Sentiment Analysis of Product Reviews Using Fine-Tuned LLaMa-3 Model: Evaluation with Comprehensive Benchmark Metrics

*Yil*i Wang

School of Computer Science and Technology, China University of Mining and Technology, Nantong City, Jiangsu Province, 226000, China

Abstract. Sentiment analysis, a crucial subfield of natural language processing, enables businesses and policymakers to understand public emotions and opinions, essential for crafting effective strategies across industries like marketing and customer service. As the volume of online reviews grows, automated sentiment classification models have become vital for efficiently processing this data. This study explores fine-tuning the LLaMA-8B large language model based on the Amazon Product Reviews dataset from Kaggle, aiming to improve sentiment classification accuracy. Using the LoRA fine-tuning approach combined with the Variant Greedy Search Technique (VGST) and TextBlob for polarity handling, the research addresses dataset size challenges. The model's fine-tuning process includes one-shot learning and chain-of-thought prompting to better capture nuanced sentiment expressions. Evaluated using comprehensive metrics, LLaMA-8B demonstrates superior precision compared to Qwen2-7B and achieves near LLaVA performance with enhanced speed. Additionally, it outperforms models like Decision Tree, SVM, Multinomial NB, and XLNet in accuracy. This work underscores the potential of large language models for sentiment analysis and sets the stage for future extensions to multimodal input scenarios.

1 Introduction

As an important branch of natural language processing (NLP), sentiment analysis has been widely used in various industries, It is widely utilized in various fields, including marketing, feedback evaluation, and customer service. By analyzing the textual content that people post on social media, comment platforms, and more, businesses and policymakers are able to gain a deeper understanding of the public's emotions and opinions, which is critical for developing more effective strategies. Especially in the analysis of book reviews, sentiment analysis can reveal readers' feelings about a certain work, provide data support for publishers, help them optimize marketing strategies and book recommendation mechanisms, and even guide the direction of new book creation.

With the rapid growth of the number of online reviews, sentiment analysis is becoming

Corresponding author: 03220957@cumt.edu.cn

increasingly important. However, in the face of such a large amount of data, manual analysis alone can no longer meet the needs of efficiency and accuracy. Therefore, it is critical to develop high-precision automated sentiment analysis models. These models are able to quickly and accurately classify emotions, enabling companies to capture consumer sentiment trends and make informed decisions in a highly competitive market.

Many sophisticated models have emerged in the realm of NLP, particularly with the advent of large-scale pre-trained language models like GPT, BERT, and LLaMa, which have transformed the landscape. These models employ a Transformer-based architecture, enabling them to comprehend and produce text that mirrors human language with exceptional accuracy. LLaMa-3, the most recent version in this series, has gained attention for its improved stability and contextual understanding, highlighting its strong potential for sentiment analysis tasks[1].

The introduction of LLaMa-3 not only improves the processing power of the model in complex scenarios but also performs well in dealing with diverse text data. Compared to earlier models, LLaMA-3 enhances its ability to accurately capture nuanced emotional shifts in text by introducing improved contextual embedding techniques and a deeper Transformer architecture. This is particularly beneficial when dealing with ambiguous or polysemous expressions. As a result, LLaMa-3 has become an important tool in the field of sentiment analysis today, demonstrating its analytical capabilities when dealing with large volumes of online comments, social media content, and other user-generated text [2].

While LLaMa-3 demonstrates significant potential, it may struggle to fully capture the complexity and nuance of subjective texts like book reviews. Current research often relies on limited benchmarks, focusing mainly on accuracy, which doesn't provide a complete picture of model performance. Metrics like precision, recall, F1 score, and AUC-ROC are equally important as they assess how well the model handles imbalanced data and its overall robustness. A multidimensional evaluation is crucial for understanding and improving the model's effectiveness in complex sentiment analysis tasks.

This research addresses these gaps by deploying and fine-tuning the LLaMa-3-8B model specifically for sentiment analysis of book reviews. The process involves quality-tested sampling and preprocessing of a book review dataset, followed by prompt engineering. The LLaMa-3-8B model is then fine-tuned using the Freeze technique and evaluated with a comprehensive set of benchmark metrics. The Freeze method, that is, parameter freezing, freezes some parameters of the original model and trains only some parameters, which can achieve the expected training effect through this technology.

2 Data and methodology

This paper utilizes a dataset from Amazon that contains product reviews. Due to issues such as duplicate and false data, as well as an excessively high proportion of positive reviews, this paper has decided to process the dataset to extract more valuable insights. This section also introduces data sources, filtering techniques, and fine-tuning methods.

2.1 Dataset Sampling

The sampling methodology for the Amazon Product Reviews dataset, which includes 500,000 product reviews provided by Kaggle, was conducted in two phases. The primary goal is to maintain data quality and consistency (DQC) and ensure data diversity and complexity (DDC) [3].

To ensure DQC, the study started by verifying the precision of the book review ratings, which range from Tier 1 to Tier 5 in the original dataset, with higher tiers signifying more positive feedback. This paper used TextBlob, a NLP tool, utilizing its polarity feature for this purpose [4]. Each review underwent sentiment evaluation, and the outcomes were compared

to the associated customer ratings to check for alignment. TextBlob provides a polarity score as a float between -1 and 1, where 1 represents the most positive sentiment and -1 is the most negative, similar to Amazon's 1 to 5 rating scale. This paper discarded reviews where the sentiment label was negative (< 0) but rated higher than Tier 3, or positive (> 0) but rated lower than Tier 3, to ensure consistency between the review content and its score. To address potential biases and ensure fair representation across different sentiment levels, this paper utilized a stratified sampling method. This approach led to an even distribution of reviews among all rating tiers. Specifically, the final dataset contains an equal number of samples for each distinct rating level, ensuring consistent representation across the sentiment range. This strategy of balanced sampling is designed to reduce class imbalance issues and improve the reliability of future sentiment analysis models. Following the data refinement process, this paper implemented a Variant of the Greedy Search Technique (VGST) to minimize information loss during sampling while maintaining a 1% sampling rate, thereby addressing the DDC principle. The algorithm iteratively evaluates each data point, ensuring that at each step (starting with an empty sample set), the addition of data maximizes token diversity until the target sample size is achieved.

In practical applications, this paper found that traditional Greedy Search Techniques (GST) tend to have high computational demands, especially when processing resources are constrained. To overcome this challenge, this paper developed an optimized approach called the VGST, which strikes a better balance between information retention and computational efficiency. VGST introduces two innovative parameters: batch_size and wishlist_len. Here's how the algorithm works: 1. Randomly select a number of data points from the dataset equal to batch_size. 2. Determine which data point in the batch maximizes token diversity. 3. Add this selected point to a wishlist. 4. Repeat steps 1-3 until the wishlist reaches the specified length, wishlist_len. 5. Choose the best data point from the wishlist as the final sample.

This strategy capitalizes on the typical token redundancy found in book reviews to balance retaining information with maintaining data diversity, all while substantially lowering computational requirements. The effectiveness of VGST relies on the careful tuning of the wishlist_len and batch_size parameters. To demonstrate VGST's performance, this paper offer comparative analyses that evaluate the computational time of VGST against GST, showcasing the method's improved efficiency. These empirical findings support the capability of VGST to preserve data quality while significantly decreasing computational load(Figure 1 illustrates the flow of the algorithm).

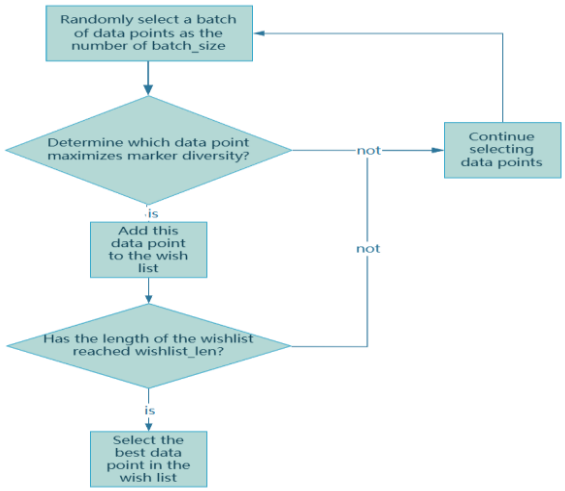

Figure 1 VGST algorithm (Photo/Picture credit : Original)

2.2 Pre-Processing

In the approach to guiding a machine learning model to generate outputs with specific types, topics, or formats, this paper employ prompt engineering during the pre-processing phase. This paper instruct the model to assess user reviews and categorize them based on sentiment using the directive: Evaluate the sentiment expressed in user reviews and classify each one according to its sentiment rating. For this task, this paper use a five-point scale to classify the reviews, where ratings of 1 and 2 are considered negative, 3 are seen as neutral, and 4 and 5 are regarded as positive. Due to the subtlety in distinguishing neutral reviews, this paper deliberately increased their representation in the dataset. By explicitly defining the criteria for emotion classification, this paper ensure the model learns effectively through these prompts.

Comments show a high level of dissatisfaction and negativity, and users may mention serious flaws, unpleasant experiences, or quality of service issues. This score is often accompanied by strong negative feedback, indicating that users have experienced great disappointment.

Although the review is still negative, the user's sentiment may be slightly softened. Comments may express dissatisfaction, but they may also mention individual areas that could be improved, or user expectations that have not been met, rather than being completely disappointed.

The review is generally neutral, probably mentioning both positive and negative aspects. The user's experience may be mediocre, with no particularly outstanding or particularly bad details, which is a more balanced attitude.

Users have a positive experience overall. Reviews may be full of praise, praising multiple aspects of a product or service, but there may be a few that leave something to be desired, indicating that the user experience is generally satisfactory but there is still room for improvement.

Showing a high level of satisfaction and positive emotions, user reviews are often full of praise and testimonials, with little to no mention of deficiencies. This type of review indicates that the user is extremely satisfied with all aspects of the product or service and is willing to use it again or recommend it to others.

This paper utilize two common methods in prompt engineering: one-shot Learning, which involves providing the model with multiple examples of tasks to help it better understand and generate appropriate output. This paper turn the sampled reviews into questions and answers and provide correct answers for the model. One-shot aims to make models learn to "learn" and be able to handle similar types of tasks [5]. The dataset takes the form of multiple rounds of conversations, and this paper set up a large number of prompts with different expressions. This approach allows the model to clearly understand the test intent while avoiding the common problem of overfitting.

Indeed, The one-shot method can improve the reasoning ability of the model to a certain extent. However, using only one-shot Learning has a limited effect. For the slightly complicated reasoning questions, the model still cannot give accurate answers. Thus, this paper use another method Zero-shot-CoT to reinforce the accuracy of prediction [6]. Zero-shot-CoT is an improved method of prompting that builds upon the one-shot approach. It adopts the concept of a thinking chain (CoT) to solve problems more efficiently. First, the model needs to understand the nature of the problem, and then generate a series of logical steps to form a Thought chain to guide the model from the problem to the solution. An important advantage of this approach is that it reduces the reliance on large amounts of labelled data because the model can fill in the gaps in knowledge through reasoning. We've taken a somewhat simplistic approach to this idea, adding a “Let's take it one step at a time” directive to some of the tips or a brief overview of the ideas contained in the dataset.

2.3 Fine-Tuning

Large language models, usually involving billions of parameters, need to be fine-tuned based on the pre-trained model to accommodate the specific application scenario. Two choices are given during the fine-tuning process. The first one is to fine-tune all parameters involved, which is highly costly. The alternative is to fine-tune only for weights and parameters of certain layers. This kind of fine-tuning can reduce storage space and accelerate deployment, but there is some loss of performance and model quality instead.

To retain the advantages of both methods, Low-Rank Adaptation (LoRA) has been utilized for the following reasons: while preserving all parameters of the base model (W) and freezing the weight of the pre-trained model, LoRA decomposes the weight update matrix (ΔW) into a product of two compact, low-rank matrices. By adjusting the ΔW represented by the product of two low-rank matrices, this manipulation not only reduces storage consumption and improves the fine-tuning speed but also retains the whole quality of the model [7].

Specifically, in the process of fine-tuning, LoRA represents the d-dimensional matrix ΔW as the product of two matrices B and A with smaller parameters, where the dimension of B is d times r, the dimension of A is r times d, and the rank r is much smaller than d. Thus, the number of parameters fine-tuned is changed from d times d to 2 times r times d, which significantly reduces the number of parameters and computational complexity. The A matrix is initialized to the Gaussian distribution matrix, and the B matrix is initialized to the 0 matrix. Thus, in the beginning, ΔW is initialized to the 0 matrix. At the same time, this paper introduces two hyper-parameters α and γ and uses the ratio α / γ to scale the update matrix and adjust its update step [8,9].

$$h = Wx + \Delta Wx = Wx + BAx \tag{1}$$

Where x represents the input, and h represents the output of input x after a series of transformations. Beside these two hyper-parameters, this paper also trained multiple hyper-parameters related to LoRA(As shown in Table 1) [10,11].

Table 1 The hyper-parameter settings

parameter	value	parameter	value
learning rate	0.00005	enable external logger	1
epochs	3	trainable layers	3
maximum gradient norm	1	LoRA rank	4
max samples	2000	LoRA alpha	64
compute type	fp16	LoRA dropout	0
cutoff length	1024	LoRA+ LR ratio	8
batch size	3	use rsLoRA	0
gradient accumulation	4	use DoRA	0
val size	0.2	Beta value	0.5
LR scheduler	cosine	Ftx gamma	2
logging steps	5	loss type	sigmoid
save steps	100	use Galore	0

warmup steps	0	Galore rank	0
NEFTune Alpha	0.02	update interval	0
optimizer	adamw_torch	Galore scale	0
resize token embeddings	0	use BAdam	0
upcast layernorm	1	BAdam mode	0
enable S^2 attention	0	switch mode	0
pack sequences	0	update ratio	0
enable LLaMA Pro	0		

3 Results

The performance of fine tuned sentiment classification model on the Amazon dataset is compared with four general baselines. In summary, the dataset comprises various filtered categories of English text, which this paper uses for training language models. Since this paper deployed only 4,000 reviews for training, an additional 4,000 reviews were randomly selected from the remaining data for testing purposes.

After obtaining the optimal hyperparameters, I conducted a comparison of multiple benchmark models. The results of this comparison are presented in Table 2.

Table 2: Performance of different models after fine-tuning

Model	predict_bleu-4	predict_rouge-1	predict_samples_per_second
LLaMA3-8B	30.2327	85.5212	6.079
LLaVA1.5-7B	30.2846	85.6835	1.563
Qwen2-7B	30.1275	85.2180	6.003

The evaluation metrics predict_bleu-4,predict_rouge-1, and predict_samples_per_second are instrumental in assessing the performance of natural language generation systems. Specifically, predict_bleu-4 measures the precision of overlapping n-grams between the generated text and reference text, focusing on 4-gram matches, which is crucial for ensuring the output's fluency and grammatical accuracy. Meanwhile, predict_rouge-1 evaluates unigram overlaps, providing insight into the lexical selection and recall ability of the system to cover expected content. These metrics, in conjunction with predict_samples_per_second, which quantifies the efficiency of the model by calculating how many samples are processed per second, collectively offer a comprehensive view of both qualitative aspects of the text generation and the model’s computational performance. This holistic evaluation is fundamental in informing iterative improvements and optimizing systems for both accuracy and efficiency.

It can be observed that the LLaMA3-8B model outperforms the Qwen2-7B model. Although it shows a slight disadvantage in accuracy compared to the LLaVA1.5-7B model, its generation speed is several times faster than that of LLaVA1.5-7B. Therefore, this paper chose the optimal LLaMA3-8B model for further analysis.

For comparative analysis, four robust baseline models were selected. The first, the Decision Tree, is widely utilized in data mining to create classification systems based on various covariates or to develop predictive algorithms for target variables. This technique constructs a tree structure with root, internal, and leaf nodes from the training data, using CountVectorizer for generating numerical word representations. Secondly, Support Vector Machine (SVM), a supervised learning model renowned for its efficacy in classification and

regression tasks, was employed. This baseline utilized TfidfVectorizer for word representation, coupled with hyperparameter tuning to enhance performance. The third model, Multinomial Naïve Bayes, is especially adept at handling discrete integer features, such as text word counts, and it also uses CountVectorizer for word representation generation. Lastly, XLNet—a language model developed to overcome several limitations associated with BERT and preceding autoregressive models—was employed for text classification and other natural language processing tasks. XLNet-large-cased, featuring 24 layers and 340 million parameters, was chosen for its superior performance over BERT on numerous benchmark datasets.

Table 3 Performance Evaluation on the Amazon Reviews

Method	Configuration	Accuracy	Precision	Recall	F1
Decision Tree		0.786	0.769	0.821	0.974
SVM	sigmoid, gamma=1.0	0.872	0.869	0.880	0.874
Multinomial NB	alpha=0.2	0.876	0.863	0.902	0.882
XLNet	XLNet-Large-Cased	0.916	0. 872	0.973	0.920
Proposed		0.926	0.953	0.926	0.933

In the conducted experiments, a cross-validation approach with K=5 was employed to train and evaluate the baseline models. Each fold included 128K samples for training and 32K for model evaluation. The final test phase utilized a separate 40K portion of the dataset. The experimental results on the Amazon dataset, as summarized in Table 3 indicate that the model achieved an accuracy of 92.6% and an F1 score of 93.3%, surpassing all selected baseline models by a significant margin. Notably, this represents an 8% improvement in precision over the previous state-of-the-art XLNet model, which has 340 million parameters, and a 9% enhancement compared to the gradient boosting classifier. Traditional models such as Decision Tree, SVM, and Multinomial Naïve Bayes yielded sub-90% accuracy, which remains a respectable outcome for general-purpose models. Specifically, Multinomial Naïve Bayes recorded an 88% F1 score utilizing chosen word representations. XLNet Large, requiring minimal optimization, demonstrated the most robust performance among baselines, nearly achieving 92% accuracy, which is 4% higher than that of Multinomial Naïve Bayes. Regarding training duration, the proposed model required an average of 50 minutes per epoch when operating on 5,000 dataset rows using a computer equipped with a GeForce RTX 4090 GPU. This training time is less than that required by the XLNet model but exceeds that of the other baseline models considered.

4 Conclusion

This paper explored the fine-tuning of the LLaMA-8B model to enhance sentiment classification capabilities using the Amazon Product Reviews dataset. The approach demonstrated significant accuracy, achieving an impressive accuracy rate of 92.6%, precision of 95.3%, recall of 92.6%, and F1 score of 93.3%. These results highlight the model's effectiveness in capturing nuanced emotional expressions within textual data. The integration of LoRA fine-tuning, VGST, and TextBlob for polarity assessment, alongside innovative prompts such as one-shot learning and chain-of-thought, was critical in overcoming challenges related to data size. The proposed model further exhibited superior performance in predicting BLEU-4 (30.2327) and predicting ROUGE-1 (85.5212), while achieving a processing speed of 6.079 samples per second, demonstrating its efficiency and robustness compared to other models.

This study not only highlights the proficiency of LLaMA-8B in outperforming several competitive models in terms of precision and processing speed but also underscores the

importance of linguistic features in refining sentiment analysis models. For future work, this paper suggests expanding the dataset to cover a broader range of scenarios, exploring model optimization methods for efficiency, and adapting these strategies for multimodal input analysis, thereby paving the path to more empathetic and responsive AI systems.

References

1. A. Lohrasebi, T. Koslowski, Modeling water purification by an aquaporin-inspired graphene-based nano-channel. J. Mol. Model. 25, 280 (2019).
2. J. Rahman, et al., Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review. Nat. Lang. Process. J. (2024): 100059.
3. A. Dubey, et al., The Llama 3 herd of models. arXiv preprint arXiv:2407.21783 (2024).
4. S. Loria, TextBlob documentation. Release 0.15, 2(8), 269 (2018).
5. W. Fan, F. Geerts, X. Jia, Improving data quality: Consistency and accuracy. ACM (2007).
6. T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are one-shot learners. In Proc. 34th Int. Conf. Neural Inf. Process. Syst. (NIPS '20), Curran Associates Inc., Red Hook, NY, USA, Art. 159, 1877–1901 (2020).
7. T. Kojima, S. Gu, M. Reid, Y. Matsuo, Y. Iwasawa, Large language models are zero-shot reasoners. arXiv abs/2205.11916 (2022).
8. J. E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, W. Chen, LoRA: Low-Rank Adaptation of Large Language Models. arXiv abs/2106.09685 (2021).
9. Y.-Y. Song, L. Ying, Decision tree methods: Applications for classification and prediction. Shanghai Arch. Psychiatry 27(2), 130 (2015).
10. H. R. Prince, A. A. Mamun, I. H. Peyal, et al., CSXAI: A lightweight 2D CNN-SVM model for detection and classification of various crop diseases with explainable AI visualization. Front. Plant Sci. (2024), 151412988-1412988.
11. Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, Q. V. Le, XLNet: Generalized autoregressive pretraining for language understanding. In Adv. Neural Inf. Process. Syst., pp. 5753–5763 (2019).