SINERGI Vol. 29, No. 2, June 2025: 385-396
http://publikasi.mercubuana.ac.id/index.php/sinergi
http://doi.org/10.22441/sinergi.2025.2.010

High-performance sentiment classification of product reviews using GPU(parallel)-optimized ensembled methods

Annaluri Sreenivasa Rao[1], Yeruva Jaipal Reddy[2] , Guggilam Navya[3], Neelima Gurrapu[4] , Jala Jeevan[5], M. Sridhar[6], Desidi Narasimha Reddy[7], Siva Kumar Pathuri[8,]*, Dama Anand[9]

[1]Department of Information Technology, VNR Vignana Jyothi Institute of Engineering and Technology, India
[2]Narasaraopeta Engineering College, India.
[3]Vignan's Foundation for Science, Technology and Research, India.
[4]Department of Computer Science and Artificial Intelligence, SR University, India.
[5]Department of Computer Science & Applications, K L E F, India.
[6]Department of Computer Applications, R. V. R & J.C College of Engineering, India.
[7]Data Consultant, Soniks Consulting, US
[8]Department of CSE, K L E F University, India
[9]Department of Computer Science & Engineering, K L E F, India

Abstract
Sentiment analysis is an important approach in natural language processing (NLP) that extracts information from text to infer underlying emotions or views. This technique entails classifying textual information into feelings like "positive," "negative," or "neutral." By evaluating data and labeling, client input may be classified on scales such as "good," "better," "best," or "bad," "worse," resulting in a sentiment classification. With the fast expansion of the World Wide Web, a massive library of user-generated data—opinions, thoughts, and reviews—has evolved, notably for diverse items. E-commerce firms use this data to gather attitudes and views from social media sites like Facebook, Twitter, Amazon, and Flipkart. The GPU-CUDA-ENSEMBLED algorithm is a GPU-accelerated method for sentiment classification, enhancing predictive performance by minimizing variances and biases. It outperforms existing algorithms like SLIQ and MMDBM, demonstrating GPU mining's efficiency. The proposed algorithm utilizes GPU-accelerated sentiment analysis to accurately predict smartphone ratings, providing valuable insights for businesses to maximize customer feedback potential.

Keywords:
GCE;
Graphics Processing Unit (GPU);
MMDBM;
SLIQ;
Smartphone Reviews;

Article History:
Received: August 2, 2024
Revised: January 1, 2025
Accepted: January 16, 2025
Published: May 1, 2025

Corresponding Author:
Siva Kumar Pathuri
Department of CSE, K L E F University, India
Email:
spathuri@kluniversity.in

INTRODUCTION

Knowing what customers think is of at utmost importance to companies seeking to measure consumer satisfaction and adjust what they provide accordingly. Especially algorithms like GPU-CUDA-ENSEMBLED that employ computational operations to help speed up sentiment analysis and lead to better insights. Conventional algorithms underpinning traditional sentiment analysis methods, coupled with the processing and computational bottlenecks, are often resolved using traditional programming methods. However, the GPU-CUDA-ENSEMBLED algorithm utilizes GPUs and parallel processing to provide a paradigm shift in feature-based sentiment analysis as applied in product evaluation systems [1].

This paper describes how the GPU-CUDA-ENSEMBLED, a brand-new design for modelling product reviews, was applied here. As a result, the algorithm enables the evaluation of feedback in near real-time with the ability to capture the nuances of consumer sentiment on a topic using a feature-based approach – all by leveraging the capabilities of GPUs. While polarity classification is rather trivial, polarity classification about a product can express more actionable needs from a business perspective. Traditional emotion study approaches rely on standard algorithms, which are usually limited by disposal limitations and processing bottlenecks.

However, with the GPU-GPU-CUDA-ENSEMBLED graphics processing units (GPUs) riches, a disruptive solution is emerging that has the ability to reshape the landscape of feature-based sentiment analysis for amount judging systems. The Proposed algorithm promises to significantly stimulate belief analysis movements, which admits real-time judgment of client feedback. Additionally, the allure feature-located approach specifies a nuanced understanding of brand emotions and goes beyond absolute opposition classification to capture the shadings of service belief.

Classification [2] involves analyzing sample records, each forming diversified attributes, inside a preparation dataset; attributes can range from numerical principles, like cost or camcorder pel, to categorical ones, usually in the direction of ordered rule principles. The CUDA-ENSEMBLED classifier invention is to redefine the production judgment example, promoting conversational resolution, making and growing customer delight. Data excavation involves distilling information from endless datasets using various techniques, such as categorization, collection rules, clustering, and most importantly, categorization.

The effectiveness of reviews and ratings is assessed based on content, with aspects such as aspect extraction and polarity analysis considered. Classification techniques are employed to analyze text-based reviews and rankings, contributing to knowledge development in data mining. The proposed approach introduces a Feature-Based Sentiment Analysis on Product Review System using the Proposed Algorithm. This hybrid classification technique aims to streamline data sorting and classification processes, thereby enhancing efficiency. By leveraging the GPU-CUDA-ENSEMBLED Algorithm, the study seeks to address the growing need for timely analysis in the fast-paced realm of the internet and technology.

In this paper, we highlight several important contributions: First, we use predictive analytics to score smartphone reviews. We find that smartphone reviews are, on average, lower than PC reviews. In addition, we analyzed the length and number of reviews of three poles: positive, negative and neutral. Understanding these statistical properties is critical to designing effective classification systems. We perform various comparative experiments to find optimal methods for analyzing short texts. These experiments include comparing polarity classification algorithms, text representation methods, and data segmentation based on word count to assess the impact on short and long texts. These experiments help identify more accurate and efficient methods of analysis. Our experiments are based on a large real-world dataset of over 5,000,000 mobile assessments from the Kaggle repository.

The main objective is to predict the star rating of mobile phones using various scoring techniques. By transferring traditional data processing to mobile devices and improving customer behavior monitoring, we want to reduce fraud by competitors. This allows mobile vendors to make informed product development decisions. Our methodology involves a two-step process: data processing and feature extraction with polar classification using the proposed algorithm, which improves model accuracy. Additionally, we implement an efficient proposed classifier on the GPU that uses multiple threads for parallel processing. GPU mining significantly speeds up computation compared to CPU, as shown by calculating the speed ratio. The data is divided into threads to calculate the number of classes, which shows the faster calculation speed of the GPU compared to the CPU. The paper consists of six parts: related work, proposed algorithm, experimental results, comparison with existing algorithms, conclusion, and future directions.

LITERATURE SURVEY

[2] Apostol, E.-S., et al. (2023). ATESA-BÆRT: A Heterogeneous Ensemble Learning Model for Aspect-Based Sentiment Analysis. This study proposes a model for aspect-based sentiment analysis (ABSA) that splits the task into two sub-tasks: Aspect Term Extraction and Aspect Term Sentiment Analysis. It utilizes an ensemble of six transformer-based learners for each sub-task and applies multi-class classification to enhance performance on reviews containing multiple aspects.

[3] Jayakody, D., et al. (2024). Instruct-DeBERTa: A Hybrid Approach for Aspect-Based Sentiment Analysis on Textual Reviews. This hybrid model integrates Instruct ABSA for aspect extraction with DeBERTa-V3 for sentiment classification, achieving enhanced accuracy and

reliability across diverse domains, with notable performance on the SemEval 2014 datasets.

[4] Sinha, S., & Narayanan, R. S. (2023). HLESV: Hybrid Lexicon Ensemble-Based Soft Voting Model. This model integrates lexicon-based methods with ensemble techniques through optimally weighted soft voting to enhance sentiment prediction in Amazon product reviews, demonstrating a 1–6% improvement in accuracy over traditional ensemble methods.

[5] WDE-CNN-LSTM (2024) A Hybrid Deep Learning Approach for Enhanced Sentiment Classification* (2024). This model combines Word Embeddings, Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) networks to enhance the accuracy of sentiment classification. It incorporates a consistency analysis between textual sentiment and star ratings, leveraging GPU acceleration for improved processing efficiency.

[7] Naveen Kumar Gondhi, undefined Chaahat, Eishita Sharma, Amal H. Alharbi, Rohit Verma, Mohd Asif Shah, Efficient Long Short-Term Memory-Based Sentiment Analysis of E-Commerce Reviews, Computational Intelligence and Neuroscience, 10.1155/2022/3464524, 2022, 1, (2022).

[8] Alsa B. Nuzi, Vassil G. Guliashki, A Survey on Optimization Methods in Business Information Systems, 2023 24th International Conference on Control Systems and Computer Science(CSCS), 10.1109/CSCS59211.2023.00050, (273-280), (2023).

[10] Vazan, M., Masoumi, F. S., & Majd, S. S. (2022). *A Deep Convolutional Neural Networks Based Multi-Task Ensemble Model for Aspect and Polarity Classification in Persian Reviews*. This study introduces a multi-task learning model utilizing Convolutional Neural Networks (CNNs) to concurrently identify aspect categories and their corresponding polarity in Persian-language reviews. Ensemble learning is employed to further improve the performance of sentiment analysis.

[11] Pathuri et al. (2020): In this study, the authors conducted predictive analysis on Smartphone review ratings using the GPU-CUDA-ENSEMBLED Algorithm. They found that Smartphone reviews generally received lower ratings compared to PC reviews. Additionally, they analyzed the length and count of reviews across different categories.

[15] Pathuri et al. (2022): In this study, the authors conducted predictive analysis on Smartphone review ratings using the CUDA-SADBM Algorithm. CUDA-SADBM classifier can classify datasets when it is implemented in parallel computing (GPU) using a large number of attributes.

[19] K. Barik, S. Misra, A. K. Ray, and A. Bokolo, "LSTM-DGWO-Based Sentiment Analysis Framework for Analyzing Online Customer Reviews," Comput. Intell. Neurosci., vol. 2023, Article ID 6348831, 2023, doi: 10.1155/2023/6348831.

[22] Bharath, P., & Venkata Lakshmi, D. (2024). *FANCFIS: Ensemble Deep Learning with a Novel Fuzzy Approach for Sentiment Analysis*. This model enhances sentiment analysis by integrating deep learning techniques with fuzzy logic, employing a Pyramid Channel-Based Feature Attention Network and Hierarchical Multi-Scale LSTM to effectively capture complex patterns in consumer reviews.

[25] FastXCatStack: A Robust Hybrid Approach with Product Context-Aware Learning* (2024). This model integrates FastText embeddings with an ensemble of XGBoost and CATBoost classifiers for analyzing Amazon user reviews. It achieves high performance across key metrics—precision, recall, accuracy, and F1-score—all at or above 0.92, surpassing the effectiveness of individual classifiers.

RELATED WORK

Product Review

Sentiment analysis, often referred to as opinion mining, is a computational method used to evaluate and classify the sentiment expressed in text as positive, negative, or neutral. This approach is instrumental in analyzing customer feedback, monitoring social media trends, and understanding public sentiment [3][4]. Below is a detailed breakdown of its core components and methodologies:

Text Preprocessing:

Text data requires preprocessing before sentiment evaluation. This involves processes like tokenization (splitting text into words or phrases), removing stop words (e.g., "the," "and"), and applying stemming or lemmatization to reduce words to their base forms.

Feature Extraction:

Extracting meaningful features from the text is vital for sentiment analysis. Common features include individual words (unigrams), word combinations (n-grams), syntactic patterns, and part-of-speech tags. These features are essential for training machine learning models to detect sentiment accurately [5]

.

Classification Techniques:

Machine learning models, particularly classification algorithms, are widely used for sentiment prediction. Traditional models like Support Vector Machines (SVM), Naive Bayes, and Logistic Regression have been effectively applied. Additionally, advanced deep learning methods, such as Recurrent Neural Networks (RNNs) and Transformers, offer improved performance on large datasets.

Sentiment Lexicons:

Lexicons, which are collections of words with associated sentiment scores (positive, negative, or neutral), play a significant role in sentiment analysis. These dictionaries enhance machine learning models by providing predefined sentiment insights.

Aspect-Based Sentiment Analysis (ABSA):

ABSA focuses on determining sentiment related to specific aspects of a product, service, or topic. It goes beyond overall sentiment detection by associating sentiment polarity with specific features or attributes mentioned in the text.

Deep Learning Advancements:

The emergence of deep learning models, such as Convolutional Neural Networks (CNNs), RNNs, and Transformer-based architectures like BERT and GPT, has significantly advanced sentiment analysis accuracy, particularly for large-scale and complex datasets.

Performance Metrics:

Evaluating sentiment analysis models involves metrics like accuracy, precision, recall, F1-score, and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC). These metrics ensure comprehensive model assessment and comparison [6]. This study compares the performance of three notable classification algorithms: SLIQ, MMDBM, and GPU-CUDA-ENSEMBLED.

SLIQ (Supervised Learning in Quest):

SLIQ is a fast and scalable decision tree classifier designed for both numerical and categorical data. It efficiently processes large datasets using a preordering technique during tree construction to minimize computation. Furthermore, SLIQ [7] employs the Minimum Description Length (MDL) principle for pruning, resulting in compact and accurate decision trees [8].

Mixed Mode Database Miner (MMDBM):

MMDBM is a versatile tree classifier that supports multiple data types, including numerical and categorical attributes. It operates in two phases: initially as a predictive classifier and subsequently as a descriptive, object-oriented implementation [9].

GPU-CUDA-ENSEMBLED:

This approach leverages GPU acceleration for the efficient classification of large datasets. It integrates aspect-based analysis and polarity classification into a two-step process. By extracting aspects and determining sentiment polarity, it achieves enhanced classification accuracy [10].

Example of Sentiment Analysis:
Consider this Amazon review:
I feel so LUCKY to have found this used phone online from someone who upgraded and sold this one. My son liked his old one that finally fell apart after 2.5+ years and didn't want an upgrade!! Thank you, Seller, we appreciate it & your honesty. Analyzing this review with text mining techniques results in the following sentiment rules [11]:

- "Lucky" → Positive
- "Liked" → Positive
- "Appreciate" → Positive

Each word contributes to the overall positive sentiment of the review. By employing methods like SLIQ, MMDBM, or GPU-CUDA-ENSEMBLED, we can accurately classify such sentiments and gain valuable insights into customer feedback. Figure 1 shows how a sentiment can be classified.

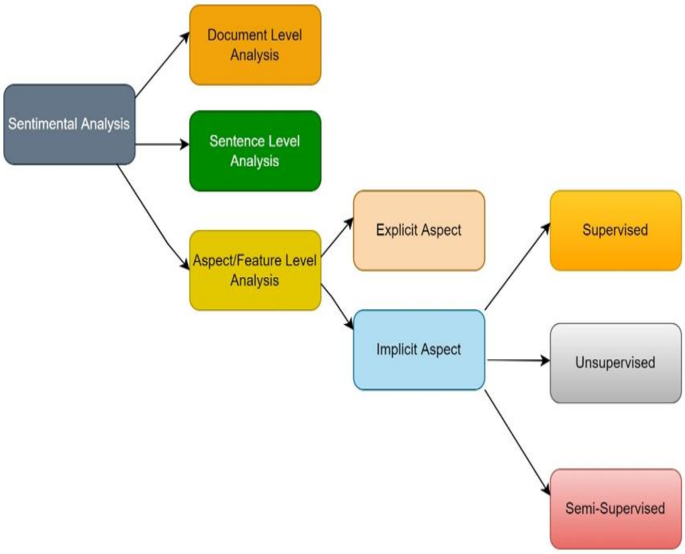

Figure 1. SA Classification

Feature Selection:

Feature selection is an important stage in developing a feature-based sentiment analysis system for product reviews. It entails extracting the most relevant features (words, phrases, or other linguistic aspects) from the text data that helps to determine the sentiment expressed. The following are the typical steps in feature selection for such a system:

Dimensional-Reduction: Feature Selection Techniques: Choosing a subset of the most informative characteristics using statistical metrics such as the chi-square test, information gain, or mutual information [12].

Principal component analysis (PCA): Reducing the dimensionality of the feature space while maintaining variance.

Singular Value Decomposition (SVD) is the process of separating the feature matrix into its most significant components.

Domain-Specific-Feature-Engineering:

Including domain-specific characteristics relevant to product reviews, such as product attributes (e.g., price, quality), sentiment lexicons, and aspect-based features. To choose a product feature, statistical analysis is essential [13].

1. Analyzing the correlation between reviews and ratings.
2. Exploring the link between feedback and ratings.
3. Examining the association between price and ranking.
4. Evaluating the rankings of distinct brands.
5. Tracking the frequency of words repeatedly used by consumers.
6. Identifying the sentiment polarity of each review.

Example: When purchasing products online, such as a mobile phone, consumers often rely on reviews and opinions provided by other users to inform their decisions. These insights are invaluable, helping individuals make informed choices about the products they intend to buy. By considering the experiences and feedback shared by customers, businesses can leverage this information to improve their offerings and achieve success [14]. Figure 2 shows the architecture of the proposed model.

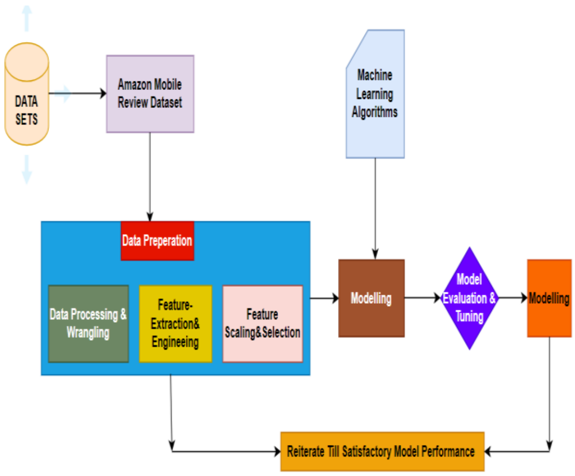

Figure 2. System Architecture

Polarity Classification:

Sorting by polarity refers to the task of determining the emotion or feeling expressed in the text. This includes categorizing the text as positive, negative, or neutral. This practice is common in natural language processing (NLP) and sentiment analysis applications, where understanding the meaning of text data is important for various tasks, such as customer feedback analysis, social media monitoring, and review analysis of products.

There are many approaches to polarity. There are many different classifications, from simple rule-based methods to more complex machine learning and deep learning methods. Rule-based methods use predefined rules or lexicons to assign polarity to words or phrases. Machine learning approaches use curated training data to train a model that can automatically classify text polarity. Deep learning methods such as recurrent neural networks (RNNs) and convolutional neural networks (CNNs) can also be used for polarity classification, often achieving state-of-the-art performance in analysis.

Example:

a = 'This is the worst mobile." Applying Vader's sentiment analysis to the above text with the help of the polarity scores(a) method, the text is classified as 'bad'.0.9, 'neutral': 0.1, "positive ": 0, "composite":
0.8404}.

METHOD

This work is based on three classification methods: SLIQ, MMDBM, and GPU-CUDA-ENSEMBLED. SLIQ (Suspended Learning In Quest) is one of the fastest growing in data mining. SLIQ is a decision tree classification method that can manage numerical and categorical data and can also classify large-scale training data. It is used to create small and accurate trees and work based on pre-section methods during tree growth to reduce the cost of evaluating annual properties. The Decision Tree Miner (MMDBM) is another that can handle numerical and attribute values in large datasets. The algorithm is divided into two parts. The first is predictive classification, which provides a detailed description of the algorithm, and the second is design-based, which provides a graphical implementation. Machine learning is an important part of AI and DS, which involves using some statistical data for analysis and telling the machine what to do. Figure 3 shows how to predict output from the designed model.

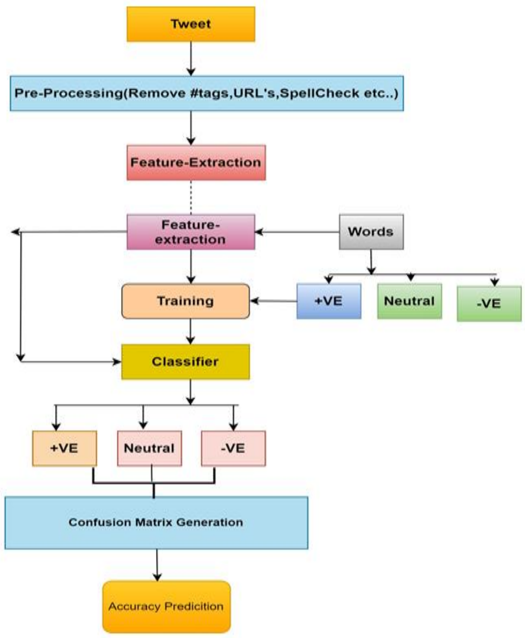

Figure 3. Workflow of Proposed Classifier

Overview of CUDA:

CUDA [15], or Compute Unified Device Architecture, is a parallel computing platform and application programming interface (API) model created by NVIDIA. It allows developers to harness the computational power of NVIDIA GPUs (Graphics Processing Units) for general-purpose processing tasks beyond just graphics rendering.

Parallel Processing Power: CUDA enables developers to tap into the massively parallel processing capabilities of modern GPUs. GPUs consist of hundreds to thousands of cores that can perform computations concurrently, making them well-suited for tasks that can be parallelized.

Performance: By offloading computation-intensive tasks to the GPU, applications can achieve significant performance gains compared to running on traditional CPUs alone. This is especially true for tasks involving large datasets or complex algorithms.

Versatility: CUDA supports a wide range of programming languages, including C, C++, Python, and Fortran, making it accessible to a broad developer community. It also provides libraries and tools for various domains, such as linear algebra, signal processing, and machine learning [16].

Deep Learning and AI: CUDA has become indispensable in the field of deep learning and artificial intelligence (AI) [17]. Many popular deep learning frameworks, such as TensorFlow, PyTorch, and MXNet, leverage CUDA to accelerate training and inference on GPUs, enabling faster experimentation and model development.

Scientific Computing: CUDA is widely used in scientific computing for tasks such as simulations, numerical analysis, and computational fluid dynamics. Its ability to perform parallel computations efficiently makes it invaluable for researchers and engineers working in fields such as physics, chemistry, and biology.

Data Processing: With the increasing volume and complexity of data in various domains, CUDA is used for accelerating data processing tasks like image and video processing, computer vision, and data analytics. Figure 4 shows the difference between CPU and GPU.

The primary objective of this paper is to apply GPU computing to Amazon Unlocked mobile data and forecast its classification accuracy. It is based on feedback given by customers of the product on online social networks using the category. Things like agility and specificity improve predictions. In this paper, we have reviewed or read a product function and qualified it as positive, negative, or neutral [19]. To achieve this, we use Vader's sentiment classification to evaluate the data at the word and sentence level. Once the binary format is complete, the polarity is expressed as 0s and 1's, the test data is classified and studied, a confusion matrix is created, and the precision, recall, and F1 scores are calculated. The goal is to see the accuracy of the data processed in the GPU software.

For this purpose, we use three types of algorithms (1) GPU-CUDA-ENSEMBLED algorithm (2) MMDBM (3) SLIQ. Convergence is used to test the hypothesis and then decide how to study it. Stacking: Stacking is the process of combining results from different models (e.g., a decision tree) to produce detailed results [18].

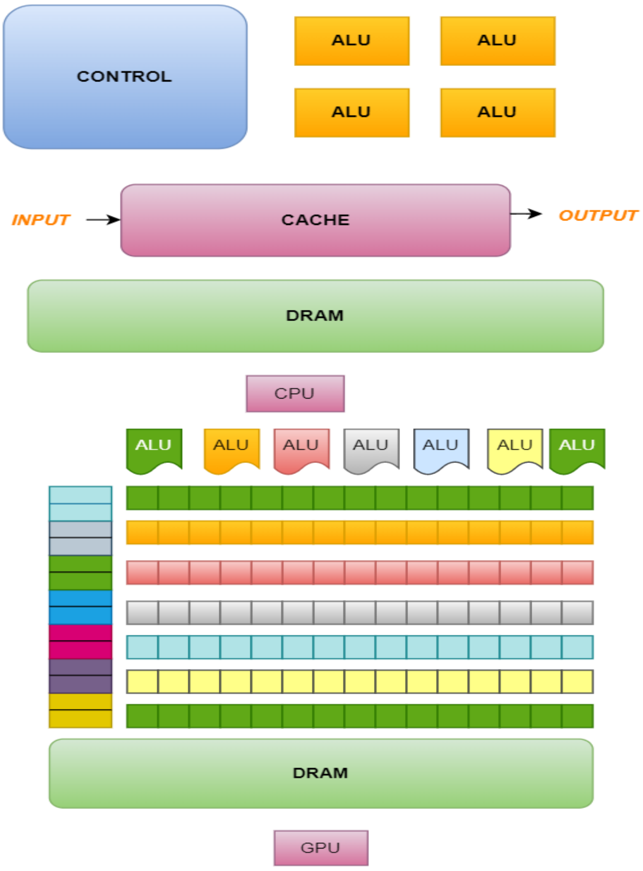

Figure 4. CPU VS GPU

In this case, the bootstrap is a sampling strategy that determines a subset of variables to analyze from the initial data set. The size of the array is equal to the size of the original array. Multiple subsets are created by selecting surrogate observations from the original data set. For each subset, a base model (weak model) was created. The final prediction is calculated by mixing all the predictions. Boosting: It is a sequential process [20]. Errors in the above examples are replaced by the following examples. In this work, we use a hybrid algorithm that combines bagging and boosting, that is, a mixed algorithm [21]. Due to its hybrid nature, this algorithm produces a classifier that removes bias at each training level.

RESULTS AND DISCUSSION

Classification algorithms such as SLIQ, MMDBM, and GPU-CUDA-ENSEMBLED are used. Also, all algorithms are compared for processing time when implemented in the CUDA [22] program. GPU mining takes less processing time compared to CPU. Therefore, when tested on different threads, GPU mining algorithms show that GPU computation is faster than CPU computation. To verify the effectiveness of the algorithm, we applied it to a randomly generated Amazon Mobile Phone dataset.

The task here is to predict the star rating of a mobile based on its rating [23]. It was done in a random database. The implementation of the algorithm can be divided into 7 steps and described as follows. In CUDA, no connection between the database and CUDA is possible. Therefore, the data for classification should be randomly generated. The first step is to generate random data, which is processed by CUDA using a built-in function called curand. CUDA [24] is very fast when generating random data. In other words, it took 0.05 seconds to generate 100,000 pieces of data. In the second step, we need to change the number and type of objects. Elements and their labels. In the third stage of implementation, the objects are represented in the form of a decision tree [25]. Here, the decision tree is a binary tree because it only classifies the completed data. The fourth step sorts the values and sets the separator for each numeric element. The fifth part of the implementation is: Model the tree-based classification rules and use them in the GPU program. The sixth step of the implementation is to allocate memory for the data on the hardware (GPU) [26]. This is done using a built-in function called cudaMalloc. Then copy the received data to the GPU device. This is done using a built-in function called cudaMemcpy. This method should be used twice. In other words, you must copy the data from the host (CPU) to the hardware (GPU) and copy the results from the hardware (GPU) to the host (CPU). The implementation is to find the precision of the results on the GPU.

The sequence occurs when only 128 threads are running for 100,000 results. This is because data is unique. The number of data sent represents the number of threads started and the process completed in microseconds [27]. Table 1 shows the various threads and counts. Table 2 shows the Acceleration Ratio time for Classifying Records using the GPU-CUDA-ENSEMBLED Algorithm.

Algorithm for CUDA g-Boost Algorithm in GPU

Input: An array containing n attributes, A = {x1, x2, ..., xn}, provided in parallel.
Output: Construction of the decision tree and the model's accuracy.
Initialize Threads: Set up threads in the GPU.
Generate Data: Randomly generate data values within the dataset.
Data Transfer to CPU: Transfer data from the GPU to the CPU host using cudaMemcpy.
Binarize the values in arrays.
Data Sorting in GPU: Copy the binarized values back to the GPU. Sort the arbitrary data within the GPU device.
Find Midpoints: Transfer the sorted data from the GPU back to the CPU. Calculate the midpoint for each attribute.
Feature Selection: Randomly select X features from the total Y features, ensuring X < Y.
Node Calculation: Using the selected X features, determine the node Z using the best split.
Node Splitting: Split these nodes into sub nodes (child nodes) using the best split criteria.
Repeat Splitting: Continue steps 6 to 8 until the W number of nodes is reached.
Build Forest: Repeat steps 1 to 9 K times to create L number of trees.
Train Model with XGBoost:
Call xgb().
Initialize the model:
model = xgb.XGBClassifier(random_state=1, learning_rate=0.01).
Train the model with training data: model.fit(x_train, y_train).
Evaluate Model: Calculate the model's accuracy: model.score(x_test, y_test).
Transfer Results: Transfer the model's accuracy score from the CPU back to the GPU.
Classify the data, compute the node count, and class count.

Table 1. Shows various threads and count

S. No.	No. of Threads	Time Take (s)
1	128	5.45
2	256	4.34
3	512	3.25
4	1024	2.82
5	2048	1.85

Table 2. Shows Acceleration Ratio time for Classifying Records using GPU-CUDA-ENSEMBLED Algorithm

No. of GPUs Times 10000	No. of. Records Sec/30000	No. of. Records Sec/50000	No. of. Records Sec/70000	No. of. Records Sec/100000	No. of. Records Sec/130000
Classification Time	0.535	1.015	1.620	2.045	2.684
CPU Time	0.710	1.130	1.740	2.350	2.900
GPU Time	0.550	1.010	1.640	2.230	2.490
Acceleration Ratio	1.296	1.118	1.064	1.054	1.16491

Return Results: Send the result from the GPU to the CPU host, which is the classifier's accuracy.
End Process:

Acceleration Ratio for GPU:

The GPU acceleration ratio [28] refers to the acceleration achieved by offloading computing tasks from the CPU to the GPU. This indicates performance improvement that can be achieved using the parallel processing capabilities of the GPU compared to running the same task on the CPU alone.
The acceleration rate can be calculated using the following formula:
Acceleration Ratio = Time taken by CPU / Time taken by GPU
Where:
"CPU time" is the time required to complete a computer task when executed on a CPU.
"GPU Time" is the time required to complete the same computing task while running on the GPU.

Performance evaluation method:

In machine learning, various performance metrics are used to evaluate the performance of models across different tasks such as classification, regression, clustering, and more. Here are some commonly used performance metrics along with their formulas:

$$Accuracy=(TP+TN)/(TP+TN+FP+FN) \quad (1)$$

Precision=TP/ FP+TP (2)

Recall=TP / FN+TP (3)

$$F1 = \frac{2(Precison * Recall)}{(Precision + Recall)} \quad (4)$$

The outcomes of the proposed approaches are assessed along with other classifiers utilizing various evaluation metrics, including the confusion matrix, sensitivity, specificity, precision, recall, F-measure, G-mean [29], balanced classification rate (BCR), and accuracy, are shown in Table 3, Table 4 and Table 5. Table 6 shows the overall comparison of all three 3 classifiers. Figure 5 shows the comparison of the two classifiers.

Table 3. Shows the accuracy of the MMDBM Algorithm

Star-rating	Precision	Recall	F-Score	Support
1	87.4	86.5	87.6	9138
5	90.5	90.5	90.5	22574
MicroAvg 87.12	87.13	87.25	31715	
MacroAvg 88.11	89.23	88.24	31715	
WeightedAvg	90.22	88.25	88.25	31715

Table 4. Shows the accuracy of the SLIQ Algorithm

Star-rating	Precision	Recall	F-Score	Support
1	85.24	87.33	87.16	9138
5	87.45	88.26	88.91	22577
MicroAvg 87.11	88.21	88.22	31715	
MacroAvg 87.21	88.27	87.42	31715	
WeightedAvg	87.11	88.21	88.44	31715

Table 5. Shows the accuracy of the Proposed Algorithm

Star-rating	Precision	Recall	F-Score	Support
1	94.24	94.5	95.6	9138
5	95.51	95.52	94.56	22574
MicroAvg	95.67	95.25	31715	
MacroAvg	94.23	94.24	31715	
WeightedAvg	95.22	95.25	95.25	31715

Table 6. Comparison of GPU-CUDA-ENSEMBLED with supervised learning methods for the Amazon Mobile Phone dataset

Methods	Precision	Recall	F-Score	Support
Proposes GPU-CUDA-ENSEMBLED	95.51	95.52	94.5	95
MBDM	90.22	90.25	90.25	90
SLIQ	87.11	87.21	87.44	87

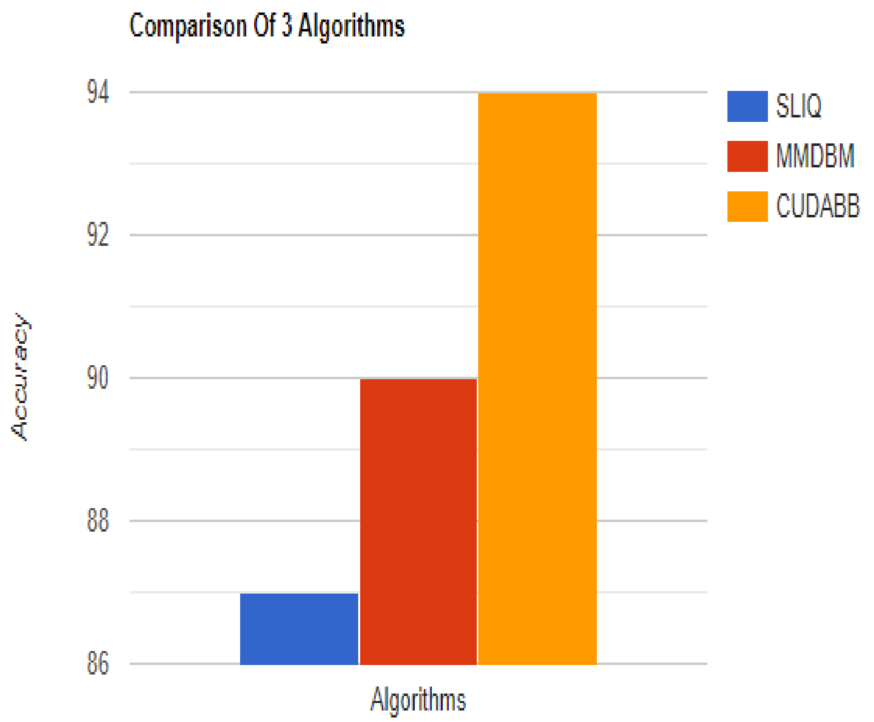

Figure 5. Comparison of three Classifiers

CONCLUSION

The proposed GPU-CUDA-ENSEMBLED algorithm is implemented using multiple objects in GPU parallel computing to classify the dataset. The analysis does not reflect the unbiased, negative, or unbiased sentiments in the reviews. The algorithms can be implemented for edge and polarity classification, identifying the results using machine learning algorithms. The results are compared with SLIQ and MMDBM using GPU, and the speedup ratio times are calculated. GPU Improves mining performance while reducing processing time. The proposed method has a better precision of 94% with 94.50% precision, 93.95% recall, 94.21% F-measure, 94.13% BCR, and 9.67%. In future research, the proposed algorithm can be applied to various real-world applications such as banking, biomedicine, and big data analysis.

REFERENCES

[1] H.L. Chang, Y.L. Liu, C.J. Keng, H.L. Jiang, “Examining Discrepancies between Online Product Ratings and Sentiments Expressed in Review Contents,” *Management Analytics and Social Insights*, vol. 1, pp. 129–144, 2024, doi: 10.22105/r4kfnt67

[2] E. S. Apostol, A. G. Pisică, C. O. Truică, Ciprian, "ATESA-BÆRT: A Heterogeneous Ensemble Learning Model for Aspect-Based Sentiment Analysis," in *Proc. IEEE*, 2023, doi: 10.48550/arXiv.2307.15920

[3] D. Jayakody et al., "Instruct-DeBERTa: A Hybrid Approach for Aspect-Based Sentiment Analysis on Textual Reviews," in *Proc. IEEE*, 2024, doi: 10.48550/arXiv.2408.13202

[4] S. Sinha and R. S. Narayanan, "HLESV: Hybrid Lexicon Ensemble-Based Soft Voting Model," in *Proc. IEEE*, 2023.
[5] S. E. Sorour, A. Alojail, A. El-Shora, A. E. Amin, & A. A. Abohany, "A Hybrid Deep Learning Approach for Enhanced Sentiment Classification and Consistency Analysis in Customer Reviews," *Mathematics*, vol. 12, no. 23, 3856, 2024, doi: 10.3390/math12233856
[6] S. K. Pathuri and N. Anbazhagan, "Prediction of Cardiovascular Disease Using Classification Techniques with High Accuracy," *Journal of Advanced Research in Dynamical and Control Systems (JARDCS)*, vol. 12, no. 02, 2020.
[7] N. K. Gondhi, C. Chaahat, E. Sharma, A. H. Alharbi, R. Verma, M. A. Shah, "Efficient Long Short-Term Memory-Based Sentiment Analysis of E-Commerce Reviews," *Computational Intelligence and Neuroscience*, 2022, doi: 10.1155/2022/3464524
[8] M.S. Hossain, M.F. Rahman, "Customer sentiment analysis and prediction of insurance products' reviews using machine learning approaches," *FIIB Business Review*, vol. 12, no. 4, pp. 386–402, 2023, doi: 10.1177/23197145221115793
[9] M. M. Baig, S. Sivakumar, and S. R. Nayak, "Optimizing Performance of Text Searching Using CPU and GPUs," in *Advances in Intelligent Systems and Computing*, vol. 1119, pp. 141–150, 2020, doi: 10.1007/978-981-15-2414-1_15
[10] M. Vazan, F. S. Masoumi, and S. S. Majd, "A Deep Convolutional Neural Networks Based Multi-Task Ensemble Model for Aspect and Polarity Classification in Persian Reviews," in *Proc. IEEE*, 2022, doi: 10.13140/RG.2.2.27089.81768
[11] S. K. Pathuri, N. Anbazhagan and G. B. Prakash, "Feature Based Sentimental Analysis for Prediction of Mobile Reviews Using Hybrid Bag-Boost algorithm," *2020 7th International Conference on Smart Structures and Systems (ICSSS)*, Chennai, India, 2020, pp. 1-5, doi: 10.1109/ICSSS49621.2020.9201990.
[12] A. Patel, P. Oza, and S. Agrawal, "Sentiment analysis of customer feedback and reviews for airline services using language representation model," *Procedia Computer Science*, vol. 218, pp. 2459–2467. 2023, doi:10.1016/j.procs.2023.01.221
[13] Z. Wang, Y. Zhu, S. He, H. Yan, Z. Zhu, "Llm for sentiment analysis in e-commerce: A deep dive into customer feedback," *Applied Science and Engineering Journal for Advanced Research*, vol. 3, no. 4, pp. 8–13, 2024, doi: 10.5281/zenodo.12730477
[14] R. Vatambeti, S. V. Mantena, K. Kiran, M. Manohar, and C. Manjunath, "Twitter sentiment analysis on online food services based on elephant herd optimization with hybrid deep learning technique," *Cluster Computing*, vol. 27, pp. 655–671, 2024, doi: 10.1007/s10586-023-03970-7
[15] S. K. Pathuri, N. Anbazhagan, G. P. Joshi, and J. You, "Feature-Based Sentimental Analysis on Public Attention towards COVID-19 Using CUDA-SADBM Classification Model," *Sensors*, vol. 22, no. 1, pp. 1–16, 2022, doi:10.3390/s22010080.
[16] N. Heidari, P. Moradi, A. Koochari, "An attention-based deep learning method for solving the cold-start and sparsity issues of recommender systems," *Knowledge-Based Systems,* vol. 256, 109835, 2022, doi: 10.1016/j.knosys.2022.109835.
[17] A. B. Nuzi, V. G. Guliashki, "A Survey on Optimization Methods in Business Information Systems," *2023 24th International Conference on Control Systems and Computer Science (CSCS)*, 2023, pp. 273-280, doi: 10.1109/CSCS59211.2023.00050
[18] L. S. Videla and M. A. Kumar, "Fatigue Monitoring for Drivers in Advanced Driver-Assistance System," in *IGI Global*, 2020, pp. 170–187, doi: 10.4018/978-1-7998-0066-8.ch008
[19] K. Barik, S. Misra, A. K. Ray, A. Bokolo, "LSTM-DGWO-Based Sentiment Analysis Framework for Analyzing Online Customer Reviews," *Computational Intelligence and Neuroscience*, vol. 1, 2023, doi: 10.1155/2023/6348831
[20] M. R. Asadabadi, M. Saberi, N. S. Sadghiani, O. Zwikael, & E. Chang, "Enhancing the analysis of online product reviews to support product improvement: Integrating text mining with quality function deployment," *Journal of Enterprise Information Management*, vol. 36, pp. 275–302, 2023
[21] T. Crnovrsanin, S. D. Bartolomeo, C. Wilson and C. Dunne, "Indy Survey Tool: A Framework to Unearth Correlations in Survey Data," *2023 IEEE Visualization and Visual Analytics (VIS)*, Melbourne, Australia, 2023, pp. 146-150, doi: 10.1109/VIS54172.2023.00038.

[22] P. Bharath and D. Venkatalakshmi, “FANCFIS: ensemble deep learning based features learning with a novel fuzzy approach for sentiment analysis,” *International Journal of Information Technology*, 2024, doi: 10.1007/s41870-024-01882-2

[23] M. Birjali, M. Kasri, A. Beni-Hssane, “A comprehensive survey on sentiment analysis: Approaches, challenges and trends,” *Knowledge-Based Systems*, vol. 226, 107134, 2021, doi: 10.1016/j.knosys.2021.107134

[24] K. L. Tan, C. P. Lee & K. M. Lim, ”RoBERTa-GRU: A Hybrid Deep Learning Model for Enhanced Sentiment Analysis,” *Applied Sciences*, vol. 13, no. 6, 3915, 2023, doi: 10.3390/app13063915

[25] E. Hashmi, S.Y. Yayilgan, “A robust hybrid approach with product context-aware learning and explainable AI for sentiment analysis in Amazon user reviews,” *Electronic Commerce Research*, 2024, doi: 10.1007/s10660-024-09896-5

[26] S. Gheewala, S. Xu, S. Yeom, S. Maqsood, “Exploiting deep transformer models in textual review-based recommender systems,” *Expert Systems with Applications*, vol. 235, 121120, 2024, doi: 10.1016/j.eswa.2023.121120.

[27] A. A. Altaye and J. S. Nixon, "A Comparative Study on Big Data Applications in Higher Education," *International Journal of Emerging Trends in Engineering Research*, vol. 7, no. 12, pp. 1–8, Dec. 2019, doi: /10.30534/ijeter/2019/027122019

[28] M.E. Alzahrani, T.H. Aldhyani, S.N. Alsubari, M.M. Althobaiti, A. Fahad, “Developing an Intelligent System with Deep Learning Algorithms for Sentiment Analysis of E-Commerce Product Reviews,” *Computational Intelligence and Neuroscience*, vol. 22, no. 1, 3840071, 2022, doi: 10.1155/2022/3840071

[29] D. Suhartono, K. Purwandari, N.H. Jeremy, S. Philip, P. Arisaputra, I. H. Parmonangan, “Deep neural networks and weighted word embeddings for sentiment analysis of drug product reviews,” *Procedia Computer Science*, vol. 216, pp. 664–671, 2023, doi: 10.1016/j.procs.2022.12.182