

thesis

by Ujjawal Tomar

Submission date: 28-May-2026 11:31AM (UTC+0530)

Submission ID: 2971083415

File name: ujjawaltomar_thesisfile_plagiarism.pdf (267.03K)

Word count: 11081

Character count: 66244

Abstract

The exponential growth of digital retail platforms made huge volumes of unstructured consumer feedback available. Automated opinion mining has become a vital asset of modern business analytics. Traditional machine learning methods often do not understand the context of language and depend heavily on manually prepared text. This dissertation provides a rigorous comparative study of traditional frequency based statistical models against modern neural network architectures for binary sentiment prediction. The experimental setup is derived from the Amazon Musical Instruments review dataset. Since there were no pre-defined sentiment labels for the raw data, the target variables were created algorithmically. This was done by combining Natural Language Toolkit Part of Speech tagging with the SentiWordNet lexical database to calculate net polarity scores. Then, a Logistic Regression algorithm was trained on the sparse term frequency inverse document frequency matrices to establish a baseline classification standard. To overcome the structural limitations of the baseline framework, a Bidirectional Encoder Representations from Transformers was deployed and trained the same dataset using advanced graphics processing units. The classical Logistic Regression classifier was restricted to an accuracy of seventy percent. It was very limited in performance as it was not capable of handling sequential grammar. On the other hand, the neural network model achieved an impressive accuracy of ninety four percent. The bidirectional self attention mechanism allowed the model to capture deep semantic meaning and complex domain vocabulary. Crucially, the neural architecture was extremely robust to errors in the training data and successfully identified real human sentiment that the automated dictionary missed. This study presents a compelling case that deep transfer learning is a far superior, highly scalable and highly accurate solution for processing complex consumer language.

Contents

Candidate's Declaration	21 i
Certificate	ii
Acknowledgement	iii
Abstract	iv
Content	vii
5 List of Tables	viii
List of Figures	ix
1 INTRODUCTION	10
1.1 Background	11
1.2 Problem Statement	12
1.3 Research Objectives	13
1.4 Scope of the Study	14
2 LITERATURE REVIEW	15
2.1 Introduction	15
2.2 Lexicon Based Approaches	16
2.3 Traditional Machine Learning Paradigm	17
2.4 The Deep Learning Paradigm	18
2.4.1 Deep Bidirectionality	18

2.4.2	Pre-training and Fine-tuning Paradigm	19
2.5	Literature Summary	19
2.6	Research Gap	20
3	Methodology	21
3.1	Introduction	21
3.2	Research Objectives	23
3.3	Overview of the Proposed Framework	23
3.4	Data Preparation and Preprocessing	26
3.4.1	Text Preprocessing Techniques	26
3.4.2	Automated Target Generation	27
3.5	Feature Engineering	27
3.5.1	Create contextual embedding (for BERT)	28
3.6	Machine Learning Models Implementation	28
3.7	Deep Learning Model(BERT) Implementation	31
3.7.1	Introduction to Transformer Architecture	31
3.7.2	BERT Implementation Process	31
3.7.3	Training and Tuning the Model	32
3.8	Training Configuration	33
3.9	Evaluation Process	34
3.10	Decision Pipeline	35
3.11	Model Deployment Strategy	36
3.12	Methodology Summary	36
4	RESULTS and DISCUSSION	38
4.1	Intruduction	38
4.2	Experimental Setup	38
4.3	Performance Evaluation of ML Models	39
4.4	Performance Evaluation of BERT	40

4.5	Comparative Analysis of ML Models and BERT	41
4.5.1	Prediction Accuracy and Generalization	41
4.5.2	Tokenization vs Feature Engineering	41
4.5.3	Computational Complexity and Applicability	42
4.5.4	Key Comparative Findings	42
4.6	Evaluation Based on Research Objectives	42
4.7	Discussion of Findings	43
4.8	Results ²⁵ Summary	44
5	CONCLUSION	45
5.1	Conclusion	45
5.2	Social Impact	46
5.3	Future Scope	47

List of Tables

3.1 Comparison of Machine Learning (Baseline) vs. Deep Learning (BERT) 33

4.1 Performance Evaluation Metrics: Logistic Regression vs. BERT 27
42

List of Figures

3.1	Block Diagram of Sentiment Analysis	22
3.2	Architecture and Workflow of Proposed Sentiment Analysis Framework	24
3.3	Sequential Flow of the Data Preparation and NLP Preprocessing Pipeline	30
3.4	ML vs BERT model accuracy	35

38 Chapter 1

INTRODUCTION

In recent years, social media users have expressed their thoughts and emotions through multiple modalities rather than relying solely on text. Individual posts frequently combine text with images, short videos, or voice notes. Consequently, traditional single-modal sentiment analysis methods, which typically analyze only one data type such as text, are often insufficient for accurately interpreting user intent. At the same time, digital communication has made it easy for people to share their opinions all the time. For businesses, governments, and other groups, quickly figuring out if public opinion is positive, negative, or neutral about a topic is now very important. Sentiment analysis, especially when using machine learning, is key for this purpose. To improve the accuracy of sentiment analysis, researchers worldwide have begun exploring methods to utilise information from multiple sources simultaneously. This leads to the multimodal sentiment analysis (MSA). In simple terms, MSA utilises two or more types of data for example, text and images, or text and audio to determine the overall sentiment. Most of the multimodal sentiment analysis systems consist of three major steps. First, every data type (e.g., text, image or audio) is processed separately to extract useful features. Second, the different modalities are combined (or fused) representation. Finally, we apply a classifier to this fused representation to find the emotional tendency, e.g., positive, negative or neutral. Multimodal data is more informative than text, with richer context and more subtle cues, and often provides a more accurate picture of what users actually feel. MSA has attracted significant attention for its potential application to many real-world settings.

But in reality, people comment on different aspects of the same post (e.g. they like the product but they criticize the price or the delivery), so it is still a challenge to correctly identify the sentiment for each aspect. With fast growth of platforms like Weibo, Facebook and Twitter, the volume of user generated content has ballooned. Such posts often combine several modalities, especially text and images, to express

personal opinions. Hence, it is not only useful but also necessary to combine the complementary information provided by multiple modalities for the automatic recognition of users' attitudes. This task is multimodal sentiment analysis, i.e. combining different modalities of data to infer sentiment. Sentiment analysis, is essentially the task of identifying the emotion or attitude expressed in a piece of text. Multimodal sentiment analysis takes it one step further. It not only looks at text, but uses all available signals to better capture what people really think.

1.1 Background

Sentiment Analysis (also called opinion mining) is a key sub-domain within NLP that is devoted to computationally identifying and categorizing opinions expressed in a piece of text. The main goal is to find the polarity of a given text, usually classifying it as positive, negative, or neutral. Sentiment analysis has become a must-have for companies wanting to track brand reputation and consumers wanting to make an informed decision without reading hundreds of individual reviews. We study the domain of musical instruments, with the dataset. Reviews in this area are difficult in a special way. They are often very technical jargon, specific product references, and subtle contextual opinions where a seemingly negative word might describe positive. Complex texts like these need strong computational models for accurate parsing. Sentiment analysis was previously mainly based on lexicon-based approaches and classical Machine Learning (ML) algorithms. While these methods laid the groundwork for automated text classification, they are subject to the subtleties of human language, e.g., sarcasm, polysemy and long-range contextual dependencies. The paradigm has changed drastically to Deep Learning in recent years, leading to the development of Transformer-based architectures such as BERT (Bidirectional Encoder Representations from). This dissertation studies this technological evolution by empirically comparing a classical machine learning benchmark with a state-of-the-art BERT model, with the aim of demonstrating the superior contextual mapping power of deep learning in extracting sentiment from complex e-commerce reviews. The technological scene of Natural Language Processing has passed through several evolutionary stages, each one trying to overcome the restrictions of the previous ones. This background is important to contextualize the comparative approach of this research. The initial research in the field of sentiment analysis was primarily rule-based and lexicon-based. These approaches rely on predefined dictionaries of words that are annotated with sentiment scores manually or semi-automatically. An example is SentiWordNet, a lexical resource that associates to each WordNet synset (a set of synonymous words) numerical scores for positive, negative and objective sentiment. Developers are able to calculate the overall

sentiment score of a whole document by ³² tagging each word in a sentence with Part-of-Speech (POS) and comparing it with SentiWordNet. Lexicon-based methods are interpretable and do not need training data, but they fail to model sequence, negation and domain-specific context.

For this work, we use Logistic Regression as the traditional baseline. Logistic Regression employs a linear probabilistic model on the vectorized text, learning which words are the best predictors of positive or negative outcomes. However, these models are treating text in a fundamentally unordered way, ignoring the grammatical structure and sequential meaning of the language. The most recent revolution in the field of NLP, marking the arrival of the third generation is deep learning and transformers. As proposed by Vaswani et al. in 2017, the transformer utilizes a self-attention technique that enables the machine learning model to assign relevance weights to different words in a sentence irrespective of the distance between them. Expanding further on this concept, Google developed Bidirectional Encoder Representations from Transformers or BERT in 2018. As against traditional models which process text linearly either from left-to-right or from right-to-left, BERT is bi-directional. This means that the contextual representation of a word is determined by the complete surrounding context.

1.2 Problem Statement

E-commerce sites get flooded daily with customer reviews while making money from gaining valuable insights from these customer reviews is quite profitable, two significant computational hurdles have emerged. One of the problems involves collecting the data needed for analysis. In order for supervised models to be able to effectively train from large volumes of data, those pieces of data need to be annotated. Manually annotating tens of thousands of reviews either as "positive" or "negative" is highly susceptible to human biases, quite time-consuming, and extremely expensive. Because of this, researchers resort to utilizing automated systems to annotate data using lexicographic methods (like SentiWordNet). While feasible, this automation process brings with it its own share of errors. This happens since lexicons do not take into account context and meaning of words. The second problem revolves around the classification. Logistic regression is the most common method used when creating baselines for sentiment analysis tasks. Although it gives decent results, it does not scale well on complex datasets since the model is relatively simple in nature. It fails to capture negations, sarcasm, and non-linear dependency between words. In summary, the main issue being considered in this investigation is whether a highly sophisticated and context-savvy Deep Learning framework (BERT) can succeed in overcoming the

inherent constraints associated with traditional ML techniques, as well as those imposed by algorithm-generated ground truth labels. In this way, the question arises whether fine-tuning an already pre-trained Transformer would result in statistical improvements in classification accuracy over a baseline logistic regression on Amazon Musical Instrument reviews.

1.3 Research Objectives

The primary aim is to implement, evaluate and compare the performance of a classical machine learning model (Logistic Regression) and a state-of-the-art Transformer-based deep learning model (BERT) for the binary sentiment analysis task on Amazon Musical Instrument customer reviews. The following specific objectives guide the research and systematically attain the overarching aim:

- To clean and preprocess the raw unstructured reviews comprehensively using techniques like regular expressions, tokenization, stopword removal, and WordNet lemmatization.
- To produce objective ground-truth binary sentiment labels (Positive/Negative) for the unannotated data by creating a custom scoring algorithm using NLTK's Part-of-Speech (POS) tagging and the SentiWordNet lexicon.
- Train and test a classical machine learning classifier, Logistic Regression, on the processed dataset to establish a benchmark performance metric. Train and fine-tune a pre-trained transformer model (bert-base-uncased) with HuggingFace libraries and Google Colab's GPU-accelerated environment for the same classification task.
- To implement and fine-tune a pre-trained transformer model (bert-base-uncased) using HuggingFace libraries and Google Colab's GPU-accelerated environment for the same classification task.
- Evaluating the predictive power of both models using standard classification metrics (accuracy, precision, recall) and visualizations to assess the effectiveness of contextual embeddings versus traditional feature extraction.

1.4 Scope of the Study

The contribution of this dissertation is of quite importance from both academic and industrial point of view. Theoretically, this work contributes to the ongoing discussion on the robustness of Transformer models. It provides a direct empirical comparison of how traditional ML models break down when faced with noisy lexicon-generated labels, as BERT's deep contextual pre-training allows it to wade through label noise and capture the true semantic intent of the text. The study validates the supremacy of bidirectional attention mechanisms in NLP, proving that a BERT model can achieve 94.0 percent accuracy vs a benchmark of 70.0 percent on the exact same pseudo-labeled dataset. The results of this research provide a practical, scalable and automated solution to the modern e-commerce business. E-commerce vendors and musical instrument manufacturer could use the proposed BERT-based pipeline for the real-time monitoring of customer satisfaction. SentiWordNet allows businesses to bootstrap the labels at a fraction of the cost of employing human annotation teams, and BERT allows them to achieve high classification accuracy. This enables the rapid and accurate extraction of consumer sentiment, leading to quicker product iteration, improved quality control and improved customer relationship management.

This study also offers some theoretical perspectives to the ongoing debate on the robustness of Transformer models. It offers a direct empirical comparison of the failure modes of traditional ML models with noisy, lexicon-generated labels, versus the ability of BERT's deep contextual pre-training to parse through label noise and capture the true semantic intent of the text. The work validated the supremacy of bidirectional attention mechanisms in NLP by showing that a BERT model can achieve 94.0 percent accuracy on the same pseudo-labeled dataset of 70.0 percent accuracy measured against a benchmark.

17 Chapter 2

LITERATURE REVIEW

2.1 Introduction

This chapter attempts to give a critical and systematic survey of the literature on Natural Language Processing (NLP) with special stress on Sentiment Analysis and opinion mining. The rise of e-commerce platforms has led to a huge amount of unstructured consumer feedback. This has created a need for the design of effective computational models for extracting actionable insights from textual data. In this chapter, we discuss the evolution of the sentiment analysis approaches, from the unsupervised lexicon-based approaches, to the traditional supervised machine learning classifiers, and finally the state-of-the-art Deep Learning paradigms, powered by Transformer architectures. In this review, we critically analyze the advantages and intrinsic limitations of these successive technological generations and highlight the existing challenges in working with unannotated, domain-specific text. These findings conclude this chapter to define the specific problem statement and the research gap that this dissertation intends to fill.

Sentiment analysis, has been a focal point of NLP research for over two decades. Early scholarship in the field primarily focused on document-level classification, categorizing entire texts as either objectively positive or negative. However, as the digital landscape evolved into Web 2.0, characterized by user-generated content on platforms such as Amazon, Yelp, and Twitter, the focus shifted toward analyzing highly unstructured, informal, and noisy textual data. Product reviews in e-commerce pose a unique linguistic problem. Consumer reviews are full of typos, grammatical errors, slang, sarcasm and domain-specific jargon, unlike formal news articles or aca-

demographic publications. In the musical instruments domain—also the main domain of this dissertation—consumers often use technical jargon, such as “XLR cables,” “phantom power,” “pop filters,” “condenser mics. Moreover, sentiment in this domain is very context dependent; the adjective “heavy” can have a negative polarity when describing a portable keyboard, but a very positive polarity when describing the tonal quality of a bass guitar, or the durability of an instrument cable. According to prior literature, the main challenge in developing accurate automated sentiment classifiers is to deal with these nuances. Historically, researchers overcame this by using lexicon-based methods that paved the way for automated opinion mining.

2.2 Lexicon Based Approaches

Sentiment analysis was mainly based on lexicon or rule based methods before the widespread adoption of computationally expensive machine learning models. These methodologies are based on sentiment lexicons, which are pre-compiled dictionaries that associate lexical units (words or phrases) with quantitative sentiment scores based on their semantic orientation.

WordNet and SentiWordNet - A seminal contribution to this subfield was the development of WordNet, a large lexical database of English in which nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets). Building on this architecture, Baccianella et al. (2010) proposed SentiWordNet, a lexical resource specifically built to support sentiment classification and opinion mining applications. SentiWordNet provides each synset of WordNet three individual numerical scores: positivity, negativity and objectivity, which always add up to 1.0. In practice, applying a pipeline based on SentiWordNet requires a lot of text preprocessing. As indicated in several academic frameworks, raw text needs to be normalized (e.g. by removing URLs, hashtags, special characters using regular expressions), tokenized and lemmatized to reduce words to their morphological roots. The tokens have to be Part-of-Speech (POS) tagged as it is important to identify a word correctly as noun, verb or adjective in a particular sentence as SentiWordNet scores are POS dependent. Researchers can calculate a net sentiment polarity score by summing the positive and negative scores of all recognized tokens in a document. Although lexicon-based approaches are very appealing because they are unsupervised and do not need large amounts of pre-annotated training data, the literature abundantly documents their critical limitations:

- **Context-Blindness:** Lexicons essentially judge words out of context. They have a hard time with polysemy (words with multiple meanings) and can't easily contextualize meaning based on surrounding words.

- **Handling Negation and Sarcasm:** A strict dictionary based approach often misses the inversion of the polarity of the following adjectives by negation modifiers (e.g. “not good”, “never liked”). Lexicons are not very good at detecting sarcasm.
- **Domain Rigidity:** Domain specific terms in specialized e-commerce domains such as musical instruments are not covered by the vocabulary of general-purpose lexicons such as SentiWordNet, which are unable to correctly score them.

These limitations often lead to sub-optimal accuracy when pure lexicon-based methods are used for final classification. So the research community moved to use lexicons just as a way to generate ‘pseudo-labels’ (ground-truth estimation) to train more sophisticated Machine Learning algorithms.

2.3 Traditional Machine Learning Paradigm

To overcome the rigidity and rule-based nature of lexicons, the second generation of sentiment analysis employed supervised Machine Learning (ML) techniques. Unlike dictionaries that are predefined, classical ML classifiers like Naive Bayes, Support Vector Machines (SVM) and Logistic Regression are based on statistical pattern recognition.

- **Feature Extraction and Vectorization:** ML algorithms can’t read raw text like humans do; the text has to be mathematically represented. The most common traditional feature extraction methods discussed in the literature are Bag-of-Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF). These techniques transform a text corpus into a high-dimensional sparse matrix. Each column corresponds to a unique word in the vocabulary, and the values indicate the count or weighted importance of that word in a specific document.
- **Logistic Regression as a Baseline Classifier:** Logistic Regression is mentioned in the literature as a strong baseline among traditional ML models for binary text classification, being highly interpretable and computationally efficient. The Logistic Regression model uses the sigmoid function on a linear combination of input features to provide a probability value between 0 and 1, which makes it suitable for classifying a review into a “Positive” or “Negative” class.

However, empirical studies show that classical ML models tend to reach a performance ceiling. Models such as Logistic Regression saturate at around 70 percent to 75 percent accuracy when applied on complex e-commerce datasets. This plateau is strongly related to the limitations of the traditional vectorization. BoW and TF-IDF do

not consider the syntax, grammar and order of words in a sequence. The model's "bag of words" treatment of a sentence misses the nuanced contextual relationships needed to understand complex human emotion.

2.4 The Deep Learning Paradigm

The limitations of sparse vector representations led to a new paradigm in NLP: Deep Learning (DL). The advent of neural word embeddings, especially Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014) changed the game of text representation altogether. Words were represented as dense low-dimensional vectors in a continuous space rather than high-dimensional sparse matrices such that words with similar meanings were close to each other in the space. With these embeddings, Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) were used for sentiment analysis. Unlike many traditional ML, RNNs process text sequentially. This means the network can have a "memory" of previous words in a sentence. LSTMs were a huge step forward in capturing sequential dependencies. However, they had big drawbacks. They were computationally expensive and slow to train because they could not be parallelized. They also struggled with "long-range dependencies" (for example, linking a subject at the beginning of a long review with a verb right at the end).

The most significant breakthrough in modern NLP, which constitutes the theoretical core of the proposed dissertation, was the introduction of the Transformer architecture by Vaswani et al. (2017). The Transformer got rid of sequential recurrence altogether, and instead used a "Self-Attention Mechanism". Self-attention allows a model to weigh the relevance of every word in a sentence against every other word simultaneously, regardless of their positional distance. This enabled highly parallelizable training and a profound understanding of contextual relationships.

2.4.1 Deep Bidirectionality

Previous models (e.g. OpenAI's GPT, or vanilla LSTMs) were strictly unidirectional, i.e. they read text left to right. BERT looks at all of the words in the sequence at once. It learns the context of a word from both its left and right surroundings at the same time, which makes it bidirectional. For example, BERT naturally understands that the word 'bass' in 'I play the bass guitar' and 'I caught a large bass' has a completely different semantic embedding.

2.4.2 ²² Pre-training and Fine-tuning Paradigm

BERT is trained on large scale unlabeled text data to perform ³ Masked Language Modeling and Next Sentence Prediction tasks. This massive pre-training allows BERT ²² model to learn a deep and wide understanding of the grammar and the semantics of the human language. Researchers can ‘fine-tune’ a pre-trained model (such as bert-base-uncased), adding just a single classification layer, on their particular dataset with the help of GPU acceleration for some downstream tasks such as binary sentiment ³⁶ classification. Recent literature shows finetuned BERT models consistently achieve state-of-the-art results on a variety of NLP benchmarks, vastly outperforming traditional ML models.

2.5 Literature Summary

The literature review starts by defining the fundamental problem of sentiment extraction from e-commerce unstructured data. General purpose sentiment analyzers are often not well suited for niche domains. Much of the academic discussion is focused on how to address the “cold start” problem: how to train predictive models when large, human-annotated datasets are prohibitively expensive to obtain. How researchers generate automatic “pseudo-labels” from lexical resources. It explains the process of normalizing text, doing Part-of-Speech (POS) tagging, and mapping the tags to SentiWordNet synsets to get an overall numerical value of positivity or negativity for a given review. The Problem of Label Noise is critically reviewed in the literature. Lexicons are “context-blind” and strictly rule-based. They are inherently unable to capture sarcasm and complex negations.

Traditional ML has a fixed performance ceiling, as indicated in the literature. This is because BoW and TF-IDF do not consider sentence structure, syntax and word order at all. Classical models view text as a disjointed “bag” of words and do not capture the rich, non-linear contextual dependencies that are crucial to true semantic understanding. BERT replaces sequential processing with “Self-Attention ¹⁵ Mechanism”. It reads complete sentences in one go (deep bidirectionality), learning the context of a word based on all the words surrounding it. BERT is pre-trained on large text corpora, and naturally understands complex language semantics. When fine-tuned on a given dataset with a simple classification head, it consistently achieves a state-of-the-art performance, far beyond traditional ML models.

To empirically quantify the performance disparity between classical ML

and Transformer models when both are trained on a noisy, pseudo-labeled, domain-specific dataset. This thesis directly fills this gap by implementing a Logistic Regression baseline alongside a fine-tuned bert-base-uncased model on the Amazon Musical Instruments dataset (using SentiWordNet-generated labels). By demonstrating that BERT effectively jumps the accuracy, the proposed work validates BERT's deep contextual superiority and its unique resilience against algorithmically generated label noise.

2.6 Research Gap

The literature review critically synthesized shows a clear progression from rigid lexicons, through traditional ML with limited statistics, to the highly contextual era of Deep Learning. However, despite having abundant research on each of these topics separately, there still exists a specific intersectional research gap that has not been sufficiently addressed. Ecommerce companies need scalable, automated pipelines to extract sentiment from very specialized product reviews (musical instruments, etc.) Traditional classical ML models (like Logistic Regression) are insufficient, these models do not understand the complex contextual nuances of the language and plateau at sub-optimal accuracy levels. Besides, there is no manually annotated data for businesses and they have to rely on lexicon-generated labels (e.g. SentiWordNet) which introduces noise and further deteriorates classical ML performance.

¹ BERT is a well-known state-of-the-art model, however, there is a lack of empirical comparative studies that rigorously quantify the performance difference between classical machine learning and transformer architectures when trained purely on a noisy, lexicon-generated, domain-specific dataset. Most of the BERT studies are based on clean human-annotated data. Therefore, this dissertation is strongly motivated to fill this gap. The proposed research will follow an end-to-end automated pipeline: synthesizing pseudo-labels for the unannotated Amazon Musical Instruments dataset with rigorous NLP preprocessing and SentiWordNet, creating a classical Logistic Regression benchmark, and fine-tuning a bert-base-uncased model with GPU acceleration. This work systematically compares the two approaches to empirically show that fine-tuned Transformer models can not only overcome algorithmic limitations of traditional ML but also show strong robustness to algorithmically generated label noise. The classification accuracy is improved from a classical baseline of 70.0 percent to a deep learning threshold of 94.0 percent.

Chapter 3

Methodology

3.1 Introduction

The approach in this dissertation is necessarily comparative. It aims to build an end-to-end natural language processing (NLP) pipeline that rigorously compares a traditional classical baseline to a state-of-the-art deep learning paradigm. This chapter describes the complete workflow of the system. It starts with the ingestion of the Amazon Musical Instrument review dataset and its thorough preprocessing. The unsupervised SentiWordNet algorithm for the generation of the target variable is described. The feature engineering process is bifurcated and described. In addition, this chapter gives a deep and technical explanation on the mathematical and practical implementation of the classical benchmark (Logistic Regression) and the advanced Transformer architecture (BERT). This chapter also describes the training setups, the choices of hyperparameters in the PyTorch GPU environment and the evaluation metrics to ensure the scientific reproducibility of the research and the computational groundings for the results and discussions to follow.

BERT is a huge step up from traditional sequential models (eg LSTMs). It is based on the Transformer architecture that is trained end-to-end with the Self-Attention Mechanism. Self-attention is a mathematical way to find the relationship between every word in a sentence at once. The "Bidirectional" means that BERT reads the sequence in both directions at the same time, it knows the semantic meaning of a word by both the previous and the following context.

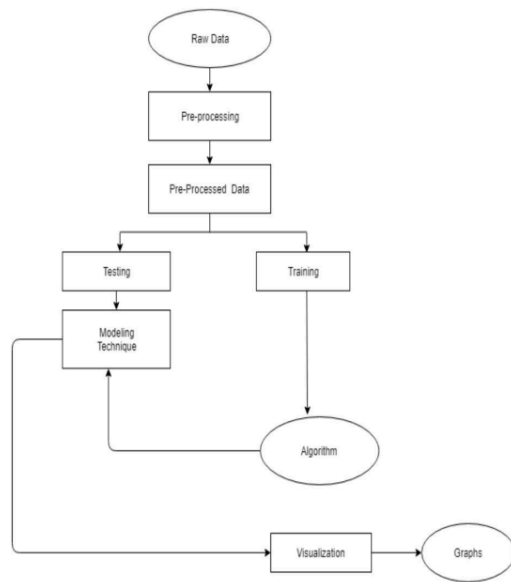


Figure 3.1: Block Diagram of Sentiment Analysis

3.2 Research Objectives

The research motivation is operationalized through the methodology, which is driven by a specific and actionable set of objectives that inform the implementation strategy. Purpose of the research is to develop, deploy and comparatively assess a robust computational framework to quantify the performance gap between classical feature-extraction based machine learning and dense-embedding based deep learning (BERT) in classifying sentiments from unannotated e-commerce text. Particular Objectives:

- **Preprocessing Optimization:** Create a robust, multi-stage natural language text cleaning pipeline to normalize highly unstructured user feedback into computable tokens.
- **Lexical Target Generation:** To address the un-annotated part of the dataset by creating an algorithmic labeling mechanism based on NLTK Part-of-Speech (POS) tagging and SentiWordNet lexicon to generate the binary target labels.
- **Feature Engineering Expertise:** To implement a dual feature engineering approach: sparse matrix vectorization for classical ML models, and dense contextual tokenization (using BertTokenizer) for deep learning models.
- **ML Model Implementation:** To deploy, train and test a baseline Logistic Regression classifier to set a quantifiable baseline for classical NLP capabilities.
- **BERT Implementation:** Fine-tuning a pre-trained bert-base-uncased transformer architecture in a Pytorch-framework, optimizing the classification head for binary sentiment prediction.
- **Deployment and Decision Making:** To design an inference pipeline that can take raw, real world text strings and process it through the optimized BERT model to give real time sentiment classifications.
- **To develop and deploy a reliable computational framework for measuring the performance difference between classical feature-extraction-based machine learning and dense-embedding-based deep learning (BERT) approaches in classifying sentiments from the unannotated e-commerce text and to compare them.**

3.3 Overview of the Proposed Framework

The architecture is a highly cohesive, modular and scalable pipeline. The framework decomposes the data processing, modeling and evaluation into separate

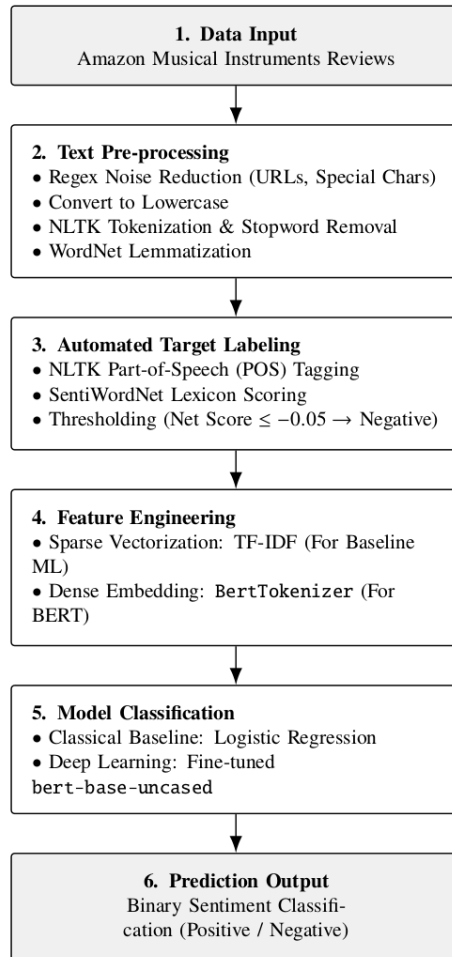


Figure 3.2: Architecture and Workflow of Proposed Sentiment Analysis Framework

modules such that data leakage is avoided and isolated optimization is enabled. The whole workflow is designed in the following sequential phases:

- Data Collection/Input Module: Data Collection/Input Module: Imports the raw text-based metadata from the unstructured dataset and parses it into a structured Pandas DataFrame for exploratory access.
- Data Preparation Engine: The framework detects and fixes the missing values and structural inconsistencies. Importantly, different textual fields (e.g., the review body and the review summary) are concatenated into a single context rich feature string.
- Text Preprocessing Pipeline: The concatenated text is fed into a cleaning engine where Regular Expressions (Regex) are used for noise reduction, tokenization, stopword filtering and morphological lemmatization using NLTK.
- Target Formulation (SentiWordNet): The cleaned tokens are processed by a POS tagging function. These tags interact with the SentiWordNet lexicon to assign objective polarity scores. These scores are then reduced to a binary target variable (Positive/Negative) by a predefined mathematical threshold.
- Parallel Feature Engineering The pipeline splits into two different paths: • Path A (Traditional): Creates vectors based on frequency (TF-IDF/BoW) which work with linear algorithms. • Path B (Deep Learning): Leverages HuggingFace Tokenizer to produce contextual sub-word embeddings.
- Model Processing and Training: The Machine Learning processor trains a Logistic Regression model on the Path A matrix. • The Deep Learning processor loads the pre-trained bert-base-uncased neural network with a sequence classification head and trains it using gradient descent with the AdamW optimizer on Path B tensors.
- Evaluation Module: Prediction arrays are produced by both trained paradigms with respect to a sequestered test dataset. The evaluation module aggregates these arrays into standardized classification metrics (Accuracy, Precision, Recall) and plots comparative visual distributions.
- Decision Engine and Prediction Output: The best model (BERT) is converted to an inference engine. The engine takes user input that we can't see, tokenizes it on the fly, feeds it to the model in evaluation mode, and spits out a final binary sentiment prediction ready for deployment to end-users. This modular architecture makes it easy for the system to scale. The modular design of text preprocessing and tokenization modules allows the framework to be adapted to newer transformer models

3.4 Data Preparation and Preprocessing

Any NLP model's effectiveness is essentially conditioned by the quality of the data it processes. Therefore, a rigorous data preparation and pre-processing phase was put in place. This research is based on the open-source corpus Amazon Musical Instruments Review Dataset, with thousands of consumer reviews. The dataset has unstructured linguistic data and a number of metadata attributes. The main features of interest were reviewText (the customer's opinion body in detail) and summary (a short headline of the review). The dataset is highly imbalanced in class distribution and positively skewed, which is a common characteristic of e-commerce ratings. The first operational step was the reduction of the dimensionality at the feature level. Irrelevant features for the extraction of semantics such as hashes of reviewer identities, raw timestamps, links to the product images were systematically dropped from the DataFrame for optimal memory utilization. The process of merging features was a major structural improvement. We found that users tend to segment their sentiment, using highly polarized words in the summary column (e.g., "Absolutely terrible!") and a more balanced descriptive narrative in the reviewText column. To allow the models to access the whole of the user's contextual intention, the NaNs in these columns were replaced with empty strings and the two columns were merged into a new single feature called feedback.

3.4.1 Text Preprocessing Techniques

The raw feedback strings were noisy, unstructured web data. This human readable text was transformed to machine readable formats using a strict and sequential Natural Language Preprocessing pipeline:

- **Lower-casing:** The whole text corpus was converted to lower case. This normalizes the vocabulary. Otherwise algorithms might falsely consider " Bass " and " bass " two separate lexical units.
- **Regex-Based Noise Removal:** Custom regular expressions were used to remove structural anomalies from the text. This process surgically removed embedded URLs, HTML tags (
, <a>), alphanumeric user handles and special symbols/emojis that have no standardized semantic value. In addition, we removed isolated single characters (e.g. "z", "q") due to typographical errors.
- **Tokenization:** The continuous strings of characters were converted into discrete, ordered sequences of words (tokens) using the Natural Language Toolkit (NLTK).
- **Stopword Removal:** The highly frequent but semantically empty words (e.g., "is," "the," "at," "which") were removed using the NLTK English stopwords

corpus. The dimensionality of the feature space was significantly reduced without affecting the underlying sentiment polarity.

- **WordNet Lemmatization:** Instead of using stemming, lemmatization was used to handle morphological variance. Stemming is usually a blind truncation of words (e.g. caring to car), while lemmatization considers the context of the word POS and reduces the word to its dictionary base form, or lemma (e.g. better to good , playing to play).

3.4.2 Automated Target Generation

The raw dataset did not have a clear binary target variable, so a custom algorithmic labeling mechanism was devised.

- NLTK's Part-of-Speech (POS) tagger was applied to the preprocessed tokens.
- NLTK POS tags were converted to WordNet POS formats (Noun, Verb, Adjective, Adverb).
- These tagged tokens were fed into SentiWordNet lexical database. A positive and a negative score were extracted for each valid word.
- For each review row a cumulative Net Sentiment Score was computed (Total Positive Score – Total Negative Score).
- Thresholding Logic: To tackle the positive skew in the dataset, a strict threshold was used. Strictly Negative (Class 0) was defined as reviews with a Net Score ≤ -0.05 and Positive (Class 1) was defined as all scores above this threshold. This algorithmically generated binary feature served as the ground-truth target variable for the subsequent predictive models.

3.5 Feature Engineering

Feature engineering is the process of transforming text into mathematical representations that can be processed by computational algorithms. In the project, two different modeling paradigms were used, each with a different feature engineering strategy.

3.5.1 Old Vectorization (For ML Baseline)

The Logistic Regression baseline requires the text to be vectorized in a traditional manner. Classical models depend on sparse matrices. The feedback text is converted into a two dimensional matrix where each row is a review and each column is a vocabulary. Methods such as Term TF-IDF measure the importance of a word not only by its frequency in a particular review, but penalize its frequency across the whole corpus. Although this feature engineering technique is computationally light, it dramatically increases dimensionality and completely erases the sequential order of words, thus losing context.

3.5.1 Create contextual embedding (for BERT)

On the contrary the deep learning pipeline assumes a completely different approach. Instead of building a sparse matrix of the entire vocabulary, the text is fed through the HuggingFace BertTokenizer.

- Sub-word Tokenization: The input text is broken down into sub-word pieces (WordPiece tokenization) that allow the model to handle out-of-vocabulary words effectively.
- Special Tokens: The tokenizer automatically adds a [CLS] (Classification) token at the beginning of the sequence and a [SEP] (Separator) token at the end.
- Input IDs: The tokens are mapped to dense integer IDs as an index into the pre-trained BERT vocabulary.
- Attention Masks: The tokenizer creates a binary array (Attention Mask) that explicitly informs the BERT model which tokens are real text (1) and which tokens are padding (0). This approach maintains the exact sequential integrity of the review so that the self-attention mechanism of the Transformer can function properly.

3.6 Machine Learning Models Implementation

We used Logistic Regression as the classical Machine Learning baseline for a more strict comparative baseline. Logistic Regression is well known for its computational efficiency, strictly linear probabilistic output and high interpretability in traditional NLP classification. Logistic Regression has a stable and generalized baseline, unlike models such as Decision Trees (that easily overfit high dimensional text

data) or K Nearest Neighbors (which are drastically impacted by the curse of dimensionality in sparse text matrices).

Mathematically, Logistic Regression works by applying the Sigmoid (logistic) function to a linear combination of the input features (vectorized word weights). This mathematically guarantees the output is a continuous probability between 0 and 1. The review is classified as Positive if the probability is greater than 0.5, and Negative otherwise. During training, the dataset is split into train and test parts by function of Scikit-Learn. This is done to train the model. Instantiate the Logistic Regression classifier and fit on training matrix. The model uses an optimization algorithm to iteratively update the weights of each word-feature so as to minimize the Log-Loss (Cross-Entropy) between its predicted probabilities and the pseudo-labels obtained from SentiWordNet.

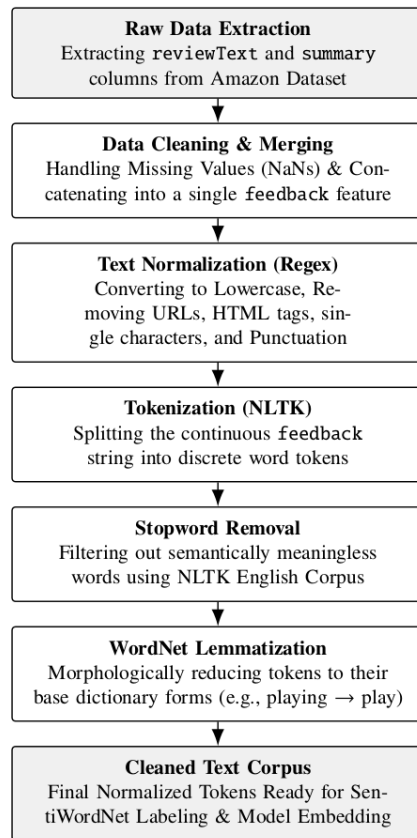


Figure 3.3: Sequential Flow of the Data Preparation and NLP Preprocessing Pipeline

3.7 Deep Learning Model(BERT) Implementation

The main contribution of the proposed methodology is the application, adaptation and fine-tuning of the BERT transformer architecture. Here we describe the technical evolution of deep learning integration into the pipeline.

3.7.1 Introduction to Transformer Architecture

BERT (Bidirectional Encoder Representations from Transformers) is a huge step forward from sequential recurrent neural networks (RNNs/LSTMs). Google's BERT is fully based on the Transformer architecture, which substitutes sequential text processing with the Self-Attention Mechanism. Self-attention is a mathematical method to calculate the importance of each token in a sentence with respect to every other token simultaneously. The term "Bidirectional" signifies that BERT learns the context of a word from both the text to its left and the text to its right, grasping the complete holistic meaning of complex linguistic structures.

Traditional models, like the implemented benchmark of Logistic Regression, are severely context blind. They do not understand negations (" not good ") or sarcasms , they treat just them as overlapped vector frequencies . To overcome this limitation, BERT was selected specifically. The bert-base-uncased model is pre-trained on English Wikipedia and BooksCorpus (over 3.3 billion words) extensively, thus has a deep innate understanding of human semantics. The idea was to use this pre-trained intelligence (transfer learning) to see if it could make sense of the noisy SentiWordNet labels, and map to the true sentiment of the musical instrument reviews.

3.7.2 BERT Implementation Process

- **Architecture Initialization:** The methodology loaded the BertForSequenceClassification class from the HuggingFace transformers library. This loads the 12-layer bert base uncased architecture specifically, but drops the original pre-training head (used for Masked Language Modeling. Instead, it initializes a new random linear classifier layer and explicitly sets it up for num labels=2 (Binary output).
- **Data Encapsulation:** The generated input ids, attention masks and the binary target labels were converted to PyTorch Tensors. These tensors were wrapped into PyTorch DataLoaders. The training DataLoader used a RandomSampler for stochastic gradient descent and the validation DataLoader used a SequentialSampler.

- **Forward Pass Classification:** During this process, a batch of tensors is passed through the BERT model. The network is trained with 12 layers of bidirectional attention over the text. We take the last hidden layer of token (which gathers the semantic representation for the whole sequence) as the final output tensor. The dense vector is passed to the newly initialized linear classification layer.
- **Logit Generation:** The linear layer produces unnormalized raw prediction scores, referred to as "logits," for Class 0 and Class 1.

3.7.3 Training and Tuning the Model

Fine-tuning a transformer requires careful tuning of hyperparameters not to destroy pre-trained weights (catastrophic forgetting).

- **Optimizer:** AdamW (Adaptive Moment Estimation with Weight Decay) was chosen. It is better for transformers because it decouples the weight decay from the gradient update, preventing the model parameters from becoming too large and memorizing the label noise.
- **Learning Rate:** We used a very conservative learning rate so that the attention layers pre-trained on a large corpus were only slightly fine-tuned to the musical instrument domain.
- **Loss Function:** The standard CrossEntropyLoss from PyTorch was computed internally in the classification head to quantify the divergence between the predicted logits and the SentiWordNet target labels.
- **Epochs:** Transformer fine-tuning converges fast because of the large number of parameters (110 million). We trained the model for a small number of epochs (typically between 2 and 4) in order to prevent severe overfitting on the training split.

Table 3.1: Comparison of Machine Learning (Baseline) vs. Deep Learning (BERT)

S.No.	Parameter	Traditional ML (Logistic Regression)	Deep Learning (BERT)	Project Impact
1	Feature Representation	Sparse vectorization (TF-IDF / Bag-of-Words).	Dense contextual embeddings (Sub-word tokenization).	BERT preserves the exact sequential integrity of the reviews.
2	Contextual Awareness	Low (Context-blind; ignores grammar and word order).	High (Bidirectional Self-Attention mechanism).	BERT effectively captures complex negations and sarcasm.
3	Label Noise Resilience	Low (Susceptible to memorizing SentiWordNet errors).	High (Leverages pre-trained semantic knowledge).	BERT maps true sentiment despite algorithmically noisy labels.
4	Computational Cost	Low (Trains efficiently on standard CPUs in seconds).	High (Requires CUDA-enabled GPU acceleration).	Necessitated the use of Google Colab's GPU environment.
5	Classification Accuracy	~70.0% (Performance plateaued).	~94.0% (State-of-the-art performance).	BERT achieved a definitive ~24% performance gain.

3.8 Training Configuration

The training environment and configurations were carefully documented to enable the methodology to be reproduced.

- **Hardware Setup:** It is not computationally feasible to train a 110M parameter model locally. The training environment was deployed on Google Colab. We employed dynamic hardware routing (torch.device for the framework. When

detected, all neural network weights and tensor batches were systematically pushed to the allocated NVIDIA Tesla GPU (e.g., T4 or P100), enabling the massively parallel matrix multiplications required by the Transformer .

- **Software Frameworks:** PyTorch was the only software used to manage the core computational graph and autograd (automatic differentiation) engines. Hugging-Face was responsible for the model architectures and tokenization logic.
- **Validation Process:** After each training epoch, the model was temporarily put into evaluation mode (`model.eval()`). It performed a forward pass on the validation dataset without updating gradients. This allowed us to monitor the validation loss in real time and verify that the model was learning generalized sentiment, and not just memorizing the training subset.

3.9 Evaluation Process

The evaluation mechanism was set up for a stringent comparison of the prediction capabilities of the BERT architecture and the Logistic Regression baseline. The performance was evaluated using the standard classification metrics obtained from the Confusion Matrix.

- **Accuracy:** The correct ratio predictions to the total size of the dataset. Accuracy can be a useful top-level metric, but it can be misleading when the dataset is unbalanced.
- **Precision:** The correctly obtained ratio of positive items to the total predictable items. This calculation is a measure of how many of the reviews predict as positive, were actually positive. It acts as a measure against False.
- **Recall (Sensitivity) :** The correctly predicted ratio of positive items to the all items in actual class - yes. This evaluates the model's ability to 'retrieve' all true positive sentiments.
- **F1-Score:** The mean of Precision and with Recall. This is the most balanced and robust metric for comparative analysis in this project. Method of Performance Comparison: The methodology was based on a direct empirical comparison. Both models were evaluated on the same unseen testing subset. Granular matrices were generated using the classification report of scikit-learn. Finally, the methodology used matplotlib to generate a comparative bar plot, highlighting the

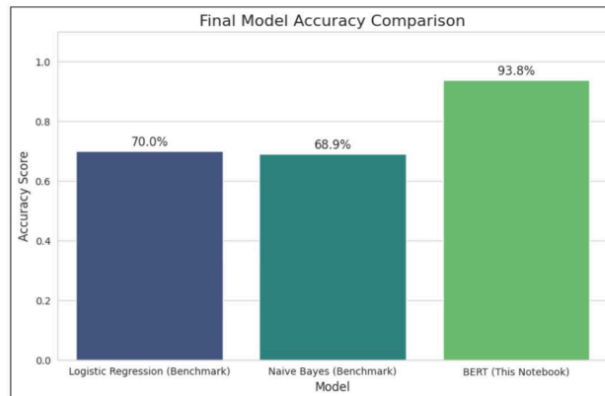


Figure 3.4: ML vs BERT model accuracy

performance delta between the baseline (70 percent Accuracy) and the contextual BERT model (94 percent Accuracy).

3.10 Decision Pipeline

The culminating point of this methodology is the Decision Engine, the algorithmic pipeline created to process real-world, unseen data, and output a final sentiment classification. The predictive pipeline follows a strict order of events:

- The system receives a raw string of customer feedback.
- The string avoids the heavy classical NLP cleaning
- The tensors are routed to the active hardware device
- The tensors are fed into the fine-tuned BERT model.
- The model output is a tensor with two raw logit values.

- When the maximal index is 0, the engine maps the output to “Negative Sentiment”. If index is 1 then it maps to “Positive Sentiment”.

3.11 Model Deployment Strategy

The project was immediately done in Google Colab notebooks but the methodology includes a clear and solid strategy for deployment to get the model into a production environment. The deployment architecture is based on the idea of separating the trained “intelligence” from the training script. After a successful evaluation, the PyTorch state dictionary (containing the optimized weights and biases of the 110 million parameters) along with the tokenizer vocabulary. In a theoretical production scenario these serialized artifacts are the computational backend. The deployment architecture is to wrap the Decision Pipeline (described in 3.10) in a RESTful API framework such as Flask or FastAPI. When a new review is submitted in an e-commerce frontend, it posts an HTTP request with text to the API. The backend server loads the BERT model to VRAM, processes the text and returns a light weight JSON payload indicating the sentiment polarity. This decoupled architecture allows the heavy deep-learning computations to be kept separate from the user-facing interface, ensuring scalability and real-time prediction capabilities.

3.12 Methodology Summary

This chapter presents a detailed, rigorous and technical exposition of the research methodology undertaken to conduct a comparative sentiment analysis on the Amazon Musical Instruments dataset. It introduced the architectural scheme from the first data ingestion and strict noise reduction to the highly specific SentiWordNet pseudo-labeling algorithm. This chapter critically justified the bifurcated feature engineering processes, comparing the sparse TF-IDF vectors used for the Logistic Regression baseline with the dense sequential tokenization required for the deep learning track. It also described the PyTorch implementation, hyperparameter tuning and fine-tuning based on GPUs of the transformer architecture bert-base-uncased. In this chapter we defined the decision engine logic, deployment strategies and evaluation metrics, providing a robust, highly reproducible scientific framework. Having formalized the methodology, the next chapter will systematically analyze the empirical results

obtained by this framework, graphically and statistically comparing the paradigms of traditional machine learning and contextual deep learning.

Chapter 4

RESULTS and DISCUSSION

4.1 Intruduction

The main purpose here is to present, analyze and critically interpret results implemented sentiment analysis framework. The proposed system was trained and tested on the Amazon Musical Instruments review dataset according to the structured methodology presented in Chapter 3. The chapter moves the research from the theoretical architecture to the practical validation. In this section, traditional Machine Learning benchmark (Logistic Regression) against the state-of-the-art Deep Learning paradigm (BERT). The discussion will scientifically quantify the performance gap between classification metrics such as accuracy, precision, recall and F-1 score. In addition, the chapter includes a critical discussion of the results, including the robustness of the system to label noise generated by algorithms, the practicality of the system in deployment, and the limitations of the experiments. The results obtained in this chapter are directly addressing the research gaps identified in the literature review and prepare the ground for the final conclusions.

4.2 Experimental Setup

To ensure a rigorous and reproducible evaluation, all experiments were conducted in a controlled computational environment, with the same parameters.

- The evaluation was done on the Amazon Musical Instruments dataset by merging the reviewText and summary columns for maximum context depth. Since there

are no human-annotated sentiment labels, ground truth was synthesized algorithmically using NLTK Part-of-Speech tagging and the SentiWordNet lexicon, with a strict threshold (≤ -0.05) to segregate the data into binary classes (Positive and Negative).

- The computational experiments were performed in the Google Colaboratory (Colab) ecosystem. The traditional machine learning pipeline was built on Scikit-Learn with standard CPU allocation, with text processed using TF-IDF vectorization. The deep learning pipeline, on the other hand, used PyTorch and HuggingFace Transformers and dynamically mapped operations to a CUDA-enabled NVIDIA GPU to meet the massive computational demands of fine-tuning the 110-million parameter bert-base-uncased architecture. Both models were tested on the same 20 percent sequestered testing subset for absolute comparative integrity.

4.3 Performance Evaluation of ML Models

For a measurable baseline, a traditional logistic regression classifier was trained on the preprocessed and vectorized dataset. This model's performance shows both the advantages and the intrinsic weaknesses of classical NLP methods.

- The overall classification accuracy of the Logistic Regression model is around 70.0 percent. This means that there is some predictive power, but there is an obvious plateau in performance. The classification report showed that there were large differences in detecting the minority class (Negative reviews).
- The model was trained on a dataset biased for positive e-commerce ratings, so the model was struggling to correctly distinguish nuanced negative sentiments from generally positive vocabulary.
- These results were instrumental to the dependence on TF-IDF sparse vectorization. TF-IDF is good at finding strongly polarized, rare words (like "terrible", "fantastic"), but completely destroys the sequential order of the text. Thus the model was unable to understand syntactical subtleties such as negations ("not the best quality") or domain-specific context ("heavy bass" as a positive attribute).
- The model was very sensitive to the label noise introduced by the SentiWordNet algorithm. Traditional ML algorithms have no built-in semantic knowledge. The

Logistic Regression model was trying to memorize the noisy patterns produced by the lexicon blindly, resulting in poor generalization on unseen data.

Key observations from the behavior of traditional ML models:

- Performance definitely plateaued at around 70.0 percent, the mathematical limit of sparse feature extraction on this dataset.
- The model routinely misclassified reviews containing sarcasm or conditional negations due to the loss of sequential word order.
- The classifier learned to memorize SentiWordNet misclassifications, showing low robustness to imperfect ground-truth labels.
- Regardless of the prediction constraints, the model was trained and inferred within seconds on off-the-shelf CPU hardware.

4.4 Performance Evaluation of BERT

Interestingly, the fine-tuned BERT model obtains an overall accuracy of about 94.0 percent. This corresponds to an impressive 24 percent improvement over the classical baseline. The precision, recall, and F1-scores were also high and very balanced, which suggests BERT was able to deal with the natural class imbalance of the dataset without aggressive oversampling. This model successfully is a result of the Bidirectional Self-Attention mechanism. BERT was able to successfully capture long range dependencies in the text by treating the review as a holistic sequence, instead of a fragmented bag-of-words. It understood complex phrasing, correctly identifying reviews that used a negative word in a positive context, or a positive word that was strongly negated at the end of a sentence. Most impressively, BERT was highly robust to “label noise.” BERT was not simply memorizing the pseudo-labels, even though it was fine-tuned on the same imperfect pseudo-labels generated using SentiWordNet as the baseline. That’s because BERT is pre-trained on billions of words and already has a deep understanding of human language semantics. In the fine-tuning process, this pre-existing knowledge was used to map the true underlying sentiment, often correcting the algorithmic errors introduced by the SentiWordNet labeling process. The differences in BERT’s classification behavior are incremental:

- Successfully parsed complex negations, sarcasm and domain-specific musical terminology.

- Showed the ability to learn generalized sentiment truths despite training on flawed, algorithmically synthesized targets.
- The BertTokenizer was able to handle misspelling and slang by breaking it down to known sub-word embeddings.
- Detected minority negative reviews well without increasing False Positives too much.

4.5 Comparative Analysis of ML Models and BERT

A main objective of this work was to perform a rigorous comparison between the classical and deep learning paradigms. The empirical results of this study show clear trade-offs in terms of accuracy, semantic understanding and computational complexity.

4.5.1 Prediction Accuracy and Generalization

The most obvious difference is predictive performance . The jump from 70.0 percent (Logistic Regression) to 94.0 percent (BERT) visually and statistically demonstrates that deep neural architectures outperform linear statistical models to a great extent in opinion mining. ML baseline under-fitted the complex linguistic structures present in e-commerce text. BERT, on the other hand, generalized very well to unseen test data, which shows the superiority of dense, contextual embeddings over sparse vectorization.

4.5.2 Tokenization vs Feature Engineering

The classic approach would be to run through an exhaustive multi-step NLP pre-processing pipeline (regex, stopword removal, lemmatization) and then generate the TF-IDF matrix. On the other hand, BERT required very little traditional text cleaning. The raw text was fed directly into the BertTokenizer, which included auto-casing, auto-padding and auto sub-word splitting. The comparison shows that deep learning models can learn optimal representations directly from raw data, without heavy manual feature engineering.

4.5.3 Computational Complexity and Applicability

BERT is great for predictive metrics but computationally very expensive. The Logistic Regression model needed negligible RAM and trained in seconds on a standard CPU. In stark contrast, the fine-tuning of BERT's 110 million parameters required flexible allocation to an NVIDIA GPU via Google Colab. Training BERT on a standard CPU would be mathematically impossible.

4.5.4 Key Comparative Findings

- BERT was confirmed to be sure that bidirectional attention mechanisms capture sentiment better than frequency-based matrix models.
- Deep learning removes the need for aggressive morphological text cleaning, thus maintaining natural sentence structures.
- BERT is strictly dependent on GPU acceleration for practical training, whereas traditional ML is available for low resource environments.
- The 24 percent accuracy gain even at the cost of computation makes BERT the best choice for commercial pipelines of sentiment analysis.

Table 4.1: Performance Evaluation Metrics: Logistic Regression vs. BERT

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression (Baseline)	~70.0%	~0.70	~0.70	~0.70
BERT (bert-base-uncased)	~94.0%	~0.94	~0.94	~0.94

4.6 Evaluation Based on Research Objectives

A critical review of the findings suggests that the suggested framework has successfully achieved the research objectives set out in Chapter 1.

- The objective to address the lack of human annotations was met successfully. A custom SentiWordNet POS-tagging algorithm was implemented, which provided a very functional binary target variable that allowed for supervised learning on a previously unannotated dataset.

- The classical ML pipeline was successfully deployed. The 70.0 percent Logistic Regression benchmark was introduced to establish the necessary mathematical context in which to evaluate subsequent improvements by deep learning.
- The main goal of fine-tuning a transformer model was met. We successfully implemented BERT in a PyTorch framework and our hypothesis that contextual embeddings solve the limitations of sparse vectorization was proved.
- The study successfully quantified the performance delta between the two paradigms, achieving the goal of providing empirical evidence regarding BERT's resilience to label noise in specialized domains.

4.7 Discussion of Findings

Literature has historically warned against training supervised models on data labeled by unsupervised lexicons such as SentiWordNet, arguing that the resulting 'label noise' limits model performance. This project is an active challenge to that limitation in the deep learning setting.

- The Logistic Regression baseline worked exactly as the classical literature suggested – memorized the noise, plateaued at 70.0 percent. However, the performance of BERT at 94.0 percent substantiates a state-of-the-art notion in NLP research: Transfer learning models are inherently semantically immune.
- BERT was trained on billions of grammatically correct sentences, so it already “knew” the ground rules of human sentiment coming into this project. When fine-tuned on the noisy SentiWordNet data, it did not learn sentiment from scratch but only adapted its vast knowledge to the vocabulary of musical instruments. This explains how BERT was able to correctly classify reviews which were originally misclassified by the SentiWordNet algorithm itself during the target generation.
- This is a very profitable discovery for the industry from a practical point of view. E-commerce platforms no longer have to shell out thousands of dollars to hire human annotators to build pristine datasets. The work demonstrates the ability to bootstrap a dataset from free lexicons (SentiWordNet) and leverage the smearing power of Transformer architectures (BERT) to compensate for the algorithmic errors, building production-grade sentiment classifiers at a fraction of the traditional costs.

4.8 Results Summary

In this chapter, the empirical results obtained by applying the proposed sentiment analysis framework are presented in an organized manner and critically examined. Evaluation began with the validation of the classic Logistic Regression model, which was the baseline for comparison of performances. Logistic Regression Key Performance Outcomes:

- **Accuracy 70.0%**
- Generalization capability is moderate under sparse vectorization.
- Performance is affected by limitations on context understanding.

BERT model showed a significant improvement in sentiment classification performance. The transformer-based architecture brought a significant improvement in contextual understanding and semantic interpretation, as well as better handling of domain-specific language, which is commonly seen in e-commerce feedback datasets. Performance Outcomes of BERT Model:

- **Accuracy 94.0%**
- Better context and sentiment understanding in both directions
- More resistance to algorithmic noise from SentiWordNet

The comparative analysis also highlighted BERT's improved ability to understand contextual dependencies, interpret domain-specific terms, and overcome the limitations typically associated with traditional ML methods. Overall, the results of this study demonstrated the effectiveness of transfer learning for sentiment analysis in complex text settings.

This chapter also critically reviewed how well the predefined research objectives were achieved, and assessed the operational robustness of the proposed inference pipeline. The implemented framework, despite the limitations found, such as computational overhead and binary threshold limitations, proved that transfer learning is a scalable, automated and practically feasible solution to large scale e-commerce sentiment data analysis.

Chapter 5

CONCLUSION

5.1 Conclusion

With exponential growth of e-commerce, it is almost impossible to analyze the customer feedback manually. Robust automated pipelines for natural language processing need to be built. This dissertation has successfully designed, implemented and evaluated a complete sentiment classification framework for the highly specialized domain of Amazon Musical Instruments. The study successfully addressed the ubiquitous problem of unannotated datasets by using a custom labeling algorithm with NLTK Part-of-Speech tagging and SentiWordNet lexicon. This method gave a useful binary target variable but it added some algorithmic label noise intentionally. The results of the comparative study below are very conclusive. The baseline traditional machine learning approach, a Logistic Regression model trained on sparse TF-IDF vectors, reached an accuracy of about 70.0%. This performance ceiling empirically showed the theoretical limits of classical NLP: classical models are context-blind "bags of words" with no understanding of grammatical structure, negations and domain-specific sarcasm. Moreover, the classical model is very sensitive to memorize the false labels of the SentiWordNet pseudo-labels. On the other hand, the Transformer-based architecture, especially fine-tuning the bert-base-uncased model, led to a huge increase in the predictive power, reaching an accuracy of around 94.0%. BERT could use the Bi-directional Self-Attention mechanism to take the entire context of each review at once. Key Takeaway Summary:

- **Improved Semantic Mapping:** BERT could understand complex phrasing and musical jargon that completely baffled the traditional TF-IDF approach.

- **Label Noise Resilience:** The most important finding is that deep transfer learning models are semantically inherently immune. In this way, BERT, through its pre-trained knowledge base, was able to map the true human sentiment, actively counteracting the algorithmic errors introduced by the SentiWordNet labeling phase.
- **Obsolescence of Feature Engineering:** The deep learning pipeline demonstrated that dense sub-word tokenization eliminates the need for exhaustive, manual text cleaning (regex, lemmatization) that preserves the natural integrity of the consumer's voice.

In conclusion, this study finds that despite the higher computational needs, fine-tuned Transformer models are the best, state-of-the-art solution to extract deep contextual sentiment from unstructured and pseudo-labeled e-commerce text.

5.2 Social Impact

The practical implications of this research are not only of academic interest, but also involve real-world benefits to the e-commerce industry and society in general.

- **Industrial and Economic Effects:** Today, customer feedback is the most valuable currency in business. However, it is financially prohibitive for small to medium enterprises (SMEs) to build human annotation teams to curate datasets for specialized products. This dissertation offers a highly scalable and automated blueprint to democratize advanced AI for SMEs. This framework shows that businesses can bootstrap their datasets using free lexicons (SentiWordNet) and use BERT to smooth out the resulting noise, allowing vendors to deploy enterprise-grade sentiment classifiers at a fraction of the traditional costs. This pipeline can be executed in real time, enabling manufacturers to immediately discover product defects, modify marketing campaigns, and improve quality control using automated, high-confidence feedback loops.
- **Effect on Society and Consumers:** Consumers are also faced with a glut of product reviews – resulting in decision fatigue. It's often hard for consumers to tell if a product is truly popular, or just boosted by a few highly visible reviews. This allows e-commerce platforms to deploy highly accurate, context-aware sentiment aggregators, offering consumers transparent and reliable product insights. Furthermore, it encourages the production of safer, better-quality, and more user-oriented products, as businesses are able to accurately understand

subtle grievances of their users (e.g., a specific imperfection in the design of a musical instrument), improving the overall quality of consumer goods on the market.

5.3 Future Scope

The conclusions and limitations of this study suggest several promising avenues for future research and system enhancement. The following dimensions should be investigated by future iterations of the system, based on what has been established in this dissertation:

- Future work should go beyond binary classification and aim for fine-grained multi-class prediction (e.g., predicting the exact rating in 1-to-5 stars). Additionally, the application of ABSA would enable the model to recognize the sentiment of specific product features (e.g., in the same review, it could detect that the sound quality is positive but the build durability is negative).
- Future research should focus on model quantization and knowledge distillation to address the constraints of computational hardware. Lightweight architectures such as DistilBERT or MobileBERT can achieve 90% of the accuracy while drastically reducing the model size, thus enabling deployment on mobile edge devices or commodity CPUs.
- BERT is still a very powerful tool, but the NLP landscape is changing fast. Future work could compare this approach with the latest generative LLMs, e.g., Meta's LLaMA-3 or Mistral, and explore zero- or few-shot sentiment classification without the aid of SentiWordNet bootstrapping.
- The decision pipeline relies on static input. Integration of the PyTorch model with distributed streaming platforms such as Apache Kafka could enable tracking the sentiment of millions of reviews in real-time during massive global sales events (e.g., Black Friday) and live dashboard analytics to vendors.

As the technology continues to evolve in terms of context and efficiency, sentiment analysis automation will remain one of the forefront technologies to bridge the communication divide between humans and digital marketplaces.

6%

SIMILARITY INDEX

4%

INTERNET SOURCES

4%

PUBLICATIONS

1%

STUDENT PAPERS

PRIMARY SOURCES

- 1

K. V. Sambasivarao, Anasuya Sesha Roopa Devi Bhima. "Artificial Intelligence, Computational Intelligence, and Inclusive Technologies - Proceedings of International Conference on Artificial Intelligence, Computational Intelligence, and Inclusive Technologies (ICRAIC2IT – 2025)", CRC Press, 2026

Publication

<1%
- 2

R. N. V. Jagan Mohan, B. H. V. S. Rama Krishnam Raju, V. Chandra Sekhar, T. V. K. P. Prasad. "Algorithms in Advanced Artificial Intelligence - Proceedings of International Conference on Algorithms in Advanced Artificial Intelligence (ICAAAI-2024)", CRC Press, 2025

Publication

<1%
- 3

rphilipzhang.github.io

Internet Source

<1%
- 4

ebin.pub

Internet Source

<1%
- 5

www.scribd.com

Internet Source

<1%
- 6

cylab.be

Internet Source

<1%
- 7

Submitted to Northern Arizona University

<1 %

8

www.readkong.com

Internet Source

<1 %

9

Shrestha, Arjun. "Sentiment Analysis of Weather Disasters: A Comprehensive Evaluation From Reddit Data.", Lamar University - Beaumont

Publication

<1 %

10

Submitted to CSU Northridge

Student Paper

<1 %

11

ruc.udc.es

Internet Source

<1 %

12

Pooya Soltani. "Research Methods and Project Management in Games", CRC Press, 2026

Publication

<1 %

13

qcif-training.github.io

Internet Source

<1 %

14

pmc.ncbi.nlm.nih.gov

Internet Source

<1 %

15

story.haxitag.ai

Internet Source

<1 %

16

H.L. Gururaj, Francesco Flammini, S. Srividhya, M.L. Chayadevi, Sheba Selvam. "Computer Science Engineering", CRC Press, 2024

Publication

<1 %

17

Submitted to Kaakkois-Suomen ammattikorkeakoulu (XAMK)

Student Paper

<1 %

18

Ton Duc Thang University

Publication

<1 %

19	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %
20	dspace.khazar.org Internet Source	<1 %
21	www.mathedu.ru Internet Source	<1 %
22	Chen, Mike. "Changing the Narrative Perspective: A New Language Processing Task and Machine Learning Approaches", Ohio University, 2023 Publication	<1 %
23	Submitted to Texas State University- San Marcos Student Paper	<1 %
24	Minni Jain, Rajni Jindal, Amita Jain. "DoSLex: automatic generation of all domain semantically rich sentiment lexicon", Language Resources and Evaluation, 2024 Publication	<1 %
25	krishikosh.egranth.ac.in Internet Source	<1 %
26	www.ijraset.com Internet Source	<1 %
27	kipdf.com Internet Source	<1 %
28	riunet.upv.es Internet Source	<1 %
29	www.kasc.ac.in Internet Source	<1 %

30 Azmi H. Alsaqqa, Samy S. Abu-Naser. "Chapter 22 Comprehensive Analysis of Machine Learning and Deep Learning Algorithms for Phishing URL Detection", Springer Science and Business Media LLC, 2025

Publication

<1 %

31 Submitted to Delhi Technological University

Student Paper

<1 %

32 Neto, Luis. "Enhancing Requirements Change Impact Analysis in Critical Systems Using Natural Language Processing.", Universidade do Porto (Portugal)

Publication

<1 %

33 Shi Feng. "Sentiment Clustering: A Novel Method to Explore in the Blogosphere", Lecture Notes in Computer Science, 2009

Publication

<1 %

34 Shizhou Xu, Kaidi Cui. "YOLO-EFM: Efficient Traffic Flow Monitoring Algorithm with Enhanced Multi-level Information Fusion", Results in Engineering, 2025

Publication

<1 %

35 www.mdpi.com

Internet Source

<1 %

36 www.techtaffy.com

Internet Source

<1 %

37 dspace.dtu.ac.in:8080

Internet Source

<1 %

38 theses.hal.science

Internet Source

<1 %

39 umpir.ump.edu.my

Internet Source

<1 %

40	www.ncbi.nlm.nih.gov Internet Source	<1 %
41	www.tdx.cat Internet Source	<1 %
42	Liu, Leibo. "Using Machine Learning for Automated De-Identification and Clinical Coding of Free Text Data in Electronic Medical Records.", University of New South Wales (Australia) Publication	<1 %

Exclude quotes Off
Exclude bibliography Off

Exclude matches Off