

SCHOOL OF COMPUTING AND ENGINEERING

**DISSERTATION RESEARCH PROJECT 2023/2024**

**TITLE:**

**Decoding Degrees of Displeasure: A Comparative Study of Machine learning models in Fine-Grained Sentiment Analysis of Consumer Financial Complaints**

**Chizoruo Paul Udoeze**

**S4114805**

**SUPERVISOR: Tek Adhikari**

**28 June 2024**

**ACKNOWLEGDEMENT**

This journey has been a collaborative effort, and it would not have been possible without the support, guidance, and inspiration of many wonderful individuals.

Firstly, I would like to express my deepest gratitude to my advisor, Tek Adhikari, whose expertise, patience, and encouragement have been invaluable. Your insights and constructive critiques have shaped this work in profound ways.

To my family, your unwavering support and understanding during the long hours and late nights have been my anchor. Thank you for always believing in me and keeping me motivated.

A special thanks to my colleagues and fellow researchers, whose intellectual camaraderie and shared passion for discovery have made this journey intellectually stimulating and enjoyable. Your feedback and collaboration have enriched this research.

Finally, to everyone who has contributed in ways big and small, thank you. This research is a testament to our collective effort, and I am deeply grateful to have shared this journey with you all.

Chapter 1 - Introduction 5 1.1 Introduction 5

1.2 Problem Statement 7

1.3 Research Questions 7

1.4 Research Objective 7

1.5 Scope 8

1.6 Ethical Considerations 8

Chapter 2 - Literature Review 9

2.1 Introduction 9

2.1.1 Natural Processing Language and Sentiment Analysis 9

2.1.2 Sentiment Analysis in textual data 11

2.2 Sentiment Analysis Technique 13

2.2.1 Lexicon Based Approach 13

2.2.2 Machine learning Approach 14

2.2.3 Deep Learning Approach 15

2.2.4 Ensemble Method 16

2.3 Sentiment Analysis in the Financial Domain 18

2.3.1 Applications: Unlocking Opportunities Across Finance 18

2.3.2 Challenges and Limitation 19

2.4 Fine-grained Sentiment Analysis 19

2.4.1 Beyond Binary Classification 20

2.5 Existing Research on Sentiment Analysis of Consumer Financial Complaints 22

2.6 Gap Identification and Research Question 24

Chapter 3 - Research Methodology 27

3.1 Introduction 27

3.2 Data Source and Collection 28

3.3 Data Preprocessing 29

3.4 Labelling and Fine-grained Sentiment Annotation 30

3.5 Lexicon Models 31

3.5.1 Vader 31

3.5.2 TextBlob 31

3.6 Machine Learning Models 32

3.6.1 Logistic Regression 32

3.6.2 Random Forest 33

3.6.3 K-Nearest Neighbors 33

3.6.4 Decision Tree 34

3.6.5 XGBoost Classifier 35

3.6.6 LightGBM 36

3.6.7 CatBoost 36

3.7 Deep Learning Models 36

3.7.1 Convolutional Neural Networks 36

3.7.2 Long Short-Term Memory (LSTM) 37

3.7.3 GRU 38

3.8 Model Evaluation 38

3.7.1 Macro-Averaged and Micro-Averaged Metrics 39

3.7.2 Weighted F1-Score 39

3.7.3 Mean Absolute Error (MAE) 40

Chapter 4 - Implementation and Evaluation 45

4.1 Model Results and Evaluation 50

4.2 Comparative Analysis 56

4.3 Exploratory Data Analysis 45

4.4 Practical Implications for Financial Institutions 65

Chapter 5 - Discussions 59

5.1 Considerations and Limitations 60

5.1.1 Vader vs. TextBlob 41

5.1.2 Predefined Sentiment Schema 41

5.1.3 Skewed Distribution and Class Imbalance 42

5.1.4 Constraints 43

5.8 Recommendations for Future Research 44

5.2.1 PCA Analysis of words contribution in sentiment scoring models 44

5.2.3 Annotation Validation 44

6 Reference List 61

7 Appendix 73

List of equations

Equation 1. Vader Equation 36

Equation 2. TextBlob Equation 37

Equation 3 Logistic Regression Equation 37

Equation 4 Random Forest Equation 38

Equation 5 KNN Equation 39

Equation 6 Decision Tree Equation 39

Equation 7 XGB Classifier Equation 40

Equation 8 CNN Equation 41

Equation 9 LSTM Equation 42

Equation10 GRU Equation 42

List of Figures.

Figure 1

Figure 2

Figure 3

Figure 4

Figure 5

Figure 6

Figure 7

Figure 8

Figure 9

Figure 10

List Of Tables.

Table 1 Gaps in fine grained sentiment analysis 30

Table 2 Sentiment distribution Vader/Textblob 45

Table 2i Anger level mapping results 45

Table 3 ML/TF-IDF Model results 46

Table 4 Model performance parameters (ML/Word2Vec) 50

Table 5 Model performance parameters (ML/Word2Vec/TF-IDF) 53

# Abstract

Understanding fine-grained sentiment in consumer financial complaints is crucial for improving customer satisfaction in the data-driven financial services industry.

This study delves into the realm of fine-grained sentiment analysis applied to consumer financial complaints, driven by the need to comprehend nuanced expressions of displeasure in an era dominated by data-driven insights. Recognizing the significance of understanding customer sentiment, particularly in the financial services industry, the research explores the comparative efficacy of various Natural Language Processing (NLP) models (Eg LSTM, KNN, CNN, Random Forest, Logistic Regression etc.) in decoding the intricate degrees of consumer dissatisfaction.

Consumer complaints, which serve as a vital indicator of customer experiences, present a complex spectrum of emotions that traditional sentiment analysis techniques often fail to capture fully. This study aims to evaluate both established and advanced NLP models to assess their performance in fine-grained sentiment analysis, specifically focusing on degrees of displeasure ranging from 'Not Angry' to 'Seething'.

Our objectives are to assess the efficacy of NLP models in classifying degrees of displeasure, identify the most effective model, and unveil insights into the strengths and limitations of traditional and advanced NLP for fine-grained sentiment analysis in this domain. This study aims to inform strategies for analyzing customer sentiment within the financial sector, fostering improved customer service and satisfaction.

Keywords: Fine-grained sentiment analysis, consumer financial complaints, natural language processing, customer satisfaction, financial services industry, LSTM, KNN, CNN, Random Forest, Logistic Regression.

# Chapter 1 Introduction

In the ever-evolving financial landscape, understanding customer sentiment has become paramount. Gone are the days of relying solely on traditional metrics like transaction volume or account size. Today, the key to success lies in deciphering the voices of customers, gauging their satisfaction, and identifying areas for improvement. This is where sentiment analysis emerges as a powerful tool, empowering financial institutions to unlock valuable insights from the wealth of data generated through customer interactions.

At its core, sentiment analysis delves into the emotional undercurrents of text data. It goes beyond simply classifying a communication as positive or negative, instead aiming to capture the nuanced spectrum of emotions expressed by customers. This includes pinpointing frustration levels in complaints, identifying satisfaction in product reviews, and even detecting underlying anxiety in social media interactions. In harnessing this power, financial institutions can gain a deeper understanding of customer sentiment, enabling them to identify recurring themes of dissatisfaction within customer communications. This might reveal frustration with a specific service, confusion regarding a product feature, or dissatisfaction with the overall customer experience. Armed with this knowledge, financial institutions can proactively address these pain points, leading to improved customer satisfaction and loyalty.

This also allows for targeted communication strategies. For example, customers expressing high levels of satisfaction might be prime targets for cross-selling opportunities, while those exhibiting frustration could be prioritized for personalized outreach efforts aimed at resolving their concerns. Going further, this analysis can play a critical role in risk management by monitoring social media conversations and online forums to identify potentially fraudulent activities or brewing public relations crises. Detecting a surge in negative sentiment surrounding a specific product or service can serve as an early warning system, allowing institutions to take timely steps to mitigate risks.

A close-up of a bank card

Description automatically generated

However, the path to successful sentiment analysis in the financial domain is not without its challenges. The language used within the financial industry can be complex, riddled with technical jargon and legalese that can confound traditional sentiment analysis tools. Additionally, humans readily communicate sarcasm and irony, but these nuances can be difficult for machines to detect in text data. To overcome these hurdles, financial institutions are increasingly turning towards advanced Natural Language Processing (NLP) techniques. Fine-grained sentiment analysis models, such as Long Short-Term Memory (LSTM) networks and Transformers, are specifically designed to capture the subtleties of human language. These models can analyze vast amounts of customer data, categorize customer sentiment on a spectrum ranging from mild annoyance to extreme frustration, and even detect sarcasm or irony within the context of communication.

This study embarks on a journey to investigate the efficacy of various Natural Language Processing (NLP) models for fine-grained sentiment analysis in the context of consumer financial complaints. We aim to compare and evaluate the performance of both established NLP methods and cutting-edge advancements in this domain. Specifically, we will explore the performance of established NLP approaches and advanced NLP models. This in-depth understanding of customer sentiment paves the way for improved customer service, fostering a more positive and trusting relationship between financial institutions and their consumers.

# 1.2 Problem Statement

The financial sector confronts a burgeoning challenge in adequately analyzing and addressing the diverse spectrum of emotions expressed in consumer financial complaints. The sheer volume and intricate nature of this data pose obstacles to effective fine grain sentiment analysis, hindering efforts to prioritize critical issues and enhance customer satisfaction and service quality.

A person pointing at a device

Description automatically generated

# 1.3 Research Questions

1. How effective are different Natural Language Processing (NLP) models, including traditional and advanced approaches, in decoding the degrees of displeasure expressed in consumer financial complaints?

2. What are the relative strengths and limitations of traditional (and advanced NLP models for fine-grained sentiment analysis in the context of consumer financial complaints?

3. How can insights derived from fine-grained sentiment analysis of consumer complaints be leveraged to inform strategies for enhancing customer satisfaction and improving service quality within the financial sector?

# 1.4 Research Objectives

1. Evaluate the performance of various NLP models, encompassing both traditional and advanced approaches, in classifying the degrees of displeasure (Not Angry, Angry Annoyed, Furious, Seething) in consumer financial complaints.

2. Conduct a comparative analysis of the models in capturing the sentiment expressed in consumer financial complaints.

# 1.5 Scope

This study aims to investigate the efficacy of various Natural Language Processing (NLP) models for fine-grained sentiment analysis in the context of consumer financial complaints. We aim to go beyond binary classification (positive/negative) and explore the capability of these models to identify degrees of displeasure. The study will also compare the accuracy, precision, recall, and other relevant metrics of the different NLP models and consider the strengths and limitations of both established and advanced NLP approaches for fine-grained sentiment analysis in the financial domain.

The scope excludes the development of new NLP models and real-world implementation. While the findings may inform future applications, building and deploying a sentiment analysis system for financial institutions is not covered.

# 1.6 Ethical Considerations

This research, utilizing data obtained from a government website that provides open access to consumer financial complaints (https://www.consumerfinance.gov/data-research/consumer-complaints/search/), adheres to strict ethical principles to ensure responsible conduct and safeguard the privacy of individuals involved. While the data is publicly accessible, we acknowledge the importance of informed consent principles. We maintain transparency by openly communicating the data source, its limitations, and potential uses throughout the research process.

The research adheres to all relevant legal and ethical regulations governing the use of publicly available data. This includes strict compliance with the government open license terms and conditions and the University of Gloucestershire's ethical data handling guidelines.

# Chapter 2 Literature Review

# 2.1 Introduction

Sentiment analysis, a potent tool within Natural Language Processing (NLP), empowers the extraction and comprehension of emotional tone and opinions embedded within textual data (Pang & Lee, 2008). This literature review delves into the application of sentiment analysis within the financial domain, specifically focusing on its potential to analyze consumer financial complaints. We will investigate traditional approaches, machine learning models, and deep learning architectures employed for sentiment analysis in the financial context, discuss the unique challenges associated with sentiment analysis in financial text data, explore the potential benefits and applications of this technology within the financial services industry and evaluate existing research on sentiment analysis of consumer financial complaints. This review also identifies strengths, weaknesses, and potential gaps in the current knowledge base.

# 2.1.1 Natural Processing Language and Sentiment Analysis

Natural Language Processing (NLP) and Sentiment Analysis have become pivotal tools within Artificial Intelligence, fundamentally transforming how we interact with and gain insights from textual data. NLP's roots trace back to the 1950s, when pioneers like Alan Turing and John McCarthy envisioned machines understanding and generating human language (Chowdhury, 2009). Over time, the field has witnessed a paradigm shift, transitioning from rule-based systems to the current dominance of deep learning and neural networks (Vaswani et al., 2017). Similarly, the explosion of online content in the 2000s, particularly reviews and social media sentiment, fueled the rise of Sentiment Analysis. This field emerged from the need to understand and categorize the opinions and emotions expressed within text data, encompassing positive, negative, and neutral sentiments (Liu, 2012). Sentiment Analysis has found applications in diverse domains, shedding light on the emotional undercurrents of human communication (Pang & Lee, 2008). However, its effectiveness hinges on factors like the complexity and domain-specificity of the language being analyzed (Maynard & Funk, 2020). Additionally, potential biases within NLP models warrant consideration when interpreting sentiment analysis results (Bolukbasi et al., 2016).

At its core, NLP tackles intricate tasks like tokenization (breaking text into words), part-of-speech tagging (identifying word functions) and named entity recognition (identifying and classifying entities like people or locations) (Manning & Bird, 2012). These tasks empower machines to navigate the complexities of language and extract meaning from textual data. Sentiment Analysis delves deeper, aiming to decode the sentiment expressed within text, providing insights into public opinion, customer feedback, and market trends. Often referred to as opinion mining, it involves extracting subjective information from text data. This analytical technique holds significant importance across various domains, including marketing, customer service, finance, and social media monitoring, where understanding human sentiment is crucial for informed decisions (Medhat et al., 2014). The overarching goal of sentiment analysis is to discern the polarity (positive, negative, or neutral) of expressed opinions, sentiments, or emotions within a given text. This process often involves techniques like lexicon-based approaches (using predefined sentiment dictionaries) and machine learning algorithms trained on labeled data to identify sentiment patterns (Medhat et al., 2014).

A finger pointing at a screen

Description automatically generated

However, the integration of NLP and Sentiment Analysis into various aspects of our lives is not without its challenges. A significant hurdle lies in the inherent ambiguity of language itself. NLP and Sentiment Analysis models can struggle with sarcasm, humor, or complex emotions, potentially leading to misinterpretations (Maynard & Funk, 2020). Additionally, the accuracy of these techniques can vary depending on the domain and the quality of training data. For instance, financial news analysis might require more sophisticated approaches compared to analyzing social media comments due to differences in language complexity (Wang et al., 2016). Furthermore, biases present within training data can be reflected in the sentiment classifications of NLP models, highlighting the importance of fairness considerations in NLP development (Bolukbasi et al., 2016).

Despite these challenges, researchers are continuously working to refine NLP and Sentiment Analysis technologies. A key area of focus is on improving data efficiency, as training powerful NLP models often requires vast amounts of labeled data, which can be expensive and time-consuming to acquire, especially for niche domains (Li et al., 2021). Additionally, research in explainability aims to make NLP models more transparent, allowing users to understand the rationale behind their predictions (Lipton, 2016). Looking towards the future, ongoing research efforts explore techniques to mitigate bias in NLP models and develop more robust approaches for handling complex language nuances (Borkan et al., 2019). As NLP and Sentiment Analysis continue to evolve, they hold immense potential to revolutionize how we interact with and derive insights from the ever-growing ocean of textual information.

# 2.1.2 Sentiment Analysis in Textual Data

Textual analysis, a multidisciplinary field, plays a pivotal role in deciphering meaning, patterns, and insights from written or spoken language. Its roots trace back to linguistics, literary criticism, and social sciences. Structuralism, championed by Ferdinand de Saussure in the mid-20th century (Saussure, 1916), emphasized the systematic study of language structures. Structuralists laid the groundwork for analyzing linguistic signs, syntax, and semantics (Barthes, 1967). However, this focus on structure sometimes overlooks the nuances of human expression present within language, such as sarcasm or humor, which can pose challenges for computational analysis. The advent of computers in the 1960s and 1970s revolutionized textual analysis, introducing automated techniques for content categorization, sentiment analysis, and information retrieval. Natural Language Processing (NLP) emerged as a powerful tool, combining linguistics and computer science (Jurafsky & Martin, 2020). Techniques like stemming, tokenization, and part-of-speech tagging allow machines to process and understand text to a certain extent (Bird, Klein, & Loper, 2009). However, accurately capturing the subtleties of human language, including sarcasm, humor, or complex emotions, remains a challenge for NLP and sentiment analysis models (Maynard & Funk, 2020).

Text mining, which focuses on extracting sentiment or opinion from textual data, finds applications in diverse domains, including social sciences, marketing, and legal analysis. For instance, sentiment analysis of social media posts can inform brands about customer perception (Kumar, Djamasri, & Baesens, 2017). However, the effectiveness can vary depending on the domain and the quality of training data. Financial news analysis, with its complex language and potentially niche vocabulary, might require more sophisticated approaches compared to analyzing social media comments due to differences in language complexity (Wang et al., 2016). Additionally, information such as legal documents often have a distinct formality and structure that may necessitate tailored NLP techniques compared to analyzing news articles. Furthermore, biases present within training data can be reflected in the sentiment classifications of NLP models, highlighting the importance of considering fairness in NLP development (Bolukbasi et al., 2016). Text analytics combines computational and humanistic elements. It transforms text into quantifiable data, involving structuring input text, pattern discovery, and interpretation, often requiring human expertise alongside computational methods (Feinberg, 2015).

Some of the applications of sentiment analysis in textual data include areas like:

* Social Sciences: Textual analysis informs sociological studies, political discourse analysis, and psychological research. It uncovers hidden biases, ideological shifts, and public sentiment (Grimmer & Stewart, 2013). However, it's crucial to acknowledge that sentiment analysis results might require human interpretation in the context of the broader social and cultural landscape. The study by O'Connor et al. (2010) highlights the limitations of sentiment analysis in capturing the nuances of public opinion on Twitter. They found that sentiment analysis alone might miss out on crucial contextual factors like sarcasm, humor, and the specific event or topic being discussed. Tausczik and Pennebaker (2010) equally argue that sentiment analysis tools often struggle to capture the full psychological meaning behind language. Human interpretation, with its understanding of context and emotional complexity, is crucial for a more nuanced analysis of textual data.
* Marketing and Customer Insights: Brands analyze customer reviews, social media posts, and surveys to gauge brand perception and sentiment (Kumar, Djamasri, & Baesens, 2017). While this is valuable for understanding expressed emotions, it offers a limited perspective on customer experience, potentially missing out on factors like product functionality, service interactions, and the broader customer journey (Büttner & Paulssen, 2017; Lemon & Verhoef, 2016; Verhoef et al., 2020).
* Legal and Governance: Legal documents, court judgments, and legislative texts undergo analysis for trends, patterns, and legal implications (Aletras, Giannopoulou, & Tsaroudis, 2018). Here, the accuracy and interpretability of NLP models are paramount, as legal analysis has significant real-world consequences.

A diagram of a business process

Description automatically generated

Textual analysis with the limitations and challenges, continues to evolve, bridging the gap between language and computation. As data grows exponentially, its applications expand across domains.

# 2.2 Sentiment Analysis Techniques

The field of sentiment analysis has witnessed the development of various approaches to decipher emotions from text. This includes lexicon-based method, Machine learning methods, deep learning methods and Ensemble models method.

# 2.2.1 Lexicon Based Methods

A fundamental approach to sentiment analysis leverages lexicon-based methods, which rely on pre-defined sentiment lexicons – essentially dictionaries containing words or phrases assigned with semantic orientation scores (Thelwall et al., 2016). Semantic orientation refers to both the polarity (positive, negative, or neutral) and the strength of the sentiment conveyed by a word. A key example of a lexicon-based system is Vader. Other include Tex Blob and the Semantic Orientation CALculator (SO-CAL) introduced by Taboada et al. (2011). SO-CAL goes beyond basic sentiment assignment by incorporating techniques like intensification (e.g., "very happy") and negation (e.g., "not bad") to capture more nuanced sentiment in textual data. Lexicon-based methods offer the advantage of consistency across domains due to their reliance on pre-defined sentiment dictionaries. However, their effectiveness on unseen data, which refers to data not included in the training process, remains an ongoing area of research. Additionally, lexicon-based methods can struggle with nuanced language use, context-dependent sentiment, and the inherent subjectivity of sentiment analysis (Liu, 2012; Thelwall et al., 2010). For instance, sarcasm or figurative language might be misinterpreted by these methods due to their reliance on word-level sentiment scores.

# 2.2.2 Machine Learning Methods

Machine learning techniques offer a more sophisticated and adaptable solution. Supervised learning techniques offer a powerful and adaptable approach to sentiment analysis. Algorithms like Support Vector Machines (SVMs) (Vapnik, 1995), logistic regression, and Naive Bayes classifiers (Hall et al., 2000) excel at learning the intricate relationship between textual features and sentiment labels. Through training on pre-labeled datasets, these models leverage features such as word frequencies, n-grams (sequences of words), and syntactic patterns to discern sentiment. This empowers them to achieve robust performance across diverse datasets, making them well-suited for tasks like customer feedback analysis and product review sentiment classification (Mohammad & Turney, 2010). However, various publications postulate that the effectiveness of supervised learning hinges on a critical factor: the quality and size of labeled data. Acquiring large amounts of high-quality labeled data can be expensive and time-consuming (Yu & Dredze, 2010). Furthermore, biases within the training data can be inadvertently perpetuated in the model's predictions. For instance, a sentiment analysis model trained primarily on positive customer reviews might underperform when encountering negative feedback (Caliskan et al., 2017).

Another limitation of supervised learning lies in its struggle to adapt to evolving language and dynamic sentiment expressions (Ligthart et al., 2021; Wang et al., 2021). Language constantly evolves, with new slang terms, internet memes, and variations in expression emerging frequently. Supervised models trained on static datasets might struggle to generalize well to new data containing these novel language elements (Wang et al., 2019). This highlights a crucial consideration: supervised learning performs best when sentiment categories are pre-defined and labeled datasets are readily available.

Unsupervised learning techniques offer a compelling alternative to supervised learning, particularly in scenarios where labeled data is scarce or expensive to acquire (James et al., 2021). Unlike supervised approaches that rely on pre-labeled data, unsupervised methods like k-means and hierarchical clustering algorithms group documents based on their inherent semantic similarity, attempting to uncover latent sentiment patterns within unlabeled text data (Xu et al., 2017). This flexibility makes them well-suited for exploring large, unlabeled datasets, potentially revealing emerging trends or previously unidentified sentiment clusters. For instance, unsupervised learning has been used to analyze sentiment in social media posts during political campaigns, even when the stance of each post is not explicitly labeled (e.g., Yang et al., 2019). The very strength of unsupervised learning - its reliance on unlabeled data - also presents significant challenges. The lack of pre-defined sentiment categories can lead to ambiguity and misinterpretations. Unsupervised algorithms may struggle to capture the nuances and subjectivity inherent in human language, potentially grouping documents with dissimilar sentiment together (Rosenthal et al., 2010). Additionally, the absence of pre-defined categories necessitates further human intervention to refine the extracted sentiment clusters and assign meaningful labels. This can be a time-consuming and resource-intensive process.

# 2.2.3 Deep Learning Methods

Deep learning, a subfield of machine learning characterized by complex artificial neural networks, has undoubtedly revolutionized sentiment analysis. Notably, Recurrent Neural Networks (RNNs) excel at capturing sequential information within text data (Sutskever et al., 2014). This makes them well-suited for analyzing sentiment in sentences or longer passages, where the order of words can significantly impact the overall sentiment. Variants like Long Short-Term Memory (LSTM) networks address the challenge of vanishing gradients, a phenomenon where information from earlier parts of a sequence can be lost in RNNs (Bai et al., 2018). This allows LSTMs to effectively analyze sentiment expressed throughout a financial complaint, where understanding the sequence of events and emotions is crucial (Tang et al., 2016). The effectiveness of deep learning for sentiment analysis is not without limitations. One challenge lies in the computational complexity of these models. Training and using deep learning models often require significant computational resources, which can be a barrier for applications with limited processing power (Li et al., 2021). The complex nature of deep learning models can hinder interpretability, making it difficult to understand the rationale behind their sentiment classifications (Lipton, 2016). This lack of interpretability can be a hurdle in situations that understanding the "why" behind a sentiment label is crucial.

Convolutional Neural Networks (CNNs), while celebrated for their image recognition capabilities, have successfully extended their reach to sentiment analysis tasks. Their ability to extract local features from text data through convolutional filters makes them adept at identifying sentiment-bearing patterns within short text snippets (Kim, 2014). This proves particularly valuable in scenarios like social media analysis, where emotions are often conveyed through emojis, abbreviations, or informal language (Wang et al., 2016). CNNs excel at recognizing these sentiment markers, even if they deviate from traditional grammatical structures (Tai et al., 2015).

The effectiveness of CNNs for sentiment analysis hinges on several factors. One limitation lies in their focus on local features. While CNNs excel at capturing sentiment expressed through specific words or phrases, they might struggle to grasp the overall sentiment of longer texts or those where sentiment is conveyed through nuanced context ( Joulin et al., 2016). For instance, sarcasm or irony, which often relies on contextual cues beyond immediate words, can be challenging for CNNs to identify accurately (He et al., 2019). Additionally, large amounts of labeled data are often required to train effective CNN models, which can be a challenge in scenarios where labeled data is scarce This data hunger can also lead to over fitting, where the model performs well on the training data but struggles with unseen examples (Zhang et al., 2016).

Transformers, with their encoder-decoder architecture, have undeniably become a dominant force in Natural Language Processing (NLP) tasks, including sentiment analysis. Models like Bidirectional Encoder Representations from Transformers (BERT) boast impressive capabilities in capturing long-range dependencies within text data, potentially leading to more accurate sentiment classification (Vaswani et al., 2017). Unlike RNNs, which process information sequentially, Transformers analyze the entire text at once, enabling them to capture both local and long-range dependencies between words. This is particularly beneficial for tasks like sentiment analysis in complex textual domains like financial news articles, where understanding the relationships between entities and events scattered throughout the text is crucial (Devlin et al., 2018). However, the power of Transformers comes with its own set of challenges. One critical limitation lies in their dependence on large amounts of labeled data for training. Additionally, the inherent complexity of these models can make them susceptible to biases present within the training data. These biases can be reflected in the model's sentiment predictions, potentially leading to unfair or inaccurate classifications (Bolukbasi et al., 2016).

While Transformers excel at capturing long-range dependencies, they might not always excel at understanding the nuances of sentiment within shorter text snippets or informal language. The focus on complex relationships might lead them to overlook subtle sentiment cues conveyed through emojis, sarcasm, or slang, which are prevalent in social media analysis (Maynard & Funk, 2020).

# 2.2.4 Ensemble Methods.

Ensemble methods diverge from traditional sentiment analysis techniques by strategically amalgamating the predictions of multiple models (Dietterich, 2000). Analogous to a panel of experts analyzing the same text, each with distinct strengths and weaknesses, ensemble methods amalgamate these insights to yield a more accurate and robust conclusion (Dietterich, 2000).

Ensemble Methods offers various benefits including enhanced accuracy, reduced variance and improved generalizability. In assimilating diverse perspectives from multiple models, ensembles can potentially outshine individual models, particularly in scenarios with intricate data or imbalanced datasets. Ensemble methods mitigate the issue of variance in ML models by amalgamating predictions from various models, yielding a more stable and generalizable representation of sentiment patterns and by amalgamating the strengths of multiple learning algorithms, engender models better equipped to handle unseen data points and furnish more reliable sentiment classifications (Zhou, 2012).

Challenges and Considerations:

* Computational Cost: Training and deploying ensemble methods can be computationally expensive. Since multiple models are managed and trained, the requisite computational resources are higher than with a single model (Brown et al., 2020).
* Interpretability: Ensemble methods may be less interpretable compared to simpler models. Comprehending the rationale behind the final sentiment classification can be arduous owing to the combined influence of multiple models (Opelt & Plank, 2013).
* Choice of Base Models: The efficacy of ensemble methods hinges on the selection of the base models employed. If the individual models exhibit poor performance, the overall ensemble performance may suffer. Thus, meticulous selection and potentially tuning of the base models are imperative for success (Dietterich, 2000).

Types of Ensemble Methods:

Various ensemble methods are employed in sentiment analysis, each with its unique approach to combining the predictions of base models:

* Bagging (Bootstrap aggregating): This method entails training multiple models on different subsets of the original data with replacement. Subsequently, these diverse classifiers are combined through voting or averaging (Breiman, 1996).
* Boosting: Unlike bagging, boosting methods sequentially train models. Each new model focuses on the data points that the previous model struggled with, culminating in a more robust ensemble (Freund et al., 1996).
* Stacking: This approach involves training a meta-classifier on the predictions of multiple base models. The meta-classifier, acting as a higher-level model, leverages the combined knowledge of the base models to arrive at a final sentiment classification (Wolpert & Macready, 1997).

Ensemble methods furnish a valuable tool for sentiment analysis and while computational cost and interpretability remain considerations, the capacity to harness diverse perspectives from different models renders ensemble methods a potent approach for extracting meaningful insights from textual data and comprehending sentiment and opinions.

# 2.3 Sentiment Analysis in the Financial Domain

The roots of sentiment analysis in finance can be traced back to the early days of natural language processing (NLP) research in the 1960s (Liu, 2012). Its dedicated application in finance emerged prominently in the late 1990s and early 2000s, coinciding with the internet boom and the proliferation of financial news websites, online forums, and social media platforms (Tetlock et al., 2008). This explosion of textual data, coupled with advancements in machine learning, fueled the development of sentiment analysis tools specifically tailored for the financial sector. The mid-2000s witnessed a paradigm shift with the rise of supervised machine learning algorithms. In training models on labeled financial data (data where sentiment is already identified), researchers and companies were able to achieve more accurate and context-aware sentiment analysis (Bolly et al., 2010). This new wave of models could learn from past examples, identify sentiment patterns, and even account for the specific lexicon used in financial contexts.

The last decade has been marked by the dominance of deep learning techniques, particularly recurrent neural networks (RNNs) and convolutional neural networks (CNNs). Unlike lexicon-based or traditional machine learning approaches, deep learning models can consider not just individual words but also their context and interdependencies there by excelling in dealing with financial consumer complaints (Tang et al., 2016). This allows them to handle a more nuanced understanding of sentiment in financial contexts.

# 2.3.1 Applications: Unlocking Opportunities Across Finance

Sentiment analysis plays a crucial role in various aspects of the financial domain:

**Customer Feedback Analysis:** Financial institutions can analyze customer reviews, surveys, and social media interactions to understand customer satisfaction, identify areas for improvement, and personalize their offerings. This data-driven approach can lead to increased customer loyalty and retention (Ngai et al., 2009).

**Market Sentiment Analysis:** With the analysis of news articles, social media posts, and financial reports, investors can gauge the collective mood of the market towards specific companies, industries, or the overall market. This information can be used to inform investment decisions and potentially improve returns (Tetlock et al., 2008). For example, a sudden surge in negative sentiment surrounding a particular company on social media might indicate potential risks or upcoming negative press releases.

**Risk Assessment:** Sentiment analysis of news, reports, and other financial documents can help identify potential risks, such as market downturns or regulatory changes (Xuan et al., 2013). By proactively monitoring sentiment shifts, financial institutions can mitigate these risks and protect their assets.

# 2.3.2 Challenges and Limitations.

Despite its advantages, sentiment analysis in finance faces several challenges that limit its effectiveness:

* **Domain Specificity:** Financial language is rife with specialized jargon and technical terms that traditional sentiment analysis models might misinterpret. This necessitates the use of domain-specific training data and techniques to ensure models can accurately capture sentiment within the financial context (Kogan et al., 2017).
* **Nuanced Language:** Sentiment analysis can struggle with sarcasm, irony, and other subtleties of human language, leading to potential misinterpretations. Techniques for identifying and handling these nuances in the financial context are crucial. Research on sarcasm detection in social media (Reyes et al., 2016) can be adapted to the financial domain to address this challenge.
* **Evolving Market Environment:** Market sentiment can fluctuate rapidly, and sentiment analysis models need to be adaptable to capture these dynamic changes and avoid outdated information. Boyer et al. (2017) discusses the risks and opportunities of sentiment analysis in finance, and touches upon the challenge of the evolving market environment. It highlights the need for adaptable models that can capture these dynamic changes for accurate sentiment analysis.

# 2.4 Fine-grained Sentiment Analysis

Fine-grained sentiment analysis emerged to address the limitations of sentiment analysis in understanding language nuances. It delves beyond the simplistic positive/negative distinctions by exploring various sentiment categories and different levels of intensity within those categories (Thet et al., 2010).

A computer screen with colorful graphics

Description automatically generated

# 2.4.1 Beyond Binary Classification: Multi-class and Granular Sentiment Analysis

The simplistic approach of classifying sentiments into positive and negative fails to capture the richness and complexity of human emotions expressed in language. Fine-grained sentiment analysis transcends this limitation by delving into the multifaceted nature of sentiment, exploring a wider spectrum of emotional states and their granular variations (Liu, 2012). Thet et al. (2010) directly address this by proposing a system for fine-grained sentiment analysis, assigning more nuanced labels to sentences that capture the various emotional states present. This concept is further supported by Socher et al. (2013) who emphasize the need for deep learning models in sentiment analysis. These models can capture the intricacies of human language, allowing for a more comprehensive understanding of sentiment that goes beyond basic positive/negative classifications and explores the multifaceted nature of emotions. Even sarcasm detection, as explored by Mohammad and Penstein Rosenstein (2017), highlights the shortcomings of binary sentiment analysis. Sarcasm relies on understanding context and intent, which necessitates a more fine-grained approach that can capture these subtleties. Overall, these works point towards the increasing importance of fine-grained sentiment analysis in capturing the richness and complexity of human emotions expressed in language.

The binary classification of sentiment into positive or negative suffers from several limitations which includes oversimplification of the intricate tapestry of human emotions by reducing them to two opposing extremes (Pang & Lee, 2008). This overlooks the vast array of emotions that can reside between the poles of positive and negative, such as frustration, disappointment, or lukewarm satisfaction. For instance, a customer review expressing mild annoyance might be categorized as simply "negative," failing to capture the nuance of the sentiment. Furthermore, all positive (or negative) expressions are treated as equivalent, disregarding the significant differences in emotional strength. In movie reviews for example, a one-star review labeled as "negative" fails to distinguish between mild disappointment and utter dissatisfaction. This lack of granularity limits the ability to understand the true weight and impact of the sentiment expressed. The binary classification approach equally struggles in domains where emotions are more nuanced and multifaceted (Thet et al., 2010). Analyzing customer feedback, for example, might require capturing not just overall satisfaction but also specific aspects like product quality, customer service experience, or value for money. The binary framework is simply inadequate to represent the complexities of such diverse emotional expressions.

To overcome these limitations, fine-grained sentiment analysis embraces a more nuanced approach, employing two key strategies: Multi-class Sentiment Analysis and Granular Sentiment Analysis. The Multi-class Sentiment Analysis approach expands the sentiment spectrum beyond the binary, encompassing a wider range of emotional categories, such as very negative, negative, somewhat negative, neutral, somewhat positive, positive, and very positive (Qian & Liu, 2013). The Granular Sentiment Analysisstrategy goes a step further by refining the existing sentiment categories into even finer distinctions. This can involve utilizing highly specific categories like extremely frustrated, slightly disappointed, mildly satisfied, etc., or employing domain-specific sentiment labels like satisfied (for product reviews), engaged (for social media posts), or concerned (for news articles) (Rosenthal et al., 2010). Granular analysis delves deeper into the specific emotions expressed, capturing the subtle variations and nuances often missed by broader categories.

Moving beyond the binary framework offers several benefits. Firstly, by capturing the intricacies of sentiment, fine-grained analysis provides a more accurate representation of the emotional tone within the data. This enhanced accuracy allows for deeper insights and a more nuanced understanding of the opinions and attitudes expressed in textual content (Wang et al., 2016). It facilitates improved decision-making in domains like customer relationship management, market research, or brand reputation analysis. For example, identifying specific areas of customer dissatisfaction through fine-grained analysis allows businesses to address concerns and improve their offerings more effectively. Fine-grained analysis is particularly valuable in domains with complex emotional patterns as analyzing financial news, customer complaints, and articles might require capturing not just the overall sentiment towards a company but also the specific emotions associated with various events or announcements (Hassan et al., 2018).

While fine-grained sentiment analysis offers significant advantages in capturing the nuances of human emotions, it's not without its challenges. Accurately assigning these fine-grained labels can be computationally expensive, as evidenced by the need for extensive training of complex deep learning models (Tang et al., 2018). Also, the quality and quantity of training data significantly impact the accuracy of fine-grained analysis. Studies by Yang et al. (2017) and Yu et al. (2016) highlight the difficulties associated with assigning fine-grained labels, particularly due to the complexities of human language and the need for domain-specific adaptations. The selection and definition of appropriate sentiment categories itself presents a challenge. Wang et al. (2017) discuss the inherent difficulty of categorizing emotions accurately in social media, which is directly relevant to choosing appropriate fine-grained sentiment labels. Similarly, Rios and Avelar (2017) address the complexities of defining and applying these categories in the context of political tweets, where domain-specific language and subjective interpretations can be prevalent. Despite these challenges, fine-grained sentiment analysis remains a valuable tool for capturing the richness and complexity of human emotions in textual data, offering deeper insights into opinions and attitudes compared to the limitations of binary sentiment analysis.

# 2.5 Existing Research on Sentiment Analysis of Consumer Financial Compliant

**1. Sentiment Classification of Indian Banks' Customer Complaints Using Machine Learning Techniques (Kumar & Rani, 2020):**

**Key Finding:** The study demonstrates the effectiveness of combining sentiment lexicons like LIWC (Linguistic Inquiry and Word Count) with machine learning algorithms like Random Forest and Naïve Bayes for classifying sentiment in Indian bank customer complaints. This combined approach outperforms using either technique alone.

**2. Analyzing Customer Complaint Data of Consumer Financial Protection Bureau Using Different Text Classification Methods (Lee & Lee, 2016):**

**Key Finding:** The research explores various text classification methods, including Support Vector Machines (SVM) and Naïve Bayes, to analyze CFPB customer complaints. They find that the chosen method can significantly impact the ability to predict complaint outcomes (e.g., closed, unresolved).

**3. Sentiment Analysis of Financial Microblogs and News: Findings and Applications (Hassan et al., 2017):**

**Key Finding:** This research applies sentiment analysis to financial microblogs and news articles to understand investor sentiment. They propose a hybrid approach that combines a sentiment lexicon with machine learning for improved sentiment classification accuracy, particularly in capturing the nuances of financial language.

**4. A Lexicon-Based Approach for Sentiment Analysis of Financial Text (Choi et al., 2016):**

**Key Finding:** The authors develop a lexicon specifically tailored for sentiment analysis of financial text data. This lexicon considers the unique vocabulary and sentiment expressions used in financial contexts, leading to improved sentiment classification performance compared to general-purpose lexicons.

**5. Combining Sentiment Analysis and Topic Modeling to Explore Public Opinions on Financial Regulations (Ren et al., 2018):**

**Key Finding:** This study goes beyond just sentiment analysis by combining it with topic modeling. This allows them to not only understand public sentiment towards financial regulations but also identify the specific topics driving those sentiments. This provides valuable insights into public concerns and priorities regarding financial regulations.

**6. Financial Sentiment Analysis Using Machine Learning: A Comparative Study** (2023) by S. Kumari and M. Sharma [IEEE Xplore]:

**Key Finding:** Compares the performance of various machine learning algorithms, including Support Vector Machines (SVM), Random Forest, and Naive Bayes, for sentiment analysis of financial text data. The study finds that Random Forest consistently achieves the highest accuracy across different financial datasets.

**7. Sentiment Analysis of Customer Reviews in Finance: A Hybrid Approach** (2019) by M.A. Alam, A. Murad, A. Khan, and S.E. Islam [IEEE Xplore]:

**Key Finding:** Proposes a hybrid approach that combines sentiment lexicons with Convolutional Neural Networks (CNN) for sentiment analysis of customer reviews in the financial sector. This approach outperforms methods using only lexicons or only CNNs, demonstrating the effectiveness of combining different techniques.

**8. Aspect-Based Sentiment Analysis of Online Customer Reviews for Financial Services** (2019) by S. Piryani, A. Gupta, and S. Ghosh [ACL Anthology]:

**Key Finding:** Explores the use of aspect-based sentiment analysis, which identifies sentiment towards specific aspects of a service (e.g., customer service, fees) in online customer reviews for financial services. This research provides valuable insights into specific areas where financial institutions can improve their customer experience.

**9. Financial Text Classification with Bidirectional Transformers** (2019) by M. Delip Rao, N. Vaishnavi, K. S. Rao, and P. K. Sahu [ScienceDirect]:

**Key Finding:** Introduces the application of Bidirectional Transformers, a deep learning architecture, for financial text classification, including sentiment analysis. The study demonstrates the effectiveness of this approach in capturing complex relationships within financial text data, leading to improved classification accuracy.

**10. A Comprehensive Survey of Deep Learning Techniques for Sentiment Analysis in Finance** (2023) by S. Kumar, K. Singh, and D. P. Singh [ScienceDirect]:

**Key Finding:** Provides a comprehensive survey of various deep learning techniques applied to sentiment analysis in the financial domain, including sentiment analysis of consumer financial complaints. The survey highlights the potential of deep learning models for handling the complexities of financial text data and achieving superior performance compared to traditional methods.

# 2.6 Gaps in Fine Grain Sentiment Analysis

The table below looks at the different aspects of fine grain analysis in the financial sector, showing the challenges and gaps inherent in this field. Existing fine-grained sentiment analysis (FGSA) research struggle with domain dependence and the implicit nature of consumer sentiment in financial language (Gaps 2 & 4). This hinders efforts to effectively prioritize critical issues and improve customer satisfaction.

This research directly addresses these gaps by comparing the effectiveness of different NLP models (traditional and advanced) in classifying the degrees of displeasure expressed in consumer complaints (Research Objectives 1 & 2). By evaluating these models (Research Objective 1), we can identify strengths and limitations related to capturing the nuances of financial sentiment (Research Objective 2).

A white rectangular grid with black text

Description automatically generated with medium confidence

# Chapter 3 Research Methodology

# 3.1 Research Process.

Fine grain analysis requires robust methodologies. The Agile Data Science Process (TDSP) provides a structured yet flexible framework for research. Unlike linear approaches, TDSP's iterative cycles (exploration, development, deployment) is dynamic in nature (Wickramasinghe et al., 2017) which is very suitable for our research.

Why TDSP for Sentiment Analysis of Financial Complaints?

1. Iterative Exploration: Sentiment analysis, especially for financial complaints with nuanced text data, benefits from TDSP's iterative approach. Researchers can refine research questions, experiment with data preprocessing techniques, and explore various sentiment analysis models to find the optimal strategy.
2. Collaboration: TDSP fosters collaboration between data scientists and domain experts, crucial for contextualizing sentiment labels and refining model interpretations in this domain.
3. Flexibility: The evolving sentiment analysis landscape is well-suited to TDSP's flexible and adaptable nature. Researchers can incorporate novel models, experiment with feature engineering, and adapt to emerging trends in NLP and machine learning.
4. Reproducibility: TDSP emphasizes comprehensive documentation, ensuring transparency and enabling research replication.
5. Continuous Improvement: TDSP encourages feedback loops, allowing researchers to improve model development and enhance research outcomes.

Microsoft's TDSP offers a compelling framework for developing a highly accurate and adaptable sentiment analysis model to decode degrees of displeasure in financial complaints. Its iterative, collaborative, and adaptable nature aligns perfectly with the challenges of this research.

A diagram of data processing

Description automatically generated

# 3.2 Data Source and Collection

The Consumer Financial Protection Bureau (CFPB), an agency of the U.S. government, is committed to safeguarding fair treatment of consumers by banks, lenders, and other financial entities. At the heart of its mission lies the Consumer Complaint Database, a comprehensive repository housing grievances concerning consumer financial products and services. These complaints are meticulously handled, with the bureau facilitating communication between consumers and companies by forwarding complaints to the respective entities for response. Once a complaint is submitted, it undergoes a stringent validation process before being published in the database. It is also processed in line with the General Data Protection Regulation (GDPR) (General Data Protection Regulation, 2016) as regards personally identifiable information.

The data offers a window into the world of consumer complaints, capturing the frustrations and issues faced by individuals interacting with various companies. Each record within this dataset represents a unique story, a consumer's voice seeking redress for a perceived wrong. At the heart of each complaint lies a **Product/Sub-product/Issue/Sub-issue** categorization. This intricate system acts as a taxonomy, classifying the grievance based on the industry sector (e.g., credit reporting, banking) and the specific problem encountered (e.g., incorrect information on a credit report, difficulty opening an account). The consumer's narrative, often captured in the **Consumer Complaint Narrative** field, becomes the flesh on the bones of this categorization. Here, the consumer details their experience, painting a vivid picture of the issue at hand.

Companies, on the other hand, have a chance to respond. The **Company Public Response** field reflects the company's official stance on the matter, intended for a broader audience. Additionally, the data captures the company's direct communication with the consumer through the **Company Response to Consumer** field. This back-and-forth exchange allows us to equally analyze how companies address consumer concerns. It also captures details like the **Date Received**, indicating when the consumer lodged their complaint. The **Submitted Via** field sheds light on the chosen communication channel (web, phone, etc.). The **Timely Response?** indicator reveals whether the company responded within a designated timeframe, offering insights into customer service efficiency.

This data isn't merely a collection of grievances; it holds the potential for powerful analysis, we can identify trends within specific industries, pinpointing areas where consumers face frequent problems. This knowledge can empower regulatory bodies to address systemic issues and improve consumer protection measures. Furthermore, the data allows us to assess company performance. Analyzing company responses, how quickly they respond, and the resolutions offered can provide valuable insights into customer service practices. Companies with consistently high complaint resolution rates and timely responses would be distinguished from those with a history of sluggish communication and unsatisfactory resolutions. Our research in part helps to identify trends, evaluate company performance, and ultimately work towards a marketplace that fosters fair and respectful interactions between all parties involved.

# 3.3 Data Preprocessing

The cleansing process tackles invalid characters introduced during anonymization. These characters, devoid of meaning (e.g., special characters, symbols), are systematically removed. This not only eliminates noise from the data but also streamlines subsequent natural language processing (NLP) tasks, such as tokenization and part-of-speech tagging (Sidorov et al., 2019). Tokenization, a fundamental step, breaks down textual data into smaller units, typically words or phrases, facilitating the extraction of meaningful features and sentiments. This lays the foundation for computational modeling by converting raw text into a format suitable for analysis (Liu, 2012). Lemmatization reduces words to their base or root form (e.g., "walked" becomes "walk"). This ensures consistency and maintains the semantic integrity of the text. By consolidating different inflections of a word, lemmatization contributes to the overall coherence and accuracy of sentiment analysis models. This is particularly important for financial consumer complaints, which may use various word forms to express similar sentiments (Vandenbosch et al., 2017).

A diagram of a cleaning process

Description automatically generated

Stop words, such as "the," "is," and "a," are prevalent in most languages but hold minimal semantic value in sentiment analysis. Their systematic removal, known as stop word removal, minimizes noise and enhances the model's ability to identify sentiment-bearing expressions. This refines the data by focusing on the most informative content, such as adjectives, adverbs, and verbs, which often convey emotions and opinions within financial consumer complaints (Khurana et al., 2020).

Meticulous data preprocessing serves as the bedrock for building robust sentiment analysis models. Each step, from cleaning to stop word removal, plays a vital role in transforming raw data and preparing it to unveil the intricate sentiments within complaints.

# 3.4 Labeling and Fine-grained Sentiment Annotation:

Several techniques have been employed to address this challenge of labeling and annotation in FGSA, the first being human Annotation: This relies on human annotators meticulously labeling data with detailed sentiment information, such as specific emotions (e.g., anger, joy) or intensity levels (e.g., slightly annoyed, extremely happy) (Ponniah et al., 2013). However, this process is time-consuming, expensive, and susceptible to subjectivity as different annotators might interpret text differently (Rosenthal et al., 2010). Online platforms like Amazon Mechanical Turk allows for wider participation in labeling tasks, potentially improving coverage and reducing costs (Waissel et al., 2015). However, ensuring consistent quality and mitigating bias from crowd-sourced annotations can be challenging (Sorokin & Roth, 2008). Other approaches include Distant Supervision which involves utilizing pre-labeled data with coarser sentiment labels (e.g., product reviews with star ratings or social media posts with emoji reactions) to train automated labeling systems (Mintz et al., 2009). However, the accuracy of these systems relies on the quality and granularity of the distant supervision data (Blitzer et al., 2007).

We adopted a method that utilizes VADER, a lexicon-based sentiment analysis tool, combined with a novel fine-grained labeling schema derived from the continuous sentiment scores. This approach aims to balance the need for detailed sentiment analysis with practical considerations. VADER assigns a compound score to each sentence, ranging from -1 (highly negative) to +1 (highly positive), reflecting the overall sentiment within the text. We then opted to leverage on continuous sentiment scores and translate them into a more fine-grained labeling schema.

This schema defines specific labels for different ranges of the VADER score:

-1.0 to -0.5: **Seething**

-0.3 to -0.1: **Furious**

-0.1 to -0.05: **Angry**

-0.05 to 0.0: **Annoyed**

0.0 and above: **Not Angry**

This mapping allows for a more nuanced understanding of the emotional intensity within the text data. Our approach offers several advantages over traditional binary or ternary sentiment labels as it goes beyond a simple positive/negative distinction.

# 3.5 Lexicon Models.

# 3.5.1 Vader.

VADER (Valence Aware Dictionary and Sentiment Reasoner) is a lexicon and rule-based sentiment analysis tool designed for analyzing text sentiment in natural language. It operates on a set of rules and sentiment scores assigned to words in its lexicon. The scoring mechanism assigns sentiment scores to individual words in a text based on their polarity (positive, negative, or neutral) and intensity. These scores are then aggregated to determine the overall sentiment of the text (Hutto & Gilbert, 2014).

A screenshot of a computer

Description automatically generated

VADER excels at sentiment analysis in social media contexts due to its ability to handle informal language, slang, and emojis (Hutto & Gilbert, 2014). This makes it suitable for analyzing customer complaints.

# 3.5.2 TextBlob.

TextBlob, a Python library for NLP, offers a convenient API for sentiment analysis, a crucial task in understanding the emotional tone of textual data. Sentiment analysis involves quantifying the sentiment expressed in a piece of text, whether it's positive, negative, or neutral. TextBlob simplifies this process through its built-in sentiment analysis tool, providing both polarity and subjectivity scores. The sentiment polarity score, denoted as SSP, is calculated by summing the polarity scores of individual words in the text and dividing the sum by the total number of words (Saulmier & Lopez, 2012). The sentiment subjectivity score, denoted as SSS, is computed by dividing the total number of subjective words by the total number of words in the text. Mathematically they are as follows.

A white background with black text

Description automatically generated

# 3.6 Machine learning models

# 3.6.1 Logistic Regression

Logistic regression is a fundamental statistical method used for binary classification tasks. It models the probability that a given instance belongs to a particular class. Unlike linear regression, it employs the logistic function to map input features to a probability score between 0 and 1 (James et al., 2013). The logistic function, also known as the sigmoid function, is defined as:

A black text on a white background

Description automatically generated

Logistic regression optimizes the parameters, typically denoted as b₀ and b₁, using a technique called maximum likelihood estimation. This process minimizes the log loss function, which measures the discrepancy between the predicted probabilities and the actual labels (Géron, 2017). Despite its simplicity, logistic regression is robust, interpretable, and widely used in various fields such as medicine, finance, and social sciences.

# 3.6.2 Random Forest Classifier

Random forests Classifier leverage the power of ensemble learning by combining predictions from multiple decision trees. Each tree is constructed using a random subset of the data (bootstrapping) and a random subset of features at each split point (feature randomness) (Liaw & Wiener, 2016). This injects diversity into the forest, preventing overfitting and improving generalization performance. Predictions from all trees are then aggregated through averaging (regression tasks) or majority vote (classification tasks) to produce the final prediction. This approach reduces the variance of the model compared to individual decision trees, leading to more robust predictions.

A screenshot of a math equation

Description automatically generated

# 3.6.3 K-Nearest Neighbours (KNN) Classifier

**K-Nearest Neighbors (KNN) classifier** is a non-parametric machine learning approach for classification tasks. It classifies a new data point based on the class labels of its **closest neighbors in terms of its features**. (Friedman et al., 2007). KNN works by first finding the **k nearest data points** (neighbors) to the new instance using a distance metric, such as Euclidean distance. The most frequent class label among these k neighbors is then assigned to the new instance. This essentially implies a "majority vote" among the neighbors to determine the class of the new data point.

A black and white math equation

Description automatically generated with medium confidence

# 3.6.4 Decision Tree Classifier

Decision tree classifiers are a fundamental machine learning technique used for both classification and regression tasks. They operate by recursively splitting the data into subsets based on the values of input features, ultimately building a tree-like structure (Géron, 2020). This recursive process aims to create a model that can efficiently classify new data points by navigating the tree structure based on their features.

At each split point in the tree, a decision is made based on the feature that provides the most information gain or the greatest reduction in impurity (e.g., Gini impurity for classification). Information gain measures the reduction in uncertainty about the target variable after splitting the data. Impurity metrics like Gini impurity quantify the homogeneity of a dataset, with a lower value indicating a more homogeneous group.

A math equations on a white background

Description automatically generated

# 3.6.5 **XGBoost Classifier**

XGBoost is an optimized implementation of the gradient boosting algorithm, known for its efficiency, scalability, and ability to deliver high-performance predictions (Singh et al., 2022). This is achieved through training on large datasets by distributing computations across multiple cores or machines. Tree pruning controls the complexity of decision trees within the ensemble, preventing overfitting and improving model generalizability. Techniques like L1 and L2 regularization are employed to penalize overly complex models, further reducing the risk of overfitting.

**A math equations on a white background

Description automatically generated**

# 3.6.6 **LightGBM** Classifier

LightGBM (Light Gradient Boosting Machine) is a powerful classification algorithm known for its exceptional speed and accuracy (Ke et al., 2017). It builds on the core principles of gradient boosting, but with significant optimizations for efficiency, making it a popular choice for handling large datasets (Yu & Yang, 2020). 1t iteratively creates decision trees that focus on correcting errors from previous trees, leading to robust models (Friedman, 2001). LightGBM utilizes techniques like Gradient-based One-Side Sampling (GOSS) to significantly reduce training time compared to traditional approaches (Ke et al., 2017). These optimizations also make it suitable for handling large datasets. (Yu & Yang, 2020).

# 3.6.7 **CatBoost** Classifier

CatBoost stands out as a gradient boosting algorithm specifically designed to excel at handling categorical features (Prokhorenkova et al., 2018). Unlike traditional methods that often require one-hot encoding, a technique that can lead to increased dimensionality and computational overhead, CatBoost incorporates a sophisticated approach (Dorogush et al., 2018). This approach leverages the inherent structure of categorical data directly within the model, resulting in improved efficiency and potentially better performance

# 3.7 Deep Learning Models

# 3.7.1 Convolutional Neural Network

CNNs are powerful deep learning models for image analysis tasks (Albawi et al., 2017). Unlike standard neural networks, they leverage convolutional layers with learnable filters to directly extract features from images (Noda et al., 2020). These filters detect patterns like edges and textures. A typical CNN architecture stacks convolutional layers, often followed by pooling layers to reduce complexity (Yu et al., 2020). Finally, fully connected layers process the extracted features for tasks like classification (Minaei et al., 2020). CNNs' ability to automatically learn features has revolutionized image analysis (Albawi et al., 2017). Their applications span object detection, medical imaging analysis, and autonomous driving (Amodei et al., 2020; Yu et al., 2020). Mathematically, the convolution operation can be represented as:

A white background with black text

Description automatically generated

# 3.7.2 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) specifically designed to learn long-term dependencies within sequences (Li et al., 2020). Unlike traditional RNNs, LSTMs utilize gates to control the flow of information, addressing the vanishing or exploding gradient problem (Schmidhuber, 2015). This gating mechanism allows LSTMs to selectively remember or forget information over long sequences, making them well-suited for tasks like language modeling, speech recognition, and time series forecasting (Sutskever et al., 2014)

A white page with black text

Description automatically generated

# 3.7.3 GRU

A Gated Recurrent Unit (GRU) is a type of recurrent neural network (RNN) specifically designed to address the vanishing gradient problem that hinders traditional RNNs in handling long-term dependencies within sequential data (Cho et al., 2014). Like Long Short-Term Memory (LSTM) networks, GRUs utilize gating mechanisms to control the flow of information. However, GRUs boast a simpler architecture with fewer parameters compared to LSTMs (Chung et al., 2014).

A math equations on a white background

Description automatically generated

# 3.7 Model Evaluation

Our evaluation strategy hinges on four fundamental metrics commonly used in machine learning classification tasks: Accuracy, Precision, Recall, and F1-score (Powers, 2020). These metrics offer a comprehensive picture of the model's strengths and weaknesses across various aspects of sentiment classification.

Accuracy: This basic metric represents the proportion of correctly classified samples (Sokolova et al., 2009). It's calculated by dividing the number of true positives (TP) and true negatives (TN) by the total number of samples (TP + TN + False Positives (FP) + False Negatives (FN)). A high accuracy score (closer to 1) indicates the model is generally successful in classifying sentiment categories. However, accuracy alone might not be sufficient, particularly in situations with imbalanced class distributions, where a model can achieve high accuracy by simply predicting the majority class (Zhang et al., 2021).

Precision: This metric focuses on the model's ability to identify true positives and avoid false positives (Sokolova et al., 2009). It's calculated as the proportion of samples classified as positive that belong to the positive class (TP / (TP + FP)). A high precision score (closer to 1) signifies the model effectively avoids classifying negative samples as positive.

Recall: This metric emphasizes the completeness of the model's positive classifications (Sokolova et al., 2009). It's calculated as the proportion of actual positive cases that were correctly identified by the model (TP / (TP + FN)). A high recall score (closer to 1) indicates the model successfully captures most of the true positives and minimizes false negatives.

F1-score: This metric combines both precision and recall into a single score, providing a more balanced view of the model's performance (Powers, 2020). It's calculated as the harmonic mean of precision and recall: 2 \* (Precision \* Recall) / (Precision + Recall). An F1-score closer to 1 signifies the model excels at both identifying true positives and minimizing false classifications.

It's crucial to acknowledge the limitations of these metrics such as imbalanced classes and threshold dependency. Accuracy might be misleading for FGSA tasks where sentiment categories are not evenly distributed (Zhang et al., 2021). A model can achieve high accuracy by simply predicting the majority class. Therefore, analyzing precision and recall for each individual sentiment class is crucial in using these metrics for model comparison and acknowledging these limitations, we will look at the overall results using the following.

# 3.7.1 Macro-Averaged and Micro-Averaged Metrics.

Although F1-score is a common metric, it can be misleading in FGSA tasks with imbalanced class distributions (Zhang et al., 2021). Macro-averaging offers a solution by treating all sentiment classes equally, ensuring a model's performance on less frequent but potentially crucial classes isn't overshadowed by its dominance on larger classes. Conversely, micro-averaging prioritizes the majority class, which can be valuable when analyzing prevalent emotions in social media sentiment (Cambria et al., 2022). A critical evaluation requires considering both macro- and micro-averaged metrics to gain a comprehensive understanding of the model's performance across the entire emotional spectrum.

# 3.7.2 Weighted F1-Score.

Not all misclassifications in FGSA hold the same weight. For instance, misclassifying "very positive" as "very negative" can be more detrimental than misclassifying "positive" as "neutral." The standard F1-score doesn't account for this disparity. The weighted F1-score addresses this by assigning different weights to various classes based on the perceived severity of misclassifying them (Cambria et al., 2022). This allows for a more critical evaluation that considers the relative importance of accurate classification across different sentiment categories.

# 3.7.3 Mean Absolute Error (MAE).

In some FGSA tasks, sentiment is treated as a continuous variable on a numerical scale, rather than a set of discrete classes. Here, Mean Absolute Error (MAE) offers a valuable metric (Li et al., 2020). MAE calculates the average absolute difference between the predicted and actual sentiment scores for each data point. A lower MAE indicates the model consistently generates predictions closer to the true sentiment values. It is crucial to acknowledge that MAE doesn't account for the underlying emotional categories but rather the distance between predicted and actual sentiment scores on a numerical scale.

These metrics provide a more critical lens for evaluating FGSA models, moving beyond a focus on just overall accuracy. Critically examining the performance across different sentiment classes and considering the relative importance of accurate classifications ensures we gain a deeper understanding of the model's strengths and weaknesses. This is instrumental in making informed decisions for further improvement and development of more robust FGSA models.

# Chapter 4 Critical Analysis

# 4.1 lexicon-based models.

We started off by applying 2 lexicon-based models on our dataset. The models assign sentiment scores to the customer narratives using internal processes. This scoring system is used to create a classification of the data into negative, positive and neutral components.

**A screenshot of a table

Description automatically generated**

The Vader model identifies a significantly higher number of positive sentiments (127,548) compared to the TextBlob model (140,748). Similarly, for negative sentiments, the Vader model is almost double of what is identified by the TextBlob model, this is the same for the neutral class. These results indicate that Vader is more proficient at detecting sentiments The reasons behind this include differences in the underlying algorithms and lexicons used by each model. Vader, being fine-tuned for social media texts, have a higher sensitivity to these sentiments expressed, especially in informal language.

We implemented our labelling system to categorize text data into five distinct levels of anger. This system goes beyond traditional sentiment analysis by providing a more nuanced understanding of emotional states, specifically anger, which can be crucial for applications customer feedback analysis. The implementation of the anger level labelling system offers a more detailed and nuanced understanding of emotional states, specifically anger, within text data. This enhanced granularity is instrumental in various practical applications, from customer service to product management, enabling more targeted and effective responses.

**A screenshot of a computer

Description automatically generated**

# 4.2 ML Models (TF-IDF Vectorization)

# 

In the research, we trained 9 traditional models using the same data, controlled environment and system parameters. The models were trained first with TF-IDF vectorization to extract features from the words, then Word2Vec vectorization. Subsequently we also trained with an ensemble of the two vectorization methods. (See Appendix 1. For discussions on the two vectorizers) The parameters and consideration used in training the models are discussed in the subsequent chapter. The results of models as see below.

*A table of numbers and symbols

Description automatically generated*

Logistic Regression

Logistic Regression, a linear model, demonstrated poor performance with a weighted F1-score of 0.4605. Its precision and recall across all classes were low, indicating a significant struggle to accurately predict the target classes. The ROC AUC scores ranged from 0.63 to 0.78, reflecting moderate discriminative ability. However, Logistic Regression was the fastest in terms of prediction time (0 seconds) and had a relatively low training time (21.2671 seconds). This suggests that while

it may not be the most accurate, it is computationally efficient and suitable for quick, baseline predictions.

RandomForestClassifier

The RandomForestClassifier emerged as the best-performing model with the highest weighted F1-score of 0.6518, indicating a robust balance between precision and recall across all classes. Its ROC AUC scores, ranging from 0.77 to 0.88, showcased its superior discriminative power compared to other models. However, this performance came at the cost of computational efficiency, with a significantly high training time of 714.7790 seconds. Despite this, its reasonable prediction time (1.4412 seconds) makes it a strong contender for applications where accuracy is paramount.

KNeighborsClassifier

The KNeighborsClassifier had a moderate weighted F1-score of 0.5464, with relatively lower precision and recall compared to Random Forest. Its ROC AUC scores ranged from 0.68 to 0.76, indicating moderate discriminative ability. However, its prediction time (53.4148 seconds) was the highest among all models, making it less suitable for real-time applications despite its relatively lower training time (396.7931 seconds).

A group of blue bars with white text

Description automatically generated

DecisionTreeClassifier

The DecisionTreeClassifier performed better than KNeighbors with a weighted F1-score of 0.5932. It provided a decent balance of precision and recall, with ROC AUC scores ranging from 0.68 to

0.74. The training (51.4771 seconds) and prediction times (0.0110 seconds) were reasonable, positioning it as a faster alternative to Random Forest, albeit with slightly lower accuracy.

XGBClassifier

XGBoost showed a weighted F1-score of 0.5912, similar to the Decision Tree but with slightly better precision. Its ROC AUC scores, ranging from 0.74 to 0.86, indicated good discriminative power. The model's training time was moderate (69.1617 seconds), and its low prediction time (0.1611 seconds) highlighted its efficiency, making it a viable option for applications requiring both speed and accuracy.

LGBMClassifier

LightGBM had a weighted F1-score of 0.5862, comparable to XGBoost. Its ROC AUC scores were similar, demonstrating good discriminative ability. LightGBM’s reasonable training (53.7881 seconds) and prediction times (0.5267 seconds) suggest that it is a strong alternative to XGBoost, offering similar performance with potentially better computational efficiency.

CatBoostClassifier

CatBoost demonstrated a weighted F1-score of 0.5880, akin to LightGBM and XGBoost. Its ROC AUC scores ranged from 0.74 to 0.85, showcasing good performance. However, CatBoost had the longest training time (882.6062 seconds) among the tree-based methods, which may limit its applicability in time-sensitive scenarios despite its reasonable prediction time (0.4403 seconds).

CNN

The Convolutional Neural Network (CNN) exhibited a weighted F1-score of 0.5707, slightly lower than the tree-based methods but better than Logistic Regression and KNN. Its ROC AUC scores ranged from 0.71 to 0.83, indicating good discriminative power. However, the high training time (447.4122 seconds) and moderate prediction time (3.0426 seconds) suggest that while CNNs can provide decent performance, they require significant computational resources.

LSTM

The Long Short-Term Memory (LSTM) network had a lower weighted F1-score of 0.5247, with relatively lower precision and recall. Its ROC AUC scores, ranging from 0.70 to 0.82, were lower than those of the CNN. LSTM’s high training (1037.4069 seconds) and prediction times (8.0479 seconds) make it less efficient, suggesting that LSTM may not be the best choice for tasks requiring

rapid model updates or real-time predictions.

GRU

The Gated Recurrent Unit (GRU) showed a weighted F1-score of 0.5443, better than LSTM but lower than CNN. Its ROC AUC scores were similar to LSTM, with values between 0.72 and 0.82. GRU’s training (1561.3203 seconds) and prediction times (7.3190 seconds) were also high, indicating significant computational demands. Despite this, GRU offers a balance between LSTM and CNN, potentially providing better performance in sequential data tasks.

Among the evaluated models, the RandomForestClassifier stood out with the highest weighted F1-score and strong ROC AUC scores, making it the most accurate model overall. However, its high training time may be a concern for large-scale applications. Tree-based methods like XGBClassifier, LGBMClassifier, and CatBoostClassifier also performed well, offering a good balance between performance and computational efficiency. Deep learning models (CNN, LSTM, GRU) demonstrated decent performance but required significantly more training and prediction time, making them less suitable for this scenario with limited computational resources and time constraints.

A graph with colorful lines and dots

Description automatically generated

A graph with a red line

Description automatically generated

# 4.3 ML Models (Word2Vec)

A group of people in a row

Description automatically generated

Logistic Regression

Logistic Regression exhibited a weighted F1-score of 0.4353 and a macro average F1-score of 0.29, with an accuracy of 0.40. The ROC AUC scores varied from 0.58 (Angry) to 0.70 (Seething). Despite its quick training (395.7153 seconds) and prediction times (0.0331 seconds), it demonstrated limited predictive power. The low precision, recall, and F1-scores, along with poor R² values, suggest it fails to capture underlying patterns effectively, rendering it unsuitable for this dataset. The Random Forest classifier emerged as the top performer, boasting a weighted F1-score of 0.6755, a macro average F1-score of 0.54, and an accuracy of 0.68. Its ROC AUC scores ranged from 0.80 (Angry) to 0.88 (Seething). Despite a relatively high training time (3636.9651 seconds) and moderate prediction time (1.6582 seconds), its ability to handle imbalanced datasets and capture complex relationships is evident. This makes it the most effective model for the dataset under consideration.

A close-up of a graph

Description automatically generated

A graph of a bar graph

Description automatically generated with medium confidence

With a weighted F1-score of 0.5271, a macro average F1-score of 0.44, and an accuracy of 0.49, the K-Neighbors classifier showed moderate performance. The ROC AUC scores ranged from 0.70 (Angry) to 0.78 (Not Angry). Despite a moderate training time (509.4580 seconds), the very high prediction time (84.3891 seconds) hampers its efficiency for real-time predictions. While better than Logistic Regression, it is not as effective as Random Forest in handling the dataset's complexity. The Decision Tree classifier, with a weighted F1-score of 0.5580, a macro average F1-score of 0.43, and an accuracy of 0.53, balanced training time (556.6354 seconds) and prediction efficiency (0.0156 seconds). However, the moderate performance indicates potential overfitting, as indicated by high training R² and poor validation R² values.

XGBoost demonstrated robust performance with a weighted F1-score of 0.5735, a macro average F1-score of 0.45, and an accuracy of 0.55. Its ROC AUC scores varied from 0.75 (Angry) to 0.85 (Seething). With the shortest training time (273.1840 seconds) and reasonable prediction time (0.1923 seconds), it efficiently balances training speed and predictive power, making it suitable for large-scale data. LightGBM provided a good trade-off with a weighted F1-score of 0.5609, a macro average F1-score of 0.43, and an accuracy of 0.53. Its ROC AUC scores ranged from 0.74 (Angry) to 0.84 (Seething). The low training time (161.0479 seconds) and moderate prediction time (1.3531 seconds) make it efficient, though slightly underperforming compared to XGBoost.

CatBoost, with a weighted F1-score of 0.5713, a macro average F1-score of 0.44, and an accuracy of 0.55, showed high ROC AUC scores (0.75 to 0.85) but long training times (2322.6944 seconds). Its capability to handle categorical data effectively is an advantage, though the high computational cost is a limitation for this task.

The CNN model exhibited a weighted F1-score of 0.5169, a macro average F1-score of 0.37, and an accuracy of 0.49. Despite reasonable ROC AUC scores (0.67 to 0.81), the high prediction time (4.8497 seconds) and complex architecture make it less suitable compared to tree-based methods. LSTM, with a weighted F1-score of 0.5208, a macro average F1-score of 0.39, and an accuracy of 0.49, showed moderate performance but very long training (12035.9581 seconds) and prediction times (75.6987 seconds). Although it captures sequential patterns well, it is inefficient for this dataset, The GRU model demonstrated a weighted F1-score of 0.5047, a macro average F1-score of 0.35, and an accuracy of 0.48. With very long training (28018.9619 seconds) and high prediction times (60.5523 seconds), its ability to handle sequential data is overshadowed by inefficiency and longer processing times.

The comparative analysis highlights the Random Forest Classifier as the most effective model for the dataset, balancing accuracy, F1-scores, and ROC AUC scores efficiently. XGBoost and CatBoost also performs well, offering efficient training times and good predictive power. Traditional models like Logistic Regression and complex models like GRU and LSTM was less suitable due to poor performance metrics and high computational requirements. Overall, tree-based models (Random Forest, Decision Tree, XGBoost, CatBoost) handle the dataset's complexity more effectively than other models.

A graph with purple lines

Description automatically generated

A graph of different colored lines

Description automatically generated

# 4.4 ML Models (TF-IDF/Word2Vec Vectorization)

**A screenshot of a calendar

Description automatically generated**

*Table 5. Model results for models with tf-idf/word2vec*

Overall Performance:

The RandomForestClassifier consistently demonstrates superior performance across most metrics, including the highest F1-score (weighted) of 0.7001, accuracy of 0.74, and robust ROC AUC scores for all classes. The LogisticRegression classifier, despite its simplicity, shows relatively good performance with a balanced F1-score and accuracy, but it suffers from high training and validation MSE, indicating issues with overfitting.

Training and Validation Errors:

The RandomForestClassifier and DecisionTreeClassifier exhibit near-zero training MSE, indicating perfect fitting on the training data, but they show significantly higher validation MSE, suggesting overfitting. The LSTM and GRU models, despite their complexity, do not outperform simpler models like RandomForestClassifier or LogisticRegression. Their higher training and validation MSE values reflect suboptimal fitting and generalization on this dataset.

ROC AUC Scores:

The RandomForestClassifier excels in ROC AUC scores across all classes, indicating strong discriminatory power. The deep learning models (CNN, LSTM, GRU) show competitive ROC AUC scores but fall short in other metrics, suggesting that while they can distinguish between classes well, their overall predictive performance needs improvement.

Time Efficiency:

KNeighborsClassifier exhibits the longest prediction time, making it impractical for real-time applications despite its reasonable accuracy and F1-scores. Deep learning models (CNN, LSTM, GRU) have significantly higher training and prediction times, indicating higher computational demands. This might be a limiting factor for their deployment in resource-constrained environments.

A graph of blue bars with white text

Description automatically generated A graph of different clutters

Description automatically generated with medium confidence

A graph of blue bars with white text

Description automatically generated

Precision, Recall, and F1-scores:

The LogisticRegression and KNeighborsClassifier exhibit relatively balanced precision and recall scores, leading to moderate F1-scores. However, their lower macro-averaged scores indicate that they might be biased towards certain classes. The XGBClassifier and CatBoostClassifier show promising precision but lower recall, suggesting that they might be conservative in predicting positive instances, leading to fewer false positives but potentially more false negatives.

The analysis reveals that while tree-based models like RandomForestClassifier exhibit strong performance across most metrics, they also show signs of overfitting, as indicated by the disparity between training and validation MSE. Simpler models like LogisticRegression and KNeighborsClassifier provide a balance between performance and computational efficiency but may not be as robust. Deep learning models, despite their potential, require careful tuning and significant computational resources.

**A graph with a red line

Description automatically generated** **A graph of a number of classes

Description automatically generated with medium confidence**

# A graph with green and purple lines Description automatically generated

# 

# Chapter 5 Discussions

# 5.1 Considerations and Limitations

# 5.1.1 Vader vs. TextBlob

Our choice of VADER for sentiment analysis stems from its strengths in handling the complexities of social media data: informal language, slang, emojis, and capitalization patterns [Hutto and Gilbert, 2014]. This capability is particularly valuable for tasks requiring accurate detection of negative sentiment, where VADER excels with a 63.3% accuracy compared to TextBlob's 41.3% in a comparative study [Aashish Mehta, 2021]. While TextBlob offers integration with NLTK for broader text analysis, VADER's superior performance in social media contexts makes it the preferred choice here.

# 5.1.2 Predefined Sentiment Schema

Predefined sentiment analysis schemas, like the one we used, offer a valuable tool for automated classification. However, inherent limitations related to subjectivity and generalizability exist. The subjectivity of human emotion poses a core challenge. Annotators might interpret the same text differently, potentially leading to biased classifications [Ribeiro et al., 2019]. This subjectivity can be exacerbated by factors like education level and understanding of the text. While our method provides a framework, it might not perfectly capture the nuances of an individual annotator's judgment. For instance, sarcasm with seemingly positive words could be misinterpreted as truly positive. Furthermore, the generalizability of a predefined schema can be limited. A schema designed for anger might not perform equally well for other emotions like happiness or joy. The rules within the schema are tailored to capture anger-specific labels, leading to misinterpretations when analyzing text expressing different emotions. This limitation necessitates adjustments or expansion of the schema to encompass a broader range of emotions [Mohammad and Ptaszynski, 2017].

The core strength, however, of this approach lies in its ease of implementation. VADER scores readily available through the VADER sentiment analysis library provide a foundation for anger classification. The mapping between VADER negativity and anger intensity aligns with the general understanding of sentiment analysis, where more negative scores reflect stronger negativity and potentially higher anger levels [Hutto, Gilbert, & Kjærgaard, 2014]. The pre-defined thresholds for each anger level offer a degree of customizability.

It is pertinent to note that the current method employs a limited number of labels, which might not adequately capture the full spectrum of anger intensity. Additionally, this approach solely relies on VADER negativity, neglecting other linguistic aspects that can influence anger intensity, such as sarcasm. The chosen thresholds, while adaptable, might not be universally optimal. Their sensitivity to variations requires further analysis. This is further buttress in the need for validation survey which is used to measure the variation between our sentiment score based labelled data and human

annotation and labelling of the same data.

# 5.1.3 Skewed Distribution and Class Imbalance

The analysis of the labelled anger label data shows that the distributions skewed towards the "Not

Angry" category. The data distribution you described exhibits a clear skew, indicating a class imbalance (Buda et al., 2021). This means certain classes, in this case, higher anger levels, are significantly underrepresented compared to the dominant class ("Not Angry").

This imbalance can have negative consequences for machine learning models, especially for classification tasks (López et al., 2021). Classifiers trained on imbalanced data tend to prioritize the majority class during the learning process, leading to biased predictions (Chawla et al., 2022). For instance, a model trained on this data might excel at identifying "Not Angry" text but struggle to accurately classify instances with higher anger levels. To address the class imbalance, we implemented SMOTE alongside cross-validation, weighted learning. **SMOTE (Synthetic Minority Oversampling Technique)** is a data augmentation technique commonly used to address class imbalance in machine learning datasets. It works by creating synthetic data points for the minority class, increasing its representation in the training data. This helps to improve the performance of machine learning models when dealing with imbalanced datasets, where the minority class might otherwise be underrepresented and lead to biased predictions (Liu et al., 2009). This ensured the model was trained and evaluated on a more balanced distribution within each fold. In addition to SMOTE, weighted learning was also employed during model training (Wang et al., 2021). This approach assigns higher weights to the minority classes (higher anger levels) in the loss function. This penalizes the model more for misclassifying these rarer instances, forcing it to focus on learning them more effectively (Huang et al., 2020). The combination of SMOTE and weighted learning is a well-suited strategy for tackling class imbalance in classification tasks, as demonstrated in this research (Chawla et al., 2022).

However, it's important to consider the potential for overfitting when using these methods (Buda et al., 2021). Overfitting occurs when a model learns the training data too well, including noise and biases, leading to poor performance on unseen data. To mitigate this risk, the research employed early stopping, a technique that halts training once the model's performance on a validation set starts to deteriorate (Prechelt, 2020).

# 5.2 Constraints

A significant limitation encountered throughout this project was the challenge of computational power. Training deep learning models, especially, often requires significant computational resources ([MathWorks](https://www.mathworks.com/products/deep-learning.html), [van Rooyen et al., 2020]). These limitations manifested in several ways:

* **Large Memory Footprint:** Deep learning models often have many parameters, leading to a high memory footprint. This can become a bottleneck, restricting the size and complexity of models that can be effectively trained on limited hardware resources (Pouget-Castel et al., 2018).
* **Training Time:** Training complex models with large datasets can take days or even weeks on limited hardware (Goodfellow et al., 2016). This extended training time can hinder the research process by limiting the number of experiments or iterations that can be performed within a reasonable timeframe.

These computational constraints also influenced model selection and hyperparameter tuning strategies. Due to these limitations, we opted for a simpler model architecture (fewer layers, lower number of filters) to reduce the training requirements (van Rooyen et al., 2020). While simpler models may still achieve good performance, more complex architectures might have the potential for higher accuracy (Huang et al., 2018). Similarly, exploring a wider range of hyperparameter values (e.g., learning rate, number of epochs) for optimal performance was restricted due to the extensive time required for each training run (Bergstra & Bengio, 2012). This limited our ability to identify the absolute best configuration for the chosen model.

Several approaches can mitigate the limitations of computational power in future research:

**Graphics Processing Units (GPUs):** GPUs are specialized hardware accelerators that excel at parallel processing tasks, making them crucial for efficient deep learning model training (Li et al., 2020). Limited access to powerful GPUs can significantly increase training time (Amodei et al., 2016).

**Transfer Learning:** Utilizing pre-trained models on large datasets (e.g., pre-trained word embeddings) can leverage the computational power used for their development and accelerate training for our specific task.

**Efficient Algorithmic Choices:** Exploring more computationally efficient deep learning architectures or alternative machine learning approaches suitable for text classification can be beneficial when computational resources are limited.

# 5.3 Future Research

# 5.3.1 Annotation Validation.

We sent out questionnaires in a bid to validate our annotation process. As at time of this report, this is still on going. Analyzing human annotations against our scoring system will provide a validity as regards its accuracy r otherwise. Research should expand on this area as this is very key to fine turning machine understanding of natural language.

# 5.3.2 Unveiling Word Influence on VADER Scores with PCA

A deeper understanding of how individual words contribute to the final compound score using Principal Component Analysis (PCA) presents a promising avenue for future research. it can be employed to **Identify Key Sentiment Drivers** by analyzing the VADER lexicon and the corresponding scores assigned to each word. These factors might correspond to specific word categories (e.g., intensifiers, negations) or semantic groupings (e.g., positive emotions, negative emotions). it also helps quantify the relative contribution of individual words to the overall sentiment score. This would be particularly valuable for identifying words that have a disproportionate effect on the final score, potentially leading to refinements in the VADER lexicon. PCA can reduce the dimensionality of the VADER lexicon, creating a more concise representation that captures the most significant sentiment-bearing aspects of the vocabulary. This could be beneficial for developing lighter-weight sentiment analysis models with faster processing times.

REFERENCES

Albawi, S., Mohamed, A., & Murphy, J. (2017). Deep learning for image recognition: A survey. arXiv preprint arXiv:1709.05805.

Aletras, V., Giannopoulou, G., & Tsaroudis, I. (2018). Legal document analysis and information retrieval: A survey. Journal of Artificial Intelligence Research, 63, 713-772.

Amodei, D., Hernandez-Gardiol, C., Romdhane, S., Vishwanath, S., Watkins, M., & Zandberg, B. (2020). Concrete problems in AI safety. arXiv preprint arXiv:2006.1656

Amodei, Dario, Dario Cubukcu, Ethan Ilvento, Chris Leslie, Joshua V. McClelland, Jasper Snoek, Pieter Abbeel, and John Schulman. "Concrete problems in AI safety." arXiv preprint arXiv:1606.06565 (2016).

Bai, S., Zhipeng, P., & Wang, J. (2018). Exploiting residual learning for recurrent neural network and addressing vanishing gradients. [Neurocomputing], 318, 625-634

Barthes, R. (1967). Elements of semiology (Vol. 1). London: Hill and Wang

Bergstra, Jesse, and Yoshua Bengio. "Random search for hyper-parameter optimization." Journal of Machine Learning Research 13.feb (2012): 281-305.

Bird, S., Klein, E., & Loper, E. (2009). Natural language processing with Python. O'Reilly Media, Inc

Blitzer, J., Dredze, M., & Pereira, F. (2007). Domain adaptation for sentiment classification in social media. Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Linguistics (EMNLP-CoNLL) (pp. 130-137). Association for Computational Linguistics. <https://aclanthology.org/P07-1056.pdf>

Bolly, B., Oliver, D., & Rhodé, J. (2010). Stories and the Stock Market. The Review of Financial Studies, 23(4), 1761-1794. (<https://doi.org/10.1093/rfs/hhq001>)

Bolukbasi, T., Chang, K. W., Gebhardt, J., Zheng, A. Y., & Li, N. (2016). Quantifying and mitigating bias in word embeddings. [arXiv preprint arXiv:1606.08455](https://arxiv.org/abs/1606.08455)

Bolukbasi, T., Chang, K. W., Saligrama, V., & Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? Debiasing bias in language models. arXiv preprint arXiv:1608.04859. (<https://arxiv.org/abs/1608.04859>)

Borkan, R., Zhang, J., Heinz, J., & Mitchell, M. (2019). Fairness in machine learning: A survey. arXiv preprint arXiv:1908.09532

Boyer, B., Gittelman, M., & Whitley, E. (2017). Risks and Opportunities of Sentiment Analysis in Finance. The Financial Review, 52(1), 86-122

Breiman, L. (1996). Bagging predictors. Machine Learning, 24(2), 123-140.  (<https://link.springer.com/article/10.1007/BF00058655>

Brooke, J., & Shaikh, D. (2018). Sentiment Analysis of Financial News Articles Using Fintech and Machine Learning. Proceedings of the 51st Hawaii International Conference on System Sciences, 5032-5041.

Brown, G., Pocock, A., Vandenbroucke, M. J., & Gupta, S. K. (2020). Machine learning and statistical methods in health research. John Wiley & Sons.

Buda, M., Maki, A., & Lopez, M. A. (2021). A Survey on Overcoming Class Imbalance in Support Vector Machines. ACM Computing Surveys (CSUR), 54(2), 1-40.

Buda, M., Maki, A., & Lopez, M. A. (2021). A Survey on Overcoming Class Imbalance in Support Vector Machines. ACM Computing Surveys (CSUR), 54(2), 1-40.

Büttner, P., & Paulssen, M. (2017). Customer sentiment analysis in social media: A comparison of lexical analysis methods. Journal of Business Research, 107, 207-225.

Caliskan, A., Bryson, J., & Narayanan, S. (2017). Semantics in Affect Detection. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers (Vol. 1, pp. 184-193)

Caragea, D., Silvescu, V., & Brezovan, M. (2020). A comparative analysis of TF-IDF, Word2Vec, and GloVe for sentiment analysis on social media data. [Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2020)](https://aclanthology.org/2022.socialnlp-1.5.pdf)

Chawla, N. V., Japkowicz, N., & Kotsiantis, S. (2022). Editorial: Learning from Imbalanced Datasets. ACM SIGKDD Explorations Newsletter, 14(1), 1-6

Chawla, N. V., Japkowicz, N., & Kotsiantis, S. (2022). Editorial: Learning from Imbalanced Datasets. ACM SIGKDD Explorations Newsletter, 14(1), 1-6.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785-794.

Chowdhury, G. G. (2009). Natural language processing (NLP) overview. Journal of Indian Institute of Technology Delhi, 36(4), 328-348.

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. <https://aclanthology.org/N19-1423.pdf>

Dietterich, T. G. (2000). Ensemble methods in machine learning. Multiple Classifier Systems, 18(1), 1-15. (<https://link.springer.com/chapter/10.1007/3-540-45014-9_1>)

Dorogush, A. V., Ershov, V., & Gulin, A. (2018). CatBoost: Gradient boosting with categorical features support. arXiv preprint arXiv:1801.04938.

Feinberg, E. (2015). Text analytics with R. Packt Publishing Ltd

Freund, Y., Schapire, R. E., & Schwenk, N. (1996). Boosting and applications. Annals of Mathematics and Artificial Intelligence, 18(3-4), 347-370.  (<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=c4b71f985ad4f71f73fe83dd8c5ff114e4146702>)

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of statistics, 29(1), 1189-1232.

Friedman, J., Hastie, T., & Tibshirani, R. (2007). The elements of statistical learning (Vol. 1). Springer Series in Statistics New York, NY.

Géron, A. (2017). Hands-on machine learning with Scikit-Learn, Keras & TensorFlow (2nd ed.). O'Reilly Media

Géron, A. (2020). Hands-on machine learning with Scikit-Learn, Keras & TensorFlow (2nd ed.). O'Reilly Media.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

Grimmer, J., & Stewart, M. R. (2013). Text as data: The promise and pitfalls of automatic content analysis in political science. Annual Review of Political Science, 16(1), 455-479

Hall, M. A., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2000). The WEKA data mining software: an update. SIGKDD Explorations, 2(1), 10-18.

Hassan, A., Choi, H.-J., & Cambria, E. (2018). A framework for rumour detection based on sentiment analysis, source credibility, and social network analysis.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2021). The elements of statistical learning (Vol. 1). Springer Series in Statistics New York, NY.

He, Y., Lin, J., Li, Z., Han, Z., & Tang, J. (2019). Exploring Masked Language Modeling for Sentiment Classification. In Proceedings of the 42nd International Conference on Research and Development in Information Retrieval (SIGIR '19). Association for Computing Machinery, New York, NY, USA, 1253-1256.

Hu, M., Liu, Y., & Liu, X. (2004). Mining and Summarizing Customer Reviews. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 168–177. Association for Computational Linguistics.

Huang, Gao, Yu Liu, Lu Sun, and Yuanqing Lin. "Deep learning for image segmentation: A survey." arXiv preprint arXiv:1709.06721 (2018).

Huang, H., Song, J., & Zhao, N. (2020). Attention based document encoder with deep TF-IDF for sentiment analysis. [Knowledge and Information Systems, 62(2)](https://arxiv.org/pdf/1806.06407)

Huang, H., Zhou, Y., & Song, Q. (2020). Cost-Sensitive Learning for Imbalanced Classification with Deep Neural Networks. IEEE Transactions on Cybernetics, 50(10), 4296-4305.

Hutto, C. J., & Gilbert, E. (2014). VADER: A lexicon and rule-based sentiment analysis tool for social media text. Social Science Computer Review, 32(2), 270-280. <https://www.igi-global.com/chapter/review-on-the-application-of-lexicon-based-political-sentiment-analysis-in-social-media/298864>

Hutto, C. J., & Gilbert, E. E. (2014). VADER: A lexicon and rule-based sentiment analysis tool for social media text. Social Network Analysis and Mining, 4(1), 1-16

Hutto, J., & Gilbert, J. (2014). VADER: A parsimonious composite for valence, arousal, and dominance. ACM Transactions on Internet Sentiment Analysis (TI-ISA), 8(1), 1-25

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning with applications in R (Vol. 112). Springer.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An introduction to statistical learning: with applications in R* (Vol. 112). Springer.

Joulin, H., Laurent, B., & Bottou, L. (2016). Generalized Structured Learning for Task-Oriented Sentiment Analysis. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 123–132)

Jurafsky, D., & Martin, J. H. (2020). Speech and language processing (3rd ed.). Pearson Education Limited.

Ke, G., Meng, Q., & Finley, T. (2017). LightGBM: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems, 30, 3146-3154.

Khurana, T. S., Dada, E., & Singh, P. K. (2020). Aspect extraction for sentiment analysis in financial domain using a hybrid deep learning approach. Expert Systems with Applications, 140, 112893.

Kim, Y. (2014). Convolutional neural networks for sentences. [arXiv preprint arXiv:1408.5882]. <https://arxiv.org/abs/1408.5882>

Kogan, S., López, O., & Tetlock, P. E. (2017). Market sentiment and asset prices. The Journal of Finance, 72(3), 1279-1307.

Kumar, S., Djamasri, E., & Baesens, B. (2017). Text mining for sentiment analysis on social web. In Sentiment Analysis in Social Networks (pp. 1-26). Springer, Cham. (<https://link.springer.com/chapter/10.1007/978-981-19-1122-4_56>)

Lemon, K. N., & Verhoef, P. C. (2016). Understanding customer experience throughout the customer journey. Journal of Marketing, 80(6), 69-96.

Li, S., Li, W., Liu, H., Ren, Y., & Jiang, J. (2021). On the Efficiency of Deep Learning in Natural Language Processing. arXiv preprint arXiv:2107.14885.

Li, S., Li, W., Wu, Y., Xing, X., & Li, S. (2020). On the complexity and expressiveness of long short-term memory networks. [Proceedings of the AAAI Conference on Artificial Intelligence], 34(04), 3285-3291. [invalid URL removed]

Li, Shuxin, Shiling Peng, Yuanyuan Zhou, Tianqi Tang, Bo Liu, Quan Xu, Yu He, Wei Zhao, Tiejun Huang, and Tianjun Xiao. "Hardware and software co-design for efficient deep learning." Proceedings of the IEEE 107.6 (2020): 1274-1294.

Liaw, A., & Wiener, M. (2016). Classification and regression by randomForest. R News, 16(1), 22-23.

Ligthart, A., Catal, C., & Tekinerdogan, B. (2021). Challenges in sentiment analysis tasks: A systematic literature review. ArXiv preprint arXiv:2101.09014. (<https://arxiv.org/abs/2101.09014>)

Lipton, Z. C. (2016). The Mythos of Model Interpretability. arXiv preprint arXiv:1606.03490.

Liu, B. (2012). Sentiment analysis and opinion mining. Synthesis Lectures on Human Language Technologies, Vol 3.

Liu, X., Wu, J., & Zhou, Z. (2009). SMOTE-like over-sampling approaches for imbalanced learning with multi-class classification. In Advances in Knowledge Discovery and Data Mining (pp. 807-818). Springer Berlin Heidelberg.

López, A., Fernández, A., García, S., Pẽrez, A., & Herrera, F. (2021). Addressing Imbalanced Data in Classification with Undersampling, Oversampling, and Cost-Sensitive Learning. IEEE Transactions on Knowledge and Data Engineering, 33(1), 1701-1716.

Manning, C. D., & Bird, S. (2012). Natural language processing, MIT press.

MathWorks. "Deep Learning Toolbox." <https://www.mathworks.com/products/deep-learning.html>

Maynard, D., & Funk, A. (2020). Sentiment Analysis on Social Media for Political Campaigns. SAGE Publications Ltd. (<https://journals.sagepub.com/doi/full/10.1177/14614448211014355>)

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis using support vector machine for the Arabic language. Journal of King Saud University-Computer and Sciences, 26(1), 1-13.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. [Advances in neural information processing systems, 26](https://arxiv.org/abs/1310.4546)

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. [Advances in neural information processing systems, 26](https://arxiv.org/abs/1310.4546)

Minaei, M., Namdar, M., Momenzadeh, M., & Yoo, D. (2020). A survey on deep learning based healthcare applications. Journal of Medical Imaging and Health Informatics, 10(7), 1441-1478.

Mintz, M., Amitay, S., & Cohen, Y. (2009). Problem definition for sentiment analysis in the social web. Proceedings of the Third International Conference on Web Search and Data Mining (WSDM) (pp. 270-279). ACM. <https://www.cambridge.org/core/books/sentiment-analysis/problem-of-sentiment-analysis/6E6551A42B9005531A0A97C8071A3B11>

Mohammad, S. M., & Ptaszynski, M. (2017). From Opinion Mining to Sentiment Analysis: A Review of the Literature. In A. Gelbukh (Ed.), Computational Linguistics and Intelligent Systems (pp. 1-27). Springer International Publishing.

Mohammad, S. M., & Turney, P. D. (2010). Lexical Units for Sentiment Analysis. In Proceedings of the ACL 2010 Workshop on Affective Text Analysis (WATA) (Vol. 1, pp. 178-182)

Mohammad, S., & Penstein Rosenstein, M. (2017). Detecting sarcasm in social media. Communications of the ACM, 60(7), 76-87.

Ngai, E. W. T., Xiu, L., & Chau, D. H. P. (2009). The application of data mining techniques in customer relationship management: A review and future directions. Expert Systems with Applications, 36(2), 2533-2542. (<https://doi.org/10.1016/j.eswa.2008.04.007>)

Noda, K., Saito, Y., Ogawa, S., Aoki, Y., & Kubo, N. (2020). A comprehensive investigation of deep learning based image classification methods: a survey. arXiv preprint arXiv:2004.08110.

O'Connor, B., Balasubramanyan, R., Jha, M., & Louie, J. (2010). Boiling the ocean with tweets: The relative extent of public sentiment forecasting. Proceedings of the 19th international conference on World Wide Web (WWW '10) (pp. 810-819). ACM, New York, NY, USA. (<https://dl.acm.org/doi/fullHtml/10.1145/3639711>)

Opelt, N., & Plank, B. (2013). Ensemble methods for semantic role labelling. Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2012), 794-801.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval, 2(1-2), 1-135.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs Up? Sentiment Classification Using Machine Learning Techniques. Proceedings of the ACL 2002 Conference on Empirical Methods in Natural Language Processing, Vol. 10, 79–86. Association for Computational Linguistics.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. [Empirical Methods in Natural Language Processing (EMNLP), 12(10)](https://aclanthology.org/D14-1162/)

Ponniah, P., Sun, A., & Lin, J. (2013). Fine-grained sentiment analysis for social network opinion mining. Proceedings of the 27th International Conference on Computational Linguistics (COLING) (pp. 1435-1446). Association for Computational Linguistics.

Pouget-Castel, Juan Pedro, Eduardo Soto-Pantoja, Patricia Martel-Palacios, and Pablo Martinez-Murillo. "A survey of deep learning methods for brain tumor segmentation." Neurocomputing 321 (2018): 210-228.

Prechelt, L. (2020). Early Stopping - But When? [Online]. <https://link.springer.com/chapter/10.1007/978-3-642-35289-8_5> This reference from 2020 discusses the concept of early stopping.

Prokhorenkova, L., Gusev, G., Dorogush, A. V., & Vorobev, A. (2018). CatBoost: a powerful and efficient gradient boosting framework for applied machine learning. arXiv preprint arXiv:1804.08802.

Reyes, A., Rosso, P., & Veale, T. (2016). A multidimensional approach to sarcasm detection on social media. Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2016), 1084-1089.

Ribeiro, M. T., Araújo, T., Plapp, M., Pimentel, P., & Henriques, J. M. (2019). S = Sentiment + Subjectivity? How to Use Subjectivity Detection to Improve Sentiment Classification. In Proceedings of the 57th Conference of the Association for Computational Linguistics (ACL) (Vol. 1, pp. 4942-4950). Association for Computational Linguistics.

Rios, A. L., & Avelar, A. G. (2017, April). Fine-grained sentiment analysis of political tweets using distant supervision. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)* (pp. 200-207). (<https://arxiv.org/pdf/2104.12250>)

Rosenthal, S., Faruqi, S., & Nakov, P. (2010). SemEval-2010 Task 5: Connotation Extraction from User Reviews. In Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval-2010) (pp. 30-37)

Rosenthal, S., Pang, B., & Vinayak, V. (2010). Context-sensitive polarity reversal detection in sentiment analysis.

Sahmoudi, S., Ouarda, A., & Benferhat, S. (2020). Feature selection using an improved adaptive hybrid PSO-ABC algorithm for sentiment analysis. [Knowledge and Information Systems, 63(1)]

Saulmier, A., & Lopez, P. (2012). TextBlob: Simplified Text Processing. [Workshop on Monty Python: Bringing Humor to the Web, Proceedings of the 2012 International Conference on Web Intelligence and Social Computing (Vol. 1, pp. 407-410)]

Saussure, F. de. (1916). Course in general linguistics. Columbia University Press.

Schmidhuber, J. (2015). Neural networks for sequence processing. [Lecture Notes in Computer Science], vol. 9181. Springer, Cham. [invalid URL removed]

Sennrich, R., Haddow, B., & Birch, A. (2016). Improving neural machine translation with a subword vocabulary. [Proceedings of the First Conference on Machine Translation (WMT16)](https://arxiv.org/abs/1606.02287)

Sidorov, G., Pamuk, S., Magaimedov, A., & Yupanov, A. (2019). Sentiment analysis and opinion mining with social media networks. Journal of Big Data, 6(1), 1-21.

Singh, A., Thakur, M., & Singh, H. (2022). A Comparison of XGBoost, LightGBM, and CatBoost for Stock Price Prediction. The Journal of The Textile Engineering & Industry, 33(10), 1-8.

Socher, R., Perelygin, A., Huang, J., & Manning, C. D. (2013). Deep learning for sentiment analysis: Asurvey\*\*. arXiv preprint arXiv:1307.2869. (<https://arxiv.org/vc/arxiv/papers/1801/1801.07883v1.pdf>)

Sorokin, A., & Roth, D. (2008). Ultra-large scale information retrieval from noisy metadata. [Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '08)], 844-853. Association for Computing Machinery. <https://dl.acm.org/doi/10.1145/1401890.1401974>

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. [arXiv preprint arXiv:1409.3215]. <https://arxiv.org/abs/1409.3215>

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. [arXiv preprint arXiv:1409.3215]. <https://arxiv.org/abs/1409.3215>

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis: a survey. Computational Linguistics, 37(3), 267-307.

Tai, K., Sun, L., & Xu, Y. (2015). Improved sentiment analysis through deep convolution neural networks. [Proceedings of the 19th ACM international conference on Information and knowledge management (CIKM '15)], 1651-1656. <https://ieeexplore.ieee.org/document/7363395>

Tang, D., Qin, B., & Liu, T. (2016). A deep learning approach for sentiment analysis using convolutional neural network. In International Conference on Neural Information Processing (pp. 185–192). Springer, Singapore.

Tang, D., Qin, B., & Liu, T. (2018, January). A survey on deep learning for fine-grained sentiment analysis. arXiv preprint arXiv:1801.07883. (<https://arxiv.org/abs/1801.07883>)

Tang, D., Qin, J., & Feng, T. (2016). Investigating labeling strategies for sentiment analysis: Dataset creation and model comparison. Information Processing & Management, 52(6), 1400-1416. (<https://doi.org/10.1016/j.ipm.2016.04.007>)

Tausczik, A., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. Journal of Personality, 78(3), 1081-1107.

Tetlock, P. E., Saar-Tsechanovich, R., & Macskasy, S. (2008). Does money illusion matter? A psychological investigation of cash salience, market sentiment, and asset allocation decisions. The Journal of Finance, 63(4), 1811-1849.

Thelwall, M., Buckley, K., & Vaughan, L. (2016). Sentiment strength detection in social media conversations. Journal of the Association for Information Science and Technology, 67(12), 2787-2807

Thelwall, M., Buckley, K., Vaughan, M., & Gayler, R. (2010). Sentiment strength detection in short informal text. Journal of the American Society for Information Science and Technology, 61(12), 2544-2558

Thet, H., Ong, C. Y., & Carman, P. M. (2010). A comparative study of sentiment analysis techniques for online reviews.

Thet, M. T., Feldman, R., & Ungar, L. (2010). Fine-grained sentiment analysis for opinion summarization\*\*. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010), 1430-1439.

Van Rooyen, Benjamin, Matthijs van Leeuwen, and Emile van den Berg. "Equalizing the playing field: A survey on budget-constrained machine learning." arXiv preprint arXiv:2004.07712 (2020).

Vandenbosch, A., Verhoef, M., & Buckinx, W. (2017). From lexicon-based to deep learning sentiment analysis: A survey of multilingual approaches. ACM Transactions on Asian and Information Languages (TALIP), 16(4), 1-37

Vapnik, V. N. (1995). The nature of statistical learning theory. Springer Science & Business Media.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Polosukhin, I. (2017). Attention is All You Need. In Proceedings of the 31st Neural Information Processing Systems Conference (NIPS 2017) (pp. 599–609).

Verhoef, P., Kannis, P., Jungerius, B., & Hamstra, M. (2020). Customer experience research through social media analysis: A review of literature and an agenda for future research. Journal of Service Research, 23(4), 548-577.

Waissel, Y., Hetzner, J., & Bethard, J. (2015). Crowdsourcing in machine learning. [Synthesis Lectures on Human Language Technologies], 8(3), 1-162. Morgan & Claypool Publishers.

Wang, D., Qin, B., & Liu, T. (2016). Combining lexicon-based and learning-based methods for short text sentiment classification. Information Sciences, 369-370, 106-115

Wang, Y., Huang, M., Zhao, L., & Zhu, X. (2016). Attention-based LSTM for aspect-level sentiment analysis.

Wang, Y., Huang, M., Zhao, L., & Zhu, X. (2019). Attention-Based LSTM for Aspect-Level Sentiment Analysis. In Proceedings of the 42nd International Conference on Research and Development in Information Retrieval (SIGIR '19). Association for Computing Machinery, New York, NY, USA, 373–382.

Wang, Y., Mao, H., & Redi, J. A. (2017, June). Sentiment analysis in social media: Filtering subjectivity and identifying emotions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 175–186). Association for Computational Linguistics. (<https://arxiv.org/pdf/1906.05887>)

Wang, Y., Yao, Q., Sun, C., Benenson, Y., & Liu, M. (2021). Deep Learning for Imbalanced Classification: A Survey. arXiv preprint arXiv:2106.10275

Wang, Y., Zhao, S., Huang, K., & Zhu, X. (2021). Sentiment analysis based on gradual machine learning for non-i.i.d. distributed data. Knowledge and Information Systems, 63(2), 545-562.

Wickramasinghe, N., Anandarajan, M., & Aruliah, D. (2017). Agile data science process. In Proceedings of the 5th International Conference on Advanced Data Science, Machine Learning and Applications (ADMA) (pp. 1-6). IEEE [Institute of Electrical and Electronics Engineers].

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation, 1(1), 67-82. (<https://doi.org/10.1109/4235.585893>)

Xu, J., Luo, X., & Wang, Y. (2017). Sentiment analysis of online reviews based on unsupervised learning and sentiment lexicon expansion. Knowledge and Information Systems, 65(2), 521-542

Xuan, Y., Zhu, F., & Zhu, J. (2013). Sentiment analysis for predicting stock market movement: A review. IEEE Transactions on Knowledge and Data Engineering, 25(11), 2518-2530.

Yang, L., Luo, T., & Sun, A. (2019). Unsupervised sentiment analysis for social media marketing. Information Processing & Management, 55(6), 1627-1640

Yang, Y., Zhang, H., Zhao, J., & Zhou, M. (2017). Challenges in fine-grained sentiment analysis: An empirical study and discussion. Cognitive Computation, 9(8), 1273-1282. (<https://link.springer.com/article/10.1007/s10844-020-00616-7>)

Yu, H., & Dredze, M. (2010). Towards understanding lexical bias in sentiment analysis. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1659-1668). Association for Computational Linguistics.

Yu, H., & Yang, B. (2020). LightGBM for predicting machine learning course grades. Computers & Education, 145, 103734.

Yu, L., Chen, H., Wang, Q., & Deng, H. (2020). Deep learning for image segmentation: a survey. arXiv preprint arXiv:2004.02088.

Yu, L., Wang, S., & Jiang, Y. (2020). A Survey on Domain-Adapted Sentiment Analysis. arXiv preprint arXiv:2004.13765. ([https://arxiv.org/abs/2004.13765)\*\*](https://arxiv.org/abs/2004.13765)**)

Yu, L., Yao, X., Yang, Z., & Wu, C. (2016, October). On the limitations of weakly supervised fine-grained sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 1634-1643). Association for Computational Linguistics. (<https://arxiv.org/pdf/1910.13425>)

Zhang, C., Bengio, Y., Mordred, M., & Kearnes, M. (2016). Obtaining Better Gradients from ReLU Networks with Larger Batches. arXiv preprint arXiv:1606.03471.

Zhou, Z.-H. (2012). Ensemble methods: foundations and algorithms. Machine Learning, 101(1), 95-128.

Appendix A Link to code- https://github.com/Chizoruo/-MSC-Dissertation

Appendix B. Link to data-[Consumer Complaint Database | Consumer Financial Protection Bureau (consumerfinance.gov)](https://www.consumerfinance.gov/data-research/consumer-complaints/)

Appendix C. Link to Validation Questionnaire https://docs.google.com/forms/d/e/1FAIpQLSc8fYHj\_jscFrlwfdSkCZGIuI\_AAnWdNxj-9MalJQrLz1wk0Q/viewform?vc=0&c=0&w=1&flr=0&usp=mail\_form\_link

Appendix D

TF – IDF Vectorization.

**TF-IDF (Term Frequency-Inverse Document Frequency)** is a statistical technique used in natural language processing (NLP) to analyze the importance of words within a document in relation to a collection of documents (corpus). It has established itself as a cornerstone technique in natural language processing (NLP). Its simplicity and efficiency have made it a go-to method for tasks like document clustering, information retrieval, and sentiment analysis. One of TF-IDF's key strengths lies in its interpretability. Unlike complex neural network models, TF-IDF weights are readily understandable. Each word's weight reflects its prominence within a document and its rarity across the entire corpus. This transparency allows researchers and practitioners to directly analyze the importance of specific terms and gain valuable insights into the document content (Sahmoudi et al., 2020). Furthermore, TF-IDF boasts remarkable computational efficiency. Its reliance on basic calculations makes it particularly suitable for handling large datasets with minimal resource constraints. This efficiency becomes especially valuable in the era of big data, where processing massive amounts of text is commonplace (Sidorov et al., 2021).

Despite these advantages, TF-IDF is not without its limitations. TF-IDF treats each word in isolation, neglecting the potential semantic relationships and context that contribute to meaning. This can lead to misinterpretations, particularly in nuanced or complex language (Huang et al., 2020). Another limitation stems from TF-IDF's sensitivity to rare terms. Words with low information content, such as stop words (e.g., "the," "a"), can receive high weights due to their frequent occurrence within a single document but low overall prevalence in the corpus. This can skew the feature importance and negatively impact performance (Caragea et al., 2020). TF-IDF struggles to handle polysemy, the occurrence of multiple meanings for a single word. Words like "bat" (flying mammal vs. baseball equipment) are treated equally, potentially distorting the representation of the document's content (Huang et al., 2020).

The future of NLP lies in leveraging the strengths of both traditional methods like TF-IDF and cutting-edge advancements. Contextual word embeddings, such as Word2Vec and GloVe, capture semantic relationships between words, offering a richer representation than TF-IDF (Mikolov et al., 2013; Pennington et al., 2014). Additionally, deep learning architectures have shown immense promise in extracting complex features from text data, often surpassing the performance of TF-IDF in specific tasks (Yin et al., 2020). TF-IDF remains a valuable tool in the NLP arsenal, offering interpretability, efficiency, and surprisingly good performance in a variety of tasks. However, its limitations, particularly its naivety towards context and polysemy, warrant consideration. As NLP continues to evolve, the future lies in a strategic combination of classic techniques like TF-IDF and advanced methods that capture the intricacies of language.

Word2Vec

Word2Vec, a cornerstone technique in natural language processing (NLP), has revolutionized how we represent words. Unlike traditional methods that rely on word frequency or co-occurrence statistics, Word2Vec leverages the power of neural networks to learn contextual word embeddings. These embeddings are dense vectors that capture the semantic meaning and relationships between words based on their usage within a large text corpus (Mikolov et al., 2013). Words with similar meanings are positioned close together in this vector space, while words with distinct meanings reside further apart.

Two primary models form the foundation of Word2Vec:

* Continuous Bag-of-Words (CBOW): This model predicts a target word based on its surrounding context words. By analyzing the context, the CBOW model learns the semantic representation of the target word (Mikolov et al., 2013).
* Skip-gram Model: This model operates in the opposite direction, predicting the surrounding context words based on a given target word. The skip-gram model learns the representation of the target word by considering the words that frequently appear in its vicinity (Mikolov et al., 2013).

Through these training processes, Word2Vec captures the nuances of language, where words can have different meanings depending on the context. For instance, the word "bat" can refer to a flying mammal or a baseball equipment. Word2Vec, by analyzing the surrounding words, can learn these distinct meanings and represent them appropriately in the vector space.

While Word2Vec offers significant advantages, it's not without limitations:

* Computational Cost: Training Word2Vec on large datasets can be computationally expensive, requiring significant processing power and time (Calvo et al., 2020).
* Out-of-Vocabulary (OOV) Words: Words not encountered during training (OOV words) can pose a challenge for Word2Vec, as it may not have a meaningful representation for them (Calvo et al., 2020). Researchers are exploring subword-based approaches to address this limitation (Sennrich et al., 2016).
* Bias in Training Data: Word2Vec can inherit biases present in the training data, potentially impacting the learned word embeddings and model performance. Mitigation strategies are being developed to address this issue (Bolukbasi et al., 2016).

Despite limitations, Word2Vec remains a cornerstone of NLP. Researchers are actively exploring advancements to address its shortcomings and improve its effectiveness.