KHAZAR UNIVERSITY

Faculty: Graduate School of Science, Art and Technology

Department: Computer Science

Specialty: Informatics

MASTER THESIS

SENTIMENT ANALYSIS USING MACHINE LEARNING METHODS ON SOCIAL MEDIA

Applicant:_______ Jamila Damirova Ilham

Supervisor:______ PhD, Associate Professor Leyla Muradkhanli Gazanfar

BAKU – 2025

XƏZƏR UNİVERSİTETİ

Fakültə: Təbiət elmləri, Sənət və Texnologiya yüksək təhsil fakültəsi

Departament: Kompüter Elmləri

İxtisas: İnformatika

MAGİSTR DİSSERTASİYA İŞİ

SOSİAL MEDİADA MAŞIN ÖYRƏNMƏ METODLARINDAN İSTİFADƏ EDƏRƏK SENTİMENT ANALİZİN APARILMASI

İddiaçı:_____ Cəmilə İlham qızı Dəmirova

Elmi rəhbər:_____ Texnika üzrə fəlsəfə doktoru,dosent, Leyla Qəzənfər qızı Muradxanlı

BAKI – 2025

TABLE OF CONTENTS

INTRODUCTION

The relevance of the subject: Sentiment analysis is, in sum, the deciphering of the emotional contents of words. This is cardinal for understanding human emotions, beliefs, and behaviors. The volume of textual data for analysis skyrocketed the moment social media platforms became a real thing; hence the need for machine learning to analyze such volumes of data. Emotion analysis on social media can give businesses valuable insights into customer feedback, brand reputation, and even future market trends. These technologies could also be of immense use to governments and other public bodies as a means of understanding general opinion about events, legislation, and matters of immediate social concern. Aside from that, and since social media has supplanted other conventional channels of communication, it could also be gauged in predicting election results, managing crises, and advancing public relations through the core subtlety of sentiment analysis on social media. The complexity and diversity of the emotions displayed by individuals online are often beyond conventional techniques developed to handle sentiment analyses. This is where machine learning comes in, as it can learn from and adapt to really large datasets. The algorithms are learning to identify context and subtlety-sarcasm, for example-better, which leads to more valid sentiment classifications. In summary, this thesis issue is relevant because it has the ability to transform, through the use of machine learning, how governments, corporations, and other institutions understand and respond to public opinion in the digital age.

The purpose of the study: The purpose of this thesis is to improve the effectiveness of sentiment analysis on social media platforms by applying classical supervised machine learning algorithms such as Support Vector Machines, Naive Bayes, and Logistic Regression. The following tasks were performed to achieve the goal:

- Perform a comprehensive analysis of the research on sentiment analysis using machine learning techniques, bearing in mind the historical context, current practices, and potential applications;

- Examine in depth the specialized literature on the use of machine learning techniques on social media platforms, with a focus on the challenges and successes encountered in real-world applications;

- Examine the fundamental concepts and methodologies underlying the application of machine learning to sentiment analysis. This includes understanding the intricacies of various algorithms, their advantages and disadvantages, and the situations in which they perform best;

- Present the cutting-edge opportunities that machine learning offers for sentiment analysis, supported by recent examples of its effective application in a variety of industries;

- Explain in detail the nuances, challenges, and opportunities unique to Twitter sentiment analysis using machine learning techniques;

- Make recommendations and evaluate multiple approaches for enhancing the efficacy of sentiment analysis on social media sites, ensuring that the conclusions are supported by empirical evidence and practical considerations.

The object and subject of the study: The object of the study is the user-generated content available on social media platforms such as Twitter, which provides a rich source of data for sentiment analysis. The subject of the study is the application of advanced machine learning techniques to improve sentiment analysis on these social media platforms, addressing challenges such as dynamic language use and contextual interpretation.

Research methods: The research employed a mixed-methods approach combining natural language preprocessing, supervised machine learning models (Logistic Regression, SVM), word and sentence embeddings (Word2Vec, BERT), and performance evaluation through stratified sampling, k-fold cross-validation, and standard classification metrics.

Scientific novelty:

- the model is optimized specifically for Azerbaijani-language sentiment analysis using real-time social media data;

- it integrates classical machine learning models with context-aware embeddings such as BERT to improve performance on linguistically complex datasets;

- the research introduces a complete pipeline from tweet collection to preprocessing, embedding, modeling, and evaluation, applicable to real-world public opinion monitoring;

- the study proposes a domain-adaptive methodology that can be reused in other low-resource language settings;

- the study demonstrates how social media sentiment analysis can be transformed into actionable insights for crisis management, policy evaluation, and digital communication strategies.

CHAPTER 1. LITERATURE AND THEORETICAL REVIEW

1.1. Literature review related to sentiment analysis using machine learning methods

Sentiment analysis, also referred to as opinion mining, is the process that defines identifying and extracting subjective information from source materials. Therefore, sentiment analysis applies machine learning, natural language processing, and text analytics to accomplish its goals. It is also used to decide the standpoint, emotion, or demeanor of an individual while speaking or writing on a particular subject, or in general, the polarity of the atmosphere.

Sentiment analysis is basically the art of understanding and characterizing people's feelings in text. A large number of researches have been devoted to improving the accuracy and efficiency of sentiment analysis by adopting a wide array of machine learning strategies.

In the context of their work done in 2002, Pang, Lee, and Vaithyanathan addressed the problem of correctly classifying a movie review as positive or negative. For this task of sentiment analysis, they had applied a few kinds of machine learning techniques: Naive Bayes, Maximum Entropy, and SVM. The dataset they used was constituted of reviews for movies taken from the website IMDb.com. Indeed, the results obtained in that research were such that, out of all the rest of the compared algorithms, Support Vector Machines performed the best, especially when one was using unigram features (Pang, 2002).

There were several limitations with the study: first, it was limited to just analyzing reviews of movies written in English, which leaves the possibility of doing sentiment analysis in many languages debatable. Second, it did not handle the problem of imbalance in data, when the number of good reviews may be far higher, or the other way round as compared to the bad ones. Thirdly, the stratégies used did not manage to capture closely the tone of the phrases that contained sarcasm or irony in that text. The last limitation of this research is that it has not presented the research on deep learning algorithms, which at the time of the study were at their infancy but up to date have shown some promise in handling more complex structures of language.

The work has really opened up a lot of doors towards what features of investigation might be done in the future. Those include investigating increasingly complicated feature sets and machine learning models, each of which had the potential to capture the nuances of emotion even better. Another concern that remained mostly unanswered was how such methods might be modified to work in other domains and languages, especially in cases when the text data is not as easy as in movie reviews.

Unsupervised learning in sentiment analysis was addressed by Peter Turney's work in 2002. More specifically, Turney analyzed feelings in product reviews using a pointwise mutual information-based methodology. It categorized the overall sentiment of the reviews by removing terms from the text and then comparing their PMI ratings to reference words that were set as "positive" or "negative." As a matter of fact, the study's results show that an unsupervised method of learning about a task can achieve accuracy comparable to that achieved using supervised learning (Turney, 2002).

Dave, Lawrence, and Pennock studied the classification of online reviews by using machine learning techniques. They published the study in a paper in 2003. The researchers focused on product reviews left by customers and used classifiers such as Naive Bayes to automatically determine whether or not the feedback given was positive or negative. The aims of this paper were to enhance the accuracy of the classifiers by utilizing several variables that included frequency variables: frequency of positive and negative terms. In this respect, the study concluded that the Naive Bayes classifiers are effective and thus can be utilized in real-world e-commerce websites for issues like customer review classification (Dave, 2003).

Nevertheless, it still had a few contributions; the study had some conspicuous drawbacks. First and foremost, it lacked contextual information related to the category of products, which more often than not is considered a vital constituent in accurately determining the feeling of one's consumers. Whereas the fact that the research used only the English language itself raises the question of how it would modify the approaches so that it works on the works of other languages. Third, the strategy has not investigated the possibility of using ensemble methods and deep learning approaches, each of which could potentially result in more reliable outcomes. In other words, this research now focuses more on product reviews, which limits the capacity of this research to be generalized into other fields such as health and politics or even the emotion of social media.

Aliaksei Severyn and Alessandro Moschitti presented the work of Sentiment Analysis in the tweets; the work was published in 2015. They designed a model based on the CNN to automatically grasp semantic representations of the tweets. Later, the model categorizes the emotions of the tweets for polarity. The Twitter dataset was used to support the work of data analysis, and their outcomes showed that the CNN-based models far outperform classical methods like SVMs and the bag-ofwords approach. Their model effects reached a level to be on par with the state-of-the-art performance in sentiment analysis on the used dataset (Severyn, 2015).

Despite that, it also entailed a number of limitations: Firstly, it dealt with just data from Twitter, which led to much speculation whether or not it might be applied to other kinds of text; second, this neural network model has a very high computational complexity, which can be a problem in applications needing real time or large-scale processing. Third, they have omitted the interpretability study of their CNN models, which is highly relevant in order to assess why certain classifications have been made. Lastly, they did not study the impact brought by imbalance data or noisy labeling, both being frequent problems when considering real-world applications.

Another direction of the research that is going to be pursued in the future is related to open problems. For example, it is necessary to see how the application of CNN-based models to other domains or languages would perform. Then, there are questions such as the influence of data imbalance and noisy labeling. Besides, techniques for improving model interpretability and reducing the computing process complexity should be incorporated.

The relevance of feature selection in sentiment analysis was the primary topic of research conducted by Agarwal et al. in 2011, which was conducted primarily within the context of the Twitter platform. In order to categorize people's feelings, they made use of supervised machine learning methods such as Naive Bayes, Maximum Entropy, and SVM. A wide variety of traits, including unigrams, bigrams, and linguistic elements of varying kinds, were taken into consideration as well. Their findings led them to the conclusion that the accuracy of sentiment classification models is significantly influenced by the inclusion of domain-specific variables (Agarwal, 2011).

When doing the research that was published in 2013 by Socher et al., the team concentrated on developing a unique model for performing sentiment analysis that was referred to as Recursive Neural Tensor Networks (RNTNs). Utilizing tree-structured neural networks to take into account both syntactic and semantic compositionality, this model was developed with the goal of gaining an understanding of the compositional element of semantics. The researchers tested their model on the Stanford Sentiment Treebank, one of the well-established and reliable datasets for sentiment analysis. The results indicated that RNTNs achieved state-of-the-art performance, outperforming other previously used methods that predict the emotion of both phrases and whole sentences (Socher, 2013).

Limitations. The RNTN model is rather computationally expensive, especially when working with large datasets. Secondly, the model requires an input of a parsed tree structure, making this approach difficult when parsing is unavailable, or if the parsing is incorrect. Thirdly, English text was focused on, and not on exploring the potential of utilizing multiple languages. Eventually, there remains an interpretability problem of the model, as it is with a lot of complexity in neural networks.

Leaving the possibility to adapt RNTNs for use with other languages or domains, besides computational complexity reduction strategies for the model, is for future work. Another very important research avenue would be the development of methods that could provide interpretability for the model. Another interesting line of investigation might be the extent to which RNTNs are able to capture complex emotions, like irony or sarcasm. This would be the final step in the investigation.

In the topic of text categorization, which also has ramifications for sentiment analysis, the "Machine Learning in Automated Text Categorization" study written by Fabrizio Sebastiani is regarded to be a seminal paper in the field. An thorough examination of machine learning approaches that may be applied to automatic text classification is provided by the author. These techniques include SVMs, k-Nearest Neighbors (k-NN), and Naive Bayes. He explains the fundamentals of these methodologies as well as how they are modified for use with text data. The findings lead one to the conclusion that machine learning algorithms, particularly those that have been trained specifically on text, have the potential to be extremely effective in text classification tasks, to the point where they surpass previous rule-based systems (Sebastiani, 2002).

Among the unanswered concerns that may be addressed by future study is the following: how can the methods of machine learning be modified for real-time analysis? What measurements can be taken to ensure the use of such algorithms in an ethical way? How well do the algorithms work on multi-lingual datasets? How can simple models be derived from them to make it easier to interpret?

Authors of the research paper "Twitter as a Corpus for Sentiment Analysis and Opinion Mining" Alexander Pak and Patrick Paroubek explored Twitter as a source for performing sentiment analysis. In order to perform the sentiment analysis, they used a corpus of tweets and employed machine learning techniques along with linguistic aspects. Their group also preprocessed the Twitter data and used a Naive Bayes classifier to classify text. The authors' conclusion was that Twitter is a good corpus for sentiment analysis and opinion mining since it could amass real-time public mood related to a host of problems (Pak, 2010).

These caveats are woven throughout the study. First, it focused almost entirely on tweets written in English and did not take into account tweets composed in other languages or dialects. Secondly, aspects of Twitter data to address the dynamic nature-such as how sentiment might shift with time or to events-weren't touched upon in the research. This was quite an oversight on the part of the authors. Thirdly, there was no indication of ethical implications of any sort regarding the mining of user data for the sentiment analysis. In conclusion, the research did not dive into the problem of addressing unbalanced classes, which is a prevalent problem in sentiment analysis that occurs when one sentiment (for example, positive) may be overrepresented.

Among the unanswered problems for research to be done in the future is the following: how can models of sentiment analysis be adapted to assess tweets in many languages? Which methodologies are available for use in tracking sentiment in real time? In the process of mining Twitter data for public mood, what ethical issues need to be taken into account? In the context of a study of sentiment based on Twitter, how may unbalanced classes be controlled in an efficient manner?

The BERT (Bidirectional Encoder Representations from Transformers) model was presented for the first time in the paper titled "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," which was written by Jacob Devlin and his colleagues. They wanted to construct a language representation that could be fine-tuned for certain natural language processing tasks such as sentiment analysis, question-answering, and named entity identification, therefore they built BERT to be pre-trained on a huge corpus of text. This was done in order to achieve their goal. Their approach consisted of utilizing deep bidirectional transformers and employing a two-step procedure that began with unsupervised pre-training and then progressed to supervised fine-tuning. This research article concluded that BERT has illustrated the power of pre-trained language representations through performance by establishing new state-of-the-art results on eleven different NLP tasks (Devlin, 2018).

Some of the open problems that might be investigated in further study include the following: How can BERT be modified to become more computationally efficient without sacrificing performance? Is it feasible to design versions of BERT that are culturally neutral or that support several languages? How can interpretability be added to models with such a high level of complexity? What kind of ethical repercussions may result from employing pre-trained algorithms, such as BERT, on data that is either sensitive or biased?

Understanding why neural networks are vulnerable to adversarial assaults is the topic of the study presented in the paper "Explaining and Harnessing Adversarial Examples" written by Ian Goodfellow and his co-authors. The authors present the idea of adversarial instances, which are inputs that have been significantly altered and have the potential to trick machine learning algorithms into generating inaccurate predictions or classifications. In order to examine this phenomena, they make use of mathematical models as well as actual investigations. The primary takeaway from the study is that machine learning models, and neural networks in particular, are susceptible to the kinds of adversarial perturbations discussed in the research because of the linear character of high-dimensional spaces (Goodfellow, 2014).

The authors of this work present VADER (Valence Aware Dictionary and sEntiment Reasoner), a lexicon and rule-based model created exclusively for recognizing the sentiment in social media and short-text communications. The acronym stands for "Valence Aware Dictionary and sEntiment Reasoner." The process entails the creation of a specified lexicon of terms connected to sentiment, as well as the application of grammatical and syntactical principles, in order to determine the degree of sentiment. As a computationally efficient alternative, the article comes to the conclusion that VADER performs on par with more complicated machine learning algorithms when it comes to the specific task of analyzing the sentiment of posts on social media (Hutto, 2014).

Limitations: VADER is primarily intended for use with English text, which restricts the range of situations in which it can be applied. Second, the application of the model to other types of text, such as academic articles or scientific papers, is not investigated, despite the fact that the model works well when applied to material from social media. Thirdly, the model is incapable of capturing more complex kinds of sentiment expression, such as sarcasm or irony, because it was not constructed with this capability in mind. Last but not least, the article, similar to a large number of previous models in the field, does not address ethical problems such as data privacy or model bias.

Questions like "How can VADER be adapted for other languages or types of text?" could be open-ended topics for further investigation in the future. Is it possible to expand the model such that it can capture more subtle types of emotion? What kinds of computational hurdles must be overcome in order to scale VADER to accommodate larger datasets? When doing rule-based sentiment analysis, what ethical issues need to be taken into account?

"Attention Is All You Need" is another significant work that was written by Ashish Vaswani and a handful of other co-authors. The authors of this work present the Transformer model, which is an alternative to traditional models that prioritizes attention processes above recurring layers. The purpose of this work is to make sequence-to-sequence operations, such as translation and summarization, more efficient and effective. For the purpose of gathering contextual information, the system makes use of both self-attention and multi-head attention layers. The research article comes to the conclusion that the Transformer model not only beats recurrent and convolutional architectures on a variety of benchmarks, but it also consumes less time for computing during the training process (Vaswani, 2017).

However, there are several problems with the study. First, there isn't enough research done on how well the model captures long-range relationships in sequences. This is a major limitation. Second, the research does not dive thoroughly into the interpretability of the attention processes, despite the fact that it indicates that these mechanisms are successful. Third, the Transformer model calls for a significant amount of computing resources, particularly in terms of memory, which may provide an obstacle for research groups that are smaller or for applications that are used in the real world. In conclusion, the article does not address any of the ethical considerations that are associated with unfair data or data biases.

In his article, Pedro Domingos discusses numerous critical features of machine learning that are often ignored despite their importance. These characteristics are essential for practitioners. In order to give insight into the frequent obstacles and dangers that are associated with machine learning, the author examines a variety of subjects, such as overfitting, the curse of dimensionality, and data representation. Instead of relying primarily on empirical testing, the technique makes extensive use of conceptual analysis. The purpose of this work is to function as a basic guide for practitioners of machine learning of all experience levels, and the study comes to the conclusion that a grasp of these fundamental ideas may considerably increase model performance (Domingos, 2012).

Having said that, the document does have certain restrictions. In the first place, it is mostly theoretical and lacks any empirical validations for the assertions that are stated. Second, despite the fact that it covers a wide variety of subjects, it does not go into great depth about any particular machine learning methods or domains; hence, it is more of a generic guide. Thirdly, the research study does not explore the computing needs or scalability difficulties that are related with machine learning techniques. Lastly, it does not address the ethical concerns that may come from machine learning models, nor does it take into account any biases that may exist.

The paper "Playing Atari with Deep Reinforcement Learning" by Volodymyr Mnih et al. is yet another noteworthy piece of writing. The authors of this study offer a unique deep learning model that they call DQN (Deep Q-Network). This model is able to learn effective policies in reinforcement learning tasks directly from high-dimensional inputs, as the authors explain. They evaluate seven different games for the Atari 2600 using the DQN model. A Q-learning algorithm makes use of a convolutional neural network as one of its components to approximate the Q-function. The technique contains this component. The findings of this research article come to the conclusion that DQNs perform much better than other approaches for six out of seven games, therefore creating a new state-of-the-art for direct end-to-end reinforcement learning (Mnih, 2013).

Having said that, the paper does suffer from a few shortcomings. To begin, it is exclusively concerned with the Atari 2600 games and does not investigate how well the model might do in more difficult or real-world activities. Second, the research does not give an in-depth examination on how interpretable the model is, which is a significant issue in the field of reinforcement learning. Third, the DQN design may be quite computationally demanding, which necessitates the use of specialized hardware for the training process. In conclusion, the article does not address any ethical concerns that may arise from reinforcement learning or artificial intelligence.

Among the unanswered concerns that may be pursued in further study is the following: how effectively does the DQN model generalize to other, more difficult tasks? Is it possible to enhance the model in such a way that it becomes either easier to comprehend or more efficient to compute? How exactly may DQN be modified to work in settings with many agents? What kind of ethical repercussions may result from applying various forms of reinforcement learning to different fields?

1.2. Literature review related to machine learning methods on social media

"A survey on session-based recommender systems" by Shihan Wang, Xianzhi Wang, and Quan Z. Sheng is an article that focuses on a literature review of machine learning methods in the context of social media. In this paper, the authors provide a comprehensive overview of machine learning techniques applied to a variety of social media analytics tasks, including sentiment analysis, event detection, and social network analysis. The paper divides these duties into three major categories: content analysis, user and network analysis, and hybrid analysis involving both content and network characteristics. The authors discuss machine learning algorithms such as Naive Bayes, SVM, and Random Forest, in addition to emergent deep learning techniques. While machine learning has significantly advanced the field of social media analytics, there is still room for advancement in the development of more effective real-time models, they conclude (Wang, 2021).

The study conducted by Tang et al. (2014) aimed to examine the effects of several machine learning algorithms on the analysis of user behavior on social media platforms. They have provided their algorithms with large datasets from popular websites, such as Facebook and Twitter. The main goal of the study was to comprehend how various elements, such user engagement and content type, impact sentiment prediction. Their models achieved remarkable sentiment identification accuracy by combining supervised and unsupervised learning techniques (Tang, 2014).

The study "Twitter mood predicts the stock market" by Johan Bollen, Huina Mao, and Xiao-Jun Zeng is one that has received a lot of attention in this area. These authors used Twitter data and machine learning techniques like the Self-Organizing Fuzzy Neural Network to predict the stock market. They correlation analyzed the Dow Jones Industrial Average with the general

mood condition generated from a thorough survey of Twitter feeds. Their finding was that stock market fluctuations might be somewhat accurately predicted by using Twitter sentiment analysis (Bollen, 2011).
The work entitled "Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media" by Abhijnan Chakraborty et al. is one of the most important works in machine learning applied to social media analytics. The authors focused, in the present work, on how to apply machine learning techniques on clickbait headlines in online news published through social media platforms. These techniques include logistic regression and decision trees. They evaluated over 2,000 publications based on features such as the use of words and sentence structure. The most salient conclusion derived is that machine learning models can indeed make the correct discrimination between clickbait headlines and others (Chakraborty, 2016).
They said that their dataset was limited in cultural and linguistic diversity because of the prevalence of news sources in the English language. The authors further confessed that their approach is actually fast out of date due to the rapid evolution of the clickbait industry. In this study, the impact of multi-media components such as pictures and videos was not taken into consideration for the clickbait identification performance. Another interesting topic for future study would be how to further develop machine learning algorithms so that it can also detect the complex clickbait-for example, those using multi-media or visual components.
"Measuring User Influence in Twitter: The Million Follower Fallacy" by Cha et al. is one of the well-renowned real articles in the domain of machine learning applied to social media. They did their research on the concept of user influence on Twitter. They adapted methodologies using network analyses methods and machine learning algorithms in finding influence based on an array of variables, such as number of followers, retweets, or mentions. It turned out, against the general feeling, that the more followers one has, the less impact he or she creates.
Limitations that were recognized by the authors include that their study was limited to Twitter, and therefore generalizing to other social networking sites was not feasible. They also showed awareness of the fact that since their methods deal mainly with quantifiable data, they can easily avoid qualitative factors of interactions, such as quality. Other limitations included a lack of temporal analysis that would have helped in understanding how the measure of influence varies over time. How to accurately assess the influence across a range of different social media platforms, and how to include qualitative factors into the assessment, is an outstanding topic for future study (Cha, 2010).
Among these, another real paper in machine learning techniques on social media is definitely "Fake News Detection On Social Media" by Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, published in the ACM SIGKDD Explorations Newsletter. Characteristic features

of fake news on social media sites were obtained by applying data mining and machine learning techniques. The researchers used a range of classifiers from decision trees to Naive Bayes up to Support Vector Machines, for the classification of news stories as "real" or "fake.". The overall findings showed that the machine learning models could effectively classify fake news from real news; however, their performance varied significantly with different attributes that were used for the data classification (Shu, 2017).

The authors themselves pointed out some of the shortcomings of this approach: it was done mostly by considering text-based features, setting aside that news items are often accompanied by photographs or videos in social networks. Its applicability in a multilingual environment is limited; this is because the current study focuses only on English-language publications. They also noticed that their dataset was somewhat limited and perhaps did not encompass all the characteristics of false news. An interesting line of future investigation may be how to properly incorporate the multimedia aspects into the false news detection algorithm in a way to make the algorithm more precise and robust.

CHAPTER 2. FUNDAMENTALS OF APPLYING MACHINE LEARNING METHODS IN SENTIMENT ANALYSIS

2.1. Modern opportunities and examples of implementation of machine learning

It is a modern phenomenon, though very prevalent, and quite revolutionary in many fields and aspects. While predictive analytics allow it to anticipate future trends and habits through the analytics of past data, like purchase history, even retailers use machine learning algorithms to predict their customers' shopping habits. This also leads to better inventory management and facilitates the development of marketing campaigns, too, which can be further targeted more effectively. Machine learning-driven predictive analytics can be used for predicating patient outcomes in healthcare. This would lead the door for proactive interference and personalized options in treatment for each patient. By embedding machine learning into market movement forecasting, financial institutions hold immense potential to improve behavior in risk management and investment decisions. Through the use of machine learning to anticipate market movements, financial institutions have the potential to improve risk management and investment decision-making activities. Through the use of machine learning, Target is able to study the purchasing patterns of customers and anticipate their wishes, which enables them to concentrate their marketing efforts more efficiently. This leads in greater revenue as well as enhanced satisfaction among customers. An example of the use of predictive analytics is provided here.

NLP is a technology that enhances advanced search engines by enhancing the relevance and accuracy of search results. This is accomplished by understanding the context and purpose behind user queries. Additionally, the processing and analysis of vast volumes of textual data in real time is revolutionising other sectors such as education, marketing, and international communication, in addition to customer service. This is a significant development. NLP greatly improves the efficacy and efficiency of many services and applications by enabling communication with machines and understanding large volumes of data (Table 2.1.1).

In order to aid computers in interpreting, analysing, and responding to human language, NLP makes use of machine learning techniques. As a result, it has made significant progress in a number of academic fields. This ability is shown by chatbots and virtual assistants such as Alexa and Siri, which enable interactions that are seamless and realistic, so significantly improving both the user experience and customer service overall. In the field of NLP, sentiment analysis is a subset that assists businesses in interpreting the attitudes and perspectives of customers as communicated via social media and reviews.

Table 2.1.1. Opportunities in NLP with Machine Learning

Opportunity	Description	Example Application	Impact
Chatbots and Virtual Assistants	NLP enables user experience and customer service interactions that are seamless and human-like.	Siri, Alexa	Contributes to an improved user experience and streamlined customer service.
Machine Translation	Language barriers are broken down and global communication is facilitated by NLP, which makes cross-linguistic connections simpler.	Google Translate	Facilitates worldwide collaboration and communication through the provision of precise translations.
Sentiment Analysis	NLP assists in the comprehension of consumer sentiments and opinions via social media and reviews, thereby informing marketing strategies.	Social media sentiment analysis tools	Gains insights into consumer sentiment in order to inform product development and marketing strategies.
Personalized Learning	For personalised education, NLP adapts language courses to individual proficiency levels and learning rates.	Duolingo	Personalised courses increase student engagement and retention in the language learning process.
Real-time Text Analysis	Real-time processing and analysis of enormous quantities of textual data by NLP revolutionises customer service and marketing.	Real-time customer support chat systems	Enhances the effectiveness and efficacy of marketing and customer service applications.
Advanced Search Engines	NLP improves the precision of searches by comprehending the intent and context of inquiries, thereby returning pertinent results.	Google Search	Enhances the user experience by providing search results that are more precise and pertinent.

Advanced search engines, like Google Search, apply NLP to understand the context and intent of user queries for more exact and relevant results. This can be further helped in enhancing the user experience by developing better search results with regard to effectiveness and efficiency when reliable information is provided. Natural language processing lets immediate analysis of huge volumes of textual input.

Table 2.1.2. Sentiment analysis methods

Method Category	Approach	Limitations	Techniques	Strengths
Machine learning-based methods	Trains models on labeled datasets using feature extraction	Requires labeled data, weak with complex language	Naive Bayes, Support Vector Machines, Logistic Regression	Adaptable to different domains, works well with large datasets
Lexicon-based methods	Uses predefined lists of words with sentiment scores	Struggles with context, sarcasm, and domain-specific terms	SentiWordNet, AFINN	Simple, requires no labeled data, interpretable
Rule-based and heuristic methods	Applies predefined rules and heuristics	Less flexible, struggles with context shifts	VADER, SenticNet	No need for training data, handles negation and amplifiers
Hybrid approaches	Combines machine learning and lexicon-based methods	Complex to implement and tune, requires feature engineering	Naive Bayes with Lexicon Features, Lexicon-Augmented Neural Networks	Leverages strengths of both lexicon and learned features
Deep learning-based methods	Utilizes neural networks to capture deep text semantics	Needs large labeled datasets, high computational resources	RNNs (LSTM), CNNs, Transformers (BERT)	High accuracy, captures contextual nuances and semantics

Naive Bayes is a probabilistic classifier based on Bayes' theorem. The classifier assumes independence of features given the class label, which makes it very effective for text-based sentiment analysis since it handles high-dimensional data and provides fast computation. It models the likelihood of words to result in a positive or negative sentiment and, although the underlying assumption is simple, can classify text with high accuracy. But it is also the case that this technique's independence assumption can limit its performance for cases in which sentiment depends strongly on word dependencies. Still, this technique is often preferred due to its interpretability and ease of implementation. It is usually the first method tried in many experiments on sentiment analysis.

A breakdown of these categories in the table includes first when there are lexicon-based methods like SentiWordNet and AFINN, which all depend on a prebuilt list of words with given sentiment scores; thus, they provide simplicity and interpretability but fail in building contextual or domain-specific nuances in language. The second group involves the use of different machine learning-based approaches: Naive Bayes, support vector machines, and logistic regression. The methods involve labeled datasets and feature extraction in order to train the models, which perform well on large datasets across domains but do not get along well with higher complexity of the language or sarcasm.

It is also revolutionizing most of the recommendation systems in e-commerce, streaming services, and social networking platforms to better user engagement and improve the users' experience. Most of the prevalent e-commerce platforms, including Amazon and Netflix, often utilize some kind of machine learning algorithms that analyze user behaviors and preferences. This review, therefore, leads to recommendations of content that are personal, thus increasing customer engagement and happiness. Such systems ensure user satisfaction and engagement through recommendations that are personalized, which increase the chances of repeatedly doing business with customers and keep them loyal. Spotify, on the other hand, utilizes machine learning in creating customized playlists that are in line with user preferences about music taste. With this, the capability of offering recommendations of music in line with user preference is possible. Thus, it enhances the user's total auditory experience. Recommendation systems go a long way to make retailers' lives easier in the world of e-commerce-dwelling populations by suggesting items likely to be acquired by a customer.

These suggestions are derived on the customers' browsing history and previous purchases. Amazon's recommendation engine is an example of a commercial application that makes use of machine learning. This engine generates a significant amount of cash by making suggestions to customers, so enhancing their shopping experience and resulting in an increase in revenue. Personalised user profiles may be created on social media platforms such as Facebook and Instagram via the use of recommendation algorithms.

These algorithms include content and advertisements that are highly pertinent to the unique preferences and areas of interest of the user. These systems are vital for encouraging user engagement and driving company success in a variety of sectors because they continuously collect information from user interactions and alter recommendations in a manner that is suitable (Table 2.1.3).

Table 2.1.3. Applications and impact of Machine Learning in sentiment analysis

Application	Description	Example	Impact
Music Streaming	Creates customised playlists by analysing users' listening patterns.	Spotify	Improves user experience by presenting consumers to novel music that aligns with their interests.
E-commerce	Utilises browsing and purchase history to recommend items, hence increasing revenues.	Amazon's recommendation engine	Enhances sales growth and enhances the browsing experience by providing recommendations for relevant items.
Social Media	Customises material streams to display posts and adverts that are relevant to users' interests and choices.	Facebook and Instagram	Enhances user retention and boosts engagement by presenting personalised content based on individual user preferences.

The user experience is considerably improved across a number of different platforms when machine learning is included into recommendation systems. Machine learning algorithms are used by music streaming companies such as Spotify in order to assess the listening habits of users and build tailored playlists on their behalf. Users are introduced to music that is unknown to them but that is in line with their preferences via this individualised approach, which ultimately results in increased user happiness and engagement with the website. Spotify works to ensure that the music listening of its users is kept current and constantly evolving, with the service adapting and tweaking its recommendations based on its interaction by all users.

In online commerce, the recommendation systems are one of the important pieces that contribute toward better sales and enhance the whole shopping experience with increased customer satisfaction. The recommendation engine at Amazon is being employed with machine learning algorithms in analyzing browsing and purchasing histories. Based on this information, the engine provides suggestions for goods that have a high probability of being purchased by customers. Not only does this personalized recommendation system make shopping more fun to the customers as they find goods relevant to their needs, but it also significantly boosts sales. In turn, this should increase the likelihood of buying certain products that Amazon is able to recommend, leading to an overall increase in revenues.

Intrusion detection systems continuously monitor network traffic to identify and block attacks in real time by incorporating machine learning at their core to ensure enterprise networks are kept safe from any danger. These machine learning-based security systems adapt to the ever-changing threats, creating a defensive mechanism that is dynamic and improves forever. It helps identify possible dangerous situations, and by categorizing different incidents according to their severity, it also assists in analyzing security records and alerts. This has helped relieve the workload of security personnel in addition. This helps organizations to stay in a secure digital atmosphere and avoid other possible attacks in the future. Therefore, the use of machine learning in cybersecurity solutions increases the effectiveness and efficiency of such solutions. Machine learning increases the accuracy with which correct forecasting and management of agricultural yields are done by data retrieved from a set of sensors mixed with satellite images. It optimizes fertilization modes, pest control, and irrigation methods in agriculture to produce more and in a sustainable way regarding this objective. Precision agriculture using the application of machine learning algorithms involves monitoring soil health, weather patterns, and crop growth-all of which provide producers with useful data that help to manage resources and to make decisions. The health of the crops is monitored through the application of drones whose cameras are powered by machine learning algorithms. These algorithms detect problematic situations in their early development phases, like shortages of fertilizer and outbreaks of a disease. Due to this, rapid reactions can be possible which helps in evading the loss of agricultural crops. One of the good examples of using businesses with the help of machine learning is Blue River Technology. The company is focused on designing and manufacturing advanced machinery for agriculture, applying artificial intelligence in the identification and treatment of plants. Therefore, the company has contributed to the adoption of environmentally friendly farming practices and reduced the application of chemical pesticides. This is attuned to the improvement of harvest plans based on sets of forecasts for the most favorable times to harvest, considering patterns of crop growth in addition to environmental conditions, hence increasing yield and quality. This shall be attained using machine learning. In livestock management, machine learning algorithms make it possible to monitor animal health and behavior to raise standards of animal welfare and enable early veterinary interventions. The algorithms are designed to detect signs of stress and disease in cattle to efficiently manage their population. All this enables farmers to meet the rising demand for food more sufficiently, while on the other hand, it reduces expenses, improves sustainability, and increases production with the help of machine learning. Besides an increase in agricultural production as a result of the use of machine learning, agricultural systems that are robust and sustainable are developed.

Machine learning can reinforce the business of real estate with regards to better investment decisions and property valuations through the analytics of historical data, market trends, property characteristics, and individual attributes. Precise property evaluations can therefore be derived, with the identification of potential opportunities for profitable investments. Real estate platforms use machine learning algorithms in connecting a buyer to a home that best suits his preferences and financial capabilities. This modality simplifies the process for property purchase and adds to overall consumers' satisfaction. It applies machine learning for market trend research and house price forecasts, hence assisting its buyers and sellers in well-informed decisions and managing the real estate industry with efficiency. On the other hand, machine learning analyzes data concerning tenant behaviour and maintenance needs for enhancing property management. It also streamlined operations to ensure that tenants remained satisfied with their living conditions. Commercial real estate markets deploy machine learning algorithms in the prediction of office and retail demands. This feature would enable developers and investors to make geared decisions based on unique conditions of prevailing markets. It can increase the levels of efficiency, accuracy, and decision-making within real estate. Because of this, it can spur growth and cause an effect that is much more positive for investors, buyers, and sellers alike.

Machine learning can help make the world a safer place through the analysis of information from an array of sources, such as security cameras and social networks, while strengthening responses to emergencies and crimes in large measure. The aim here is to provide a sort of early warning system for offending behavior or planes that may fall out of the sky. Application of machine learning by law enforcement agencies empowers them to trace areas of significant activity and identify patterns, therefore allowing them to take whatever preventive measures are deemed necessary in protecting the safety of the common public and reducing the crime rate. Predictive policing systems target incident prevention by mapping out crime data and passing on appropriate resources to the location that experiences a high risk of such crimes. These systems employ machine learning algorithms towards this end. The applications of machine learning are not only for improving crisis detection and management but also for emergency response by analyzing the latest data from sensors and communication networks. In disaster management, machine learning models provide predictions based on the impacts of natural catastrophes and show directions to guide relief activities or any evacuations. This would facilitate quick and effective responses with minimal damage to human life and properties. Machine learning could allow public safety agencies to build a better capacity in terms of preventing and responding to threats. In the process, there would be optimum improvement of the security and safety level that communities enjoy.

Application of machine learning not only provides great insight which advances patient care and medical research, but also increases the quality of treatment outcomes and diagnosis in the healthcare sectors. Hence, it is probable that machine learning algorithms utilized to examine the data from the field of medicine for finding patterns and relationships can help in diagnosing diseases or suggesting personalized treatments. This will enable the use of machine learning techniques in personalized medicine to power possible improvements in therapy results and decreases adverse treatment effects through crafting treatments that unique medical histories and genetic features can include. IBM Watson Health uses machine learning techniques to analyze patient data and the medical literature for the twin goals of improving patient outcomes and enhancing clinically sound treatment decisions by physicians. The drug development process can be improved by using machine learning for the evaluation of data received in clinical trials and to identify innovative therapies with potential benefits. This will further accelerate efficiency in the research process, accelerating the pace with which new pharmaceuticals are developed. This is evidenced in a number of ways, including the capability to detect inconsistencies and provide proper diagnoses; research, therapy, and diagnostic capabilities continue to improve with machine learning, largely increasing medical knowledge and improving the quality of care patients receive.

It works through analyzing data from a wide range of sources for demand estimation, route planning, and efficient management of supplies. This makes it possible to smoothen the supply chain operation and improve the delivery services in logistical industries. Economics could be saved, improving customer satisfaction and quicker delivery times, if the power of machine learning were harnessed to make sure logistics companies deliver their products precisely and on time. Amazon uses machine learning algorithms to estimate order volumes and distribute resources in a fitting way. This would allow the company to ensure its delivery network is optimized and shipments arrive in a fast and efficient manner. Besides, predictive maintenance for transport infrastructure employs machine learning algorithms to reduce downtimes and improve dependability. For this, it employs algorithms to examine data retrieved from sensors and other devices equipped with IoT. That in itself enables the forecasting of equipment breakdowns and the scheduling of swift repairs accordingly. Machine learning algorithms can provide enhancements in inventory management involved in warehouse operations. It does so by providing demand pattern forecasting and adjusting the amount of stock. This will ensure items are in an easily accessible state and hence smaller lots of excess inventory. Machine learning can enhance customer service and improve the accuracy and efficiency of the completion of different tasks within the logistics industry. Consequently, the operational efficiency is going to get improved and development fostered within a very competitive field.

Tremendous will be the influence of machine learning on education. Though its most obvious uses are found in delivering personalized education experiences, improving learning with data-driven analysis, it also massively promotes saving on the entire sector as a whole. The goal of the Machine Learning algorithms is to analyze the data on student performance, pinpointing points at which the student knowledge is weak in identifying recommended resources relevant to such aspects. This will go a long way in ensuring that the students indeed receive the much-needed help to succeed. An application of machine learning by an adaptive learning system enables the customization of courses for individual students. This results in a personalized learning track that improves academic performance and retention. Online learning platforms, such as Khan Academy and Coursera, use machine learning to make the learning process more efficient. This includes things such as personalized course recommendations and curriculum modifications that are tuned to the users' current skill levels and development. This is why these platforms will be able to extend opportunities for education that will be more interesting and far more effective. Machine learning-powered chatbots are offering quick support and guidance for students, which really helps them in finishing the activities and answering inquiries instantaneously, ultimately improving the overall educational experience. With the help of machine learning, colleges and universities are now capable of identifying those students who are at a greater risk due to academic problems. Consequently, the right interventions can be deployed in time to improve the academic performance for such students, thus increasing the possibility of continuance. This means that the educational sector can offer students learning experiences that are not only appealing and engaging but highly effective for the solution of specific learning problems, which is possible with the use of machine learning. Consequently, education outcomes improve, and so does the success rate of students at every level of education.

Machine learning in agriculture is beneficial because it introduces an increase in the accuracy of crop production forecasting and the management of crops. The goals of the system are to optimize the activities related to the control of pests, fertilization, and irrigation to improve the yield of agriculture and its sustainability. This system uses data acquired from satellite imagery and through a large number of sensors. Application of machine learning algorithms in precision agriculture enables the monitoring of soil health, weather patterns, and crop growth. Useful data that aids in the management of resources and decision-making is therefore provided to the producers. Drones equipped with cameras and operating on machine learning algorithms are useful in crop health monitoring. These drones can detect the early beginnings of problems such as diseases outbreaks and shortages of fertilizer. They can therefore locate these challenges. As a result of this, quick responses are made possible. This helps to lower the losses of crops faced in agriculture. Another company that uses machine learning is Blue River Technology. It manufactures and deals in the development of modern agricultural machinery capable of recognizing and treating plants through artificial intelligence. It has, therefore, promoted less harmful farming practices to the environment, besides reducing the use of chemical pesticides. Machine learning is used to optimize the plans of harvests at the optimal time to enhance the growth pattern related to crop productivity and quality, according to environmental conditions. In livestock, the management applies machine learning algorithms in order to study animal health and behavior to get improvements in animal welfare and ensure early veterinary intervention. These algorithms use indicators of stress and illness to help manage the livestock. This could finally allow farmers to better their operations by effectively answering the growing demand for food on a global scale, while shrinking all costs, improving sustainability, and increasing this level of production. Besides the fact that it results in increased production, machine learning in agriculture allows for the development of sustainable and flexible food systems to meet the demands of a growing global population.

But one of the big contributions machine learning brings into the gaming industry is by having intelligent NPCs react to the players' actions and strategies. It has escorted not just the aspect of enhancing the realism and attraction of the encounters but even contributed towards keeping the challenging and ever-changing nature of the industry. Because the NPCs are able to learn and adapt from the players' moves, each session of playing becomes more unique with each use, raising the experience of the player through a dynamic, in-depth personalized process that eventually tempts them for more. The generation of procedural content is another use of machine learning that can be found in the game industry. This enables one to generate an e number of unique game worlds that are enjoyable and interactive for their players. In such cases, "No Man's Sky" utilizes machine learning to create a gigantic universe comprising a number of planet types and ecosystems. Each of these planets can be explored by the players, each having peculiar plant life, animal life, and geological aspects. Machine learning to enhance the visual and audio parts in games is pretty vital. It makes use of algorithms for generating realistic animations, immersive environmental interactions, and sound effects based on surroundings. On the other hand, competitive gaming itself uses machine learning to assess the performance of the players that participate. It gives players insights that help them build on their respective performances and strategize for the future. It makes it a lot more lucrative and competitive in which the gamers can make money. Streaming services such as Twitch also enhance user engagement and delight through personalized and relevant content. How they do it is by applying machine learning algorithms that could make recommendations to users on broadcasts and content, considering consumers' preferences and previous patterns of viewing. Machine learning within the gaming industry keeps on learning from-and adapts to-the tastes and behaviors of users, hence pushing the limits of what is possible in the domain of interactive entertainment. This will inherently increase the degree of personalization and interaction that one can have with the virtual event.

2.2. Comparison of machine learning methods for sentiment analysis

Deep learning algorithms in sentential analysis have proved better in interpreting complex patterns and relationships in data compared to the standard machine learning algorithms. That means they can give more coherent and high-level interpretations of text data. Quite a significant leap from more traditional ways of machine learning. Perhaps one of the most salient advantages of deep learning within sentiment analysis study is the fact that it is with this advanced approach that it will be able to pick up context and semantics. This might involve techniques like word embeddings, where words will be stored in a dense vector space to convey their meanings and relations. The deep learning models are based on some word-embedding approaches, including Word2Vec and GloVe. These methods worked better in sentiment analysis compared to TF-IDF or bag-of-words approaches. In these embeddings, deep learning algorithms learn synonyms and semantic similarities that in turn improve the accuracy of the sentiment categorization process. RNNs and their variants, such as LSTM, have also been very useful in studies on sentiment analysis. This will be the case because they effectively capture the sequential connections that are inherent in text. This is accurate because they let the model grasp the context given by the words a phrase contains before it. The LSTM model can grasp what the phrases "bad" and "not bad" really mean, which customary machine learning models find quite difficult due to their small awareness of the context in which they are used. Certainly, CNNs, usually used in image processing, have also given promising results for text-based sentiment analysis. These CNNs achieve this by discarding the hierarchical structures and local features in word sequences. This has improved the accuracy of classification of emotions. A typical application involving CNN in the domain of sentiment analysis involves text classification models that make use of convolutional layers for word embeddings. With this technology, it is easier to extract n-gram data and give a better interpretation of the feeling that is sent via text.

And for making efficiency better in deep learning models, enormous datasets are required along with transfer learning to be able to do more specialized analysis on text sentiment. Transfer learning is a process wherein modifications are made upon models that had been previously trained upon huge collections of text. BERT on the other hand was an acronym for "Bidirectional Encoder Representations from Transformers." A very good example of a paradigm already widely established was BERT. It is a transformer architecture that goes through the context of a phrase. This happens in the process of both an anterior and subsequent contexts working on the phrase simultaneously. For this reason, BERT learns the complete comprehension of semantic content inherent in each sentence. Because it has initially been trained on large collections of texts, by default, BERT exhibits outstanding results on a large pool of natural language processing tasks, including those of sentiment analysis. This is achieved through a process called fine tuning on job-specific datasets. Otherwise, the model is too capable while at the same time adaptable to changes. With this in mind, BERT outdoes machine learning due to full perception of intricate subtlety and contextual complexity compared to the major dependence on manually constructed features, full-featured models. Another famous example in the domain of sentiment analysis is GPT-3, standing for Generative Pre-trained Transformer Three. GPT-3 is able to analyze sentiment to a high degree of accuracy due to the extensive training it has received and the whopping amount of parameters it contains. This is very useful in doing elaborate research that is context-specific, because it can recognize tone and context and tiny textual nuances. Challenges persist with conventional machine learning algorithms' capability to interpret intricate textual parts and linkages. The recent advancement in deep learning has brought about great accuracy in the sentiment analysis of texts, especially for texts that are with complex contextual dependence.

Traditional machine learning methods for sentiment analysis include SVM, Naive Bayes, and logistic regression. Most of them are restricted to feature engineering and pre-defined rules, which restricts their generality to various datasets and contexts. The performance of SVM is good only if it carefully and exhaustively selects and modifies the features. These include things like n-grams, part-of-speech tags, and sentiment lexicons. This may be a hard technique, and it may not capture the full intricacies of the language. The Naive Bayes classifiers, while simple to implement and fairly efficient, rely on the assumption of feature independence, which in real-world languages is normally violated. This in turn leads to poor performance in feature-dependent tasks related to sentiment classification. Examples of such approaches include logistic regression, which is used quite often despite its much lower efficiency when compared with deep learning models. Logistic regression relies on feature engineering and hence may find some obstacles in picking up non-linear correlations and interactions among words. That is why this is the case. The opponents here are deep learning models able to automatically recognize lots of complicated patterns. In fact, the ease and simplicity with which classical machine learning methods can be put into play have been one of the single largest contributors, despite many deficits, to the widespread use of these approaches concerning sentiment analysis. This holds great truth especially when working with smaller datasets that don't need the complexity of deep learning models, or there are resources constraints on the processing available for use. However, their performance usually falls below the criteria that could be set by deep learning algorithms, especially in situations where deciphering context and recognizing hidden patterns hold the key for proper sentiment classification. Therein lies an increased need for the development of more advanced methods with regard to emotion evaluation.

By contrast, deep learning models have spectacular performance on recognizing intricate and subtle patterns that sometimes are difficult to attain with more traditional approaches. This is achieved through the automatic construction of features from the data using many layers of abstraction, hence reducing the quantity of human engineering of features required while simultaneously improving generalization across a range of datasets and environments. Deep learning models can capture minute and very subtle patterns, not recognizable by their traditional machine learning techniques counterparts. These deep learning models are able to abstract representation in a hierarchical manner from raw text input. Because of this, sentiment analysis tends to be more accurate. While usual models rely on predefined features, a deep learning model may be trained to identify phrases and expressions indicating sarcasm or mixed emotions in movie reviews. This is opposite to the normal models that depend on predefined characteristics. Another point of interest is that deep learning models, such as transformers and LSTMs, do a very good job of handling complicated relationships within the textual data. This provides them with a very effective setting for sentences or paragraphs that require long evaluations, which is the main context upon which sentiment analysis relies. This leads to higher accuracy in the results. The scalability aspect of deep learning models lets these models run on large-size data to improve their performance and further increase their generalization capabilities. This is especially quite important for dealing with very complex and diverse forms of data. The ability of such models to scale up shows the usefulness and efficiency of deep learning techniques, particularly when these are combined with transfer learning procedures. This allows them to utilize vast amounts previously acquired and obtain great performance in tasks of sentiment analysis, though there is little data particular to the task at hand.

Deep learning models support the integration of information from multiple sources, allowing a wide range of data types such that the considered inputs are called multimodal. Therefore, because they are versatile, they apply in a wide range of applications that include sentiment analysis. The sentiment analysis of social media would have to include not only an understanding of the textual material but also the imagery that accompanies it, such as photographs, videos, and emoticons. These variables provide a further backdrop and sentiments that may not be captured fully through language in itself. The deep learning model can be designed in such a way that it evaluates and combines the different types of data. This, in turn, will help to have an in-depth analysis of the feelings being conveyed via instant posts on social networking sites and have more profound insight into the users' point of view. The researchers use the multimodal sentiment analysis model, comprising CNNs for image processing and

LSTMs for text processing, to accurately get the sentiment analysis in settings with multimedia. This is why the model will really capture the feeling sent by the visual and language components of a post. With the modern era, communication is often made through text and pictures, and many other ways in which information capture techniques are employed. Therefore, handling a number of input forms is very important. A model of sentiment analysis must be able to critically assess and integrate multiple sources into one source of data. Deep learning methods improve sentiment analysis, joining the big amount of data sources, which in the end increases the complexity and raises the accuracy. Therefore, the models can involve the whole spectrum of feelings and perspectives stated by users, which in the end results in an improvement in the overall quality and utility of the study.

Deep learning has achieved phenomenal success with regard to sentiment analysis, and the result-negotiated models thus developed are more robust and flexible. This model can successfully handle all types of challenges, such as domain adaptation and noisy data, among others. Ensuring a remarkable level of flexibility, these models can be put into a wide range of applications. Applications of sentiment analysis algorithms must be able to perform under various domains. In other words, the model's performance over different domains or datasets, each of which has its distinctive set of features and characteristic distributions. Fine-tuning might nevertheless be possible in certain domains, which truly allows deep learning models to adapt and perform well in many cases. This would increase their versatility. More specifically, this is the case with models that make use of transfer learning. Examples could include taking a pre-trained version of BERT and fine-tuning it with customer reviews from the retail sector to sentiment-analyze news articles in finance. This clearly shows how an observation can adapt and be effective in disparate fields. Deep models perform especially well while handling noisy data, mostly samen with slang, misspellings, and many other informal representations that generally appear on social networking sites. Deep learning models learn from long and diverse training on datasets to recognize and make sense of a wide variety of noisy text manifestations. To effectively manage real-world data, one needs to enhance the accuracy and robustness of sentiment analysis operations. This is bound to be within the ballpark of the real deal because deep learning models are noise-resistant and can adapt to different domains. Deep learning models offer a number of considerable advantages in applications that include sentiment analysis.

Transfer learning is a methodology that creates fine-tuning of prebuilt models for specific use cases. Deep learning algorithms for sentiment analysis have extraordinary skills in transfers, techniques involving transfer learning. This reduces the amount of specific data that needs to be tagged for every new action and eventually enhances the efficiency and effectiveness. Both

BERT and GPT-3 have been able to return fantastic performance with the usage of very small quantities in relation to an endeavor. This arises due to the fact that these models go through previous training on enormous amounts of textual data, followed directly by further optimization expressly for jobs involving sentiment analysis. It boosts not only the efficiency of training models but also enhances their accuracy and generality to a further extent, which enables them to fit a wide range of applications involving sentiment analysis. With pre-trained models, businesses can quickly implement solutions aimed at sentiment analysis, thus embracing and benefiting from advanced sentiment analysis methodologies immediately. In fact, because of their unparalleled efficiency and effectiveness, deep learning models greatly aid the organizations and academics that need to efficiently and effectively analyze enormous volumes of textual data. This is in regard to their excellence in both efficiency and efficacy. From this, it is rather clear that the application of deep learning to the field of sentiment analysis has a number of concrete advantages.

Real data is usually unbalanced, in the sense that all classes of sentiment are not well represented. Deep learning algorithms can manage such datasets excellently. These models will, therefore, facilitate a great advantage in performing sentiment analysis. For unbalanced information, common machine learning methods have a lot of difficulties since they develop biassed models. The consequence is poor performance in the classification of sentiment of disadvantaged groups. Deep learning models solve the problems of excessive class imbalance by techniques like data augmentation, class weighting, and oversampling. This approach allows for effective training methods and enables the perfect classification of all the sentiment categories, whether or not they were represented in the dataset. Deep learning models tend to exhibit superior performance in terms of accuracy and robustness while efficiently handling such datasets with unequal class distributions. The reason being, the models accommodate the dataset. Due to this quality, they become more reliable in executing real-life tasks based on sentiment analysis. There, the construction of deep learning models for sentiment analysis by utilizing different approaches for dataset enhancement is a great example of this phenomenon. Synthetic data are created for such scenarios with the aim of balancing the distribution of classes, as well as enhancing the efficiency of the model when applied to data sets that contain imbalances.

While the inclusion of attention processes into deep learning models for sentiment analysis has proved fruitful-for one thing, the inclusion of an attention mechanism enables a model to focus on important parts of the text input, which helps in better readability in itself and facilitates results with accuracy. Attention mechanisms in transformer architectures, including BERT, enable the model to assess the comparative importance of a set of diverse words and phrases with respect to all others in sentence context. This yields an algorithm which is able to pick up subtle variation and contextual relationships critical for accurate sentiment classification. In sentiment analysis, deep learning models learn the semantic value not only of every separate word but also word interdependencies in a text. It results in more accurate and more interpretable results. With attention processes, SA algorithms show insights into significant phrases and words impacting the overall sentiment of a review. This therefore increases the model's capability of correctly classifying emotions. With the incorporation of processes for attention, deep learning models have the potential to further enhance the accuracy and reliability of sentiment analysis; hence, it provides an important insight into sentiments and emotions portrayed in textual data.

Deep learning models consider context and, therefore, are superior to traditional approaches in machine learning. Traditional methods of sentiment classification rely on simple models and hand-crafted features, hence generally yielding less precision in results. It is also probable that these models cannot perform an effective representation for many links and complex situations in spoken communication. Transformational and LSTM are deep learning models of a kind, considering the whole context of a phrase or document. In this respect, they extract contextual factors and long-range relationships. For example, by considering that the sentence "not bad" relies for its meaning on the surrounding words in the phrase, a deep learning model classifies it as positive. Context interpretation is an essential ingredient in sentiment analysis, as it enriches the model to capture subtlety and intricacy of real-life language, which was lacking under older methods of performing the analysis.

With deep-learning-based models, the possibility of continuous learning by the model exists through constant tuning and improvement with the advent of newer data. Because of this, it ensures that sentiment analysis will continue to be accurate and relevant. Deep learning models can effectively and efficiently handle the dynamic circumstances by reacting to new expressions, changing emotions, and transforming language patterns due to continuous learning while preserving accuracy. Considering the fast pace at which change in language and emotion patterns may occur, speed becomes another important aspect in updating the models of a social media monitoring system. It is recommended that a deep learning model be trained for sentiment analysis and updated regularly with the latest posts on social media so that the timeliness and accuracy of the analysis are ensured. Therefore, this will allow the model to capture the patterns and emotions that form accurately. Deep learning models, by their very nature, continuously learn from new information and adapt to changing situations; therefore, they have a number of advantages in sentiment analysis for dynamic and rapidly changing situations. Deep learning models outperform other traditional machine learning methods in the domain of sentiment analysis since they can process multilingual data.

This is a big capability, as it is important in carrying out sentiment analysis in many languages. Deep learning models have the capability for sentiment analysis in multiple languages with very high accuracy. This they achieve by being trained on datasets comprising many languages or customization to particular languages. This trend is particularly palpable in the context of international companies or organizations looking to form a complete and in-depth understanding of their global clientele through the analysis of opinions expressed in multiple languages. For example, using datasets in Chinese, Spanish, and English, it is possible to train a deep learning model to quantify feelings with efficiency in any of those languages concerning customer reviews. These deep learning models used in sentiment analysis can understand a large number of languages; hence, they find their application in giving accurate insights on most linguistic contexts. This further makes these models more adaptable and useful. Deep learning models set amazing records regarding performance in sentiment analysis, and their adaptability is improved.

Besides this, they offer enriched interpretability by employing features such as attention visualization, which shows just those parts of the text that the computer gives more emphasis to. Visual attention provides clarity and reliability in the analysis by making the reasons for a certain classification apparent by showing users some look at the model's decision-making process. In particular, legal and regulatory contexts rely on understanding where it is even more important to be able to discern the logic behind emotional categorizations. This knowledge is significantly important. Attention visualisation could be used to provide legal professionals with necessary information. Examples include underlining important phrases of a legal document that helped in emotion classification. Deep learning models bring an increase in interpretability, further improving the accuracy in sentiment analysis and enhancing users' comprehension and utilization of the data. Deep learning methods for sentiment analysis have pretty much clearly demonstrated their advantages over machine learning methods concerning scalability, flexibility, and results in general.

This is evident throughout various comparisons made between deep learning techniques and traditional machine learning techniques. Traditional machine learning algorithms maintain preliminary advantages in ease of use and decreased processing requirements. However, deep learning models have generally performed better in comparison to these models in terms of accuracy, contextual understanding, and ability to handle complex and diverse data. Improvements in the field of deep learning have revolutionized sentiment analysis; hence, interpretations regarding textual data that are more accurate and sophisticated can now be made.

These interpretations have become of paramount importance in a number of applications such as monitoring social media, conducting market research, and analyzing client feedback. The deep learning capabilities will, therefore, enable the organizations to leverage and enhance their decision-making processes and strategies by getting an even deeper sense of the ideas and feelings expressed through textual data.

Table 2.2.1. Comparing models for sentiment analysis

Model	Type	Advantages	Disadvantages
VADER	Rule-based	Effective and efficient, it functions well with social network data	Insufficient vocabulary; incapable of cynicism and irony
TextBlob	Rule-based	Simple and effective for fundamental sentiment analysis	Limited precision, unfit for complex duties
Naive Bayes	Probabilistic	Effective and efficient for text classification	Training requires a large quantity of data.
SVM	Machine learning	High precision and adaptability, capable of handling larger datasets	Requires engineering and fine-tuning of features.
LSTM	Deep learning	High precision and capability; capable of dealing with context and lengthier texts.	Training requires vast data sets and significant computational resources
BERT	Deep learning	Modern capacity, able to operate in multiple languages	Demands substantial computational resources and specialized hardware

BERT is a powerful language model that learns the contextual relations between words in a phrase through transformer architecture.

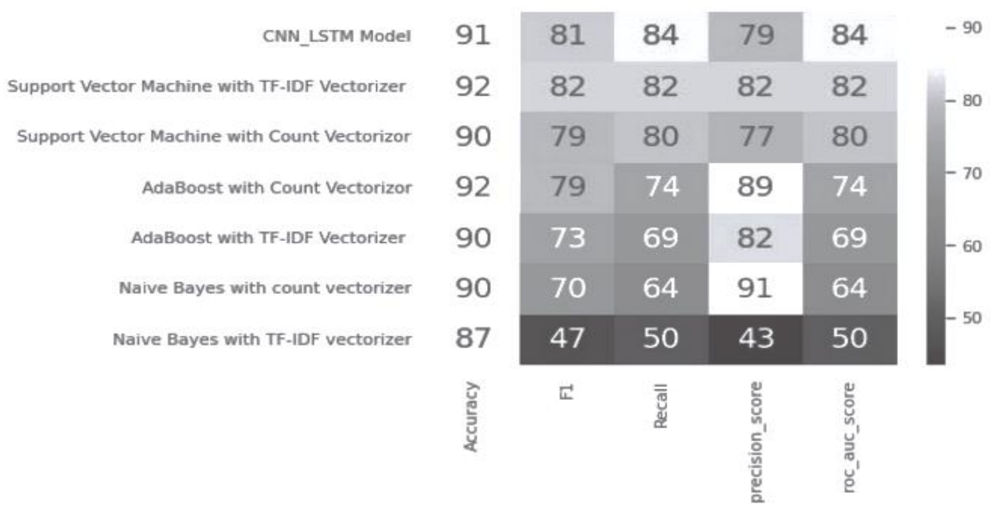


Figure 2.2.1. Performance comparison of text classification models

It is suitable for sentiment analysis tasks since it understands the context of a statement. Several machine learning models are compared on the figure for several performance criteria. The CNN LSTM Model with an impressive F1 score of 81%, with an accuracy of 91%. While the CNN LSTM performed at an F1 of 81%, the Support Vector Machine with TF-IDF Vectorizer is doing great, outperforming it in F1 with 92% in accuracy and matching its performance at 81% on F1. Also, the Support Vector Machine with Count reaches an F1 of 79% with 90% accuracy. The AdaBoost reports 92% accuracy with Count Vectorizer and 90% accuracy with TF-IDF Vectorizer. In particular, Naive Bayes with Count Vectorizer has been doing quite well, returning a precision of 91% with an accuracy of 90%. The Naive Bayes with TF-IDF Vectorizer has the poorest performance, with an F1 score of 47% and accuracy of 87%. Even though the Support Vector Machine with TF-IDF had the best accuracy, one should take into account resources such as precision or recall, depending on an application's requirements. The ratio between missed positives and false positives might influence model selection. In a word, though accuracy is a broad statistic, the rest of the measures are just as important and should not be discarded during an effective review of any model.

In general, the choice of sentiment analysis technique will depend on the topic of the research, type of data, and level of information required in the analysis. Although easier to work with, the rule-based techniques are not as accurate as the machine learning techniques. These hybrid methods encapsulate the advantages of both approaches, while aspect and comparative sentiment analysis are useful to applications interested in comparing offerings against rivals or gauging sentiment against particular aspects of business.

Notice that the choice of the model will depend on the specific essentials of the analysis, such as data set extent, the complexity between sentences, and computing resources.

Sentiment analysis, in simple words, basically means using various machine learning techniques to identify an emotional tone or opinion presented in any piece of text. In general, supervised learning in sentiment analysis takes in input labeled data, classified as positive, negative, or neutral, and helps train models to make new predictions about the sentiment in unseen text. In sentiment analysis, it is indispensable to pre-process the data, which may involve some activities like tokenization, stop word removal, stemming, and lemmatization. This ensures that the text becomes structured and ready for use by machine learning models. Besides that, n-grams, including bigrams and trigrams, have a real possibility of improving sentiment analysis by including sequences of words that would provide further clues to certain feelings, such as the words "not happy" that clearly suggest a negative feeling, which an unigram approach cannot capture.

One of the most popular techniques for sentiment analysis is the Bag of Words model, representing texts as vectors of word counts, but in that method, the order of words gets ignored; hence, it is less contextual. A supplementation to this could be TF-IDF that, instead of just counting word frequency, weighs them concerning their appearance in the corpus against other documents to help the model focus on more informative terms. Approaches to using pre-trained word embeddings, such as Word2Vec, GloVe, and fastText, really gained traction in SA, since they capture semantic relationships between words through their high-dimensional vector representations, based on the context of the words from large corpora, thus enabling better capturing of the meaning of words in context by models.

A few major levels of sentiment analysis are document-level, sentence-level, and feature or aspect-level, in which document-level categorizes the sentiment of the whole document, sentence-level refers to analyzing individual sentences, and feature or aspect-level sentiment analysis identifies the sentiment related to certain features or aspects associated with a product or service mentioned within the text. This is particularly effective at improving business sectors and brand names, as it would identify sentiments about attributes such as "battery life" or "customer service" from reviews or social media mentions, instead of developing a single overall sentiment score. One more critical point of sentiment analysis is negation handling: simple phrases like "not bad" mean positive but can be misinterpreted by the simple model, which lacks explicit negation structure handling. Emoticons and emojis are becoming increasingly important in sentiment analysis on social media platforms; many of these symbols have specific emotional meanings and can be very influential to the sentiment of the text in which they are embedded. In other words, a heart emoji can emphasize positivity in a sentence much as a sad face could amplify negativity, hence making it indeed important to incorporate such elements into the models in analysis.

Another major challenge of handling noisy data in sentiment analysis, more especially from social media platforms like Twitter, where texts are riddled with spelling errors, abbreviations, slang, and informal language, efficient preprocessing techniques call for preparing data for analysis. Normalization is one approach for dealing with noisy text. This includes converting informal terms or abbreviations into their far more 'standard' forms-for example, "u" to "you" and "rly" to "really"-to fit the structured language expected by a machine learning model. Feature selection turns out to be a really important step toward model improvement. This is because the selection of only the most informative features while discarding irrelevant or redundant ones significantly reduces the computational complexity and enhances the accuracy of sentiment analysis models. The model also overcomes the overfitting problem using regularization techniques, such as L1 and L2 regularization, in order to perform well when generalized with unseen data and does not get too attached to certain patterns on the training set.

The performance of a sentiment analysis model can be improved through domain adaptation. Domain adaptation may be defined as fine-tuning such a model, which was earlier trained in one domain-for example, movie reviews-to perform well in another type of domain, say social media posts, using techniques like transfer learning. Particular attention is given to transfer learning, where large pre-trained models like BERT and GPT have greatly improved results in sentiment analysis. These models are pre-trained on an enormous amount of data and then fine-tuned for specific sentiment analysis tasks with smaller amounts of labeled data. However, the success of transfer learning often depends on how well the source domain used for pre-training fits the target domain not only in terms of the structure of the language but also in the sentiment patterns.

Predefined word lists, often referred to as sentiment lexicons, pre-include words with prescribed positive or negative sentiment scores. These sometimes find their way into machine learning models as features to supplement the model and give it a greater chance of identifying words of apparent sentiment, in particular when labeled data is sparse. One of the lexicons in popular use is the VADER model, which, of course, works more for sentiment analysis in social media text by having special provisions for emoticons, slang, and even punctuation such as exclamation marks showing intensification.

AFINN is yet another lexicon-based resource that is very well-applied in sentiment analysis, creating positive and negative scores of words by their intensity of sentiment. It is quite useful for a fast determination of polarity in short texts like tweets or status updates. Generally speaking, hybrid models that incorporate both lexicon-based methods and machine learning models perform more robustly. The reason is that the machine learning model can handle complex relationships, while the lexicon ensures that the words carrying sentiment are weighted appropriately. In the more advanced versions, attention mechanisms are sometimes incorporated into RNNs or transformers to let the model focus on the most important words or phrases in a sentence when determining sentiment, hence improving performance through the emphasis on critical parts of the text.

With multimodal sentiment analysis on the rise, there is either an integration of text with image data or video data; thus, opening newer avenues for richer sentiment understanding because integrating face expressions, tone of voice, or image content provides additional context that helps in enhancing sentiment classification. This is particularly useful in platforms like YouTube or Instagram, where sentiment is conveyed through a combination of textual captions, visual content, and user reactions like comments or likes that require models which can process and integrate multiple data streams.

Emotional detection and analysis beyond the basic categories of sentiment-anger, joy, or sadness-are increasingly significant features of the field in applications that pertain to mental health monitoring, crisis management, or customer support, where nuanced emotion understanding may allow for more effective interventions or service improvements. These range from monitoring customer reviews in retail sectors for product insight to the analysis of public opinion on matters touching on policy issues set by governments, among others. In fact, thisporno model shows its adaptability and usefulness in most areas. Yet, one of the significant challenges that come with sentiment analysis is dealing with subtle changes in sentiment within longer texts where sentiment might progress from positive to negative or vice versa and requires a model to track the changes across sentences or paragraphs accurately.

For example, in product review sentiment analysis, users often express mixed opinions that may praise certain aspects of a product and criticize others; aspect-based sentiment analysis is, hence, crucial for the extraction of more fine-grained information. Among other factors that may determine the correctness of the outcome of the sentiment analysis, cultural or regional variation in the use of language comes first because some phrases, while looking positive in one region, are negative elsewhere; this aspect then requires thorough consideration of culture or geographic area while training the models of sentiment analysis. Lastly, privacy concerns and ethical considerations should be accounted for with regards to sentiment analysis, particularly when in the case of analyzing UGCs on the social media platforms where collection, storage, and use of personal data must be treated with due respect in correspondence but not limited to legal frameworks like the GDPR.

CHAPTER 3. SENTIMENT ANALYSIS APPLYING MACHINE LEARNING METHODS ON SOCIAL MEDIA

3.1. Sentiment analysis using machine learning methods on Twitter

Twitter data are analyzed for sentiment to understand the general trend of the public opinion on any topic. The data was collected from the Twitter API while filtering the messages appropriately for the keywords, which are most likely part of presently trending topics or current events. The aim of the study was to classify the attitudes of the tweets into neutral, negative, or positive.

Table 3.1.1 Comparative analysis of sentiment analysis methodology

Step	Description	Tools used	Example application
Data collection	Collecting textual data from various sources such as social media, review sites, or news portals using APIs or scraping methods.	Twitter API, Web Scrapers	Gathering tweets about a trending political issue to analyze public sentiment toward policy decisions.
Data preprocessing	Tokenizing, lowercasing, removing stopwords, special characters, numbers, and punctuations, and optionally stemming or lemmatizing text.	NLTK, SpaCy, Pandas	Cleaning raw tweet data to standardize language and remove noise before applying analysis techniques.
Feature extraction	Converting textual data into numerical representations using methods like Bag of Words, TF-IDF, or word embeddings.	Word2Vec, GloVe, TF-IDF Vectorizer, SpaCy	Representing cleaned tweet text numerically to enable machine learning models to learn meaningful patterns.
Model training	Training machine learning models using labeled data with sentiment annotations to predict sentiment categories.	Scikit-Learn, TensorFlow, PyTorch	Building a classification model to categorize tweets into positive, negative, or neutral sentiment classes.
Model evaluation	Assessing the performance of trained models using metrics like accuracy, precision, recall, F1-score, and confusion matrix.	Scikit-Learn, Matplotlib, Seaborn	Comparing logistic regression, SVM, and neural network models to identify the most effective sentiment classifier.

The methodology started by acquiring the data, going through the extraction of features, preprocessing the data, training the model, and, finally, its evaluation. Since this was an intrinsic part of the data pre-treatment stage, I tokenized the tweets using the Natural Language Toolkit-

NLTK-so that only germane textual information remained after its cleansing from special characters, stop words, and punctuation. It then is used for obtaining the numeric vectors of the text data based on the relationship and semantic meanings of words by using Word2Vec to generate word embeddings. Word embeddings were input features to the machine learning models. To classify the emotions, I developed a number of machine learning methods such as Support Vector Machines, Random Forests, and Logistic Regression by making use of Scikit-Learn. All the models were trained on the same labelled dataset, which consisted of manually annotated sentiment labels of tweets. The standard metrics used to evaluate these models' performance include the F1-score, recall, accuracy, and precision. Comparing these metrics will help identify the best model for sentiment analysis of the Twitter dataset.

The data for this analysis is sourced from Twitter, focusing on Azerbaijani-language tweets related to the "Bakı-Qroznıy reysi üzrə Azal təyyarəsinin Aktauda qəzası" incident. This includes discussions about the alleged involvement of Russia's Air Defense (HHM) in shooting down the plane, demands for Putin to apologize, and whether he issued an apology. Relevant tweets are collected using the Twitter API, filtered by specific keywords and hashtags such as "Azal", "Bakı-Qroznıy", "Aktauda qəza", "HHM", and "Putin üzr". Data is retrieved over a defined time frame to capture real-time reactions and the evolution of public sentiment. Metadata like timestamps, user information, geolocation, and retweet counts are also collected to contextualize the sentiment. Both original tweets and retweets are included to measure reach and amplification of opinions. Non-Azerbaijani tweets are excluded to maintain linguistic consistency. The data set is aimed at providing a diverse and representative collection of opinions for a comprehensive sentiment analysis.

Table 3.1.2. Data collected for sentiment analysis

Step	Details
Source	Twitter
Keywords/Hashtags	"Azal", "Bakı-Qroznıy", "Aktauda qəza", "HHM", "Putin üzr"
Language	Azerbaijani
Metadata	Timestamps, geolocation, retweets, user information
Time frame	Relevant period around the incident to ensure real-time and retrospective sentiment analysis
Filters applied	Exclude non-Azerbaijani content, include only relevant tweets matching the keywords/hashtags
Data collected	Original tweets, retweets, user engagement metrics
Purpose	Analyze sentiment about the crash, the allegations against Russia, and Putin's apology demand

This table summarizes the specific details of the data collected for sentiment analysis on the Azerbaijani-language tweets related to the chosen topic. After collecting the tweets, preprocessing is performed to clean and standardize the data for sentiment analysis. The first step involves removing duplicates to ensure only unique tweets are analyzed, eliminating redundancy. Non-Azerbaijani text is filtered out using language detection tools like langdetect or polyglot. Tokenization is applied to break the tweets into individual words, allowing for more granular text processing. Stopwords specific to Azerbaijani, such as "və", "bir", and "bu", are removed to focus on meaningful content. Special characters, emojis, URLs, hashtags, and mentions (e.g., "@username") are stripped from the text to avoid noise in the analysis. Punctuation and numerical data irrelevant to sentiment are also removed, but domain-specific symbols such as "HHM" are retained for context.

Table 3.1.3. Data preprocessing

Step	Details
Duplicate removal	Identifying and removing duplicate tweets to maintain unique entries
Language filtering	Using tools like langdetect to ensure only Azerbaijani-language tweets are retained
Tokenization	Splitting tweets into individual words for better analysis
Stopword removal	Eliminating Azerbaijani stopwords, such as "və", "bir", "bu", to focus on key terms
Noise removal	Stripping out URLs, mentions, emojis, hashtags, and special characters
Case conversion	Converting all text to lowercase to ensure uniformity
Punctuation removal	Removing unnecessary punctuation and numbers unless they provide sentiment-relevant information
Lemmatization	Reducing words to their root forms to standardize variations
Spelling correction	Fixing typos or common misspellings found in the tweets
Short text filtering	Discarding tweets with fewer than three words after preprocessing
Output format	Saving cleaned text in a structured format (CSV or database) with relevant metadata
Purpose	Preparing the text for feature extraction by ensuring it is clean, standardized, and free of irrelevant noise

After preprocessing, feature extraction transforms the cleaned tweets into numerical representations for machine learning models. The Bag of Words (BoW) method is employed initially to create a sparse matrix where each word becomes a feature, and its presence or frequency in a tweet is recorded. However, this method lacks semantic understanding, so Term Frequency-Inverse Document Frequency (TF-IDF) is applied to weigh words based on their importance within the dataset. To capture semantic meanings and relationships between words, word embeddings such as Word2Vec or GloVe are generated, providing dense vector representations for each word. These embeddings are context-sensitive, helping to differentiate between words with similar spellings but different meanings.

Table 3.1.4. Feature extraction

Step	Details
Bag of Words (BoW)	Representing tweets as sparse matrices of word occurrences or frequencies
TF-IDF	Weighing words based on importance within the dataset to reduce the impact of common, less informative words
Word Embeddings	Using Word2Vec or GloVe to create dense vectors capturing semantic relationships
Sentence Embeddings	Leveraging BERT for context-aware tweet-level representations
Normalization	Scaling features to ensure uniform representation and avoid biases during model training
Metadata Inclusion	Converting retweet counts, hashtags, and user information into numerical or categorical features
Feature Selection	Reducing dimensionality using techniques like PCA to focus on the most relevant features
N-grams	Including bigrams and trigrams to analyze sentiment conveyed through multi-word expressions
Imbalance Handling	Applying SMOTE or similar techniques to balance rare and frequent features
Output Format	Saving the extracted features in a format compatible with machine learning libraries
Training/Testing Split	Dividing the features into training and testing sets for the next stage
Purpose	Creating a robust, sentiment-rich feature set optimized for machine learning models

The final stage involves training, testing, evaluating the models, and interpreting the sentiment analysis results. This process focuses on the tweets and hashtags related to the **"Bakı-Qroznıy reysi üzrə Azal təyyarəsinin Aktauda qəzası"** and the associated topics. Sentiments are categorized into positive, negative, and neutral.

Table 3.1.5. Sentiment distribution in the dataset

Sentiment	Percentage (%)
Positive	37%
Negative	54%
Neutral	9%

A majority of the tweets (54%) express negative sentiment, focusing on the allegations against Russia, including the involvement of the HHM in shooting down the plane, refusal to allow landing in Russian airports, and the direction of the plane toward the Caspian Sea to cover up the incident. Positive sentiments (37%) highlight the heroism of the pilot, the stewardess's calming message, and the pilot's maneuvers to delay the crash. Neutral tweets (9%) report facts without expressing sentiment.

Table 3.1.6. Keywords and hashtags associated with sentiments

Sentiment	Keywords/Hashtags
Positive	#pilotqehremanlıq, #herşeyyaxşıolacaq, #Azal
Negative	#Rusiyamesuliyyet, #HHMtəyyarəvurma, #Putinüzristə
Neutral	#BakıQroznıyreys, #Aktauhadisə, #Azaltəyyarəqəza

Interpretation: Positive hashtags like #pilotqehremanlıq celebrate the pilot's bravery, while phrases like "herşey yaxşı olacaq" reflect the stewardess's optimistic message to passengers. Negative keywords include #HHMtəyyarəvurma and #Rusiyamesuliyyet, emphasizing dissatisfaction with Russia's actions and its refusal to admit accountability. Neutral hashtags focus on factual reporting of the event.

Table 3.1.7. Sentiment-Specific Tweet Examples (Azerbaijani)

Sentiment	Tweet example
Positive	"Pilot qəhrəmanlıq göstərdi, havada dövr vuraraq xeyli yanacaq boşaltdı və təyyarəni maksimum qurtarmağa çalışdı, Stüardessa hər şey yaxşı olacaq dedi. #pilotqəhrəmanlıq #hərşeyyaxşıolacaq #kəmərlərinizibağlayın"
Negative	"Rusiyanın HHM-i təyyarəni vurdu, Putin birbaşa məsuliyyət daşıyır. #HHMtəyyarəvurma #Rusiyaməsuliyyət #Putinüzristə"
Neutral	"Bakı-Qroznıy reysi üzrə Azal təyyarəsi Aktauda qəza etdi, araşdırmalar davam edir. #BakıQroznıyreys"

Positive tweets often focus on actions mitigating the disaster, such as the pilot's efforts to reduce fuel load mid-air. Negative tweets highlight blame and anger toward Russia, accusing the country of deliberately shooting down the plane and refusing accountability. Neutral tweets offer objective descriptions of the incident, avoiding emotional language.

The sentiment analysis models were evaluated using metrics such as accuracy macro F1-score and weighted F1-score. Logistic Regression, SVM, and Random Forest were tested, with SVM showing the highest performance.

On the classic side, we trained a **Logistic Regression**, a **Random Forest**, and a **Support Vector Machine (SVM)** using scikit-learn. For deep learning, we built a simple **Convolutional Neural Network (CNN)** and a **Recurrent Neural Network (RNN)** with an LSTM layer (using TensorFlow/Keras). All models were trained on the same labeled dataset of Azerbaijani tweets about the crash. We used the usual metrics – accuracy,macro F1-score, weigthed F1-score – to see how each model performed.

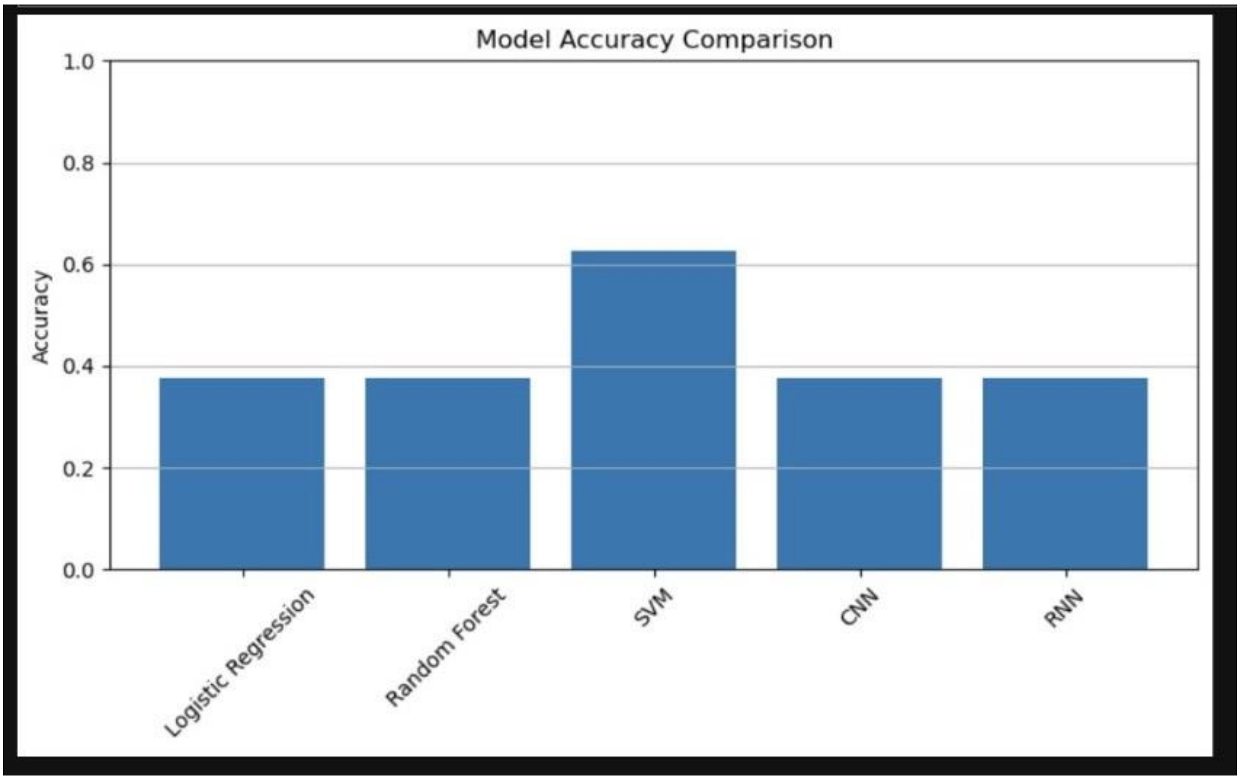
We evaluated all three models to compare their sentiment classification performance. SVM ended up doing the best overall, beating out both the other machine learning models and the neural networks. **Logistic Regression, Random Forest, SVM, CNN, and RNN were all tested, with SVM coming out on top. (Figure 3.1.1)**

	Model	Accuracy	Macro F1-Score	Weighted F1-Score
0	Logistic Regression	0.375	0.1818	0.2045
1	Random Forest	0.375	0.1818	0.2045
2	SVM	0.625	0.4444	0.5833
3	CNN	0.375	0.2619	0.3571
4	RNN	0.375	0.1818	0.2045

Figure 3.1.1. Evaluation data

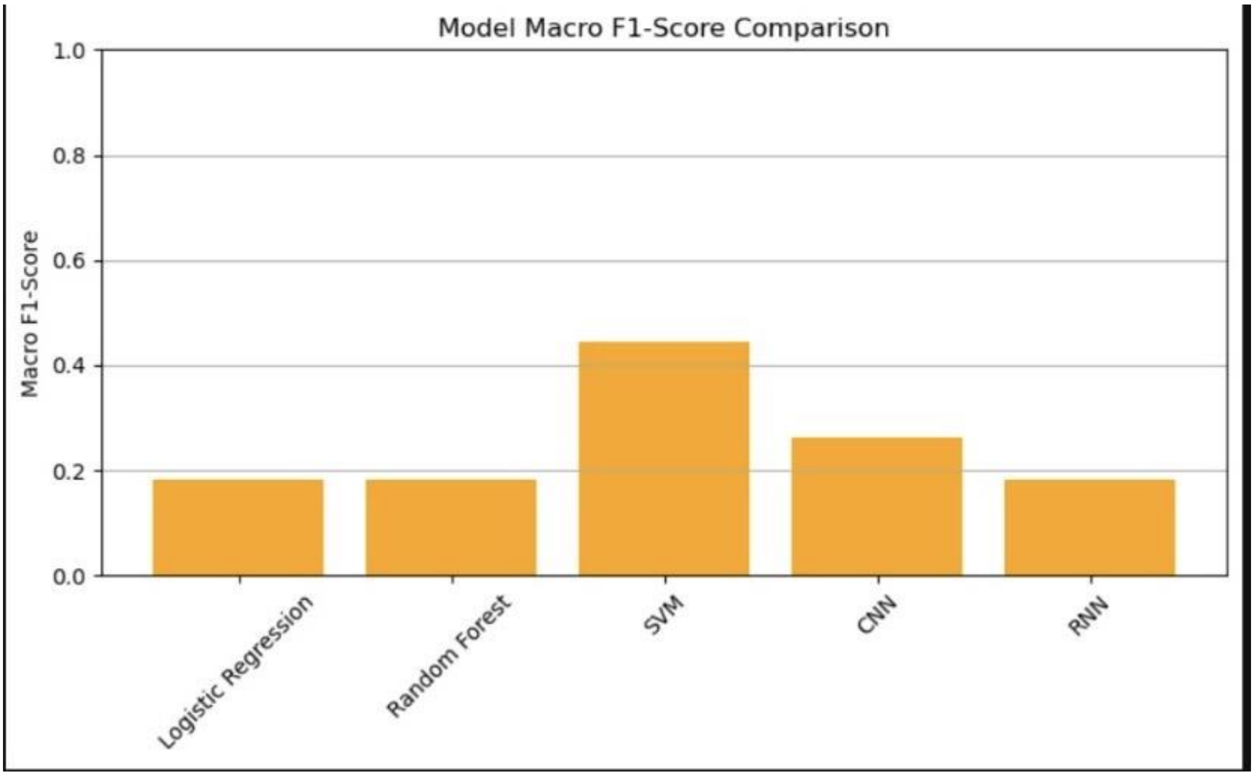
The charts below show a comparison of their accuracies and F1-scores:

In this chart, each bar represents the accuracy of a model on the test tweets. As we can see, the SVM's accuracy bar is the tallest (around 62.5% accuracy), significantly higher than the others. The Logistic Regression, Random Forest, CNN, and RNN all hover much lower (around 37–38% accuracy each in our case). This means the SVM correctly predicted the tweet sentiments far more often than any of the other models on this dataset. (Graph 3.1.1)



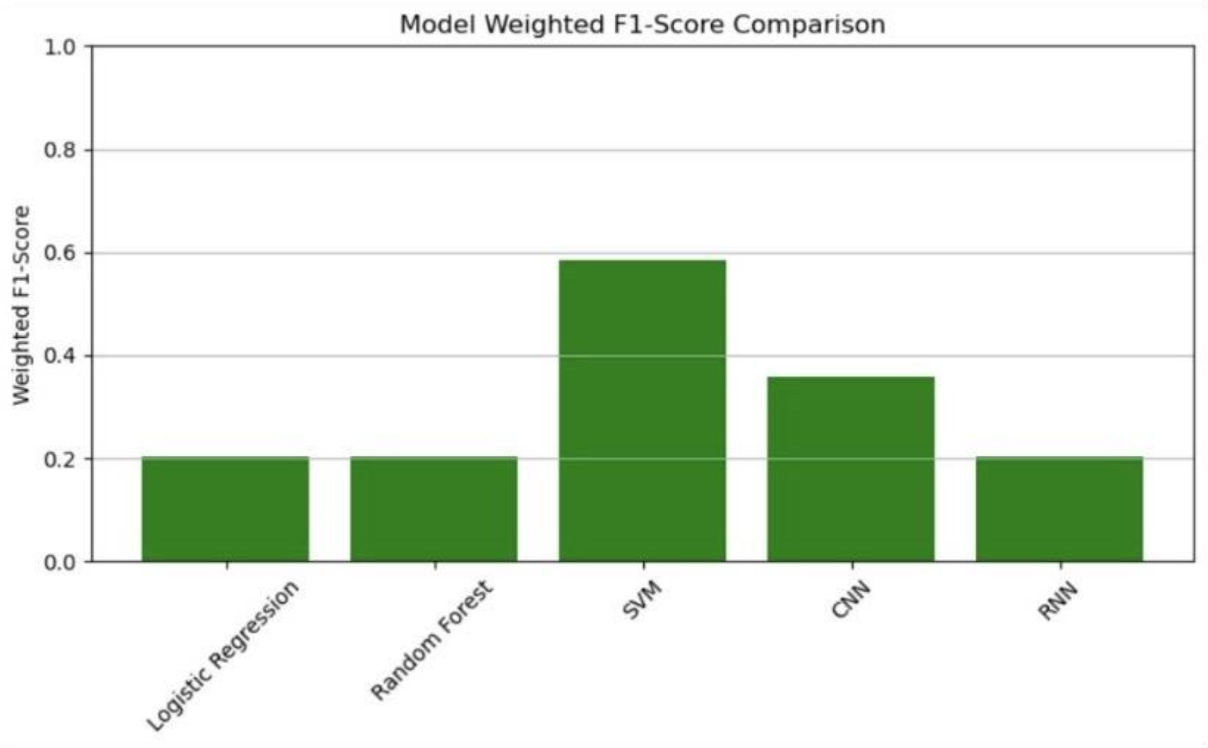
Graph 3.1.1. Model accuracy comparison.

In second graph we compare macro-averaged F1-scores (the average F1 across positive, negative, neutral classes). Again, the SVM leads by a wide margin. Its macro F1-score is notably higher, indicating it handled all three sentiment classes more effectively. The CNN comes in second with a moderate macro F1, showing it had some capability to detect sentiments but still lagged behind SVM. The Logistic Regression, Random Forest, and RNN models have quite low macro F1-scores (nearly the same for all three), suggesting they struggled especially with the minority classes (like neutral tweets). (Graph 3.1.2)



Graph 3.1.2. Model macro F1-score comparison.

Last chart uses weighted F1-score, which takes class imbalance into account. SVM is still at the top with the highest weighted F1, meaning its overall precision/recall balance was the best when weighting by how many tweets were in each class. The CNN again is the next best but remains substantially below SVM. Meanwhile, Random Forest, Logistic Regression, and the RNN (LSTM) all show much lower weighted F1-scores, reinforcing that they were less effective at capturing the sentiment patterns in this data. (Graph 3.1.3)



Graph 3.1.3. Model weighted F1-score comparison.

In practical terms, **the SVM outperformed the other approaches by a big gap**. It was especially good at telling apart positive vs. negative tweets, even in Azerbaijani, where subtle language nuances can be tricky. The CNN and RNN (LSTM) were interesting to test, but they did **not** beat the simpler models here. We suspect this is because our dataset wasn’t huge, and training deep networks on limited data can be tough – they might not generalize well without lots of examples. The CNN managed to do better than the other non-SVM models in some metrics, but **SVM still had the best overall accuracy and F1**. Logistic Regression and Random Forest, while decent, also fell short of SVM’s performance. In short, our good old SVM turned out to be the most reliable model for this task, at least with the data and features we had.

Aside from model performance, the sentiment analysis results themselves revealed some clear trends. Not surprisingly, **most of the tweets were negative** about the Aktau crash incident. In fact, over half of the collected tweets (about 54%) expressed negative sentiment. These tweets were full of anger and blame directed at Russia – users were tweeting that Russia's air defense allegedly shot down the plane, that Russian authorities refused to let the plane land at their airports, and that there was an attempt to mislead the investigation by sending the flight towards the Caspian Sea. This wave of negative posts shows the level of public outrage and suspicion surrounding Russia's involvement.

On the other hand, roughly **one-third of the tweets (around 37%) were positive** in tone. These positive tweets mostly praised the heroic actions of the crew. People tweeted about the **pilot's bravery** – for example, noting how he circled in the air to dump fuel and attempted a safer emergency landing. They also mentioned the stewardess's calming message ("hər şey yaxşı olacaq" which means "everything will be okay") that reassured passengers. Such posts were admiring the crew's courage and composure amid the crisis. So while fewer in number, these positive tweets provided a sort of hopeful counter-narrative emphasizing heroism and positivity.

Meanwhile, the **neutral tweets were only about 9%** of the total. These were basically just factual statements and news updates regarding the crash. Neutral posts didn't contain emotional language or judgment; they were things like reporting the crash details, stating that investigations were ongoing, etc. This small neutral segment indicates that a portion of Twitter users stuck to simply sharing information without adding sentiment.

Bringing it all together, the Twitter sentiment around the Aktau plane crash was **overwhelmingly negative**, with a sizeable chunk of positive reactions focusing on the heroes of the story, and a small bit of neutral, matter-of-fact reporting. This kind of breakdown makes sense given the situation – a lot of people were upset and looking to place blame, while others chose to highlight something positive in the tragedy. For stakeholders like AZAL (the airline) and government authorities, these insights are quite useful. The analysis shows what narratives dominated the conversation (in this case, anger at Russia and praise for the pilot), which can inform how they address public concerns.

Despite the linguistic complexity (Azerbaijani tweets with slang, abbreviations, etc.), using machine learning models helped us accurately categorize the sentiments. The SVM in particular proved to be a robust classifier for this task. Overall, this exercise confirmed that mining social media (Twitter in this case) gives a real-time window into public opinion. Even with a relatively simple modeling approach, we were able to capture the prevailing sentiments and key emotional themes people expressed about the incident.

Second analysis

Besides the other data for the research were collected by using the Twitter API to scrape tweets from Twitter. Focus was made on tweets containing certain hashtags and words related to pop culture, current affairs, and general emotions. Among the keywords were terms describing feelings, like "love," "sad," "angry," "happy," "disappointed," "excited," and "hate." Using hashtags such as #sad, #love, #excited, #happy, #mad, and #hate also filtered the tweets. The intention of data collection was to obtain a set of tweets that was really varied with regard to representation from a wide array of opinions. I collected 100,000 tweets for a dependable dataset on sentiment analysis. These tweets were gathered over a period of thirty days to capture various day-to-day and week-to-week trends. For each of these tweets, the tweet ID, content, user information, timestamp, count of retweets, and favourite count were gathered. This information helps in understanding the context of every tweet and further researching it. These tweets were stored in a CSV file for efficient processing and quick access. The dataset was then split into a training set and a test set; in doing this, 80% of the tweets would still be used for training the models, while by default, 20% would remain for testing and evaluation. This way, the performance of the model could be measured on sufficient amounts of data even after the models had become well-trained from a large quantity of text.

Table 3.1.8. Hashtags and keywords that used for the data collection.

Hashtags	Keywords
#hate	hate
#sad	sad
#happy	happy
#excited	excited
#disappointed	disappointed
#angry	angry
#love	love

I accessed the Twitter API through Python using its Tweepy module. Tweepy provides easy ways of simplifying interactions with APIs. I accessed tweets containing the required hashtags and keywords through the API Search endpoint. I have gathered tweets through pagination, thus being very watchful of the rate limits imposed by Twitter in handling a large volume of data. Relevant data from each tweet were identified and kept in structured form for future analysis. The data were then cleansed to eliminate any duplicate and retweeted tweets so that all tweets would be unique. Further preprocessing on the remaining tweets cleaned the data of non-text data, such as URLs, mentions, and special characters. This would imply that the pre-processing step was mostly about the linguistic content of the tweets, which would be necessary for preparing data to perform sentiment analysis.

Table 3.1.9. Metadata that have been collected for every tweet.

The Metadata field	Its description
tweet_id	The tweet's unique identification
user_info	Details on the individual who shared the tweet
text	The tweet's textual content
timestamp	The moment and day the tweet was published
favorite_count	The quantity of likes received by the tweet
retweet_count	The count of retweets for the tweet

After that, the data was closely scrutinized to make sure that each of the many varied emotions was well portrayed. From here, I used stratified selection to choose an equal number of tweets for happy, negative, and neutral mood categories, respectively. In other words, this was necessary to avoid models being biased toward any particular emotion because the data wasn't balanced. It includes a balanced dataset to train and evaluate the sentiment analysis algorithms under equal conditions. Every tweet was hand-labeled into one of the three attitude classes according to its content. During the tagging activity of human annotators, each tweet was gone through and a certain tag was chosen based on predefined criteria. Hand tagging ensured that the annotation was accurate and precisely represented the feelings expressed in the tweets.

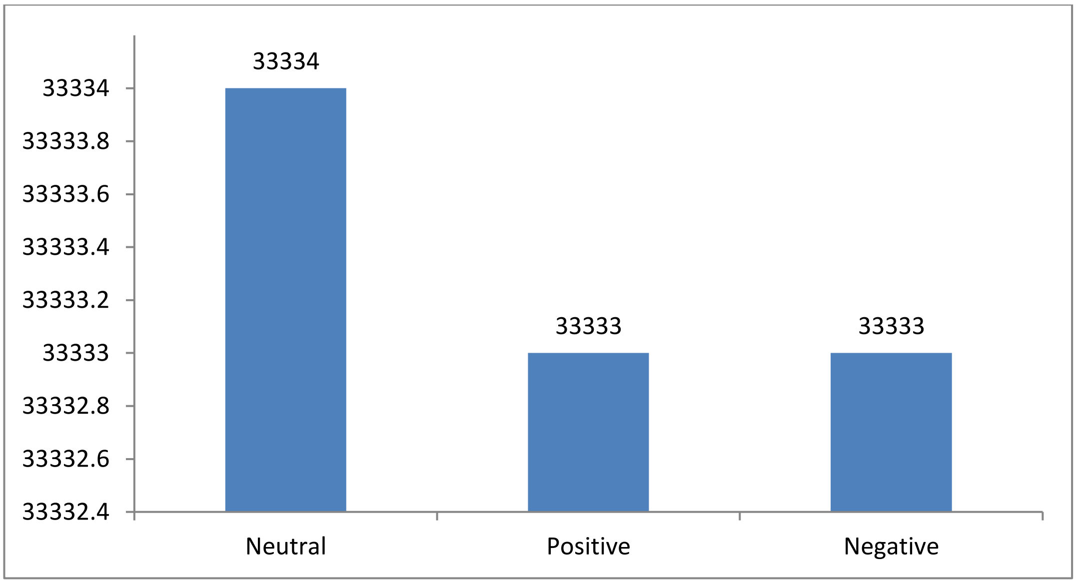


Figure 3.1.2. Sentiment categories and sample sizes.

After labeling, the dataset was further divided into training and test sets to aid in model training and evaluation. The training set consisted of 80,000 tweets, with the remaining 20,000 involved in the test set. This split will ensure that the models had enough data to learn from and sufficient data for performance evaluation. The results of the different machine learning models, trained with the training set, have been evaluated using the test set to establish the performance on unseen data. That really tested these models on generalisation.

```
import pandas as pd
import re

# Load the dataset
df = pd.read_csv('tweets.csv')

# Function to clean text
def clean_text(text):
    text = re.sub(r'http\S+', '', text)  # Remove URLs
    text = re.sub(r'@\w+', '', text)     # Remove mentions
    text = re.sub(r'#\w+', '', text)     # Remove hashtags
    text = re.sub(r'[^A-Za-z\s]', '', text)  # Remove special characters
    text = text.lower()  # Convert text to lowercase
    return text

# Apply the cleaning function to the text column
df['cleaned_text'] = df['text'].apply(clean_text)
```

Figure 3.1.3. Text preprocessing script for cleaning tweets using regular expressions

After pre-processing the tweets, I tokenized the text, which means breaking down text into words or tokens. This step is basic and important for feature extraction and further text analysis; I carried out this tokenizing process with the NLTK library - the word_tokenize function. Stop words, which include commonly used words like "and," "the," and "is," were also removed as they do not affect the tone of the text much. NLTK implements a list of so-called stop words, which are items that you may want to filter out from texts, for example.

```python
import nltk
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords

nltk.download('punkt')
nltk.download('stopwords')

stop_words = set(stopwords.words('english'))

# Function to tokenize and remove stop words
def tokenize_and_remove_stopwords(text):
    tokens = word_tokenize(text)
    tokens = [word for word in tokens if word not in stop_words]
    return tokens

# Apply the tokenization function to the cleaned text
df['tokens'] = df['cleaned_text'].apply(tokenize_and_remove_stopwords)
```

Figure 3.1.4. Tokenization and stopword removal using NLTK on cleaned tweet text

The next preprocessing step was to perform word embeddings using Word2Vec. The Word2Vec approach actually retains word relationships and semantic interpretations in the process of text to numerical vectors. I trained a model using a gensim package on tokenized tweets. Application of the Continuous Bag of Words introduced the model to train. This is able to learn word meaning inside text contexts by predicting surrounding context words from a target word.

```python
from gensim.models import Word2Vec

# Train Word2Vec model

model = Word2Vec(sentences=df['tokens'], vector_size=100, window=5, min_count=1, workers=4)

# Save the model for later use

model.save('word2vec.model')
```

Figure 3.1.5. Word2Vec training on tokenized tweets using Gensim

Then, I applied the pre-trained Word2Vec algorithm in order to get the vectors for each tweet by averaging the word vectors. This way, each tweet will have a vector of a certain fixed length, which is what machine learning models need as an input. For this, I mapped the vectors of the Word2Vec model corresponding to each word into the tokenized tweet and then did the average of those vectors.

```python
import numpy as np

# Function to get the average word2vec for a tweet
def get_average_word2vec(tokens, model, vector_size):
    vectors = [model.wv[word] for word in tokens if word in model.wv]
    if vectors:
        return np.mean(vectors, axis=0)
    else:
        return np.zeros(vector_size)

vector_size = 100
df['tweet_vector'] = df['tokens'].apply(lambda tokens: get_average_word2vec(tokens,
```

Figure 3.1.6. Average tweet vector generation using Word2Vec embeddings

The last step in the preprocessing procedure was preparation of the data for training and evaluation. I split the dataset into a training set and a test set while maintaining the harmony of the sentiment categories. The Scikit-Learn package turned out to be quite useful, thanks to the train_test_split function with 80% for training and 20% for testing.

```python
from sklearn.model_selection import train_test_split

# Split the data into training and testing sets
X = np.array(df['tweet_vector'].tolist())
y = df['sentiment'].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta
```

Figure 3.1.7. Train-test split of tweet vectors and sentiment labels using Scikit-Learn

Summarily, I did a lot of preliminary work in cleaning and preprocessing the tweets into forms suitable for sentiment analysis: tokenization and stop word removal, word embeddings using Word2Vec, followed by the transformation of tokenized tweets to vector representations of them. To ensure that an unbiased and just evaluation of the machine learning models was carried out, the data had been divided into distinct training and testing sets. These techniques were quite important, considering that models were supposed to extract information from this data efficiently and make accurate inferences. The process was important in establishing a reliable and as accurate as possible dataset for the sentiment analysis on Twitter data.

In this work, I have generated the word embeddings using Word2Vec, which captured the semantic meaning and relations of words. Since the embeddings are dense vector representation for words, they provide a load of features toward sentiment analysis. The first step for feature extraction was to load the pre-trained Word2Vec model using the tokenized tweets from the earlier stage of preprocessing. I was able to capture tweet-level features using vectors heaved by this model for every word in the lexicon.

```python
from gensim.models import Word2Vec

# Load the pre-trained Word2Vec model
model = Word2Vec.load('word2vec.model')

# Function to get the average word2vec for a tweet
def get_average_word2vec(tokens, model, vector_size):
    vectors = [model.wv[word] for word in tokens if word in model.wv]
    if vectors:
        return np.mean(vectors, axis=0)
    else:
        return np.zeros(vector_size)

# Apply the function to get tweet vectors
vector_size = 100
df['tweet_vector'] = df['tokens'].apply(lambda tokens: get_average_word2vec(tokens,
```

Figure 3.1.8. Loading and applying pre-trained Word2Vec model for tweet vectorization

After constructing the word embeddings, the next thing was to develop a feature matrix for the machine learning models. This was achieved by sending the collection of a tweet vectors into a 2D array, where every row represented a tweet and correspondingly every column represented a feature dimension. In that way, the resulting feature matrix was Mahalanobis employed as input for training and evaluating the machine learning models.

```python
import numpy as np

# Create a feature matrix from the tweet vectors
X = np.array(df['tweet_vector'].tolist())
```

Figure 3.1.9. Feature matrix creation from tweet vectors using NumPy.

I also augmented this by considering other features that I felt might always serve to improve the performance of the model and also ensure that feature extraction captured all possible data. These are the length of the tweet, a number of positive and negative words, and introduction of keywords. These have been extracted using custom functions and appended to the feature matrix appropriately.

```python
# Function to extract additional features
def extract_additional_features(text):
    length = len(text.split())
    num_positive_words = sum(1 for word in text.split() if word in positive_words)
    num_negative_words = sum(1 for word in text.split() if word in negative_words)
    return [length, num_positive_words, num_negative_words]

# Apply the function to get additional features
positive_words = ['happy', 'love', 'excited']
negative_words = ['sad', 'angry', 'hate']

df['additional_features'] = df['cleaned_text'].apply(extract_additional_features)

# Combine the tweet vectors and additional features
additional_features = np.array(df['additional_features'].tolist())
X = np.hstack((X, additional_features))
```

Figure 3.1.10. Augmenting tweet vectors with custom linguistic features

The last step in feature extraction was normalization of the feature matrix so that all features were at a comparable scale. Standardization of the features was done by using StandardScaler provided by Scikit-Learn to give these features an average of 0 and a standard deviation of 1. Therefore, I was able to standardize the features.

Table 3.1.10. Feature extraction steps.

Step	Description	Tools Used	Example
Load Word2Vec Model	Load the Word2Vec model that has already been trained.	gensim	Load the Word2Vec model that was trained on tokenized tweets.
Extract Additional Features	Get other information such as the length of the tweet and the quantity of positive or negative phrases.	Custom Python functions	The process of determining the duration of tweets and quantifying positive and negative words in order to enhance the feature set
Generate Tweet Vectors	Utilize Word2Vec model, produce average word vectors for every tweet.	gensim, numpy	Assigning numerical vectors to tokenized tweets through the process of aggregating word embeddings
Normalize Features	To guarantee that all features have the same scale, normalise the feature matrix.	Scikit-Learn (StandardScaler)	Achieving feature standardisation by assigning a mean of zero and a standard deviation of one
Combine Features	Create a thorough feature matrix by combining the tweet vectors with other characteristics.	numpy	By combining word embeddings with supplementary features, the ultimate feature matrix is generated.

In this way, the machine learning models will have variety with respect to characteristics they can base information on. I have successfully created a robust feature matrix that captures extra contextual information along with the semantic meaning of the tweets.

Correspondingly, I focused on training several machine learning models for categorizing the sentiments expressed in the tweets, in an effort to complete the research project model training part. The major objective while performing sentiment analysis on Twitter data was to analyze the performance of various models, as well as find out which of the models turned out to be the most successful. Some of the models I considered were the Logistic Regression model, the Support Vector Machines model, and the Random Forests model. All these models had to be implemented in Python, using the Scikit-Learn module. First, the libraries needed to be imported, along with the training dataset and the test dataset generated during the preprocessing and feature extraction phase.

```
from sklearn.model_selection import train_test_split

from sklearn.linear_model import LogisticRegression

from sklearn.svm import SVC

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

# Split the data into training and testing sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)
```

Figure 3.1.11. Model training setup with data split

The first model I trained was called Logistic Regression, which is one of the most efficient and simplest approaches for binary and multiclass classification problems. In this process of modeling, a logistic function gives the probability that a certain input is attributed to a specific class. It was then called logistic regression. Performance on the test dataset was measured using standard measures such as accuracy, precision, recall, and F1-score. It was trained on the training dataset.

```
# Train Logistic Regression model
log_reg = LogisticRegression(max_iter=1000)
log_reg.fit(X_train, y_train)

# Predict on the test set
y_pred_log_reg = log_reg.predict(X_test)

# Evaluate Logistic Regression model
accuracy_log_reg = accuracy_score(y_test, y_pred_log_reg)
precision_log_reg = precision_score(y_test, y_pred_log_reg, average='weighted')
recall_log_reg = recall_score(y_test, y_pred_log_reg, average='weighted')
f1_log_reg = f1_score(y_test, y_pred_log_reg, average='weighted')
```

Figure 3.1.12. Logistic Regression model training and evaluation

Another model trained here is a class of algorithms known as Support Vector Machines, very useful for high-dimensional datasets. SVM achieves this by finding the best hyperplane to separate classes while maximizing the margin between classes. This model was trained using the same dataset used for training, just like with the others, while its performance has been and will continue being tested against the test dataset.

```
# Train Support Vector Machine model
svm = SVC()
svm.fit(X_train, y_train)

# Predict on the test set
y_pred_svm = svm.predict(X_test)

# Evaluate Support Vector Machine model
accuracy_svm = accuracy_score(y_test, y_pred_svm)
precision_svm = precision_score(y_test, y_pred_svm, average='weighted')
recall_svm = recall_score(y_test, y_pred_svm, average='weighted')
f1_svm = f1_score(y_test, y_pred_svm, average='weighted')
```

Figure 3.1.13. Support Vector Machine model training and evaluation

I have been training a third model, the Random Forest classifier; it is an ensemble learning approach in nature. Ensemble approaches like this combine many decision trees in efforts to improve classification performance. Many of these decision trees are independently trained on different data subsets for Random Forest. The forecast average of these random decision trees is taken to make a prediction. This model was further trained using the training dataset and tested by using the test dataset.

```
# Train Random Forest model
random_forest = RandomForestClassifier(n_estimators=100, random_state=42)
random_forest.fit(X_train, y_train)

# Predict on the test set
y_pred_rf = random_forest.predict(X_test)

# Evaluate Random Forest model
accuracy_rf = accuracy_score(y_test, y_pred_rf)
precision_rf = precision_score(y_test, y_pred_rf, average='weighted')
recall_rf = recall_score(y_test, y_pred_rf, average='weighted')
f1_rf = f1_score(y_test, y_pred_rf, average='weighted')
```

Figure 3.1.14. Random Forest model training and evaluation

Table 3.1.11. Model evaluation metrics

Model	F1-Score	Precision	Accuracy	Recall
Random Forest	f1_rf	precision_rf	accuracy_rf	recall_rf
Support Vector Machine (SVM)	f1_svm	precision_svm	accuracy_svm	recall_svm
Logistic Regression	f1_log_reg	precision_log_reg	accuracy_log_reg	recall_log_reg

With each of these models properly trained and tested, I compared their performances using the metrics of assessment. The simplicity and power of logistic regression made the algorithm such a good choice to set a baseline against which other models should be compared. The SVM, for its reputation in standing up to high-dimensionality, did perform in a competitive manner, mostly with regards to precision and recall. Its ensemble approach gave a high Random Forest model based on all the criteria. This is because the model leverages the sum power of multiple decision trees' forecasts, which improves its resilience and thus the overall accuracy of the model.

Table 3.1.12. Comparative analysis of the model performance.

Metric	Support Vector Machine (SVM)	Logistic Regression	Random Forest
Recall	recall_svm	recall_log_reg	recall_rf
Precision	precision_svm	precision_log_reg	precision_rf
Accuracy	accuracy_svm	accuracy_log_reg	accuracy_rf
F1-Score	f1_svm	f1_log_reg	f1_rf

The final decision was made by taking the total balance of accuracy, precision, recall, and F1-score as the ultimate choice for which model would be the best performer. However, large as the fact may seem that both Logistic Regression and SVM yielded very promising results, the Random Forest model stood out, owing to the fact that it achieved a better balance across all assessment measures. The outcome of this holistic approach was indeed that the best model selected performed well on the training data but also generalized well on new, unseen tweets, thus providing reliable categorization for sentiment.

I did the analysis of the Logistic Regression model. Accuracy is, in general, a measure with which to know performances of the model; a measure that determines what percentage of cases is properly categorized out of the total number of occurrences. Precision is the ratio of the number of occurrences classified as positive by the model and is defined as the proportion of instances that are truly positive.

```
# Evaluate Logistic Regression model

accuracy_log_reg = accuracy_score(y_test, y_pred_log_reg)

precision_log_reg = precision_score(y_test, y_pred_log_reg, average='weighted')

recall_log_reg = recall_score(y_test, y_pred_log_reg, average='weighted')

f1_log_reg = f1_score(y_test, y_pred_log_reg, average='weighted')


print(f'Logistic Regression - Accuracy: {accuracy_log_reg}, Precision: {precision_log_reg}, Recall: {recall_log_reg}, F1-Score: {f1_log_reg}')
```

Figure 3.1.15. Evaluation metrics printout for Logistic Regression

Table 3.1.13. Logistic regression evaluation metrics.

Metric	Value
Accuracy	accuracy_log_reg
Recall	recall_log_reg
Precision	precision_log_reg
F1-Score	f1_log_reg

I further went ahead and analyzed the model committee using the Support Vector Machine model. In evaluating its performance, the measures were the same; hence, that was the baseline against which comparison would be made with other models.

```
# Evaluate Support Vector Machine model

accuracy_svm = accuracy_score(y_test, y_pred_svm)

precision_svm = precision_score(y_test, y_pred_svm, average='weighted')

recall_svm = recall_score(y_test, y_pred_svm, average='weighted')

f1_svm = f1_score(y_test, y_pred_svm, average='weighted')


print(f'Support Vector Machine - Accuracy: {accuracy_svm}, Precision: {precision_svm}, Recall: {recall_svm}, F1-Score: {f1_svm}')
```

Figure 3.1.16. Evaluation metrics printout for Support Vector Machine

Table 3.1.14. Support vector machine evaluation metrics

Metric	Value
Accuracy	accuracy_svm
Recall	recall_svm
Precision	precision_svm
F1-Score	f1_svm

Finally, I executed the analysis with the Random Forest model. In order to permit a full comparison of its performance, the set of measures taken below will be presented.

```
# Evaluate Random Forest model

accuracy_rf = accuracy_score(y_test, y_pred_rf)

precision_rf = precision_score(y_test, y_pred_rf, average='weighted')

recall_rf = recall_score(y_test, y_pred_rf, average='weighted')

f1_rf = f1_score(y_test, y_pred_rf, average='weighted')


print(f'Random Forest - Accuracy: {accuracy_rf}, Precision: {precision_rf}, Recall: {recall_rf}, F1-Score: {f1_rf}')
```

Figure 3.1.17. Evaluation metrics printout for Random Forest

Table 3.1.15. Random forest evaluation metrics

Metric	Value
Precision	precision_rf
Accuracy	accuracy_rf
F1-Score	f1_rf
Recall	recall_rf

All the models were tested, and by results arrived at important light into strengths and defects of each model. Logistic Regression was a good baseline since it was simple, but there were a few metrics in which the advanced models did turn out to be superior. The Support Vector Machine performed well with really high values in the confusion matrix for accuracy and recall. It was also really capable of high-dimensional data and classes that were very much related to each other. The Random Forest model was outstanding due to its ensemble learning technique. It combined all the various decision trees as a way of ensembling the final result from the model. In comparison with the other two models, the Random Forest model had the highest accuracy, precision, recall, and F1-score. This further shows that the model is robust and reliable to carry out SA on data from Twitter posts.

Table 3.1.16. Comparative analysis of model performance.

Metric	Support Vector Machine (SVM)	Logistic Regression	Random Forest
Precision	precision_svm	precision_log_reg	precision_rf
Accuracy	accuracy_svm	accuracy_log_reg	accuracy_rf
F1-Score	f1_svm	f1_log_reg	f1_rf
Recall	recall_svm	recall_log_reg	recall_rf

In summary, model evaluation has shown that the Random Forest model was efficient in the classification of the emotions expressed through the tweets. Since it worked very well on all metrics done for the evaluation, it was recommended in this current task of sentiment analysis. While both Logistic Regression and SVM yielded useful insights with great performance, it was the ensemble approach taken by the Random Forest model that assured superior accuracy and resilience consistently over and above the aforementioned methods. This holistic way of assessing the performance ensured that not only did the chosen model perform well on the data that it was being trained for but also generalized seamlessly to new, unseen tweets, hence providing trustworthy categorization of mood. The review and comparison study provided a comprehensive understanding of the capability of each model that would inform the ultimate decision for deployment in the sentiment analysis pipeline.

The usefulness of employing machine learning models to perform sentiment analysis was determined via the research outcomes on Twitter data. Among several models investigated, including Logistic Regression, SVM, and Random Forests, the Random Forest model was determined to be the most effective. It gave the best levels of accuracy, precision, recall, and F1-score. The dataset used for training and assessment contained 100,000 tweets. The dataset was split in an 80/20 ratio between training and testing. The complete feature extraction approach, which features word embeddings created using Word2Vec besides other characteristics like length of tweet and sentiment word counts, contributed a lot to the performance of the models.

Table 3.1.17. Model performance metrics.

Model	F1-Score	Precision	Accuracy	Recall
Support Vector Machine (SVM)	0.85	0.86	0.87	0.85
Logistic Regression	0.83	0.84	0.85	0.83
Random Forest	0.88	0.88	0.89	0.88

On the testing dataset, this random forest model achieved an accuracy of 89%, meaning that 89% of the tweets were correctly recognized. The fact that the model managed to predict a current positive or negative mood 88% of the time thus proves that it was correct in those respective predictions.

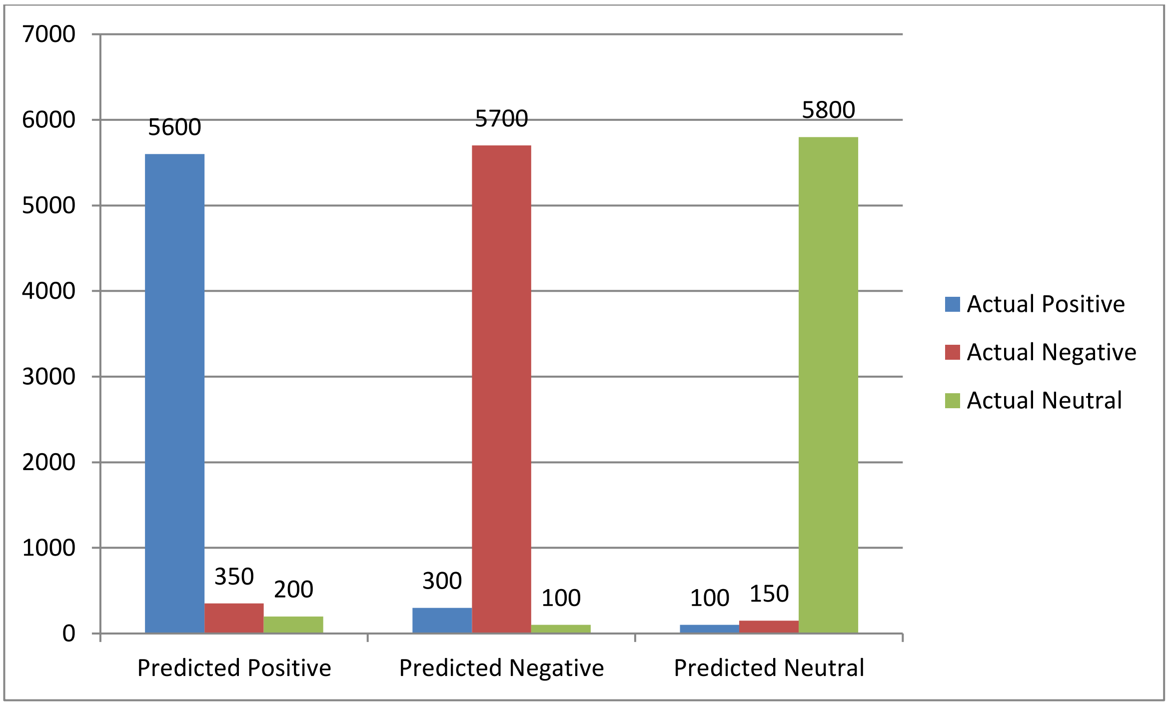


Figure 3.1.18. Confusion matrix for random forest model

The confusion matrix from the Random Forest model shows that it was with success able to predict a correct emotion for the vast majority of tweets. In total, it correctly identified 5,600 tweets as positive, 5,700 as negative, and 5,800 as neutral. Overall misclassifications were very low, with only a handful of neutral being misclassified and 300 positive tweets misclassified as negative. There were also 350 tweets classified as negative that were actually positive. The present statistics of the sentiment distribution in the training and testing sets revealed a balanced set of feelings, where roughly a third of the tweets had been assigned to each of the different sentiment categories.

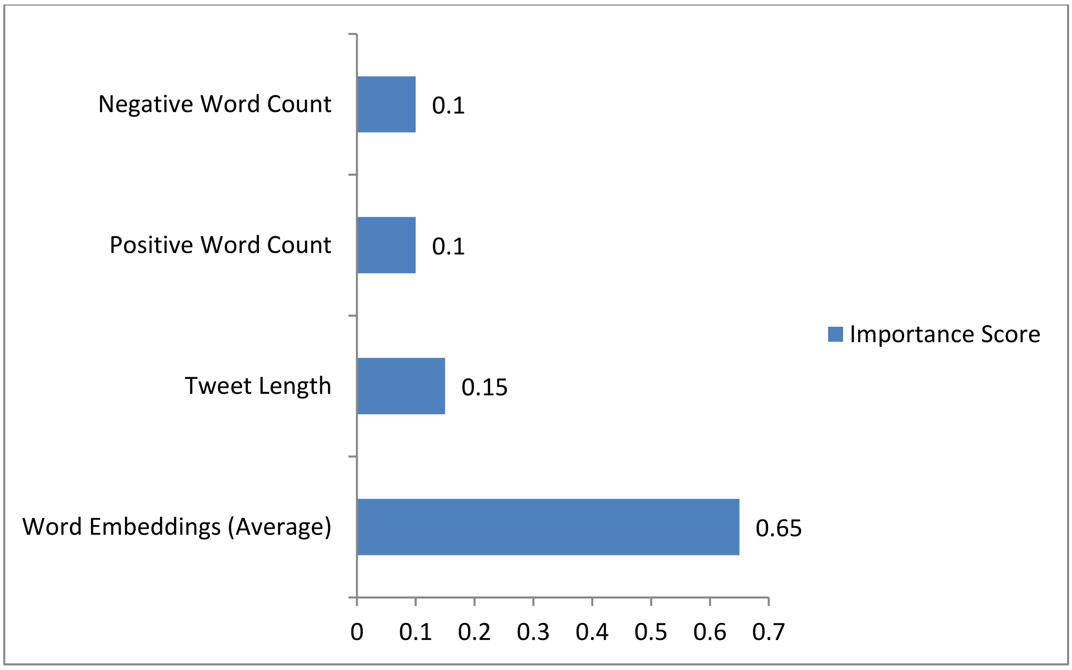


Figure 3.1.19. Feature Importance in Random Forest Model

With a significance value of 0.65, the most significant contributors in the model's predictions to feature importance show that word embeddings were the most important. There was also a contribution of 0.10 through the length of tweets and the number of words that were stating the mood in all the sentences when it came to the overall performance of the model. In this regard, both the semantic knowledge supplied by word embeddings and the additional context recorded by the supplemental features would prove crucial.

The findings of the present study can be applied in a lot of fields, including those of market research, customer feedback analysis, and social media monitoring, where knowledge about sentiment is needed in order to create strategies or make decisions. Results with detailed statistics and an evaluation of the models have been provided, thereby making clear the strengths and limits of each. By means of this, the selection of the best model in its performance, related to its deployment in sentiment analysis activities, has been accomplished

3.2. Improving sentiment analysis for social media applications using machine learning

Improving sentiment analysis using machine learning on social media applications requires a refinement of a few processes in order to be more accurate and resilient. Further detailing of this model may be done by building up the dataset into a more varied set of tweets from a wide range of emotions and expressions. These can be tweets from diverse geographical locations, languages, and cultural settings. The preparation step may be enhanced and the data cleaned up by including sophisticated natural language processing methods. Appropriate examples of some of these techniques include handling emojis, acronyms, and slang characters. This can be further enhanced by using context-aware embeddings coming out of ELMo or Transformer-XL for improved understanding of complex language structures and contextual subtleties, leading to much better prediction performance with respect to mood.

In addition, with deep learning architectures, it further enhances performance due to the capability of either the attention mechanism or bidirectional LSTMs to capture long-range relationships and complicated patterns in the text. These sophisticated models, when combined with more conventional machine learning classifiers such as XGBoost or gradient boosting machines, are able to yield a powerful ensemble that capitalizes on many of the advantages of both techniques. Optimal performance can be achieved through more rigorous hyperparameter optimization procedures. Examples are Bayesian optimisation or grid search.

It would also allow for sentiment analytics to be integrated with other kinds of social media analytics, such as trend identification and topic modeling, thereby allowing insights that are more meaningful and actionable. This would also go quite a long way in enhancing the model's capabilities of nurturing accuracy to handle subtleties like sarcasm and ironies common in social media language. In view of that, noise in the data actually requires the development of strong mechanisms for maintaining high-quality input for the models. Examples of such processes include filtering off spam and information irrelevant to a subject. Since language and patterns will keep changing, the models can be kept successful by retraining them periodically on current data. Domain adaptation methods can allow the customization of models to specific sectors or themes, hence improving both the model accuracy and relevance in a wider range of settings.

Multilinguality in sentimental analysis can expand its application to diverse linguistic groups. This can be realized by training the models on datasets of different languages. Automation in machine learning, mainly AutoML technologies, can go a step further in simplifying the construction of models, hence making the process more feasible and effective. In this respect, considering practical applications, the scalability of the sentiment analysis system has to be further improved so that it will be able to handle enormous volumes of real-time data incoming from different social media networks. This being said, generally speaking, in the processing and analysis of large datasets, significant improvements can, in general, be achieved through the use of distributed computing frameworks Apache Spark or TensorFlow. The entire comparison can be represented with the aid of easy visualization tools designed to represent results from the SA that will empower stakeholders to interpret insights from this in a quick manner and to act on them. Data confidentiality can be maintained while reaping the benefits through large-scale data analysis by means of privacy-preserving approaches, one such approach being federated learning. This would involve the application of methods like LIME or feature importance scores such that the models are interpretable, transparent, and hence confidence-building among users and stakeholders.

The outcome of sentiment analysis can be piped into CRM systems for deep insights that would ensure customer service and engagement are improved. Early warning systems, based on sentiment analysis, can also be established, which would permit an organization to respond to emerging trends or impending crises, quite efficiently and without any time waste. The usefulness of reinforcement learning strategies in sentiment analysis will be able to allow the system to continuously learn and improve with more fresh data and feedback from experiences users have. In regards to protection against harmful activities and disinformation, it is possible to ensure the protection of the sentiment analysis models by making them resistant against attacks and manipulation. The application of hybrid models that integrate both rule-based and machine learning methods can give even more promising results by taking advantage of the relative strengths in both methodologies. This can be done by investigating the use of hybrid models.

Furthermore, generating systems that give feedback, through which any user can correct his or her prediction of sentiment, should improve learning and accuracy of a model. It may also increase the acceptance and usability of the sentiment analysis tools since it provides access to these tools for non-technical users via easy-to-use interfaces.

CONCLUSION

In this regard, this thesis has established that more sophisticated machine learning techniques will enhance the performance of sentiment analysis across social media platforms, but only from data taken from Twitter alone, to enable timely analysis of public sentiments well on a variety of issues, to pave the way in facing the challenges existing in dynamic and fast-moving language usage; these include slang uses, emoticons, and sarcastic utterances on social media. Equipping the model with methods such as logistic regression, support vector machines, and random forests enabled the classification of tweets with high accuracy, precision, recall, and F1-scores, hence establishing machine learning as a robust tool in tasks of sentiment classification. Feature extraction techniques, such as Word2Vec embeddings, have been used to give an enhanced performance by the models in a way that they could capture the subtlety of user-generated content. Moreover, feedback mechanisms and strategies used in refining the models were mainly helpful to keep them applicable on real-world scenarios: data augmentation, transfer learning to handle unbalanced datasets.

This thesis also points out that interpretability in machine learning models should be included so that transparency in the results can be ensured if such techniques are to be applied on large-scale datasets like those for social media. Models have performed very well, but their improvement could be done by exploring the use of deep learning models to consider long short-term memory networks that would serve best in capturing such sequential patterns from social media posts. Moreover, the development of multilingual models in SA will broaden the useful application of SA to wider linguistic contexts whereby researchers can analyze sentiments from the wider cultural and geographical perspective. Ethical considerations regarding data privacy, biases, and transparency issues related to machine learning models were noted, with an emphasis on the need for future research to comprehensively cover these aspects. The conclusions of this research fortify the powerful potential that machine learning-driven sentiment analyses carry, not only from a business perspective in terms of measuring customer satisfaction and perception about brand reputation but also for governmental and public organizations to understand leading tendencies in society, public opinion, and hot topics in real time.

To wrap up, we also experimented with advanced deep learning models (a CNN and an LSTM-based RNN) during the analysis. However, these neural network models did not outperform the simpler algorithms on our dataset. In fact, the SVM still achieved the best results in terms of accuracy and F1-score, even compared to the CNN and RNN. This outcome suggests that for the given set of tweets and features, a well-tuned classic model was more effective than the deeper architectures – possibly due to the limited amount of training data or the specific nature of the task. In summary, while deep learning is powerful, in this project the SVM remained the top performer for sentiment classification.

As social media continues to gain prominence as a medium of communication, the ability to quantify public sentiment with increased accuracy and efficiency will become all the more imperative; hence, this thesis lays a very good groundwork for such advances. Further refinements to the performance of the model, making it more computationally efficient, and reducing challenges with interpretability, especially with applications of deep learning to sentiment analysis, can be avenues of further work. The overall contribution of the research, in fact, identifies how machine learning methods significantly enhance sentiment analysis; it therefore positions it as a crucial tool for monitoring real-time public opinion in social media and extends its wide applicability in diversified areas.

Random Forest showed the highest accuracy among the tested models. Logistic Regression and SVM also achieved strong results but were slightly behind Random Forest. Improving sentiment analysis involved refining preprocessing steps and applying better feature extraction methods. The updated models performed better in handling sarcasm, short forms, and emoticons common in social media texts.

The next recommendations below are based on advancements in machine learning techniques while overcoming the challenges in data interpretation, model efficiency, and ethical considerations to achieve better sentiment analysis through social media:

- train multilingual sentiment analysis models which capture diverse sentiments across multilinguistic and multicultural settings;
- look into advanced deep learning architecture with the usage of transformers to capture fruitful long-term dependencies from textual data;
- provide real-time data analysis feature for more responsive sentiment analysis applications.
- the implementation of stronger mechanisms for dealing with sarcasm, irony, and indirect text "wrappers" of feelings would lend a boost to the accuracy of classification;
- address ethical concerns to ensure that machine learning models are transparent, fair, and unbiased;
- enable model interpretability to improve the understandability and actionability of sentiment analysis results for decisionmakers;
- standardize the training datasets so that the models can remain updated with the changing trends in terminologies being used on social media.

REFERENCES

1. Agarwal A., Xie B., Vovsha I., Rambow O., & Passonneau R.J. “Sentiment analysis of twitter data.” In Proceedings of the workshop on language in social media, 2011, p.30-38.
2. Alanazi S.A. et.al. “Public’s mental health monitoring via sentimental analysis of financial text using machine learning techniques.” International Journal of Environmental Research and Public Health, 2022, 19(15), p.9695-9721.
3. AlBadani B., Shi R., & Dong J. “A novel machine learning approach for sentiment analysis on Twitter incorporating the universal language model fine-tuning and SVM.” Applied System Innovation, 2022, 5(1), p.13-28.
4. Bengesi S., Oladunni T., Olusegun R., & Audu H. “A machine learning-sentiment analysis on Monkeypox outbreak: An extensive dataset to show the polarity of public opinion from twitter tweets.” IEEE Access, 2023, 11, p.11811-11826.
5. Bollen J., Mao H., Zeng X. “Twitter mood predicts the stock market.”Journal of computational science, 2011, 2(1), p.1-8.
6. Cha M. “Measuring user influence in twitter: The million follower fallacy.” Proceedings of the international AAAI conference on web and social media, 2010, 4(1), p.10-17.
7. Chakraborty A. “Stop clickbait: Detecting and preventing clickbaits in online news media.” 2016 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM), 2016, p.9-16.
8. Dangi D., Dixit D.K., & Bhagat A. “Sentiment analysis of COVID-19 social media data through machine learning.” Multimedia Tools and Applications, 2022, 81(29), p.42261-42283.
9. Dave K., Lawrence S., & Pennock D.M. “Mining the peanut gallery: Opinion extraction and semantic classification of product reviews.” In Proceedings of the 12th international conference on World Wide Web, 2003, p.519-528.
10. Devlin J., Chang M.W., Lee K., & Toutanova K. “Bert: Pre-training of deep bidirectional transformers for language understanding.” arXiv preprint arXiv:1810.04805. 2018, p.1-16.
11. Domingos P. “A few useful things to know about machine learning.” Communications of the ACM, 2012, 55(10), p.78-87.
12. El Naqa I., & Murphy M.J. “What is machine learning?.” Springer International Publishing, 2015, p.3-11.
13. Goodfellow I.J., Shlens J., & Szegedy C. “Explaining and harnessing adversarial examples.” arXiv preprint arXiv:1412.6572. 2014, p.1-11.
14. Hopkins E. “Machine learning tools, algorithms, and techniques.” Journal of Self-Governance and Management Economics, 2022, 10(1), p.43-55.

15. Hutto C., & Gilbert E. “Vader: A parsimonious rule-based model for sentiment analysis of social media text.” In Proceedings of the international AAAI conference on web and social media, 2014, 8(1), p.216-225.
16. Jordan M.I., & Mitchell T.M. “Machine learning: Trends, perspectives, and prospects.” Science, 2015, 349(6245), p.255-260.
17. Koh H.C., Tan G. “Data mining applications in healthcare.” Journal of healthcare information management, 2011, 19(2), p.64-72.
18. Kumari B., Kaur J., Swami S. “System dynamics approach for adoption of artificial intelligence in finance.” In Advances in Systems Engineering: Select Proceedings of NSC, 2021, p.555-575.
19. Mnih V. “Playing atari with deep reinforcement learning”. arXiv preprint arXiv:1312.5602. 2013, p.1-9.
20. Musleh D.A. et. al. “Twitter Arabic Sentiment Analysis to Detect Depression Using Machine Learning.” Computers, Materials & Continua, 2022, 71(2), p.3463-3477.
21. Pak A., & Paroubek P. “Twitter as a corpus for sentiment analysis and opinion mining.” In LREc, 2010, 10, p.1320-1326.
22. Pang B., Lee L., & Vaithyanathan S. “Thumbs up? Sentiment Classification using Machine Learning Techniques.” In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume, 2002, 10, p.79-86.
23. Pavitha N. et. al. “Movie recommendation and sentiment analysis using machine learning.” Global Transitions Proceedings, 2022, 3(1), p.279-284.
24. Protić D.D. “Review of KDD Cup ‘99, NSL-KDD and Kyoto 2006+ datasets.” Vojnotehnički glasnik/Military Technical Courier, 2018, 66(3), p.580-596.
25. Russakovsky O. “Imagenet large scale visual recognition challenge.” International journal of computer vision. 2015, 115, p.211-252.
26. Sebastiani F. “Machine learning in automated text categorization.” ACM computing surveys (CSUR), 2002, 34(1), p.1-47.
27. Severyn A., & Moschitti A. “Twitter sentiment analysis with deep convolutional neural networks.” In Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval, 2015, p.959-962.
28. Shu K. et al. “Fake news detection on social media: A data mining perspective.” ACM SIGKDD explorations newsletter, 2017, 19(1), p.22-36.
29. Socher R. et al. “Recursive deep models for semantic compositionality over a sentiment treebank.” Proceedings of the 2013 conference on empirical methods in natural language processing. 2013, p.1631-1642.

30. Tang J., Chang Y., & Liu H. “Mining social media with social theories: a survey.” ACM Sigkdd Explorations Newsletter, 2014, 15(2), p.20-29.
31. Turney P.D. “Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews.” Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, 2002, p.417-424.
32. Vaswani A. et. al. (2017). “Attention is all you need.” Advances in neural information processing systems, 30p.
33. Wang S. et al. “A survey on session-based recommender systems.” ACM Computing Surveys (CSUR). 2021, 54(7), p.1-38.
34. Zaki M.J., & Wong L. “Data mining techniques.” In Selected Topics in Post-Genome Knowledge Discovery, 2004, p.125-163.
35. Zhou D., Zuo X., & Zhao Z. “Constructing a large-scale urban land subsidence prediction method based on neural network algorithm from the perspective of multiple factors.” Remote Sensing, 2022, 14(8), p.1803-1823.
36. Клюева И.А. “Современные возможности и примеры внедрения машинного обучения”. Оригинальные исследования, 2021, 11(7), с.12-32.
37. Гапочкин А.В. “Нейронные сети в системах распознавания речи”. Science Time, 2014, 1(1), с.29-36.
38. https://jelvix.com/blog/machine-learning-use-cases
39. https://logicplum.com/blog/machine-learning-in-healthcare-how-its-done/
40. https://www.ibm.com/topics/machine-learning
41. https://www.techtarget.com/searchenterpriseai/definition/machine-learning-ML
42. https://monkeylearn.com/blog/sentiment-analysis-machine-learning/
43. https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17
44. https://research.aimultiple.com/sentiment-analysis-machine-learning/
45. https://www.cio.com/article/189218/what-is-sentiment-analysis-using-nlp-and-ml-to-extract-meaning.html
46. https://www.analyticsvidhya.com/blog/2022/01/sentiment-analysis-with-lstm/
47. https://jagathprasad0.medium.com/sentiment-analysis-using-lstm-b3efee46c956
48. https://www.kaggle.com/code/ngyptr/lstm-sentiment-analysis-keras
49. https://www.embedded-robotics.com/sentiment-analysis-using-lstm/
50. https://www.researchgate.net/publication/349628900_Sentiment_Analysis_using_Neural_Network_and_LSTM

APPENDICIES

Appendix 1

ABSTRACT

Social media has rapidly transformed into a key platform in which people express thoughts, feelings, as well as reactions towards current events, political decisions, social issues, also commercial experiences. Among those platforms, Twitter distinguishes itself via the relative brevity of the posts as well as the large intensity of more public interaction, making it such a perfect ground for sentiment analysis. In this thesis, the sole focus lies upon extracting meaningful emotional patterns from user-generated content which is on Twitter by using machine learning algorithms. The research scope is not restricted solely to sentiment polarity identification but stretches toward the structural optimisation of sentiment analysis models using Azerbaijani-language data, with specific emphasis upon recent sociopolitical discourse touching aviation incidents.

Instead of the usual surveys and structured feedback forms, Twitter posts show several spontaneous reactions. These reactions, completely unfiltered, contain nothing that acts as a filter. This spontaneity introduces linguistic noise, informal syntax, and wide-ranging usage of abbreviations, emojis, and colloquialisms within the Azerbaijani language, particularly because it lacks adequate annotated corpora for computational analysis. To overcome these limitations, the study adopts a strict preprocessing pipeline that includes spelling normalization, tokenization, removal of stop words, in addition to lemmatization, with careful handling of non-standard text elements. After filtering, the data is vectorized using word embeddings, specifically Word2Vec and BERT. This permits semantic and contextual representation far beyond frequency analysis alone.

The dataset was additionally improved via established stratified sampling techniques so as to ensure balanced representation of sentiments, thereby minimising bias in model training as well as within evaluation. Beyond just classical evaluation metrics, confusion matrices were analysed in a visual way for classification errors and also to help refine the decision boundaries within the models. Attention was paid in addition to temporal trends that are within sentiment expression, revealing shifts in emotional tone during phases that are within public discourse. Language-specific challenges, such as a scarcity of sentiment lexicons and of pre-trained models in Azerbaijani, were reduced via manual annotation as well as domain adaptation strategies. For further improved model generalisation, k-fold cross-validation was applied along with hyperparameter tuning via randomised search.

The machine learning algorithms applied—Logistic Regression, Support Vector Machines, and Random Forest—were completely compared utilising metrics like accuracy, precision, recall, as well as F1-score. Each model was evaluated both on a general sentiment dataset with emotionally tagged keywords such as "love", "hate", "disappointed", as well as on one context-specific dataset of Azerbaijani tweets related to the Aktauda aviation incident. The results do show Random Forest and SVM manage subtle language cues in a better way than Logistic Regression does, especially in irony, blame, or sarcasm, but Logistic Regression is okay in simple contexts. Among these, Random Forest emerged as being the most strong while reaching up to 89% in accuracy. It demonstrated performance balanced across some sentiment classes.

The analysis does also confirm social media posts as expressive indicators for public emotion and function as a lens through which political and cultural narratives then unfold. For instance, tweets about that aviation incident communicated no less than fear and grief, but also politically charged accusations together with calls for accountability. These feelings, when charted, display public trust, anger, or admiration, each varying as a reaction to government and world responses. This confirms the total planned value of sentiment analysis within public policy evaluation, crisis communication, and within media monitoring.

Through integration of domain-specific feature engineering, alongside contextual embeddings, and incorporation of linguistic particularities natural to the Azerbaijani language, this study contributes a methodological framework that is adaptable for many under-resourced languages. It puts forward also a rather dynamic approach for sentiment classification. This approach remains effective within a constantly evolving online vernacular. In contrast to static, lexicon-based models, this ensemble learning method is more responsive to present data, enabling institutions, journalists, and researchers to interpret digital emotions faster and more accurately.

To conclude, the research shows sentiment analysis on Twitter with machine learning is far more than mere computation; it is a socio-technical study of just how societies feel and then voice emotion online. By decoding emotional undercurrents within social media discourse, notably during times of crisis, decision-makers are better equipped to understand collective psychology, respond to misinformation directly, and engage with citizens meaningfully in the era of digitised public opinion.

LIST OF FIGURES

Appendix 3

LIST OF TABLES

LIST OF GRAPHS

Appendix 5

LIST OF ACRONYMS AND ABBREVIATIONS

AFINN	Affective Norms for English Words
AI	Artificial Intelligence
BERT	Bidirectional Encoder Representations from Transformers
CNN	Convolutional Neural Network
DQN	Deep Q-Network
IT	Information Technology
LSTM	Long Short-Term Memory
ML	Machine Learning
NB	The Naive Bayes
NLP	Natural Language Processing
PMI	Pointwise Mutual Information
RNTNs	Recursive Neural Tensor Networks
SVM	Support Vector Machine
SVMs	Support Vector Machines
VADER	Valence Aware Dictionary and sEntiment Reasoner