Optimal Encoder-Decoder Pairs for Enhanced Sentiment Analysis in Restaurant Reviews

Venkata Vaibhav Parasa
VVP23

Sai Teja Yapuram Ramesh
SY23F

Hasan Angel Bazzi Sabra
HB21H

1. Introduction

In the era of advanced language models such as ChatGPT 3.5 and other large language models (LLMs), our interest lies in exploring their integration into recommendation systems. Notably, LLMs like GPT, unequipped with internet access, often have constraints, requiring purchase, waiting lists, or facing unavailability. Driven by these considerations, our goal is to investigate their potential integration into our existing models.

Initially targeting the IMDb movie dataset, we faced a crucial limitation. The dataset exclusively consisted of movie reviews and binary sentiment labels, making it impractical and inaccurate for validating user-input movies. Consequently, we shifted our focus to the Yelp dataset[19].

We carefully selected four encoders and six decoders, forming pairs to experiment with various combinations. The encoders generate numeric representations (embedding vectors) for textual reviews, which are then used as input for the decoders. Our aim is clear: identify the combination that yields optimal results. We intend to recommend restaurants based on the overall sentiment in user reviews, acknowledging that average star ratings might not fully capture nuanced user sentiments.

2. Literature Survey

Sentiment analysis automatically identifies and extracts the sentiment expressed within the text, serving valuable applications like customer feedback analysis and social media monitoring [16]. Early approaches relied on simple techniques like keyword matching but lacked accuracy and nuance [14].

The emergence of large language models (LLMs) and pre-trained embeddings like Word2Vec revolutionized sentiment analysis [16, 2]. Embeddings provided LLMs with compressed representations of words and their relationships, enabling them to analyze the semantic meaning and context within the text for improved sentiment prediction accuracy [3]. While the specific meaning of an embedding may shift slightly depending on the operation applied, subtraction or addition operations generally preserve the relationships between embedding vectors, allowing them to retain their overall semantic meaning [12].

Contextual encoders like BERT and BART further enhanced LLM capabilities by analyzing the context surrounding each word to capture subtle meaning changes [16, 13]. This led to even more accurate sentiment analysis by allowing LLMs to understand the nuances and complexities of human language [16].

Recent research explores LLM-specific embedding techniques tailored to the architecture and capabilities of each LLM, potentially leading to further improvements in accuracy [4]. Sentiment analysis with embeddings and LLMs has become incredibly effective, enabling accurate sentiment identification in various text data and enhancing applications like customer feedback analysis, social media monitoring, targeted marketing, and fake news detection [16, 4]. This technology continues to evolve, promising even more impactful applications across various domains in the future [16].

3. Methodology

3.1. Data:

The Yelp database we utilized has specific attributes that enabled us to identify, preprocess, and subsequently integrate it to meet our specific requirements. Key attributes include business_id (hashed data type), ratings (numeric scale ranging from 1 to 5), which are the labels, and reviews, all collectively contributing to the derivation of our final results.

3.2. Encoders:

Word2Vec:

This encoder captures semantic relationships between words within a corpus, generating dense vector representations for each word.[11]

BERT:

This encoder utilizes a bidirectional transformer architecture, analyzing each word's context within a sentence to create comprehensive vector representations.[3]

BART:

This encoder leverages a sequence-to-sequence model, translating text input into a hidden representation suitable for various NLP tasks, including sentiment analysis.[13]

T5:

This encoder employs a massive pre-trained transformer model that excels at text summarization and translation, also generating effective vector representations for sentiment analysis tasks.[16]

3.3. Decoders:

Logistic Regression:

This classic decoder applies a logistic function to predict the probability of a specific sentiment category (e.g., positive, negative) based on the encoded input.[19]

Support Vector Machines (SVM):

This decoder implements a hyperplane to separate different sentiment classes in the encoded data space, making predictions based on the class closest to the input vector.[20]

Multi-Layer Perceptron (MLP):

This decoder utilizes a multi-layer perceptron network to learn complex non-linear relationships between the encoded input and the predicted sentiment label.[5]

Convolutional Neural Network (CNN):

This decoder employs convolutional layers to extract patterns and features from the encoded data, subsequently predicting the sentiment category.[1]

Gradient Boosting:

This decoder combines multiple weak learners into a strong ensemble, iteratively improving its accuracy in predicting the sentiment based on the encoded input.[15]

Random Forest:

This decoder leverages a collection of decision trees trained on different subsets of the encoded data, making predictions based on the majority vote of these trees.[9]

4. Implementation

4.1. Data Extraction and Preprocessing:

Analyzing the labels (star ratings) of the Yelp database, we encountered a significant data imbalance, as can be seen in figure one. This imbalance skewed the data towards positive ratings, posing a threat to the accuracy and fairness of our machine-learning model. To mitigate this issue, we implemented a balanced dataset approach, sampling an equal number of reviews from each rating class, ensuring the model receives fair exposure to all sentiment categories, reducing bias, and enhancing its capacity to generalize to unseen data [21]. This balanced dataset, a concept further explored by [18, 6], creates a foundation for a more robust and reliable sentiment analysis model capable of providing richer and more nuanced insights into the diverse perspectives expressed within the Yelp reviews.

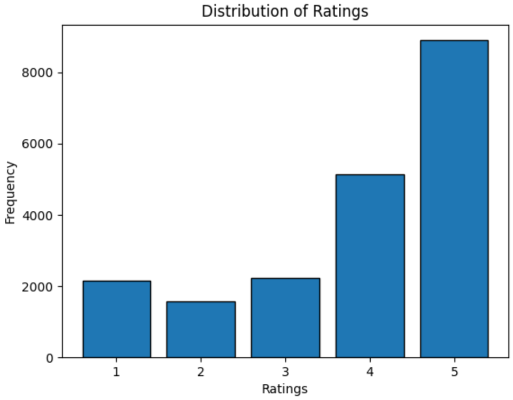

Fig 1 : Frequency of classes before preprocessing for YELP data set.

4.2. Embeddings Extraction:

We use the power of pre-trained embedding models like BERT, BART, and Word2Vec to represent Yelp review text as dense vector representations. This captures the semantic meaning and relationships within the reviews, enabling efficient and insightful analysis [14, 8].

For each model, we implement custom embedding extraction functions to ensure accurate conversion of text to vector representations [10]. These embeddings are saved as pickle files for convenient access and future use in the sentiment analysis pipeline.

This approach empowers us with robust and informative sentiment analysis by leveraging the strengths of pre-trained embeddings to unlock the hidden meaning within the vast collection of Yelp reviews [7].

4.3. Model Training and Evaluation:

Machine learning models, including Support Vector Machines (SVM) [20], Logistic Regression [19], CNN [1], and others, are trained and evaluated using the processed embeddings. These models learn to predict the sentiment of a review based on its underlying features.

The training process involves splitting the data into training and testing sets and iteratively adjusting the model parameters to minimize the prediction error [18]. Evaluation metrics such as accuracy, precision, recall, and F1-score [18] are recorded and analyzed to assess the performance of each model [8]. Thus identifying the optimal encoder-decoder pair.

4.4. Optimal Model Inference:

After selecting the optimal encoder-decoder pair, we utilize it to predict whether a user should visit a particular restaurant. The process involves averaging all the embeddings associated with the user-entered matching IDs. Subsequently, these averaged embeddings are passed to the decoder model. The decoder, in turn, generates a prediction on a scale from 1 to 5, where 1 suggests avoiding the restaurant and 5 indicates strong encouragement to visit

5. Experiments & Results

5.1. Results:

Encoders	Logistic Regression	SVM	MLP	CNN	Gradient Boosting	Random Forest
Word2Vec	0.524	0.483	0.498	0.426	0.421	0.398
BERT	0.569	0.582	0.549	0.384	0.5	0.488
BART	0.583	0.612	0.561	0.43	0.514	0.51
T5	0.554	0.41	0.581	0.255	0.455	0.427

Table 1. : Accuracy Table displaying accuracy scores for all combinations of encoder and decoder pairs.

Embeddings	Logistic Regression	SVM	MLP	CNN	Gradient Boosting	Random Forest
Word2Vec	0.525085667	0.50116	0.50574	0.39382	0.420763272	0.389913215
BERT	0.576110078	0.59401	0.57443	0.45286	0.500476504	0.470976518
BART	0.591872183	0.62207	0.56612	0.43672	0.519005419	0.505947995
T5	0.564081414	0.43734	0.59017	0.24745	0.471058075	0.422051699

Table 2. : Precision Table displaying precision scores for all combinations of encoder and decoder pairs.

Embeddings	Logistic Regression	SVM	MLP	CNN	Gradient Boosting	Random Forest
Word2Vec	0.524	0.483	0.498	0.426	0.421	0.398
BERT	0.569	0.582	0.549	0.384	0.5	0.488
BART	0.583	0.612	0.561	0.43	0.514	0.51
T5	0.554	0.41	0.581	0.255	0.455	0.427

Table 3. : Recall Table displaying recall scores for all encoder and decoder pair combinations.

Embeddings	Logistic Regression	SVM	MLP	CNN	Gradient Boosting	Random Forest
Word2Vec	0.524039918	0.4844	0.4986	0.3802	0.419294734	0.381832057
BERT	0.571850623	0.5859	0.5544	0.3850	0.499815498	0.469034388
BART	0.585122163	0.6153	0.5630	0.4128	0.51388516	0.497817705
T5	0.556153553	0.4160	0.5751	0.2398	0.456066595	0.410301019

Table 4.: F-1 score Table displaying the F-1 scores for each combination of encoder and decoder pairs

Class	Precision	Recall	F1-Score	Support
1	0.721393	0.732323	0.726817	198
2	0.5888325	0.574257	0.5814536	202
3	0.5602094	0.554404	0.5572917	193
4	0.4396135	0.532164	0.4814815	171
5	0.75	0.648305	0.6954545	236

Table 5. : Classification report of SVM classifier with BART embeddings

5.2 Analysis:

Accuracy Performance:

Looking at table-1, BART consistently outperforms, achieving 61.2% accuracy with SVM, highlighting its efficacy in capturing sentiments. SVM exhibits resilience across encoders, whereas Word2Vec performs moderately, trailing BART and SVM. T5 demonstrates variable accuracy, underscoring the encoder's pivotal role in sentiment analysis accuracy. BART and SVM emerge as standout options, offering valuable insights for optimization.

Precision Analysis:

Table-2 rows represent different embeddings (Word2Vec, BERT, BART, and T5), and columns denote classifiers (Logistic Regression, SVM, MLP, CNN, Gradient Boosting, and Random Forest). BART consistently excels, attaining a peak precision of 0.622 with the SVM classifier, showcasing its effectiveness in identifying positive sentiments. SVM maintains robust precision across embeddings, emphasizing its reliability. While Word2Vec exhibits decent precision, it lags behind BART and SVM, highlighting the impactful role of embedding choice in precision.

Recall Analysis:

Table-3 shows us that BART consistently excels, especially with an SVM classifier, achieving a recall of 0.612. SVM maintains commendable recall across embeddings, showcasing reliability in identifying positive sentiments. Word2Vec exhibits decent recall but falls behind BART and SVM. This emphasizes the pivotal role of embedding choice in sentiment analysis, with BART and SVM proving a potent combination for adeptly capturing positive sentiments, offering insights for optimization.

F1-Score Analysis:

According to Table-4, BART consistently excels, especially with SVM, showcasing effectiveness in balancing precision and recall. SVM proves reliable, delivering balanced F1 scores across embeddings. While Word2Vec performs decently, it lags behind BART and SVM's consistent performance. T5 exhibits variable F1-scores, suggesting sensitivity to the classifier choice. This underscores the crucial role of embeddings in sentiment analysis, with BART and SVM standing out as potent options and providing insights for optimization in specific tasks.

Classification Analysis of SVM classifier with BART embeddings:

The analysis of the SVM classifier with BART embeddings reveals commendable performance across various classes. Class 1 demonstrates a high precision of 72.14%, indicating a significant proportion of correctly identified positive instances out of the total predicted positive instances. The recall for class 1 is 73.23%, suggesting the model effectively captures the majority of actual positive instances for this class. The corresponding F1-score of 72.68% reflects a balanced measure of precision and recall. Similar trends are observed across classes, with class 5 showing particularly strong precision (75%) and a balanced F1-score of 69.55%. However, class 4 exhibits a lower precision of 43.96%, indicating a notable proportion of false positives, impacting the overall F1-score. Overall, the SVM classifier with BART embeddings demonstrates robust performance, particularly in effectively identifying positive instances, with variations across different classes warranting attention for further optimization.

6. Conclusion

In summary, our exploration of encoder-decoder pairs reveals the SVM classifier coupled with BART embeddings as a consistent frontrunner, showcasing exceptional recall performance in discerning positive sentiments. BART embeddings, distinguished for their robust semantic representation capabilities, adeptly address subtleties in user opinions. Beyond the commendable recall score, the SVM-BART synergy demonstrates adaptability to new data and is a pivotal attribute for recommendation systems requiring timely updates.

In the future, our focus should focus on refining performance metrics, emphasizing precision and sensitivity enhancements through meticulous model adjustments, and integrating additional contextual features. Subsequent evaluations across diverse datasets and application domains will provide nuanced insights into the strengths and limitations of the SVM-BART combination.
In conclusion, this study, conducted within the context of sentiment analysis for recommendation systems, underscores the distinct competitive advantage of the SVM classifier with BART embeddings. The robust performance signals present efficacy and lays a solid foundation for future optimizations, ensuring this combination's enduring applicability and effectiveness in various real-world scenarios within the domain of sentiment analysis and recommendation systems.

7. References

1. Ali, Jehad & Khan, Rehanullah & Ahmad, Nasir & Maqsood, Imran. (2012). Random Forests and Decision Trees. International Journal of Computer Science Issues(IJCSI). 9.
2. Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data 8, 53 (2021). https://doi.org/10.1186/s40537-021-00444-8
3. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
4. Evgeniou, Theodoros & Pontil, Massimiliano. (2001). Support Vector Machines: Theory and Applications. 2049. 249-257. 10.1007/3-540-44673-7_12.
5. G. Singh and M. Sachan, "Multi-layer perceptron (MLP) neural network technique for offline handwritten Gurmukhi character recognition," 2014 IEEE International Conference on Computational Intelligence and Computing Research, Coimbatore, India, 2014, pp. 1-5, doi: 10.1109/ICCIC.2014.7238334.
6. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T., Bagheri, M., & Castro, M. (2017). FastText.zip: Compressing text classification models. arXiv preprint arXiv:1612.03651.
7. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension.
8. Liu, B. (2012). Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies, 5(1), 1–167. doi:10.2200/s00416ed1v01y201204hlt016
9. Liu, Y. (2019). Learning to compare: Relation extraction as a ranking problem. arXiv preprint arXiv:1909.03206.
10. Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (Vol. 1, pp. 142–150). doi:10.3115/1229956.1229992
11. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. Google Inc., Mountain View, CA.
12. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).
13. Natekin, Alexey & Knoll, Alois. (2013). Gradient Boosting Machines, A Tutorial. Frontiers in neurorobotics. 7. 21. 10.3389/fnbot.2013.00021.
14. Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval, 2(1–2), 1–135. doi:10.1561/1500000011
15. Peng, Joanne & Lee, Kuk & Ingersoll, Gary. (2002). An Introduction to Logistic Regression Analysis and Reporting. Journal of Educational Research - J EDUC RES. 96. 3-14. 10.1080/00220670209598786.
16. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.
17. Tang, D., Qin, B., & Liu, T. (2015). Document Modeling with Gated Recurrent Neural Network for Sentiment Classification. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1422–1432). doi:10.18653/v1/d15-1167
18. Wang, W. Y. (2017). "Why should I trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144).

19. Yelp, Inc. (2022, March 17). Yelp dataset. Kaggle. https://www.kaggle.com/datasets/yelp-dataset/yelp-dataset
20. Zhao, Wayne Xin, et al. “A Survey of Large Language Models.” arXiv.Org, 24 Nov. 2023, arxiv.org/abs/2303.18223.
21. Zhang, Y., Wallace, B. C., & Feng, S. (2019). A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification. arXiv preprint arXiv:1510.03820.