groups are created using a dependency and constituency parser. The constituency parser generates trees by dividing the text into phrases to help sort information with sentiment in a structured manner. In contrast, the dependency parser joins words in the text through their binary relationship.

The processed text is used for input for the Recursive Neural Network. The recursive unit better understands the grammatical and semantic structure in the text. The output of the recursive unit is then used as input for the Recurrent Neural Network. The recurrent unit learns the effects of time in the sequence of the text. Accuracy is used as the sole evaluation metrics in the experiments, with
the authors reporting significantly better results than the related works.

2.7 Challenges of Sentiment Analysis

This section will introduce challenges in the Sentiment Analysis research area. Significant challenges can range from the technique or approach used to the application of the Sentiment Analysis System. These challenges are elaborated on in the following subsections.

2.7.1 Domain/Language Specific

The bulk of research done in the field is targeted at the English Language; hence, most tools and libraries (sentiment lexicons or corpus) are built for the English Language (Shahnawaz and Astya, 2017). Applying these resources to other languages does not yield desirable results. In addition, lexicon-based techniques can be limited in scope. Sentiment analysis systems developed with such techniques can not handle diverse data from various domains (Shayaa et al., 2018).

2.7.2 Quality Data

Data with accurately labelled sentiment is often complex and costly to source (Shahnawaz and Astya, 2017). This is a direct result of the strenuous task of annotating sentiment labels in sourced data. Various sentence types can pose a challenge for annotating sentiment; for instance, sarcasm, ridicule, rhetorical questions (Mohammad, 2017) and negation. Researchers, however, circumvent this by utilising semi-supervised and unsupervised Machine Learning.

2.7.3 Bias and Fairness

Building Accountable and Transparent Algorithmic Systems is a popular emerging area in Machine Learning. The principles for such systems aim to provide fairness in the decisions made by their algorithms. Sentiment Analysis is utilised in various sensitive fields like healthcare, customer retention, marketing etc. it is therefore vital to ensure fairness and remove bias. Bias should be identified at various stages in the process of the development of Sentiment Analysis systems, it can range from bias in training data, word embeddings, and the architecture of the machine learning model used (Poria et al., 2020).
2.8 Conclusion

This chapter of this research project gives a detailed understanding of sentiment analysis, evolution of sentiment analysis as research shows that the first work on sentiment analysis was done post-war and was done primarily to investigate countries that had suffered from the war. Analysis of sentiments increased tremendously in the past decade and is now used in various life areas, including: education, politics, businesses, religion. The two (2) techniques for sentiment analysis, including lexicon-based approach, machine learning approach and hybrid approach, were also discussed. The chapter further explained the challenges of sentiment analysis and gave detailed explanations of various authors’ previous work.

Following a thorough and in-depth analysis of the state-of-the-art, for the purpose of achieving the research objective, the following models are chosen: Convolutional Neural Network and Long Short-Term Memory. The models are selected because of their performance in the related work and their widespread presence in the literature.
Chapter 3

Data Mining Methodology

This Chapter introduces the Data Mining Methodology applied to solve the research objectives. This project adopts the Knowledge Discovery in Databases (KDD) methodology. The KDD process allows for easy uncovering of vital knowledge from any collection of data. This process consist of the following steps: Data Selection, Data Preparation, Preprocessing and Transformation etc. In figure 3.1 the KDD Process Flow Diagram is shown. The rest of this chapter discusses each steps in the methodology.

Figure 3.1: Knowledge Discovery in Databases Process Flow Diagram
3.1 Data Selection

The data used for the research experiments is sourced from Amazon (Amazon, 2015). The reviews are related to the Products available on Amazon.com. The reviews are from customers over a period of two decades (1995-2015). The Electronics Products reviews are selected for this project. The data was obtained in a tsv format, where each row represents a single review. The data has attributes such as review_id, product_id, product_category, rating etc.

3.2 Data Preparation & Preprocessing

Following the data selection is the Data Preparation and Preprocessing Stage in the KDD Methodology. This stage involves preliminary investigation of the sourced data. In the following subsections, the steps taken at this stage are discussed.

3.2.1 Exploratory Data Analysis

Exploratory Data Analysis is performed on the selected data to examine the relationships within the attributes in the dataset. It is an initial way of discovering patterns in the data.

There are five distinct class of Ratings in the dataset, figure 3.2 displays the Rating Countplot in form of a bar chart, this diagrams shows that the distribution of the ratings are negatively skewed.
Figure 3.2: Ratings Count Plot - The x-axis reveals the five classes of Ratings whilst the y-axis shows the count for each Rating class.

Figure 3.3 shows a pie chart of the brands in the dataset, the legend reveals the brand with the highest number of devices is Amazon, whilst the least is Amazon Digital Services Inc.

Figure 3.3: Brands Pie Chart

A Word Cloud of the most prevalent words in the reviews is shown in figure 3.4. Some of the
words include Amazon, Kindle, Disappointed, Fire, Grand, etc.

The Exploratory Data Analysis stage reveals that the class imbalance in the target variable, Figure 3.5a shows the class imbalance present in the target variable, while figure 3.5b displays the target variable after the class imbalance was treated.

![Review Word Cloud](image)

Figure 3.4: Review Word Cloud

![Target Variable](image)

(a) Target Variable: Class Inbalance  
(b) Target Variable: UpSampled

Figure 3.5: Target Variable

### 3.2.2 Data Cleaning & Feature Selection

The dataset is further scrutinized. Null Ratings are removed, this is because null ratings do not have any valuable information and can not be transformed into the sentiment label. Checks are carried
out to ensure there are no duplicate reviews in the data. Lastly, the Features used to build the deep learning models are chosen.

### 3.3 Data Transformation

In order to training the deep learning models, The data undergoes a data transformation stage. Machine Learning models do not understand text data, therefore, at this stage the reviews are transformed from text to numerical vectors.

### 3.4 Data Mining

After taking careful consideration of the related work in the sentiment analysis research area, this research paper investigates the use of two Deep Learning Models (Convolutional Neural Network and Long Short-Term Memory) to automatically detect and classify sentiment polarity in the reviews.

1. **Convolutional Neural Network**: A convolutional neural network is a type of feed-forward neural network introduced in 1989; it was initially used in computer vision, recommender systems, and natural language processing. The application of CNN in Neutral Language Processing was initiated by (Collobert and Weston, 2008).

   Raw data is inputted and produces embedding, convolutional, and pooling or subsampling layers that typically provide input to a fully connected classification layer to extract features. This convolutional layer is applied as a feature detector to extract feature maps, while the pooling layer is a dimension reduction layer used to reduce unnecessary information and avoid overfitting (Dang, Moreno-García and De la Prieta, 2020).

   According to Habimana et al. (2019), Convolutional Neural Networks excel at learning local contextual features using filters in sentiment analysis. However, CNN’s are limited in modelling long-term dependencies in sentiment analysis because they must be profound, making them time-consuming. Also, their effectiveness is dependent on a large number of training samples, which are not always accessible.

2. **Long Short-Term Memory**: Long Short-Term Memory (LSTM) is a particular type of RNN proposed by (Hochreiter and Schmidhuber, 1997) to face the vanishing gradient issue by
extending the classic RNN with a gating mechanism. LSTM is capable of using long memory as the input of activation functions in the hidden layers. Similar to the CNN Model, the input layer is pre-processed to reshape the data for embedding, the second layer is the LSTM layer the the fully connected layer for text classification. The sigmoid layer is then activated to reduce the vector height to an output vector of one.

3.5 Design Specification

This section highlights the technical design and project architecture utilized to implement this research project.

3.5.1 Overview of Project Architecture

The architecture for the project is structured into three tiers; the topmost layer is the presentation tier, here, the insights and results derived from the experiments are displayed to the end users. The next layer features the Logic of the experiments, this accommodates the Deep Learning models employed for the Sentiment Analysis. The last layer is the Data Persistent layer, that comprises the data used in the experiments and the tools used to manipulate, retrieve and store the data. Figure 3.6 provides a diagrammatic representation of the Project Architecture.
### Table 3.1: System Configuration

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>OS</td>
<td>Windows 10</td>
</tr>
<tr>
<td>Memory</td>
<td>16 GB</td>
</tr>
<tr>
<td>CPU</td>
<td>Intel® Core™ i5-10210U processor</td>
</tr>
<tr>
<td>Software</td>
<td>Jupyter Notebook, Google Colaboratory</td>
</tr>
<tr>
<td>Python Version</td>
<td>3.7.9</td>
</tr>
<tr>
<td>Keras Version</td>
<td>2.4.3</td>
</tr>
<tr>
<td>TensorFlow Version</td>
<td>2.3.1</td>
</tr>
</tbody>
</table>

#### 3.5.2 Environment Settings

The section details the system configuration the experiment was performed under. The project is implemented in Python, with deep learning libraries such as Keras and TensorFlow. Table 3.1 above describes the system configuration information.
Chapter 4

Experimentation

This Chapter elaborates on the processes performed to execute the project and accomplish the research objectives. In the Implementation stage of the project, first, the dataset for the experiments is obtained. The following subsections provide a description of the remaining steps.

4.1 Data Preparation & Preprocessing

In this step, the data is prepared for the training of the models. The dataset is initial examined to discover the relationships present in it. Details are present in Section 3.2.1. After, the data is cleaned, null ratings and duplicate reviews are removed as noisy data can have adverse effect on the model. Finally, the clean data is stored in a PostgreSQL database.

4.2 Data Transformation

Machine Learning models are incapable of handling data in raw text form, therefore, the data is transformed. The following are the steps taken to transform the text data.

1. The reviews are converted to lowercase, special characters like square brackets, numbers and punctuation are removed. Subsequently, stop words are filtered out.

2. The cleaned reviews are broken down to tokens, next, the tokens are transformed to vector sequences, this is followed by padding the vector sequences.

3. Word Embedding Techniques are applied to the vector sequences. This produces vector representation the tokenized words, Word Embeddings are created with GloVe (Pennington,
Sentiment Labels are generated using the Rating in the dataset, 5 and 4 stars ratings are considered positive, while 1 and 2 star ratings are negative. The dataset is split in training and test sets in the ratio of 80:20. Furthermore, the training set is split 80:20 to create a Validation set.

4.3 Data Mining

Two Deep Learning algorithms are applied on the vector representation of the reviews. There are four experiments performed.

The architecture of the algorithms, batch size, number of epochs remain the same across all four experiments to allow comparison of the two models. The model architecture are described below.

1. **Convolutional Neural Network**: The Convolutional Network Model has an Embedding Layer, followed by a 1D Convolution Layer with 128 neurons, kernel size of 5 and ReLU activation, a 1D Global Max Pooling layer, this layer is followed by a Flatten layer and a fully connected layer with ten neurons, the final layer is the Output layer with a Sigmoid Activation function. This model is compiled with the Binary Cross Entropy loss function and Adam optimizer.

2. **Long Short-Term Memory**: The Long Short-Term Memory model begins with an Embedding Layer, the next layer is a LSTM layer with 128 units and then a fully connected Output layer for classification. To compile the model, the loss function is set to Binary Cross Entropy, and Adam is used as an optimizer.

4.4 Hyperparameter Tuning

Hyperparameter Tuning is implemented on the baseline Models to examine the performance of the models. **Random Search** is utilised to perform the Hyperparameter Tuning. In Random Search, the parameters in the search space are chosen using probability. The purpose is to find an optimal solution fast, without exhausting the whole search space. The hyperparameters of the models like the activation functions, filters and neurons in each layer and learning rates are tuned. The details from the experiments are highlighted in the next Chapter of this report.
Chapter 5

Evaluations and Results

This Chapter present the results of the experiments performed on the dataset introduced in 3.1. Accuracy, Precision, Recall and F1 Score are the chosen metrics used to evaluate the models. These metrics as well as the results and observations from the experiments are expanded below.

i Accuracy: Accuracy can be defined as a metric that is used to evaluate ML classification models. Accuracy is calculated as the proportion of correct model predictions to total number of model predictions. The Accuracy formula is given in the equation 5.1 below.

\[
    Accuracy = \frac{TruePositive + TrueNegative}{TruePositive + FalsePositive + TrueNegative + FalseNegative} \tag{5.1}
\]

ii Precision: Precision can be defined as a metric that evaluates the classification model’s ability to detect relevant cases. Precision is measured by dividing true positives by the sum of true positives and false positives. The formula for Precision is presented in equation 5.2.

\[
    Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{5.2}
\]

iii Recall: The percentage of overall relevant cases correctly identified by any classification model is known as Recall. Recall is calculated by multiplying True Positives by the sum of True Positives and False Negatives. The definition is represented mathematically in equation 5.3.

\[
    Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{5.3}
\]

iv F1 Score: The F1 Score is a metric which give balance to the model’s Precision and Recall. Here, both metrics: Precision and Recall have equal weights. The F1 Score is calculated using the equation below.

\[
    F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5.4}
\]
5.1 Experiment 1 - Baseline Models

Table 5.1 gives information of the results from the baseline model experiment. In this experiment, the models are trained with a class imbalance present in the data. The CNN model obtains an Accuracy of 93%, with a Precision of 73%, Recall and F1 Score of 59% and 63% respectively. In contrast, the LSTM model acquires 93% Accuracy, 72% Precision, 66% Recall and F1 Score of 69%.

Table 5.1: Baseline Model Results

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy (%)</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F1 Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>93</td>
<td>73</td>
<td>59</td>
<td>63</td>
</tr>
<tr>
<td>LSTM</td>
<td>93</td>
<td>72</td>
<td>66</td>
<td>69</td>
</tr>
</tbody>
</table>

Figure 5.1: Baseline CNN Model Confusion Matrix

Figure 5.1 presents the confusion matrix for the Baseline CNN Model. The model correctly predicts 93 instances in the test set as Negative and 6,370 instances as Positive. Additionally, the model incorrectly predicts 84 instances as Negative and 379 instances as Positive.
The confusion matrix of the Baseline LSTM model is shown in figure 5.2, the model accurately classifies 168 Negative and 6271 Positive instances. The baseline LSTM model also misclassifies 304 instances as Positive and 183 instances as Negative in the test set.

5.2 Experiment 2 - Models with UpSampled Data

Table 5.2: UpSampled Model Results

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy (%)</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F1 Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>98</td>
<td>98</td>
<td>98</td>
<td>98</td>
</tr>
<tr>
<td>LSTM</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
</tr>
</tbody>
</table>

The details of the results in the UpSample experiments are provided in Table 5.2. In the Upsampled Data Experiment, the class imbalance in the target variable is treated.

- The Accuracy, Precision, Recall and F1 Score of the CNN Model are 98%. Figure 5.3 displays the confusion matrix for the CNN Model. The model accurately predicts 6529 cases as Negative, 6193 cases as Positive. Along with the accurately predicted cases, the model predicts incorrectly 205 cases as Negative.
• The LSTM Model’s Accuracy, Precision, Recall and F1 Score are 96%. In figure 5.4 shows the confusion matrix for the LSTM Model. Here, the LSTM model correctly classifies 6417 Negative and 5942 Positive cases, furthermore, the model incorrectly classifies 112 cases as Positive and 456 cases as Negative.

Figure 5.3: UpSample CNN Model Confusion Matrix

Figure 5.4: UpSample LSTM Model Confusion Matrix
5.3 Experiment 3 - Hyperparameter Tuning of Baseline Models

The Hyperparameter Tuning experiments aimed at improving the overall accuracy of the baseline models. Apart from the previous stated goal, the general hypothesis here was that the experiment might prevent the models from overfitting the training data.

The best CNN Model obtains 93% Accuracy, 75% Precision, 59% Recall and 63% F1 Score as opposed to the best LSTM Model which acquires an Accuracy of 94%, Precision of 82%, Recall of 60% and a F1 Score of 65%. This information is presented in a tabular format in table 5.3.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy (%)</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F1 Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>93</td>
<td>75</td>
<td>59</td>
<td>63</td>
</tr>
<tr>
<td>LSTM</td>
<td>94</td>
<td>82</td>
<td>60</td>
<td>65</td>
</tr>
</tbody>
</table>

The confusion matrix for the CNN Model in the Hyperparameter Tuning Experiment is shown in figure 5.5. The figures shows the best CNN model predicts correctly 95 Negative and 6,376 Positive cases. However the model incorrectly predicts 377 Negative cases as Postive and 78 Positive cases as Negative.

Figure 5.5: CNN Model Confusion Matrix - (Hyperparameter Tuning Experiment)
Figure 5.6 displays the LSTM model confusion matrix in the Hyperparameter Tuning experiments. The model is able to accurately classify 99 instances as Negative and 6,411 cases as Positive. Furthermore, it misclassifies 373 Negative instances as Positive and 43 Positive instances as Negative.

![Figure 5.6: LSTM Model Confusion Matrix - (Hyperparameter Tuning Experiment)](image)

Tables 5.4 and 5.5 contains details of the random search hyperparameter tuning performed on the baseline models. The table includes information of the hyperparameter for each model, their initial values, the hyperparameter search space and the optimal values obtained after the experiments.

The best CNN Model in the Hyperparameter Tuning Exercise possesses a Convolutional Layer with an optimal values for the Filter, Kernel Size and Activation Function of 80, 4, ReLU. The optimal number of neurons in the Dense Layer is 35, this Dense layer has an ELU activation function as the optimal value. Finally, the optimal activation function in the output layer is Sigmoid and the optimal learning rate is 0.00045.

The best Hyperparameter LSTM Model has an optimal value of 128 units in the LSTM layer with a Tanh activation. The Output layer also has a Tanh activation function and the optimal learning rate is 0.00176.
Table 5.4: CNN Hyperparameter Tuning Settings

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Initial Value</th>
<th>Hyperparameter Search Space</th>
<th>Optimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convolutional Layer Filter</td>
<td>128</td>
<td>16, 32, 48, 64, 80, 96, 112, 128</td>
<td>80</td>
</tr>
<tr>
<td>Convolutional Layer Kernel Size</td>
<td>5</td>
<td>4, 6, 8, 10, 12, 14, 16</td>
<td>4</td>
</tr>
<tr>
<td>Convolutional Layer Activation Function</td>
<td>ReLU</td>
<td>ReLU, Tanh, ELU, Sigmoid</td>
<td>ReLU</td>
</tr>
<tr>
<td>Dense Layer Neurons</td>
<td>10</td>
<td>10, 15, 20, 25, 30, 35, 40, 45, 50</td>
<td>35</td>
</tr>
<tr>
<td>Dense Layer Activation Function</td>
<td>ReLU</td>
<td>ReLU, Tanh, ELU, Sigmoid</td>
<td>ELU</td>
</tr>
<tr>
<td>Output Layer Activation Function</td>
<td>Sigmoid</td>
<td>Tanh, Sigmoid</td>
<td>Sigmoid</td>
</tr>
<tr>
<td>Learning Rate</td>
<td>0.001</td>
<td>0.0001 - 0.01</td>
<td>0.00045</td>
</tr>
</tbody>
</table>

Table 5.5: LSTM Hyperparameter Tuning Settings

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Initial Value</th>
<th>Hyperparameter Search Space</th>
<th>Optimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM Layer Units</td>
<td>128</td>
<td>16, 32, 48, 64, 80, 96, 112, 128</td>
<td>128</td>
</tr>
<tr>
<td>LSTM Layer Activation Function</td>
<td>ReLU</td>
<td>ReLU, Tanh, ELU, Sigmoid</td>
<td>Tanh</td>
</tr>
<tr>
<td>Output Layer Activation Function</td>
<td>Sigmoid</td>
<td>Tanh, Sigmoid</td>
<td>Tanh</td>
</tr>
<tr>
<td>Learning Rate</td>
<td>0.001</td>
<td>0.0001 - 0.01</td>
<td>0.00176</td>
</tr>
</tbody>
</table>

**5.4 Discussion**

The experiments sought to automatically classify sentiment polarity in electronic reviews, to achieve this two deep learning models are built. In the baseline model experiment, the LSTM model outper-
forms the CNN Model, however there is a class imbalance in the target variable which causes both baseline models to overfit the training data. As a result of the overfitting of the training data in the baseline models, the class imbalance is treated in the next experiments, here the minority class in the target variable is upsampled. In this experiment, the models do not overfit on the training data, in addition, the CNN Model attains an overall performance over the LSTM model.

It is worthy to note that the models trained with the class imbalance dataset have a high False Positive rate (type 1 error), whereas, the models trained with the upsampled data have a high False Negative rate (type 2 error). The Hyperparameter Tuning experiment for the LSTM show a slight improvement in the metrics, however, no tests are performed to ensure this improvement is statistically significant. In the case of the CNN Model, the metrics remain the same as the Hyperparameter Tuning does not leave any impact.

Without a doubt, there are a number of reasons why the experiments produced this kind of results: for instance, the technique used to obtain vector representations from the text. In most cases, feature extraction is vital to obtaining desirable results. To conclude, from recent literature, Bidirectional Encoder Representations from Transformers (BERT) models are delivering promising results, it would be worthwhile to explore this model to obtain improved accuracy and low false positive rates.
Chapter 6

Conclusions and Future Work

6.1 Conclusion

The field of Sentiment Analysis and Opinion Mining examines individuals’ emotions, point of view and sentiment in relation to a variety of things. Information from textual data is analysed to understand the connection of what is being said and the sentiment. This research paper addresses the problem of sentiment polarity detection and classification. Electronic products reviews retrieved from Amazon.com are obtained for the experiments.

The literature review chapter gives adequate answers to the first and third research question. The results from the experiments indicate the models are able to automatically detect and classify the sentiment polarity in the reviews. The LSTM model performs better in most of the experiments, Therefore, the research objective have been achieved successfully.

6.2 Future Work

To build up on this research work, the following future aspects can be considered. More meaningful characteristics of speech can be investigated to derive additional features for the Machine Learning models. With the appropriate labelled dataset, Aspect Based Sentiment Analysis can be also be probed to detect and categorize what is being discussed in the reviews. Models that can deal with multiple languages is another noteworthy area to explore. In conclusion, the models built here can be further extended to develop a Content-Based Recommendation System.
References


Appendix A: Data Preparation, Preprocessing & Transformation

A.1 Import Libraries

The figure below shows the code snippet used to import the Python libraries used to prepare and transform the data.

![Import Libraries](image)

Figure A.1: Import Libraries

A.2 Reading Data

After, importing the Python libraries, the dataset is read, this is done using the Pandas library. Figure A.2 shows the code snippet.

![Read Data from Disk](image)

Figure A.2: Reading Dataset
A.3 Word Cloud

The Word Cloud showing the most frequently used words in the dataset is computed using the code snippet in figure A.3.

![Word Cloud Code Snippet](image)

Figure A.3: Word Cloud Code Snippet

A.4 Data Transformation

The conditions for used to transform the ratings in the dataset are displayed in the code snippet below.
A.5 UpSampling

The UpSampling of the minority class in the target variable is shown in the code snippet in figure A.5.

```python
In [23]: # Separate majority and minority classes in training data for upampling
data_minority = clean_df[clean_df['label'] == 'Negative']
data_majority = clean_df[clean_df['label'] == 'Positive']

print('majority class before upsample:', data_majority.shape)
print('minority class before upsample:', data_minority.shape)

# upsample minority class
data_minority_upsampled = resample(data_minority, replace=True, # sample with replacement
n_samples= data_majority.shape[0], # to match majority class
random_state=123) # reproducible results

# Combine majority class with upsampled minority class
data_upsampled = pd.concat([data_majority, data_minority_upsampled])

# Display new class counts
print('After upSampling:
', data_upsampled.label.value_counts(), sep = '')

majority class before upsample: (32316, 3)
majority class before upsample: (2311, 3)
After upampling
Negative    32316
Positive    32316
Name: label, dtype: int64
```

Figure A.5: UpSample Code Snippet
A.6  Text Processing

The function in the code snippet below is used to clean the reviews, the function converts all the words into lowercase, removes special characters etc.

```
def cleanText(text):
    text = text.lower()
    text = re.sub('["]', '', text)
    text = re.sub('[\w\-\_\s]', '', text)
    text = re.sub('[\n\r\t]', '', text)
    return text

cleaned = lambda x: cleanText(x)
```

Figure A.6: Text Cleaning Code Snippet

A.7  PostgreSQL

The Data Preparation, Preprocessing & Transformation stage concludes with storing the cleaning and processed dataset in a PostgreSQL database. In figure A.7, a database called Research Project is created, figure A.8 contains the code snippets that show the population of the Baseline Data and UpSampled Data tables in the Research Project database.

```
try:
    dbConnection = psycopg2.connect(
        user = "postgres",
        password = "password",
        host = "127.0.0.1",
        port = "5432",
        database = "postgres")
    dbCursor = dbConnection.cursor()
    dbCursor.execute("CREATE DATABASE ResearchProject;")
    dbCursor.close()
    print("Connection Established, Query Executed Successfully")
except (Exception, psycopg2.Error) as e:
    print("Error while connecting to PostgreSQL", e)
finally:
    if(dbConnection): dbConnection.close()
```

Figure A.7: PostgreSQL 1 Code Snippet
Creating and populating PostgreSQL database with cleaned Dataset

```
In [31]:
    engine = create_engine('postgresql://postgres:password@127.0.0.1:5432/researchproject')

Populating BaselineData table in ResearchProject Database in PostgreSQL with the cleaned Dataset

In [32]:
    clean_df.to_sql('baseline_data', engine, if_exists='replace', index=False)

Populating UpsampleData table in ResearchProject Database in PostgreSQL with the cleaned upsampled Dataset

In [33]:
    data_upsampled.to_sql('upsampled_data', engine, if_exists='replace', index=False)
```

Figure A.8: PostgreSQL 2 Code Snippet
Appendix B: Experimentation - Models and Evaluation

B.1 Baseline CNN Model

B.1.1 Model Definition

Figure B.1 displays the code block in which the Baseline CNN Model is defined.

* CNN Model

Figure B.1: CNN Model Definition - Code Snippet

B.1.2 Training and Validation Accuracy & Loss

The Training and Validation Accuracy and Loss are shown in the line graphs in figure B.2. The graphs display the Baseline CNN Model model overfitting the training data.

Figure B.2: CNN Training and Validation Graphs: Accuracy/Loss are on the y-axis, the number of Epochs are on the x-axis
B.1.3 CNN Model Classification Report

The Classification Report for the Baseline CNN Model is highlighted in the figure below.

Figure B.3: CNN Model Classification - Code Snippet

B.2 Baseline LSTM Model

B.2.1 Model Definition

The Baseline LSTM model is defined in the code snippet shown in figure B.4

Figure B.4: LSTM Model Definition - Code Snippet

B.2.2 Training and Validation Accuracy & Loss

The Training and Validation Accuracy and Loss for the Baseline LSTM Model are shown in the line graphs in figure B.5.
Figure B.5: LSTM Training and Validation Graphs: Accuracy/Loss are on the y-axis, the number of Epochs are on the x-axis

B.2.3 LSTM Model Classification Report

The Classification Report for the Baseline LSTM Model is shown in the figure B.6 below.

Figure B.6: LSTM Model Classification - Code Snippet