Text Sentiment Analysis for JD.com Based on Machine Learning

Hanyu Wang[a]
Global Sun School of Business and Management, DongHua University, Shanghai, China

Keywords: Text Sentiment Analysis, Long Short-Term Memory, Machine Learning, Natural Language Analysis.

Abstract: One of the most important uses of Natural Language Processing (NLP) is text sentiment analysis. It is the process of processing and classifying textual content that has been infused with subjective attitudes. The final result is the identification of public sentiment patterns toward specific topics or products. To elevate both accuracy and efficiency in sentiment analysis, the research simultaneously assesses the effectiveness of several models, promoting a detailed understanding of their individual benefits and limitations. Notably, the investigation showed that the Long Short-Term Memory (LSTM) model was a strong competitor. The LSTM model demonstrated its effectiveness in sentiment analysis tasks by achieving an excellent accuracy rate of 87.29% during rigorous training and testing with tens of thousands of datasets. This work then uses this model to analyze user reviews for certain digital products on JD.com, providing an example of the usefulness of LSTM in practical settings. This paper highlights the promising potential of LSTM networks in addressing complex sentiment analysis problems and pushes the boundaries of sentiment analysis approaches.

1 INTRODUCTION

In recent years, online shopping has become an integral part of daily life, with JD.com standing out as a major player in China's thriving e-commerce landscape (Araque et al., 2024). The wealth of user comments on JD.com, rich in sentiment expressions, offers valuable insights for businesses seeking to understand consumer preferences and opinions. Comprehending this feedback is crucial for gauging customer satisfaction levels and refining marketing strategies. Traditional methods, such as sentiment dictionaries (Wu et al.,2017) and machine learning algorithms like Naive Bayes, K-Nearest Neighbour (KNN), and Support Vector Machine (SVM), often fail to capture the nuanced semantics embedded in user comments (Bonaccorso, 2018; Shekhawat, 2024; Fangxu & Jianhui, 2024). This necessitates the pursuit of more advanced techniques to accurately analyze consumer sentiment.

This research delves into the utilization of Long Short-Term Memory (LSTM) neural networks for sentiment analysis. As a specialized type of Recurrent Neural Network (RNN), LSTM excels at managing sequential data and capturing long-term dependencies, offering unique advantages for this task. This study aims to benchmark LSTM's performance against KNN and SVM models, utilizing JD.com user comments as a real-world testbed. By conducting rigorous empirical analysis, the paper aims to demonstrate LSTM's ascendancy in sentiment analysis, leveraging its proficiency in sequential data processing to reveal deeper sentiment insights that traditional methods may overlook.

This paper begins by examining the limitations of current sentiment analysis methods, particularly in capturing the intricate nuances of user sentiment. It then introduces the LSTM model and its unique abilities in this domain, emphasizing its proficiency in modeling temporal dependencies and contextual information. The study meticulously details the dataset used, comprising JD.com user comments, along with the preprocessing steps taken to guarantee data quality. It also outlines the experimental setup for comparing LSTM with KNN and SVM models.

The experimental results demonstrate LSTM's superior performance over KNN and SVM in sentiment analysis, with higher accuracy and effectiveness. LSTM's proficiency in capturing long-term dependencies and comprehending contextual nuances within user comments facilitates more precise sentiment classifications. These findings

[a] https://orcid.org/0009-0007-6035-4491

Wang and H.
Text Sentiment Analysis for JD.com Based on Machine Learning.
DOI: 10.5220/0013515400004619
In *Proceedings of the 2nd International Conference on Data Analysis and Machine Learning (DAML 2024)*, pages 257-260
ISBN: 978-989-758-754-2

emphasize LSTM's potential in addressing intricate sentiment challenges within e-commerce environments like JD.com, where user comments exhibit diverse sentiment expressions.

In conclusion, the findings highlight the significance of LSTM in providing deeper insights into customer sentiment for businesses. The paper underscores the need for further exploration, suggesting hybrid models combining LSTM with other machine learning techniques to enhance sentiment analysis capabilities. Overall, this research contributes to advancing sentiment analysis techniques, demonstrating LSTM's potential as a crucial tool for understanding and leveraging customer sentiment in e-commerce.

2 EXPERIMENTAL DATASETS

To ensure the quality of text data, the author used Python's `re` module with regular expressions to preprocess the crawled data. Key steps involved removing @replies, usernames, {%xxx%} tags, and [xx] contents. Additionally, special characters, emojis, and non-Chinese symbols were removed, while exclamation marks and question marks were replaced with appropriate sentiment-conveying words.

Subsequently, the author undertook preprocessing of the text data, removing stopwords- common yet non-substantive words - to enhance relevance. Utilizing the Harbin Institute of Technology's compiled stopword list, these unnecessary words were effectively eliminated from the text dataset.

Finally, the author employed word embedding to represent the text data. Traditional one-hot coding - a bag-of-words model - suffers from limitations such as ignoring word order, assuming word independence, and resulting in discrete, sparse features. To address these issues, the author adopted word embedding, a neural network-based distributed representation approach. This method converts vector elements from integers to floating-point numbers, enabling representation across the entire real number range. It also condenses the original sparse, high-dimensional space into a more compact, lower-dimensional one. Leveraging Python's Keras framework and word2vec, the author constructed a 150-dimensional word vector space encompassing nearly all Chinese vocabulary from the cleaned and tokenized Wikipedia corpus (Tang et al., 2020).

3 METHODS BASED ON MACHINE LEARNING

The text information is first processed for features, and then the model undergoes supervised learning training. The trained model is then used to predict the sentiment polarity of new text information. The working method is as follows: initially, labeled text data is utilized for feature extraction, from which key information is derived. Subsequently, these features are employed to generate sentiment polarity labels, serving as the foundation for model training. Through the machine learning training process, a model capable of recognizing sentiment polarity is constructed. This model can receive new unlabelled sentences, perform feature extraction again, and predict the sentiment polarity based on the trained model, ultimately outputting the prediction results. The entire process, from data preparation to model prediction, achieves efficient and accurate sentiment analysis.

Based on different classification algorithms, methods can be divided into KNN, SVM, Naive Bayes, Maximum Entropy, etc (Chen. S & Chen. J, 2024).

3.1 K-Nearest Neighbour

The K-Nearest Neighbours (KNN) classification algorithm is a simple yet effective method in data mining classification. This algorithm classifies records by examining the labels of the KNN and assigning the most frequent label. Easy to understand and implement, it's sensitive to K-value selection and the distance metric used (Li et al., 2024).

3.2 Support Vector Machine

The Support Vector Machine (SVM) is a generalized linear classifier renowned for binary classification using supervised learning. It identifies the hyperplane with the maximum margin - the distance to the nearest data points from each class - to maximize separation. This approach prevents overfitting, ensuring a robust and accurate classifier (Fang et al., 2024).

3.3 Long Short-Term Memory

Long Short-Term Memory (LSTM) is a kind of RNN specifically crafted to deal with the hardships faced during the training of long sequences. Traditional RNNs frequently encounter problems such as vanishing and exploding gradients, resulting in poor long-term memory preservation. In contrast, LSTM networks possess unique mechanisms that permit them to manage and remember information effectively over long durations. This characteristic makes LSTM a great choice for tasks involving the processing of long data sequences, like sentiment analysis of extensive texts. By resolving the issues related to gradient vanishing and exploding (Staudedfzmeyer & Morris, 2019), LSTM provides more accurate and reliable results, especially when handling complex and large datasets. The application of LSTM technology in various fields has been shown to improve the efficiency and precision of data processing.

The complex design of an LSTM cell enables it to handle long-term dependency issues effectively. Its specialized architecture shows its ability to overcome challenges associated with retaining information over extended periods (Yu & Zhou, 2018).

The first tier is known as the Forget Gate. It is the initial stage of LSTM for determining which data should be removed from the cell state. This decision is determined via a sigmoid network layer called the "forget gate layer". It takes the current input ($\mathrm{X_t}$) and the previous hidden state ($\mathrm{h_{t-1}}$) as inputs, and for each number in the cell state ($\mathrm{C_{t-1}}$), it returns a value between 0 and 1. The number of 1 denotes "accept this fully", whereas a value of 0 denotes "totally ignore this" (Yang & Wang, 2019).

The input gate is the next layer. Its purpose is to ascertain what fresh data will be kept in the cell state. There are two parts to this process. The "input gate layer", a sigmoid layer, determines which values will be updated first. A layer then creates a fresh candidate value vector to be included in the state. Following that, an update for the state is created using these two pieces of data.

Next, there is the third layer, namely the Cell State Update Gate. The paper uses the new cell state ($\mathrm{C_t}$) to replace the old cell state ($\mathrm{C_{t-1}}$). The author multiplies the old state by the output of the forget gate (f_t) to discard the information that has been decided to be forgotten. Subsequently, this paper brings in new candidate values and adjusts them according to the degree of update determined for each state. Finally, the output value is determined by the filtered cell state. A sigmoid layer is utilized to output the specific part of the cell state. Then, it processes the cell state through a tanh function (producing a value from -1 to 1 and multiplies it by the output of the sigmoid gate.

Eventually, output the selected portion. In this way, the state of the hidden layer from the previous moment is integrated into the calculation process of the current moment. In simpler terms, the selection and decision-making take into account the previous state, addressing the long-term dependency issues that regular RNNs encounter.

4 EXPERIMENT RESULTS

Despite its simplicity, the KNN model only achieves a moderate performance in sentiment analysis, with an accuracy of 0.5847. The model's F1 Score of 0.5506, along with balanced precision (0.5558) and recall (0.5513) values, indicate its struggle in accurately distinguishing between positive and negative sentiments. This limitation can be attributed to KNN's sole focus on feature space proximity, overlooking the sequential dependencies present in text data. Therefore, while KNN is user-friendly and straightforward, it falls short of effectively analyzing sentiment due to its inherent design flaws and lack of consideration for textual nuances. The experiment results are shown in Table 1.

The SVM is well-known for its strong generalization capabilities, surpassing KNN in performance. However, when it comes to sentiment analysis, SVM falls short despite achieving an accuracy of 0.6301. This is evident in the imbalance between its precision (0.7979) and recall (0.5345), resulting in an F1 score of 0.4480.

This disparity highlights SVM's cautious approach toward classifying positive samples, prioritizing precision over effectively capturing genuine positives. The model's struggle with recognizing sequential patterns essential for sentiment analysis further accentuates its limitations in this area. In conclusion, while SVM showcases moderate success, its shortcomings in sentiment analysis are apparent. The LSTM model stands out as a superior choice for sentiment analysis tasks due to its exceptional performance across various metrics. In comparison to other models like KNN and SVM,

Table 1: Experimental results.

Model	Accuracy	F1	Recall	Precision
KNN	0.5847	0.5506	0.5513	0.5558
SVM	0.6301	0.4480	0.5345	0.7979
LSTM	0.8729	0.8960	0.8943	0.9061

LSTM excels with an accuracy rate of 87.29%, accurately classifying a high percentage of samples. Additionally, its F1 Score of 0.8960 reflects a harmonious balance between precision and recall, showcasing its proficiency in identifying positive sentiments while minimizing false positives and false negatives.

One of LSTM's key strengths lies in its ability to process sequential data, allowing it to capture nuanced sentiment orientations and tendencies within the text. This unique capability significantly contributes to its high classification accuracy and overall exceptional performance. In contrast, models like KNN and SVM struggle to capture the sequential nature of text data, therefore hindering their effectiveness in sentiment analysis tasks. Ultimately, this study conclusively establishes LSTM's superiority in handling text data with intricate sequential patterns for sentiment analysis. When faced with complex textual data, prioritizing LSTM or similar sequence-processing models is crucial to ensure optimal performance and accuracy. By leveraging LSTM's capability to understand context and dependencies within text sequences, researchers and practitioners can enhance the accuracy and effectiveness of sentiment analysis tasks.

5 CONCLUSIONS

This paper emphasizes the crucial significance of sentiment analysis for understanding customer feedback, especially on e-commerce platforms like JD.com. Through analyzing user reviews of specific digital products, the study compares advanced machine learning techniques (such as LSTM networks) and traditional algorithms (like KNN and SVM). LSTM is highlighted for its remarkable ability to achieve high accuracy in sentiment analysis, especially in handling sequential data and extracting detailed contextual semantic information from long texts. The research evaluates the performance of LSTM, KNN, and SVM in sentiment analysis of JD.com's user reviews. LSTM emerges as the most effective model, showing its value in helping businesses understand customer satisfaction levels and guiding strategic decisions on product quality improvement, customer service optimization, and marketing strategy refinement. However, LSTM models have limitations in handling long sequences. While they are good at processing short sequences, dealing with sequences exceeding 1000 elements poses computational challenges and time constraints due to the complexity of LSTM cells. Future research should focus on optimizing and enhancing LSTM architectures to address these limitations. Possibilities include developing more efficient LSTM variants for long sequences, using parallel processing techniques, and leveraging hardware accelerators. Hybrid approaches combining LSTM with other algorithms also hold promise. In conclusion, integrating LSTM in sentiment analysis of JD.com's user reviews has demonstrated its potential. As research continues, LSTM-based sentiment analysis will be important for driving customer satisfaction, building brand loyalty, and contributing to the success of JD.com and other businesses in the e-commerce field.

REFERENCES

Araque, O., Corcuera-Platas, I., Sánchez-Rada, J. F., Iglesias, C. A., 2017. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. Expert Systems with Applications, 77, 236-246.

Bonaccorso, G., 2018. Machine Learning Algorithms: Popular algorithms for data science and machine learning. Packt Publishing Ltd.

Chen, S., Chen, J., 2024. Research on Sentiment Analysis Model of Online Course Reviews Based on R-Boson. Modern Information Technology, 16, 107-112.

Fangxu, Y., Jianhui, W., 2024. A sentiment recognition model for Weibo comments based on SVM and Word2vec. Modern Computers, 10, 60-64.

Li, Y. W., Chen, Y. X., Hu, G. X., 2024. Recognition and detection of apple leaf diseases based on KNN and multi-feature fusion. Food and Fermentation Technologies, 4(04), 25-32.

Shekhawat, B. S., 2019. Sentiment classification of current public opinion on BREXIT: Naïve Bayes classifier model vs Python’s TextBlob approach (Doctoral dissertation, Dublin, National College of Ireland).

Staudedfzmeyer, R. C., Morris, E. R., 2019. Understanding LSTM--a tutorial into long short-term memory recurrent neural networks. arxiv preprint arxiv:1909.09586.

Wu, J., Lu, K., Su, S., Wang, S., 2019. Chinese micro-blog sentiment analysis based on multiple sentiment dictionaries and semantic rule sets. IEEE Access, 7, 183924-183939.

Yang, Q., Wang, C. W., 2019. Research on global stock index prediction based on deep learning LSTM neural network. Statistical Research, 03, 65-77.

Yu, W., Zhou, W. N., 2018. Sentiment analysis of product reviews based on LSTM. Computer Systems & Applications, 08, 159-163.