# Classifying Ecommerce Product Reviews with Deep Learning

**Naveen Sukumar, Ramakrishna Ramadurgam**

Natural Language Processing - W266, University of California, Berkeley

{naveensukumar@berkeley.edu, ramadurgan@berkeley.edu}

## Abstract

Our research study explores the effectiveness of novel deep learning architectures in predicting the rating of Amazon product reviews on a scale of 1 to 5. In this paper, we present two approaches, a CNN with multi-head attention model and a CNN - bidirectional LSTM hybrid model, that are able to capture the semantic and syntactic features of reviews. We trained and evaluated our models on two different datasets: reviews from the electronics category and reviews from the clothing, shoes, and jewelry category. Throughout our study we ran different experiments to test our novel architectures on a variety of conditions. We try addressing class imbalance with data balancing techniques and test the generalizability of our models on unseen product categories. Our results demonstrate that our models outperform traditional baseline models and achieve state-of-the-art performance on a variety of metrics.

## 1 Introduction

With competitive prices and efficient delivery, e-commerce is a dominating force in today's society that has revolutionized the consumer experience. Online product reviews are a vital source of information for sellers and consumers as they provide feedback on the quality, features, and utility of a product. Analyzing product reviews helps customers make informed decisions and helps sellers improve their products. Developing methods to automate the analysis of product reviews is a valuable yet challenging task.

A key aspect of product reviews is the rating, which is a score on a scale of 1 to 5 that reflects the overall opinion of the reviewer. Ratings are one of the most important pillars for ecommerce as they influence customers' perspectives as well as the ranking and visibility of products. However, ratings alone do not capture the full sentiment and reasoning behind a review. It would be beneficial to have a system that can predict the rating of a review based on its actual content.

Predicting ratings is difficult for many reasons. The content of reviews can drastically vary in length, style, tone, and vocabulary. Different types of information can also have different effects on the rating. Lastly, the standards and criteria for a rating can differ between individuals. Building a model that can take all these nuances into account is quite a challenge.

## 2. Background and Motivation

Many researchers have explored using models such as Support Vector Machines (Liu et al. 2007) and Radial Basis Networks (Gupta et al. 2014) for tasks such as predicting the helpfulness or validity of a

review (Liu et al. 2007). Others focused their efforts on using Transformer-based models such as BERT (Devlin et al. 2019) for summarizing reviews (Yuan et al. 2020). All of these models have shown state of the art results.

We take a different approach in our study by predicting the rating from the content of the review itself. This allows us to gain a deeper understanding of the nuances and sentiment in text compared to binary classification or summarization models. We accomplish this by implementing two models. The first is a CNN with a Multi-Head Attention Mechanism. CNNs (Kim 2014) have been proven to perform very well for NLP tasks and do an excellent job of extracting local features. Multi-head attention (Voita et al. 2019) can focus on different inputs simultaneously and has proven to do well at modeling long-range dependencies. The second model is a CNN and Bi-directional LSTM hybrid model. Bi-directional LSTMs (He et al. 2017) have shown to also do well in modeling long-range dependencies because of their gating mechanism and ability to process input from forward and backward directions. Please reference Figures 4 and 5 in the appendix which contain diagrams of their architectures.

## 3. Dataset

Jianmo Ni, a former PhD student at UC San Diego, and his research team pulled and compiled a database of Amazon product reviews (Ni 2018). In this database are 5-core datasets of Amazon product reviews by product category. From these categories we randomly selected to train our models on the Electronics dataset, which contains 6,379,590 reviews, and the Clothing, Shoes and Jewelry dataset, which contains 11,285,464 reviews.

Before using these datasets, we performed a series of data preprocessing steps. These steps included removing all but the columns pertaining to the review and its rating, deleting duplicate and empty rows, stripping html tags and stop words, and deleting reviews that are less than a word long.

## 4. Exploratory Data Analysis

**Figure 1: Distribution of Ratings**

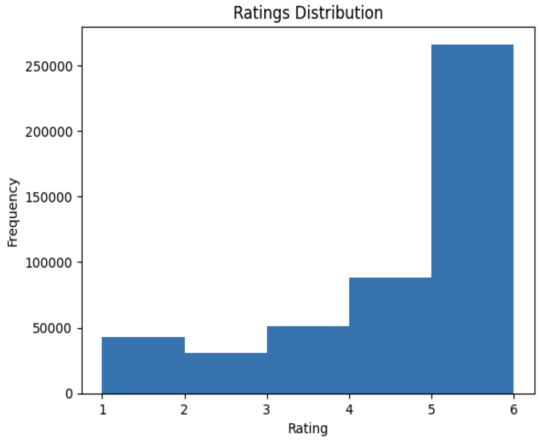
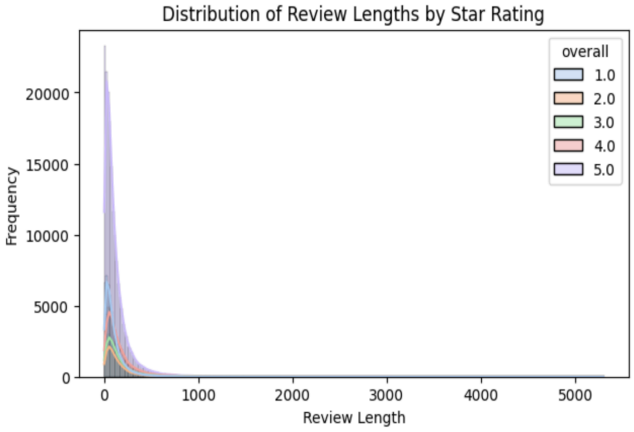



**Figure 2: Distribution of Review Lengths**




To better understand our dataset, we performed an exploratory data analysis. Figure 1 is a histogram of the counts of reviews by ratings for the electronics dataset. As can be seen, there is a heavy class

imbalance where there are significantly more 5 star reviews than 1 star, 2 star, 3 star, and 4 star reviews. This poses a challenge for multi-class classification as the model may learn to perform well for 5 star reviews, but poorly for the other ratings. We noticed the same distribution occurring for the Clothing, Shoes, and Jewelry dataset.

Figure 2 is a histogram of the lengths of reviews by star ratings for the Clothing, Shoes, and Jewelry data set. This distribution is very similar to the distribution for the Electronics dataset and has an average length of 132 characters. This shows that it is important to have models that can capture short and long range dependencies.

We also analyzed the top 10 words that appear in reviews by rating. Please refer to Table 1 and Table 2 below, which contain the top 5 words for 5 star and 1 star reviews in the Electronics dataset. As can be seen, 5 star reviews tend to contain more words indicating positive sentiment such as “great”, “like”, “good”, and many more in comparison to 1 star reviews. These are trends that are also present in the other dataset that we expect our models to capture to perform well.

**Table 1: Top 5 words for 5 star ratings**

| Word | Counts |
|---|---|
| great | 82663 |
| one | 82347 |
| use | 71674 |
| like | 61901 |
| good | 57412 |

**Table 2: Top 5 words for 1 star ratings**

| Word | Counts |
|---|---|
| one | 24131 |
| would | 19233 |
| work | 17469 |
| get | 17176 |
| product | 13979 |

## 5. Models and Methodology

### 5.1 Random Forest Classifier

For our baseline model we decided to implement a random forest classifier. A random forest classifier is an ensemble method that makes a classification based on a combination of different decision trees (Kontonatsios et al. 2013). Given that 5 star reviews tend to have more positive words, we used a count vectorizer from Scikit-learn to convert the input into a vector of counts that we can use to train the model.

### 5.2 CNN with Multihead Attention

The first advanced model that we implemented was a convolutional neural network with a mult-ihead attention layer. We instantiated our embedding layer with pretrained Word2Vec embeddings (Mikolov et al. 2013). We decided to use them because models with Word2Vec embeddings have been shown to consistently perform better than models without pre-trained embeddings. Before training the model, we used a tokenizer to convert the text into tokens, and then mapped tokens to ids that the model can use to represent the input they receive.

### 5.3 CNN with Bi-Directional LSTM

The second model that we implemented was a convolutional neural network followed by a Bi-directional LSTM. This model uses the same embedding layer and tokenization process as the first advanced model.

## 6. Model Evaluation

**Table 3: Baseline and Advanced Model Results**

| Model | Dataset | Precision | Recall | F1-Score | Accuracy |
|---|---|---|---|---|---|
| Random Forest Classifier | Electronics | 0.62 | 0.53 | 0.36 | 0.5265 |
| Random Forest Classifier | Clothing, Shoes and Jewelry | 0.74 | 0.56 | 0.4 | 0.5558 |
| CNN - Attention | Electronics | 0.58 | 0.66 | 0.6 | 0.6579 |
| CNN - Attention | Clothing, Shoes and Jewelry | 0.64 | 0.67 | 0.63 | 0.671 |
| CNN - Bi LSTM | Electronics | 0.62 | 0.67 | 0.62 | 0.6736 |
| CNN - Bi LSTM | Clothing, Shoes and Jewelry | 0.64 | 0.67 | 0.64 | 0.6656 |

### 6.1 Metrics

Due to the heavy class imbalance, we decided to use weighted average precision, weighted average recall, and weighted average F1-score (Harbecke et al. 2022) as our primary metrics. Weighted metrics use a weight for each class based on its size to compute a score that takes the class imbalance into account.

### 6.2 Baseline models

We can see above that the baseline model performs very poorly with a low F1-score. The heavy class imbalance is most likely causing the model to overfit on the 5 star reviews and underperform on the other ratings, causing the F1-score to be low despite the precision and recall being higher. Using a count vectorizer on the inputs can also create sparse vectors, which random forest models handle very poorly.

### 6.3 Advanced models

The CNN - attention model and CNN - bidirectional LSTM models both perform much better than the baseline models across all metrics. Both of their weighted F1-scores are at least 0.2 points higher than the baseline models for both data sets. This conveys that the models have learned patterns that help them perform significantly better on the minority classes in comparison to the baseline models. Tuning hyperparameters such as learning rate, convolutional filter size, convolutional kernel size, L2 regularization, LSTM units size, and hidden dimension layer size also played a role in their success.

Between the advanced models, their performances are very similar. This can be ascribed to their architectures serving the same purpose. The multihead attention in the first model and the bidirectional LSTM in the second model both aid in widening the context window that their models can learn trends

from. Both models performed relatively similarly across the datasets as well. This indicates that reviews from both product categories have similar trends or patterns that the models learned.

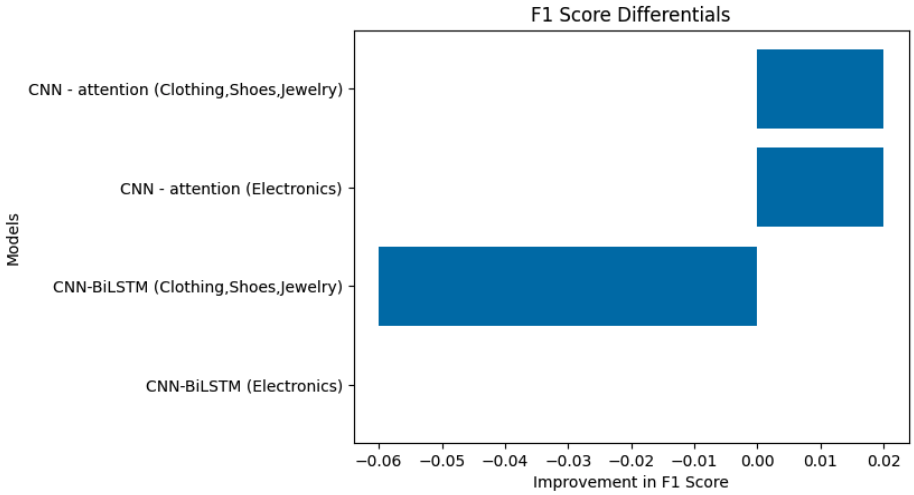
## 7. Experiment 1: Balancing Datasets

Improving the models' performance however from this point is difficult. The heavy class imbalance poses data quality issues that any model would find difficulty handling. In an effort to overcome this hurdle, we tried data augmentation techniques to balance the dataset and introduce variations that the model can learn to make more accurate predictions for minority classes. Our main approach was to undersample the 5 star reviews to about 85% of its current size and over sample the other reviews to about 150% - 200% of their current size.

**Table 4: Advanced Models Results on Balanced Dataset**

| Model | Dataset (Balanced) | Precision | Recall | F1-Score | Accuracy |
|---|---|---|---|---|---|
| CNN - Attention | Electronics | 0.63 | 0.62 | 0.62 | 0.6311 |
| CNN - Attention | Clothing, Shoes and Jewelry | 0.66 | 0.64 | 0.65 | 0.6396 |
| CNN - Bi LSTM | Electronics | 0.62 | 0.62 | 0.62 | 0.6115 |
| CNN - Bi LSTM | Clothing, Shoes and Jewelry | 0.61 | 0.65 | 0.58 | 0.6497 |

**Figure 3: F1 Score Differentials between Balanced and Standard Datasets**




Unfortunately this experiment brings inconclusive results. The F1-scores for the CNN with multihead attention have improved for both datasets by 0.02 points while the F1-score for the CNN - BiLSTM stayed the same on the electronics dataset and decreased by 0.06 points on the other dataset. This may be due to over-sampling re-introducing reviews the model has already seen, causing it to not learn any new valuable patterns. While this indicates that better data balancing techniques than oversampling have a better chance of boosting the performance, re-sourcing the dataset in a way that balances ratings is one of the best remedies for extremely heavy class imbalance.

## 8. Experiment 2: Stress - Testing Generalizability

Seeing how models across datasets perform similarly, we wanted to test if our models were robust enough to accurately predict ratings of reviews from categories it has never seen. We chose two random categories, Automotive and Pet Supplies, and tested our best performing CNN - BiLSTM hybrid model and CNN with multihead attention model on these cleaned datasets to see how they generalize.

**Table 5: Testing Advanced Models on New Datasets**

| Model | Source Dataset | Testing Dataset | Precision | Recall | F1-Score |
|---|---|---|---|---|---|
| CNN - BiLSTM | Electronics | Automotive | 0.66 | 0.72 | 0.67 |
| CNN - BiLSTM | Electronics | Pet Supplies | 0.64 | 0.69 | 0.63 |
| CNN - attention | Clothing, Shoes, Jewelry (Balanced) | Automotive | 0.66 | 0.67 | 0.65 |
| CNN - attention | Clothing, Shoes, Jewelry (Balanced) | Pet Supplies | 0.59 | 0.64 | 0.59 |

We can see above that all models are actually generalizing well to the new datasets as their F1-scores are within the 0.59 to 0.68 range just like the F1-scores in the previous model evaluations. This shows that patterns and features that the model found for the Electronics dataset and Clothing, Shoes and Jewelry dataset can be used to make predictions on unseen product categories. More importantly, it proposes the possibility of training a model on amazon product reviews and utilizing it for related transfer-learning tasks.

## 9. Limitations

One of the biggest limitations in our study was the heavy class imbalance. As demonstrated before, we tried data augmentation techniques but none seemed to improve precision, recall, and F1-score by much. Another limitation was a lack of access to compute resources. We wanted to fine-tune large models such as DistillBERT, XL-Net, and RoBERTa, but were unable to do so due to the large amount of compute resources it would take to train them for even one epoch.

## 10. Conclusion

We presented two novel-architectures for product review classification, a CNN with multihead attention, and a CNN - bidirectional LSTM hybrid model. These advanced models perform better than the baseline models with an F1-score at least 0.2 points higher. A challenge that we noted earlier on was the heavy class imbalance causing our advanced models to perform as well as we would like them to. The balanced datasets did not improve the models' performances however, conveying the need for a better-sourced dataset. Finally, we stress-tested the robustness of our models by making them predict reviews of product categories they have never seen before. The models' performed just as well showing how generalizable one dataset is to the rest of the product categories. Looking ahead we look forward to the opportunity of enhancing our approach. Our future plans involve sourcing a more diverse dataset and exploring larger models like BERT to boost performance.

## Acknowledgements

We thank Amit Bhattacharyya, Mark Butler, Gurdit Chahal, and the rest of the 266 course staff for their guidance and advice throughout the course of our project.

## References

David Harbecke, Yuxuan Chen, Leonhard Hennig, and Christoph Alt. 2022. Why only Micro-F1? Class Weighting of Measures for Relation Classification. In *Proceedings of NLP Power! The First Workshop on Efficient Benchmarking in NLP*, pages 32–41, Dublin, Ireland. Association for Computational Linguistics.

Dhruv Gupta and Asif Ekbal. 2014. Determing Trustworthiness in E-Commerce Customer Reviews. In *Proceedings of the 11th International Conference on Natural Language Processing*, pages 196–205, Goa, India. NLP Association of India.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Georgios Kontonatsios, Ioannis Korkontzelos, Sophia Ananiadou, and Jun'ichi Tsujii. 2013. Using a Random Forest Classifier to recognise translations of biomedical terms across languages. In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora*, pages 95–104, Sofia, Bulgaria. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jianmo Ni. 2018. Amazon Review Data. *University of California, San Diego.* San Diego, California

Jingjing Liu, Yunbo Cao, Chin-Yew Lin, Yalou Huang, and Ming Zhou. 2007. Low-Quality Product Review Detection in Opinion Summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 334–342, Prague, Czech Republic. Association for Computational Linguistics.

Peng Yuan, Haoran Li, Song Xu, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2020. On the Faithfulness for E-commerce Product Summarization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5712–5717, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tomas Mikolov, et al. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* (2013).

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Yuanye He, Liang-Chih Yu, K. Robert Lai, and Weiyi Liu. 2017. YZU-NLP at EmoInt-2017: Determining Emotion Intensity Using a Bi-directional LSTM-CNN Model. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 238–242, Copenhagen, Denmark. Association for Computational Linguistics.

## Appendix

**Figure 4: CNN with Multi-head Attention Model Architecture**

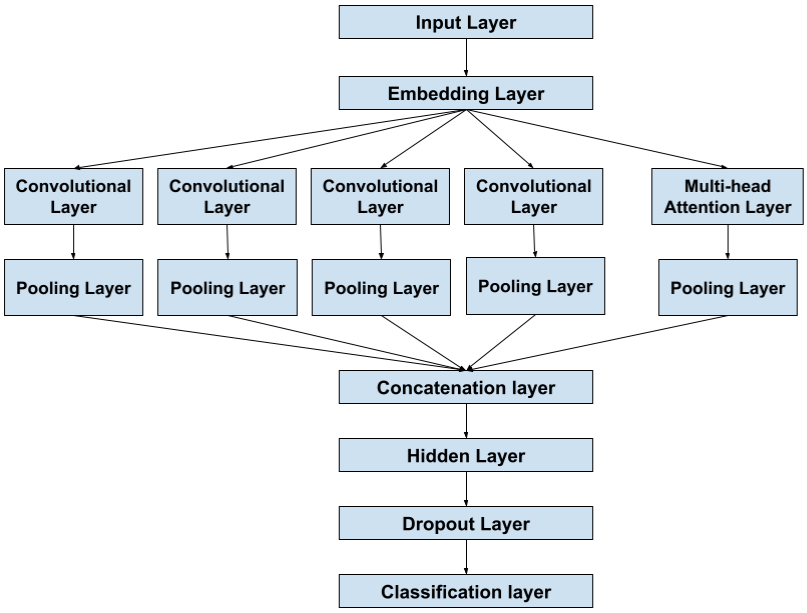



**Figure 5: CNN and Bi-directional LSTM hybrid Model Architecture**