**Sentimental Analysis of Movie Reviews**

Challa bhanu kiranComputer Science Engineering

Lovely professional university

Jalandhar, Phagwara

[Challabhanu2004@gmail.com](mailto:Challabhanu2004@gmail.com)

Kotagiri Kulbhushan

Computer Science Engineering

Lovely professional university

Jalandhar, Phagwara

[kulbhushankotagiri@gmail.com](mailto:kulbhushankotagiri@gmail.com)

**Abstract:**

Sentiment analysis is a rapidly growing field in natural language processing that aims to extract subjective information from text data. One of the most common applications of sentiment analysis is in the movie industry, where it is used to gauge public opinion on films. In this research paper, a sentimental analysis of movie reviews has been presented using a dataset of over 25,000 reviews collected from various sources. A machine learning model with different classifiers was built using Naïve Bayes, Logistic Regression for classifying movie reviews as positive, negative or neutral. A comparison of three popular machine learning algorithms was made. After pre-processing the dataset by removing stop words, a stemming technique was applied to reduce the dimensionality of the dataset. The recognition algorithms were evaluated in terms of performance matrices such as accuracy, precision, recall and F1-score.The results of this analysis demonstrated the effectiveness of the model in accurately classifying movie reviews and provided valuable insights into the current state of public opinion on films. The comparison of the two algorithms provided insight into the best algorithm to be used for a specific dataset and scenario.

Keywords: Movie reviews, machine learning, Sentiment analysis (SA), Naïve Bayes, Logistic Regression , accuracy.

# 1.INTRODUCTION:

Movie reviews are a rich source of subjective opinions and are of great interest to researchers in sentiment analysis. They provide a valuable benchmark for evaluating the effectiveness of different sentiment analysis techniques, as well as a real-world application for such techniques. In recent years, there has been a proliferation of online movie reviews platforms, such as Rotten Tomatoes and IMDb, which have made it easier for researchers to access large datasets of movie reviews. However, the analysis of movie reviews was also a challenging task due to the variability and subjectivity of natural language. Movie reviews could vary greatly in terms of length, structure, and language use, and they may contain subtle or ironic expressions of sentiment that were difficult for machines to interpret.

In addition, movie reviews often contain domain-specific language and references, which can make it difficult to transfer sentiment analysis models from one domain to another. To address these challenges, researchers have developed a variety of techniques for sentiment analysis of movie reviews. These techniques can be broadly classified into two categories: rule-based approaches and machine-learning approaches. Rule-based approaches rely on predefined rules or dictionaries to identify and classify sentiments in movie reviews. They are generally simple and fast to implement, but they are limited in their ability to adapt to new domains or handle subtle or irony expressions of sentiment. Machine learning approaches, on the other

hand, rely on machine learning algorithms to learn patterns in movie review data and classify sentiments automatically.These approaches are more flexible and adaptable than rule-based approaches, but they require a larger dataset and more computational resources to train. Common machinelearning techniques used in sentiment analysis of movie reviews include support vector machines, decision trees, and neural networks. In this paper, a novel approach has been proposed for improving the performance of sentiment analysis of movie reviews that combines the strengths of rule-based and machine-learning approaches. This approach involved training a machine learning classifier on a large dataset of movie reviews and then fine-tuning the classifier using a smaller dataset of movie reviews annotated with domainspecific rules and dictionaries. We evaluate our approach on a benchmark dataset of movie reviews and show that it outperforms both rule-based and machine-learning approaches alone.

In this study, we apply word embeddings to sentiment analysis on movie reviews, using logistic regression and Naive Bayes classifiers. The objective is to evaluate the effectiveness of word embeddings in transforming unstructured text data into a structured form that enhances classification accuracy. By comparing logistic regression and Naive Bayes, we aim to determine which model more effectively leverages these embeddings for binary sentiment classification. Through this approach, we hope to highlight the benefits of combining word embeddings with traditional machine learning classifiers in NLP tasks.

**Literature Review:**

* Text Classification Techniques:  
  Overview of popular sentiment analysis techniques, including bag-of-words, TF-IDF, and machine learning models. Traditional approaches face challenges in understanding semantic similarity and handling context within language.
* Word Embedding Models:  
  Detailed review of word embedding models such as Word2Vec, GloVe, and FastText, which represent words in a continuous vector space. These models capture semantic relationships, making them useful for NLP tasks.
* Logistic Regression and Naive Bayes in NLP:  
  Brief explanation of logistic regression and Naive Bayes as effective classifiers for binary classification tasks in NLP, highlighting their simplicity and interpretability.

**2. Methodology:**

Sentiment analysis, often referred to as sentiment mining, is a subfield of natural language processing (NLP) and machine learning (ML) that focuses on identifying and classifying sentiment in text. Its goal is to determine whether the opinion expressed in a document is positive, negative, or neutral. This ability is more useful in many areas such as business, customer service, finance, and social care, where understanding the public's sentiment is important. The desire to make good decisions.

***A diagram of a process

Description automatically generated***

**3.Data Collection:**

Data collection is the process of collecting raw data or analysis for use in machine learning. This is important step because the quality and quantity of data directly affect the performance and reliability of the final model.

**Importing Libraries:**

1. **Pandas:**

Pandas is a powerful data an d analytics library. It provides a data structure suitable for creating data structures such as Data Frame.

1. **NumPy:**

NumPy is a simple package for computing with Python. Provides support for multi dimensional arrays and various arithmetic operations

1. **Scikit-learn:**

It is a popular machine learning library in Python. It provides simple and effective tools for data exploration and data analysis. The train\_test\_split function is used to split the dataset into random training and testing subsets.

1. **Matplot:**

Matplotlib is a powerful Python library for creating static plots, visualizations, and dialogs in Python. It provides a variety of drawing tools commonly used in fields such as data science, machine learning, engineering, and scientific research.

**4.Data Preprocessing:**

Reading data is an important step in data analysis or machine learning. In Python, the Pandas library is often used to read and manipulate data, especially when creating data such as CSV files, Excel files, and SQL databases.

* **Removing the stop-words**

Sentimental analysis is an important part of natural language processing (NLP) and involves identifying the emotional tone behind the text. An important first step in emotional assessment is to eliminate stop words. Abandoned words are usually words that do not have any significant meaning and do not reflect the main meaning of the text.

Abandoned words are words that occur frequently in a language but contribute little to the meaning of a sentence. Examples include objects (the, a, one), prepositions (in, of, of), conjunctions (and, but, or), and pronouns (he, her, it). While these words are useful grammatically, they often do not provide useful information when analyzing the thought or content of a text. Important points for improving data quality Words that help define ideas or topics.

* **Tokenization**

Tokenization is an important step in natural language processing (NLP), especially when analyzing sentiment expressed in tweets. Given the unique characteristics of Twitter profiles, effective tokenization is essential for the dissemination of meaningful comments. This article explores the tokenization process in Twitter’s understanding of identity

Tokenization is the process of breaking text into smaller pieces called tokens. These tokens can be words, phrases, or symbols. In the context of Twitter sentiment analysis, tokenization typically focuses on a wordlevel tag that is important for understanding the sentiment expressed in tweets.

* **Stemminng or Lemmatization**

Stemming and lemmatization are two simple methods in natural language processing (NLP) that aim to reduce a word to its base or root form. Both methods are important for activities such as text analysis, data retrieval, and sentiment analysis because they help generate informative data by identifying morphological changes. However, the methods differ in terms of accuracy and computational requirements.

Stemming and lemmatization play an important role in natural language processing by reducing words to their simplest forms. Stemming provides speed and simplicity through heuristics, while lemmatization provides clarity and expressiveness through complex word analysis. Understanding these differences allows practitioners to choose appropriate methods based on specific applications and needs in word processing tasks.

* **Feature Extraction**

Feature extraction is the process of identifying and extracting relevant features from raw data to create more informative data. The goal is to reduce the complexity (often referred to as “dimensionality”) of the data while preserving as much relevant information as possible. This simplification helps increase the efficiency and effectiveness of machine learning algorithms and simplifies the analysis process.

It is an important technique in machine learning and natural language processing that involves transforming raw data into a format suitable for analysis. Specifically in the context of text, feature extraction technology converts unstructured text into a digital representation that can be processed by machine learning algorithms.It will provide an indepth look at various extraction methods, their applications.

**Feature Extraction Using Word Embeddings:**

In this study, feature extraction is accomplished through the use of word embeddings, specifically GloVe embeddings, to transform textual data into numerical form. Word embedding is a representation of words in a continuous vector space, where semantically similar words are mapped to similar vectors. This technique captures both syntactic and semantic information, allowing models to better understand the relationships between words, which is particularly useful in natural language processing tasks like sentiment analysis.

**Why Word Embeddings?**

Traditional feature extraction methods such as bag-of-words and term frequency-inverse document frequency (TF-IDF) represent text in high-dimensional and sparse vectors, which often lack semantic meaning and context. Word embeddings address this limitation by representing words in dense, low-dimensional vectors that encode relationships between words. By leveraging word embeddings, we can provide machine learning models with more meaningful and compact representations of text data.

**Process of Feature Extraction:**

1. **Loading Pre-trained GloVe Embeddings**:
   * We use pre-trained GloVe embeddings, which are trained on a large corpus and capture both the statistical and semantic properties of words. These embeddings are stored in a dictionary format where each word is associated with a fixed-length vector (e.g., 100-dimensional).
   * GloVe embeddings are particularly useful because they are trained on a broad range of contexts, making them well-suited for transfer learning in various NLP tasks, including sentiment analysis on movie reviews.
2. **Tokenization and Vector Conversion**:
   * Each review in the dataset is tokenized into individual words, removing punctuation and irrelevant tokens to streamline the process.
   * For each word in a review, we retrieve its vector representation from the GloVe embedding dictionary. If a word does not exist in the GloVe vocabulary, it is skipped or replaced with a zero vector.
3. **Averaging Word Vectors**:
   * Since reviews are composed of multiple words, we calculate a single vector that represents each review by averaging the word vectors of all words in the review.
   * The averaging approach provides a straightforward yet effective way to aggregate word-level information into a single fixed-length vector for each review. This “sentence embedding” captures the overall semantic tone of the review.
   * By using averaging, we maintain a consistent input vector size (e.g., 100 dimensions if using GloVe’s 100-dimensional embeddings) regardless of the review length, making it easier for machine learning algorithms to process the data.
4. **Handling Missing Words**:
   * Words that do not have a corresponding embedding in the pre-trained GloVe set are omitted from the averaging calculation. While this can result in minor information loss, the impact is typically minimal due to the extensive vocabulary in the GloVe embeddings.

**Benefits of Averaging Word Embeddings for Sentiment Analysis**

Averaging word embeddings is a powerful yet efficient approach for sentiment analysis. Although it simplifies the structure of the text by condensing multiple word vectors into a single sentence-level vector, this process retains enough information to capture the sentiment of the review effectively. This is particularly valuable in cases where a machine learning model, such as logistic regression or Naive Bayes, relies on numerical features as inputs. The averaging method is computationally inexpensive and works well with relatively simple classifiers, offering a balance between performance and interpretability.

**Example Calculation**

For a review like "This movie was truly amazing," we calculate an average vector as follows:

* Retrieve vectors for "movie," "truly," and "amazing" from GloVe.
* Average these vectors to create a single vector that represents the entire review.
* This vector serves as input to the classifier models, which use it to predict the sentiment (positive or negative) of the review.

**Advantages:** Captures meaning and relationships between words, similar words have similar vector representations. **Disadvantages:** Requires substantial amounts of data for training; may not perform well on small datasets.

**Label Loading:**

* For supervised tasks, labeled images indicating objects like lanes, other vehicles, pedestrians, and traffic signs are also loaded. This labeling ensures the model learns to distinguish between different classes in the environment, essential for tasks such as lane following and object detection.

**Splitting Data into Training and Testing Sets**

* To evaluate model performance effectively, split the standardized dataset into training and testing subsets. A common practice is to allocate 80% of the data for training and 20% for testing.

**5. Model Training:**

* After splitting the data into training and testing, the next step is to train the model using the training data. Here are the basics of training machine learning models in Python using scikitlearn.Model selection in Twitter sentiment analysis is an important step that affects the efficiency and accuracy of the sentiment classification process. Model selection depends on many factors as the nature of the product, the complexity of the analysis task, and the performance evaluation to be performed.

**Models for Sentiment Analysis:**

In this study, two widely used machine learning classifiers **logistic regression** and **Naive Bayes** are employed to predict sentiment polarity (positive or negative) from movie reviews. Both models are well-suited for binary classification tasks and offer distinct advantages when paired with word embeddings as feature inputs. Here, we outline each model's theoretical foundation, practical application in sentiment analysis, and any specific tuning used in this project.

**Logistic Regression:**

Logistic regression is a statistical method used for binary classification; The goal here is to estimate the probability that a sample belongs to a particular class. Despite its name, logistic regression is not a regression algorithm, but a classification algorithm.

**explanation of how the logistic regression works in a model:**

**Input variables:** Logistic regression takes input variables (also known as features or independent variables) and assigns weights to them. Input variables can be continuous, categorical or binary.

**Logistic function:** Logistic regression uses the logistic function (also known as the sigmoid function) for a linear combination of input variables and their weights. The logistic function transforms the output into a value between 0 and 1 that represents the probability that the sample belongs to the positive class.

**Decision boundary:** Logistic regression calculates a decision boundary (or threshold)that separates events in to two groups based on their predicted probability of occurrence. In general, if the predicted probability is greater than 0.5, the sample is classified as a good class.

**Model training:** Train a logistic regression model using optimization techniques to find the most weighted model that minimizes the difference between the predicted probability and the class map in the training data.

**Prediction:** Once the model is trained, it can predict a list of new events by applying the learned weights to the communication variables and feed the min to the logistics study.

Logistic regression is widely used in many fields, such as medicine(e.g., predicting disease incidence), finance(e.g., credit risk assessment), business(e.g., stopping people guessing), and more. It is interesting for its simplicity, interpretation and efficiency, especially considering that the relationship between different inputs and outcomes is linear.

**Naïve Bayes**

Naive Bayes is a popular classification technique based on Bayes' theorem and is widely used in many applications including text classification, spam detection, and sentiment analysis. This topic will take an indepth look at the principles behind Naive Bayes, its types, advantages, limitations, and practical applications.

Naive Bayes is a family of bestfit methods that use Bayes' theorem and the "naive" assumption of independence of each pair of features. This means that the presence of a feature in a category is assumed to be independent of the presence of other features. Despite this simplicity, Naive Bayes often performs very well in practice, especially for large datasets.

**Bayes Theorem:**

The basis of Naive Bayes is Bayes' theorem, which describes the probability of a hypothesis based on prior knowledge of events associated with the hypothesis. It can be represented mathematically as:A math equations and formulas

Description automatically generated with medium confidence

**Types of Naive Bayes Classifiers**

**Gaussian Naive Bayes:**

The features follow a normal (Gaussian) distribution. It is suitable for continuous data.

**Multinomial Naive Bayes:**

Used for discrete counts (e.g., word counts for text classification). It is particularly effective for document classification tasks.

**Bernoulli Naive Bayes:** Similar to Multinomial Naive Bayes but assumes binary features (e.g., presence or absence of a word).

**6.Model Evaluation:**

Model evaluation is an important step in the machine learning process and helps measure the effectiveness and quality of the training model. This process involves using various measures and methods to understand how the model performs on invisible objects, making it perform better than just memorizing information. Below is a detailed description of the measurement model, including its importance, common metrics, measurement methods, and best practices.

**Importance of Model Evaluation**

**Performance evaluation:**

Allows practitioners to evaluate how well the model predicts based on new data, which is important for determining the effectiveness of the model in real-world use in practice.

**Identify strengths and weaknesses:**

By evaluating the model, data scientists can identify its strengths and areas for improvement.

**Avoid overfitting:**

Testing performance on a separate test helps ensure that the model not only remembers the training data but can also generalize to new situations.

**Decision making:**

information:  Information obtained from evaluating the model can inform decisions about using the model, improving it, or abandoning the model altogether.

**Evaluation Metrics**:  
Accuracy, precision, recall, and F1-score are used to evaluate model performance. These metrics provide a balanced view of model effectiveness in predicting both positive and negative reviews.

**1. Accuracy:** The proportion of correct predictions made by the model out of all predictions.

**Accuracy = (True Positives + True Negatives)/Total Predictions**

**2. Precision:** The ratio of true positive predictions to the total predicted positives, indicating how many selected items are relevant.

**Precision = (True Positives)/(True Positives + False Positives)**

**3. Recall (Sensitivity):** The ratio of true positive predictions to all actual positives, indicating how many relevant items were selected.

**Recall = (True Positives)/(True Positives + False Negatives)**

**4. F1 Score:** The harmonic mean of precision and recall, providing a balance between the two metrics.

**F1 = 2⋅(Precision⋅Recall/Precision+Recall)**

**Table 1. Accuracy of each classifier.**

|  |  |
| --- | --- |
| **Algorithm** | **Accuracy (%)** |
| Logistic Regression | 87.07 |
| Naïve Bayes | 76.74 |

**7.Conclusion:**

In conclusion, sentiment analysis of movie reviews was a useful tool for understanding the public's perception of a particular film. It can be used by movie studios and marketers to gauge the success of a film and adjust its strategy accordingly. There are many potential future applications for sentiment analysis in the entertainment industry and beyond. For example, it could be used to predict the success of a movie before it was released or to analyze reviews of television shows or music albums. In addition, sentiment analysis could be applied to other types of text data, such as customer reviews of products or services, to help businesses understand their customers' opinions and needs. Sentiment analysis of movie reviews research is at its right mark but in future, the process can be used for positive and negative comments and reviews on e-commerce products and application ratings, as analysis of comments on social media. It also can be useful in the analysis of the tone of the paragraph i.e., a truly professional approach of the writer which was written in terms of blogs, tweets, podcasts and research articles. Deep learning can also be used with neural network analysis for better accuracy. The said approach is useful for the comments in binary form i.e., positive and negative comments, but by enhancing the data, the approach can be evaluated to classify different types of emotions of the reviewers in detail. The analysis can be used for real-time training and analysis of data for the classification of new reviews. As machine learning techniques continued to improve, it was likely that sentiment analysis will become an increasingly important tool for businesses and organizations seeking to understand and respond to public sentiment.

**8. References :**

1. R. Parikh and M. Movassate, “Sentiment Analysis of User- GeneratedTwitter Updates using Various Classi\_cation Techniques",CS224N Final Report, 2009

2. A.Pak and P. Paroubek. „Twitter as a Corpus for Sentiment Analysis and Opinion Mining". In Proceedings of the Seventh Conference on International Language Resources and Evaluation, 2010, pp.1320-1326

3. Bifet and E. Frank, "Sentiment Knowledge Discovery inTwitter Streaming Data", In Proceedings of the 13th InternationalConference on Discovery Science, Berlin, Germany: Springer,2010, pp. 1-15.

4. Dmitry Davidov, Ari Rappoport." Enhanced Sentiment Learning Using Twitter Hashtags and Smileys". Coling 2010: Poster Volumepages 241{249, Beijing, August 2010

5. Pablo Gamallo, Marcos Garcia, “Citius: A Naive-Bayes Strategyfor Sentiment Analysis on English Tweets", 8th InternationalWorkshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland,Aug 23-24 2014, pp 171-175.

6. R. Xia, C. Zong, and S. Li, “Ensemble of feature sets and classification algorithms for sentiment classification,” Information Sciences: an International Journal, vol. 181, no. 6, pp. 1138–1152, 2011.

7. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., &Stede, M..“Lexicon basedmethods for sentiment analysis”. Computational linguistics, 2011:37(2), 267-307.

8. Li, S., Xue, Y., Wang, Z., & Zhou, G..“Active learning for cross-domainsentiment classification”. In Proceedings of the Twenty-Thirdinternational joint conference on Artificial Intelligence (pp. 2127-2133).AAAI Press,2013

9. Bollegala, D., Weir, D., & Carroll, J.. Cross-Domain SentimentClassification using a Sentiment Sensitive Thesaurus. Knowledge andData Engineering, IEEE Transactions on, 25(8), 1719-1731,2013

10. Pang, B.and Lee, L. “A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts”. 42nd Meeting of the Association for Computational Linguistics[C] (ACL-04). 2004, 271-278.

11. V. M. K. Peddinti and P. Chintalapoodi, “Domain adaptation in sentiment analysis of twitter,” in Analyzing Microtext Workshop, AAAI, 2011.

12. https://arxiv.org/pdf/1601.06971