Journal Homepage: -www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

Article DOI:10.21474/IJAR01/22148
DOI URL: http://dx.doi.org/10.21474/IJAR01/22148

ISSN (O): 2320-5407
ISSN (P): 3107-4928

RESEARCH ARTICLE

COMPARING SENTIMENT ANALYSIS METHODS: FLIPKART REVIEWS

Veena Ghuriani

1. Department of Computer Science, Maitreyi College, University of Delhi, India.

Manuscript Info

Manuscript History
Received: 08 September 2025
Final Accepted: 10 October 2025
Published: November 2025

Key words:-
Online reviews, sentiment analysis, machine learning models, lexicon-based models, large language models

Abstract

Sentiment analysis is widely applied to examine opinions, evaluations, attitudes, judgments, and emotions toward a product. In this study, online reviews of the e-commerce portal Flipkart have been analyzed. The dataset contains 189874 rows and 5 columns of product information such as product name, product price, rate, and review and summary of 104 different types of products. Natural Language Processing has been used to carry out the sentiment analysis of online reviews. These techniques are used to conclude the amount of positive and negative reviews received by a product and further identify the opinions of consumers towards the product. In this research, three approaches of sentiment analysis - machine learning, unsupervised lexicon-based analysis and large language models have been used. Machine learning classifiers like logistic regression, support vector machine, decision trees, and eXtreme Gradient Boosting(XGBoost); lexicon-based approaches -Valence Aware Dictionary and sEntiment Reasoner(VADER) lexicon and SentiWordNet; and large language models(LLMs) like Generative pre trained Transformer(GPT)and Bidire ctional Encoder Representations from Transformers(BERT) are used. These approaches have been compared based on accuracy, F1-score, recall, precision, and kappa, to determine their effectiveness in sentiment analysis, revealing that machine learning and lexicon-based approaches provide robust performance, while the large language models are computationally intensive, time-consuming, and show comparatively lower accuracy.The identification of context-specific limitations of LLMs in sentiment classification is a significant finding of this work. This comprehensive evaluation of different approaches can help us select the most suitable model for sentiment analysis.

Introduction:-

Sentiment analysis, often referred to as opinion mining, is a technique of Natural Language Processing (NLP). It helps to identify the human sentiment- positive, negative, or neutral, expressed in the textual data [1]. In the age of big data, this technique is vital, as it enables the understanding of customer opinions shared across various social media sites, e-commerce portals, and review forums. The reviews help the business understand the product strengths, flaws and

Corresponding Author:-Veena Ghuriani
Address:-Department of Computer Science, Maitreyi College, University of Delhi, India.

preferences. The feedback of customers in the form of reviews is instrumental in helping businesses decide their future directions.

Sentiment analysis efficiently processes large volumes of customer feedback in the form of unstructured data [2]. Sentiment analysis uses a blend of traditional linguistic techniques and modern machine learning methods [2]. The machine learning models use labelled datasets to train algorithms that classify text [3] as positive, negative, or neutral. Lexicon-based approaches [4][5] make use of a predefined corpus of sentiment-bearing words. Recent advancements in deep learning and transformer-based architectures [6], such as Bidirectional Encoder Representations from Transformers (BERT) [7] have significantly enhanced the accuracy and applicability of sentiment analysis.

The comparative analysis of sentiment analysis approaches—machine learning, lexicon-based, and large language models (LLMs)—reveals varied performance depending on the contexts. Traditional machine learning models are simple, robust and offer foundational accuracy. Transformer models, like Generative pre-trained Transformer(GPT) and BERT, give higher performance in handling complex linguistics, but they are computationally intensive. Lexicon-based approaches rrequire predefined lexicons, a collection of terms that belong to a particular subject or language. They often lack the precision of machine learning and LLMs.

Earlier studies focused on comparing only limited machine learning algorithms—logistic regression, random forest, and naive bayes for sentiment analysis, without addressing lexicon-based or large language models, thus limiting its scope to these traditional machine learning approaches [8]. Another comparative study in financial services domain, highlighted that neural networks perform well, suggesting that texts rich in sentiment words yield more reliable sentiment evaluations[9]. Subsequent work compared Support Vector Machine (SVM) and Long Short-Term Memory (LSTM) with BERT, and highlighted that the BERT model performed best on a balanced dataset [10]. Another research compared LLMs like ChatGPT, GPT-4, and Llama-2-chat against machine learning models for sentiment analysis of app reviews, revealing that LLMs are more promising for structured text and not reviews [11]. Another study highlighted that the machine learning and deep learning models perform well in performing sentiment analysis of customer feedback data [12]. A similar study showed that SVM outperforms LSTM and BERT [13]. Another study evaluated and compared the effectiveness of Naive Bayes (machine learning), TextBlob (lexicon-based), and LSTM (deep learning) on a multi-source dataset from various social media platforms [14].A recent study highlights the improved accuracy of BERT and GPT on informal text on social media platform X[15]. Another study compared the performance of GPT-4o and FinBERT on financial news data and demonstrated the strength of GPT-4o better performance[16].

In the present study, sentiment analysis has been done using machine learning models, lexicon-based models and large language models. Machine learning models like Logistic Regression(LR), Decision Tree(DT), Random Forest(RF), SVM and XGBoost; lexicon-based models Valence Aware Dictionary and sEntiment Reasoner(VADER) and SentiWordNet and large language models BERT and GPT2 have been deployed for sentiment analysis of customer reviews. These models were evaluated based on various performance metrics to determine the most effective approach for extracting insights from customer reviews. The present work employs all 3 above approaches and works on a dataset, which is not balanced. Also, the customer reviews are unstructured text; hence, the text doesn't provide context. This comparative analysis helps in choosing the appropriate accordingto the specific needs of businesses and data characteristics.

Material and Methodology:-

In this research, a dataset of product reviews of the e-commerce portal Flipkart available at the Kaggle repository[17] has been analysed using sentiment analysis. This dataset contains 189874 rows and 5 columns of product information, such as product name, product price, rating(on a scale of 1 to 5), review and summary of 104 different types of products. The "summary" column has been considered for sentiment analysis.

Dataset Preprocessing:

The dataset consisted of unstructured and unformatted data which had to be converted into a structured format. Data without a proper framework and structure is difficult to work with and causes unnecessary errors while running programmes. Structured data draws attention to the characteristics in reviews so that tokenization becomes easier for the algorithm. Tokenization is when text is broken down into smaller units to make it meaningful to the machine without losing the text's initial essence[18]. The data set was cleaned by converting the text into lower case to maintain uniformity and punctuation marks were replaced with spaces to ease tokenization. The missing values,Not a

Number (NaN) were replaced with spaces and non-alphanumeric characters were removed. The common stop-words were filtered out using Python's Natural Language Toolkit(NLTK) English stop words list[19]. Tokenization was performed which divides text into smaller pieces, such as phrases or words. The Python function WhitespaceTokenizer()was used for tokenization. This was followed by lemmatization, which returns the words to their original form. The NLTK library was utilized to complete this procedure[20]. The Python function WordNetLemmatizer()was used for lemmatization.

After preprocessing, feature extraction was performed to convert unprocessed raw text data into numerical features suitable for processing while retaining the data from the original dataset [21]. To extract features from the corpus, i.e. the "cleaned" text data was transformed into a sparse matrix. For sentiment analysis, the machine learning model has to know the sentiment score of every unique word in the text data, and its frequency of occurrence. The features and their target values were specified to train the machine learning models. The features are the transformed text data using Term Frequency(TF) and Inverse Document Frequency(IDF) vectorizer [22][23].

Machine Learning Models:-

The study uses different types of machine learning models, Decision Trees, Random Forest, Logistic Regression, XGBoost and SVM to predict sentiment labels. The sentiment labels were calculated using TextBlob [24] as the initial dataset did not contain any information about sentiment polarity, which is required for supervised ML models. TextBlob is an established lexicon-based model, well-suited for short phrases(customer reviews) given in the "Summary" column. The models are trained using TF-IDF vectors and evaluated based on precision, accuracy, F1-score, and recall [25]. Their performances are compared.Logistic Regression is used to predict results based on probability [26]. It is a machine learning model that is quite popular and is simple to understand and use for binary classification tasks such as positive/negative reviews. In our work, Word-Level Logistic Regression has been employed. This technique uses the individual words (unigrams) as features. Words are converted into numerical features using methods like TF-IDF. A decision tree is used for tasks concerning classification and is a non-parametric supervised learning algorithm. It operates by splitting the data into various subsets based on the most important features. It creates a structure similar to that of a tree made of decisions. Each node in the tree denotes a feature [27].

XGBoost is a popular classification algorithm, an implementation of gradient-boosting decision trees, that is suitable when data training is involved. It is an implementation of gradient boosting machines (GBM) which is known as one of the best-performing algorithms utilised for supervised learning [28]. It is designed for speed, convenience and performance on large datasets.SVM seeks to find a hyperplane that best divides data into different classes. In a sentiment analysis setting, SVM tries to find a boundary between negative and positive reviews that are as far from each other as possible. SVM works well in high-dimensional spaces and is, therefore, appropriate for text data that can be represented by thousands of features, such as TF-IDF scores of words or n-grams. It is also known for efficiently working with unbalanced datasets and preventing overfitting, which explains its quite good accuracy in your results[29].Random Forest classifier can be described as a collection of tree-structured classifiers. Each tree classifies the input into a class based on its features. The classification with the highest votes is selected by the forest (over all the trees in the forest). The random forest is a classification technique made up of several decision trees[30].

Lexicon-based Approaches:-

The lexicon-based sentiment analysis or unsupervised sentiment analysis leverages predefined dictionaries of words (lexicons) to assess the sentiment of text by evaluating the polarity of individual words. Polarity scores are numerical scores ranging from 1 (most positive) to -1 (most negative) that indicate the overall sentiment and tone of a phrase or word[31]. A column mentioning the polarity score was added to the dataset, which categorised each review as positive or negative based on the score. In our work, VADER (Valence Aware Dictionary and Sentiment Reasoner) and SentiWordNet libraries have been used. The VADER lexicon [32] is a curated vocabulary of words and phrases for sentiment analysis and their magnitude of the polarity. VADER's sentiment scoring mechanism involves splitting the text into individual tokens like words, phrases, emoticons and retrieving its sentiment intensity score. Sentiment scores are adjusted based on contextual elements such as punctuation, capitalization, degree modifiers, negations, and conjunctions[33]. The adjusted scores are aggregated to compute an overall sentiment score for the text and lastly normalized to produce a compound score ranging from -1 (most negative) to +1 (most positive).The SentiWordNet approach [34] uses SentiWordNet, a lexical resource built on the WordNet database, to assign sentiment scores to words and phrases. It is widely used for analyzing text to determine its sentiment polarity (positive, negative, or neutral). TF-IDF has been used for feature engineering [35]. Sentiment Analysis with SentiWordNet also tokenizes the input text into words and performs part-of-speech (POS) tagging to identify whether a word is verb, noun, adjective, etc.[36]. For each word, its synsets are identified in WordNet based on the POS tag. The sentiment score from SentiWordNet is retrieved and the aggregate score is calculated by aggregating the score of all words and using weighting methods to give importance to adjectives and adverbs.

Large Learning Models(LLMs):-

LLMs like OpenAI's GPT [37] and Google's BERT [38]have been used for sentiment analysis. These models are capable of understanding context, tone, and even humor, detect sarcasm and even identifying sentiment shifts within a single document as they have been trained on extensive datasets. In the research, pre-trained LLMs GPT2 [39] and BERT have been used. In addition to the data preprocessing mentioned above, some additional steps were performed. The BERT and GPT2 tokenizer functions were employed in this study's tokenizing procedure. The corpus utilized is "microsoft/DialogRPT-updown" for GPT2 and "bert-base-uncased" for BERT. The "microsoft/DialogRPT-updown" corpus has 50,257 words and 1,024 token classes, while the "bert-base-uncased" corpus has 30,522 words and 768 token classes. The quality of the generated tokens may be impacted by the differing tokenization techniques used by BERT and GPT2.

This was followed by the encoding function from BERT and GPT2. Word tokens are transformed into tensor-formatted vectors by this technique. After that, padding is done to make the vector length for all of the data 32. The dataset was divided into 80% and 20% for training and testing respectively. GPT2 model's learning process employed the AdamW() optimization algorithm, utilizing a 2e-5 learning rate, 1e-8 as eps(Adams epsilon) and processing data in groups of 75930 training batches and 18983 testing batches. In this work, BERT pre-trained model "bert-base-uncased" was employed, the dataset was divided into 70%, 15% and 15% for training, validation and testing respectively. The AdamW() optimizer was used. The number of epochs used for both is 10.
The comparison between the classifiers was done on the basis of various performance metrics namely accuracy, precision, F1-score, confusion matrix, recall and kappa statistics [40].

Results and Discussion:-

The dataset has the "summary" column which has the customer review, it has been analysed for sentiment analysis. The TextBlob python library has been used to calculate the polarity and subjectivity of the "summary" column, then the column "sentiment label" is calculated as positive or negative. The neutral sentiment has not been considered in this study. The top 10 words in positive and negative reviews in the dataset are shown in Figure 1 and2.

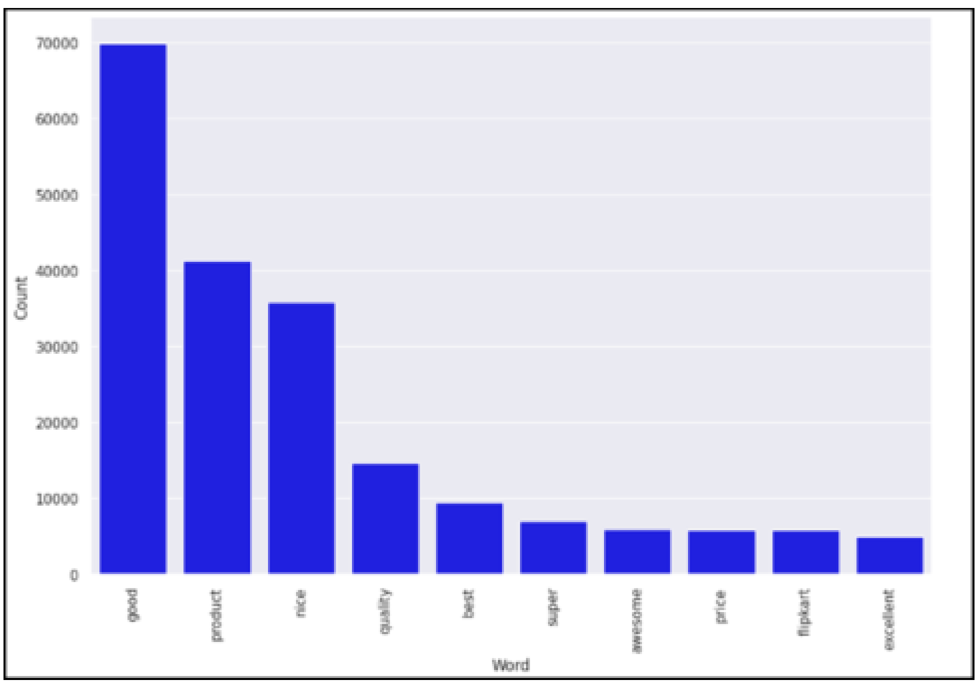


Figure 1:Words used most frequently in positive reviews

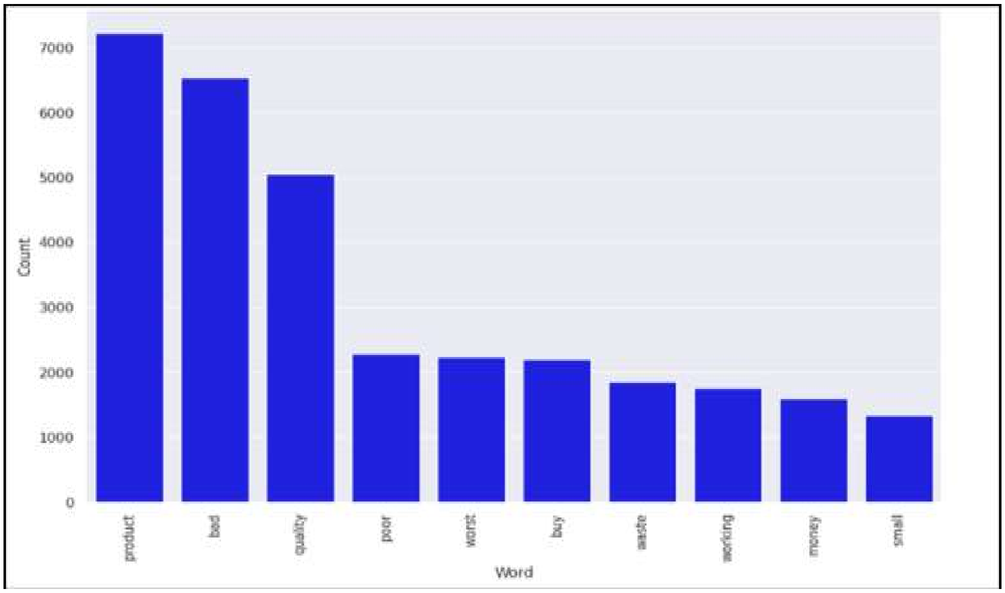


Figure 2:Words used most frequently in negative reviews

The frequently occurring positive and negative words in customer reviews were identified and plotted as word clouds. The word clouds [41]for positive and negative words are shown in Figure3and4.

Figure 3: Word clouds of frequent words in positive reviews

Figure4:Word clouds of frequent words in negative reviews

In this research, traditional machine learning techniques, lexicon-based techniques and large language models were used. The models were run on the dataset and their performance was compared.

Traditional Machine Learning Algorithms:-
The traditional machine learning algorithms employed TF-IDF for feature extraction. The classification algorithms employed —logistic regression (LR), decision tree (DT), SVM and two ensemble methods—random forest (RF) and XGBoost through TF-IDF features. The function tfidVectorizer() from Python library sklearn was used to generate the TF-IDF feature set. The dataset was split into training and testing sets. The results of sentiment analysis using various traditional classifiers are given in Table 1.

Table 1:Performance of machine learning classifiers for sentiment analysis

Machine learning models	Avg. Accuracy (%)	Avg. Precision (%)	Avg. Recall (%)	Avg. F1- score (%)	Kappa (%)
LR	96	87	93.5	90	80
DT	98	93.5	94	94	87
RF	98	92.5	97	94.5	89
SVM	97	90.5	93.5	92	84
XGBoost	**98**	**92.5**	**96.5**	**94.5**	**89**

It can be observed that the accuracy given by classifier is 96% (LR), 98% (DT), 98% (RF), 97% (SVM) and 98% (XGBoost), also the performance of the ensemble methods - RF and XGBoost is slightly higher,XGBoost has higher accuracy (98%)as compared to other classifiers, taking other performance metrics into consideration.

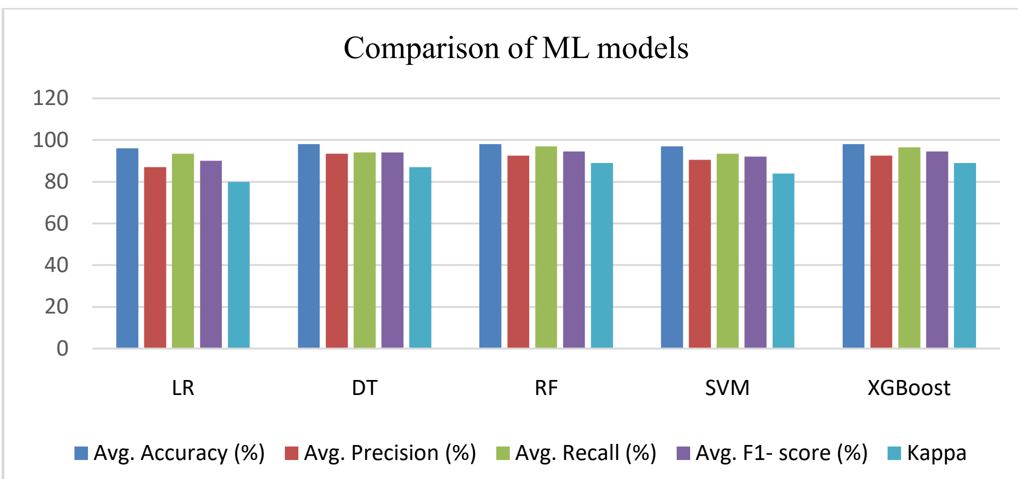


Figure5:Performance of MLmodels

RF and XGBoost showed the best overall performance with high accuracy, precision, recall, and F1-score as is clear from Figure 5. The ensemble models outperform as they combine multiple base models to provide better results and reduce overfitting. These methods can capture the variations and ambiguity in language more effectively than other traditional machine learning models.

Lexicon-based Approaches:-
For lexicon-based sentiment analysis, VADER and SentiWordNet classifiers were used. TF-IDF has been used for feature extraction. The results of the lexicon-based approach using VADER and SentiWordNet are summarized in Table 2 and 3.

Table 2:Performance of classifiers using VADER lexicon-based approach

Machine learning models	Avg. Accuracy (%)	Avg. Precision (%)	Avg. Recall (%)	Avg. F1-score (%)	Kappa (%)
LR	96	94.5	94.5	94.5	89
DT	97	96	95.5	95.5	90
RF	97	95	96	95.5	91
SVM	97	95.5	94.5	95	90
XGBoost	97	96.5	95	95.5	91

Table 3:Performance of classifiers using SentiWordNet lexicon-based approach

Machine learning models	Avg. Accuracy (%)	Avg. Precision (%)	Avg. Recall (%)	Avg. F1- score (%)	Kappa (%)
LR	98	95	92	93.5	87
DT	97	92.5	93	93	85
RF	98	95.5	92	94	88
SVM	98	95	95	95	90
XGBoost	97	95	90	92.5	85

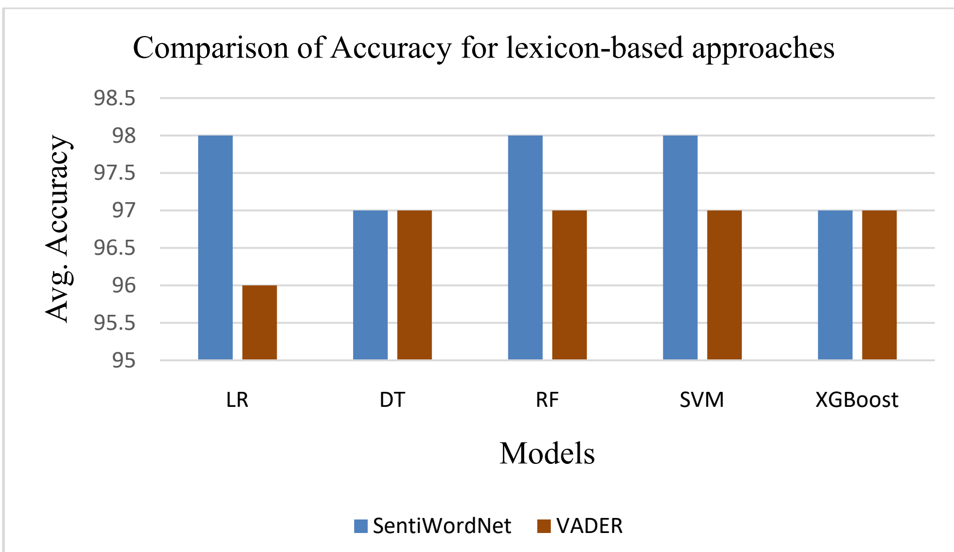


Figure6:Comparison of accuracy of VADER and SentiWordNet lexicon-based approaches

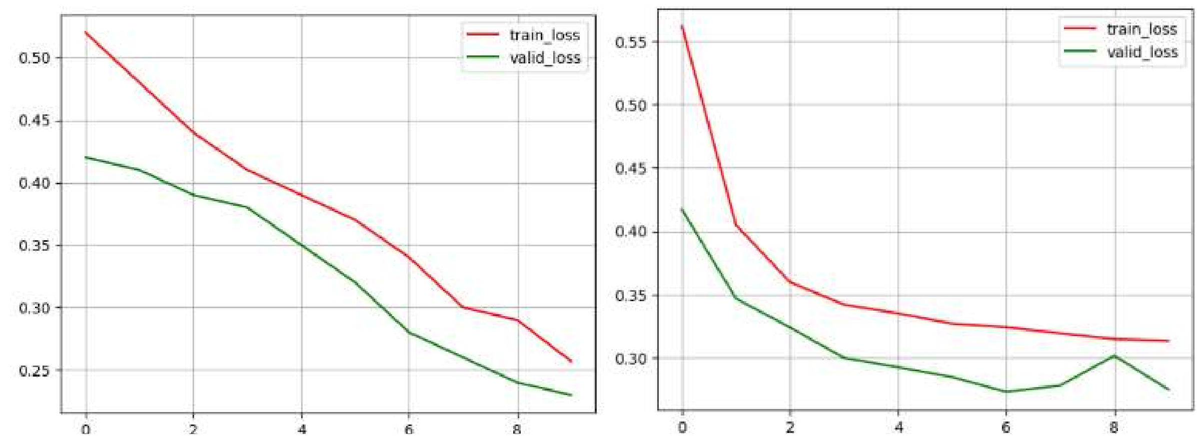
It can be observed from Figure6 that lexicon-based technique using SentiWordNet gives better accuracy for the classifiers; LR, RF and SVM all of which give 98% accuracy and for DT and XGBoost, the accuracy remains unchanged(97%). SentiWordNet can assign partial sentiment scores to different parts of a sentence, offering more granular analysis. It performs better than VADER because of its lexical richness, contextual sensitivity and adaptability to specific domains. VADER, while good with slang and social media language, often struggles with such context-dependent meanings.

LLMs:-

For executing the LLM models, the GPU available in Google Colab was deployed. The LLM model BERT used the model "bert-base-uncased", the batch size was 16, learning rate 1e-5 and number of epochs 10, gave an accuracy of 93%. The training loss and validation loss were 0.274 and 0.282 respectively. The model used for GPT was GPT2, the learning rate was 2e-5, epsilon parameter was 1e-8 and epochs 10, it resulted in accuracy of 95%. The training loss and validation loss were 0.257 and 0.23 respectively. The performance of both these models is summarized in Table 4.

Table 4: Performance of classifiers using LLMs

LLMs	Avg. Accuracy (%)	Avg. Precision (%)	Avg. Recall (%)	Avg. F1- score (%)	Kappa (%)
GPT2	95	88.5	94.5	91	82
BERT	93	86.5	91.5	88	77



(a) (b)

Figure 7: Training and validation loss of GPT and BERT in 10 epochs

It can be observed that the GPT2 model is performing better than the BERT model, although the execution time taken by GPT2 is longer as compared to the BERT model. The graphs given in Figure7(a) & (b) indicate that the training and validation loss for both models is decreasing and hence there is no overfitting. GPT2 was trained using the "microsoft/dialogrpt-updown" corpus, which is tailored for opinion ranking and response modeling, aligning more closely with review sentiment tasks, whereas BERT's "bert-base-uncased" lacks domain-specific fine-tuning. It can be concluded that fine-tuned GPT2 model can surprisingly handle classification tasks well at the sentence-level, especially when large datasets are involved.

Conclusions:-

This study presented a comparative evaluation of sentiment analysis on thecustomer review dataset using traditional machine learning classifiers, lexicon-based approach and deep learning models. The performance of traditional models, LR, DT, SVM, RF and XGBoost, lexicon-based approaches VADER and SentiWordNet and deep learning models GPT2 and BERT has been compared based on accuracy, F1-score, recall and kappa statistics.The studydemonstrated that the ensemble classifiers such as RF and XGBoost have outperformed other traditional ML models. Lexicon-based models were slightly less robust than ML models, but SentiWordNet demonstrated stronger contextual sensitivity than VADER. On the contrary, large language models like GPT2 and BERT underperformed in comparison to traditional models in this specific application. The lack of domain-specific fine-tuning, high computation requirement and direct binary sentiment classification task, could be the reasons for their reduced performance. Customer reviews often contain complex expressions of sentiment, such as sarcasm, mixed feelings or indirect sentiment. Despite their strength in understanding context, LLMs may still miss these subtleties or generate inaccurate sentiment predictions.The identification of context-specific limitations of LLMs in sentiment classification is a significant finding of this work. Our findings show that traditional models perform better in domain-specific applications like sentiment analysis of customer review as compared to LLMswhich have shown good performance in several research.The work can be extended by includingthe neutral sentiments and aspectbased sentiment analysis, which could offer more granular insights. The data imbalance could be handled using undersampling or oversampling techniques in future work.

References:-

[1]B. Liu, Sentiment Analysis and Opinion Mining. Morgan & Claypool Publishers, 2012.
[2]E. Sullivan, "Understanding from Machine Learning Models," Br. J. Philos. Sci., vol. 73, no. 1, pp. 109–133, Mar. 2022, doi: 10.1093/bjps/axz035.
[3]E. Cambria, B. Schuller, Y. Xia, and C. Havasi, "New Avenues in Opinion Mining and Sentiment Analysis," IEEE Intell. Syst., vol. 28, no. 2, pp. 15–21, Mar. 2013, doi: 10.1109/MIS.2013.30.
[4]V. Bonta, N. Kumaresh, and N. Janardhan, "A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis," Asian J. Comput. Sci. Technol., vol. 8, no. S2, pp. 1–6, Mar. 2019, doi: 10.51983/ajcst-2019.8.S2.2037.
[5]V. Singh, G. Singh, P. Rastogi, and D. Deswal, "Sentiment analysis using lexicon based approach," in Proc. 2018 Fifth Int. Conf. Parallel, Distributed and Grid Computing (PDGC), Solan, India, Dec. 2018, pp. 13–18, doi: 10.1109/PDGC.2018.8745971.
[6]A. Gillioz, J. Casas, E. Mugellini, and O. A. Khaled, "Overview of the transformer-based models for NLP tasks," in Proc. 2020 Federated Conf. Computer Science and Information Systems, 2020, pp. 179–183, doi: 10.15439/2020F20.
[7]J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. 2019 Conf. North American Chapter of the Association for Computational Linguistics, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
[8]V. M. Abia and E. H. Johnson, "Sentiment Analysis Techniques: A Comparative Study of Logistic Regression, Random Forest, and Naive Bayes on General English and Nigerian Texts," J. Eng. Res. Rep., vol. 26, no. 9, pp. 123–135, Sept. 2024, doi: 10.9734/jerr/2024/v26i91268.
[9]A. Garg, S. Dhanasekaran, R. Gupta, and J. Logeshwaran, "Exploring machine learning models for intelligent sentiment analysis in financial services," in Proc. 2024 IEEE Int. Conf. Contemporary Computing and Communications (InC4), Bangalore, India, Mar. 2024, pp. 1–6, doi: 10.1109/InC460750.2024.10649215.
[10]A. M. ElMassry, A. Alshamsi, A. F. Abdulhameed, N. Zaki, and A. N. Belkacem, "Machine learning approaches for sentiment analysis on balanced and unbalanced datasets," in Proc. 2024 IEEE 14th Int. Conf. Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, Aug. 2024, pp. 18–23, doi: 10.1109/ICCSCE61582.2024.10695972.
[11]F. A. Shah, A. Sabir, R. Sharma, and D. Pfahl, "How Effectively Do LLMs Extract Feature-Sentiment Pairs from App Reviews?," 2024, arXiv. doi: 10.48550/ARXIV.2409.07162.
[12]T. Ananth Kumar et al., "Machine learning and sentiment analysis: Analysing customer feedback," in Advances in Marketing, Customer Relationship Management, and E-Services, IGI Global, 2024, pp. 245–262, doi: 10.4018/979-8-3693-2165-2.ch014.
[13]N. J. Xiang et al., "Sentiment analysis using learning-based approaches: A comparative study," in Proc. 2023 11th Int. Conf. Information and Communication Technology (ICoICT), Melaka, Malaysia, Aug. 2023, pp. 469–474, doi: 10.1109/ICoICT58202.2023.10262604.
[14]K. Kapur and R. Harikrishnan, "Comparative Study of Sentiment Analysis for Multi-Sourced Social Media Platforms," 2022, arXiv. doi: 10.48550/ARXIV.2212.04688.
[15]H. Sharma and V. Kakran, "Sentiment analysis: Analyzing Flipkart product reviews using NLP and machine learning," in Proc. 2024 Int. Conf. Computing, Sciences and Communications (ICCSC), Ghaziabad, India, Oct. 2024, pp. 1–7, doi: 10.1109/ICCSC62048.2024.10830421.
[16]J.-W. Kang and S.-Y. Choi, "Comparative Investigation of GPT and FinBERT's Sentiment Analysis Performance in News Across Different Sectors," Electronics, vol. 14, no. 6, p. 1090, Mar. 2025, doi: 10.3390/electronics14061090.
[17]H. Sharma and V. Kakran, "Sentiment Analysis: Analyzing Flipkart Product Reviews using NLP and Machine Learning," in 2024 International Conference on Computing, Sciences and Communications (ICCSC), Ghaziabad, India: IEEE, Oct. 2024, pp. 1–7. doi: 10.1109/ICCSC62048.2024.10830421.
[18]A. Erkan and T. Güngör, "Analysis of deep learning model combinations and tokenization approaches in sentiment classification," IEEE Access, vol. 11, pp. 134951–134968, 2023, doi: 10.1109/ACCESS.2023.3337354.
[19]D. Yogish, T. N. Manjunath, and R. S. Hegadi, "Review on natural language processing trends and techniques using NLTK," in Recent Trends in Image Processing and Pattern Recognition, Singapore: Springer, 2019, pp. 589–606, doi: 10.1007/978-981-13-9187-3_53.
[20]N. Hardeniya et al., Natural language processing: Python and NLTK. Packt Publishing, 2016.
[21]R. Ahuja et al., "The impact of features extraction on the sentiment analysis," Procedia Comput. Sci., vol. 152, pp. 341–348, 2019, doi: 10.1016/j.procs.2019.05.008.

[22]A. Aizawa, “An information-theoretic perspective of TF–IDF measures,” Inf. Process. Manag., vol. 39, no. 1, pp. 45–65, Jan. 2003, doi: 10.1016/S0306-4573(02)00021-3.
[23]S. Qaiser and R. Ali, “Text mining: Use of TF-IDF to examine the relevance of words to documents,” Int. J. Comput. Appl., vol. 181, no. 1, pp. 25–29, July 2018, doi: 10.5120/ijca2018917395.
[24]S. Loria, “Textblob Documentation. Release 0.15,” 2018.
[25]N. W. S. Wardhani et al., “Cross-validation metrics for evaluating classification performance on imbalanced data,” in Proc. 2019 Int. Conf. Computer, Control, Informatics and its Applications (IC3INA), Tangerang, Indonesia, Oct. 2019, pp. 14–18, doi: 10.1109/IC3INA48034.2019.8949568.
[26]A. Poornima and K. S. Priya, “A comparative sentiment analysis of sentence embedding using machine learning techniques,” in Proc. 2020 6th Int. Conf. Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, Mar. 2020, pp. 493–496, doi: 10.1109/ICACCS48705.2020.9074312.
[27]L. Rokach and O. Maimon, “Decision trees,” in Data Mining and Knowledge Discovery Handbook, Springer, 2005, pp. 165–192, doi: 10.1007/0-387-25465-X_9.
[28]T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016.
[29]T. Evgeniou and M. Pontil, “Support vector machines: Theory and applications,” in Machine Learning and Its Applications, Springer, 2001, pp. 249–257, doi: 10.1007/3-540-44673-7_12.
[30]L. Breiman, “Random forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
[31]D. H. Abd, A. R. Abbas, and A. T. Sadiq, “Analyzing sentiment system to specify polarity by lexicon-based,” Bull. Electr. Eng. Inform., vol. 10, no. 1, pp. 283–289, Feb. 2021, doi: 10.11591/eei.v10i1.2471.
[32]D. C. Youvan, “Understanding sentiment analysis with VADER: A comprehensive overview and application,” 2024, doi: 10.13140/RG.2.2.33567.98726.
[33]C. Hutto and E. Gilbert, “VADER: A parsimonious rule-based model for sentiment analysis of social media text,” Proc. Int. AAAI Conf. Web Soc. Media, vol. 8, no. 1, pp. 216–225, 2014..
[34]A. Cernian, V. Sgarciu, and B. Martin, “Sentiment analysis from product reviews using SentiWordNet as lexical resource,” in Proc. 2015 7th Int. Conf. Electronics, Computers and Artificial Intelligence (ECAI), Bucharest, Romania, 2015, pp. WE-15–WE-18, doi: 10.1109/ECAI.2015.7301224.
[35]S. Akuma, T. Lubem, and I. T. Adom, “Comparing bag of words and TF-IDF with different models for hate speech detection from live tweets,” Int. J. Inf. Technol., vol. 14, no. 7, pp. 3629–3635, Dec. 2022, doi: 10.1007/s41870-022-01096-4.
[36]S. Baccianella, A. Esuli, and F. Sebastiani, “SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining,” in Proc. LREC, 2010.
[37]K. I. Roumeliotis and N. D. Tselikas, “ChatGPT and Open-AI models: A preliminary review,” Future Internet, vol. 15, no. 6, p. 192, 2023, doi: 10.3390/fi15060192.
[38]R. Sudharsan, Getting started with Google BERT: Build and train state-of-the-art natural language processing models using BERT. Packt Publishing, 2021.
[39]M. Hanna, O. Liu, and A. Variengien, “How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model,” arXiv preprint, 2023, doi: 10.48550/ARXIV.2305.00586.
[40]M. Aldwairi and A. Alwahedi, “Detecting fake news in social media networks,” Procedia Comput. Sci., vol. 141, pp. 215–222, 2018, doi: 10.1016/j.procs.2018.10.171.
[41]F. Heimerl, S. Lohmann, S. Lange, and T. Ertl, “Word cloud explorer: Text analytics based on word clouds,” in Proc. 2014 47th Hawaii Int. Conf. System Sciences, Waikoloa, HI, 2014, pp. 1833–1842, doi: 10.1109/HICSS.2014.231.