What insights can CheckList provide about the cross-domain robustness of sentiment classifiers?

Yasmine Benmessaoud, `yabe@itu.dk`
Cæcilie Abildgaard Jeppesen, `cjep@itu.dk`
Luke O'Neill, `luon@itu.dk`

Abstract

Measuring accuracy on held-out data for evaluating the performance of NLP models doesn't provide enough insight into how robust and sensitive to systematic biases they are. CheckList is a set of evaluation tools proposed to provide more insight into NLP models than traditional metrics. In order to asses this claim, we performed two Named Entity Recognition related tests from CheckList on a State-of-the-art sentimental analysis model. We also tested whether cross-domain accuracy could be explained by CheckList's insights. Our findings give credibility to CheckList, whilst also discussing it's shortcomings. To replicate our testing results, utilise our repository.[1]

1 Introduction

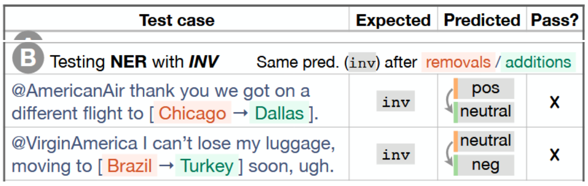
We live in an increasingly digital age, with an ever increasing amount of online communication. Providing some robust method for analysing the sentiment of text could provide solutions to the challenges faced in online spaces such as bullying (Tika Dewi Amelia and Rania Balqis, 2023) as well as some more commercial uses.

Machine learning models, that analyse the sentiment of text, can be used as a proxy for human emotional assessment. (Kenyon-Dean et al., 2018) These can then be used to efficiently find text in large corpora that contain particular sentiment.

Historically, when testing an NLP model's success, some straightforward analyses are performed, commonly an accuracy measure. In categorical sentiment analyses, this is occasionally taken a step further, with an F-test to compare category accuracy variance. Simple methods such as these can lead to overestimation or misunderstanding of an NLP model's performance. (Ribeiro et al., 2020)

CheckList is proposed as a method to analyse the nuance of how an NLP model is succeeding or failing, allowing for an analysis that better informs the directions for improvement. A section of the tests suggested by CheckList relate to Named Entity Recognition (NER).

Generally, humans should have no preconceived sentiment attached to an unknown Named Entity (NE). In this way, we do not want a sentiment analysis model to attribute sentiment categorisation to NEs. When reading text, the linguistic indicators of sentiment should come from adverbs, adjectives and participles. With this understanding, CheckList proposes Invariance (INV) as one of the tests to assess the performance of an NLP model. If a NE is replaced in text with another in such a way that has no syntactic change, does the predicted sentiment of the text change? Examples of how this type of testing works is shown in figure 1.



Test case	Expected	Predicted	Pass?
B Testing **NER** with ***INV*** Same pred. (`inv`) after removals / additions			
@AmericanAir thank you we got on a different flight to [Chicago → Dallas].	`inv`	pos neutral	X
@VirginAmerica I can't lose my luggage, moving to [Brazil → Turkey] soon, ugh.	`inv`	neutral neg	X

Figure 1: INV test examples from the CheckList Evaluation paper (Ribeiro et al., 2020)

Conversely, the test of Directional Expectation (DIR), focuses on the ability of adverbs, adjectives and participles to affect a sentiment model's certainty of prediction. (Ribeiro et al., 2020) This gets measured by the change in the sentiment model's prediction confidence after additional text, with an expected prediction shift, is appended to the review.

In short, these two tests, INV and DIR, will respectively look at how a sentiment analysis model responds to changes in NEs, and how to changes in language with sentiment indication, whilst NEs remains unchanged. In this way we can provide analysis of how NEs affect the outcome of a sentiment analysis model.

Furthermore, we aim to study the ability of a sen-

[1] https://github.com/LJONeill/NLP.git

timent model to correctly predict sentiment from a domain of which it wasn't trained. This was in the hopes that we could simulate the real world challenge of a sentiment model being used on new, unseen data. In this way we could counteract the bias created by using only a held-out dataset, but also assess the level of bias contained in our sentiment model by comparing in and out of domain tests. Our in-domain is movie reviews from the IMDB dataset (Maas et al., 2011) whilst out-of-domain is a Yelp reviews dataset (Anonymous, 2023)

We aim to research the benefit that CheckList can provide in the analysis of sentiment classifiers across domains. We do this using the NER subset of CheckList tests alongside traditional metrics.

2 Experimental setup

2.1 Data setup

We use the IMDB movie reviews dataset, reviews of movies and TV shows, categorised as positive or negative sentiment.

The original creators of this dataset used the star ratings to generate their sentiment labels. Star ratings on IMDB are given on an integer scale from 1 to 10. A negative label was assigned to star ratings of 4 or less, and a positive label was assigned to star ratings of 7 or more. So although it has been argued that the dataset is highly polarised, we felt that the range of polarity within categories would be well suited for studying the effect of DIR perturbation, whilst also being effective at INV testing. (Maas et al., 2011)

Another indicator that led us to utilise this dataset was that it had previously been successfully used to train a similar sentiment analysis model (Jia et al., 2019).

Generally both sentiment categories have similar distributions of word counts as seen in figure 2

Data cleaning was mostly motivated by looking at word frequency and searching for anomalous characters. The most common issue being html encodings, specifically line breaks being retained as strings, and so needing removal. Furthermore, we performed reg-ex based identification of punctuation. html tags, urls, and common emojis and then removed these items. (Al Sharou et al., 2021) The last cleaning method we performed was identification and removal of duplicates.

The dataset initially contained 50,000 reviews, with 25,000 reviews in both sentiment categories. After processing, the dataset consists of 24,884 positive reviews and 24,697 negative ones.

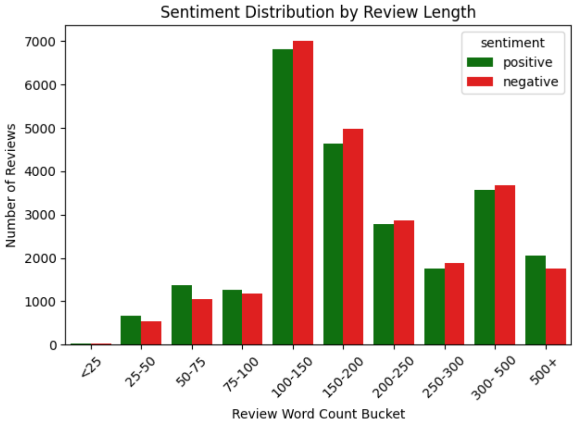


Figure 2: Review length distributions by sentiment category

Additional figures from the EDA process are available in the Appendix.

2.2 Named Entity Recognition Model Setup

Prior to the CheckList evaluation testing methods, we need a deeper analysis of the information within our IMDB dataset. Specifically we needed to perform Named Entity Recognition (NER) on the words within the reviews' text.

For this task we selected the roberta-base-squad2 NLP model, seeing as it's recognised as being a state-of-the-art NLP model (Liu et al., 2019). This would help us to generate the most accurate NER. We trained this NLP model on data from the English Word Treebank (Plank, 2021), and upon an unseen data test, it scored accuracy of 84.64%.

2.3 Creating Test Data

In order to perform the CheckList evaluation tests on our in-domain data we chose to hold-out 10% of the reviews. We also maintained balance between sentiment categories in this test-split. Our intention was to maximise the amount of training data fed to the sentiment analysis model.

Using following perturbation methods described by CheckList, we then generated three testing sets from the 10% held-out:

- An original unperturbed test set,
- an INV set,
- a DIR set.

2.3.1 Invariance test set

Having performed NER on our review data, we systematically isolated and replaced the named entities with ones from two json libraries of tagged Named Entities. One for organisations (ORG)[2], and another for both persons (PER) and locations (LOC).[3]

This was done within categories, meaning that a PER tagged named entity would be replace with a PER named entity, and likewise for ORG and LOC entities. For PER entities created from more than one token were also ensured to be replaced with randomly generated entities of equivalent length so that potential random differences according to review length would not be introduced. An example of INV perturbation for a PER is shown in table 3.

2.3.2 Directional Expectation test set

In order to perform directional expectation, we created 35 positive and 33 negative sentiment short sentences without named entities. According to the sentiment model's prediction of the sentiment label, we appended an equivalent sentiment from our list of short sentences to the review. This was done in line with the ground truth of the review's sentiment. An example of this being performed for a negative sentiment is shown in table 4.

3 Sentiment Analysis Model

For our sentiment analysis model, we chose the most up to date versionn of bert-base-uncased. This choice was informed based on research into testing of various large language models, and finding that an earlier version of bert-base-uncased performed well (Devlin et al., 2019). Our primary intention was to find an NLP model with claims to being state-of-the-art, with the intention to used the CheckList evaluation methods to verify this claim. The bert-base-uncased NLP model fulfilled this criterion.

3.1 Sentiment Model training

After training three versions of our sentiment model, using 10%, 50% and 90% of the data, we tested these sentiment models' accuracy on their respective held-out test data. Ultimately, as indicated by the size of our test data, we chose to move ahead with the sentiment model trained on 90% of our IMDB dataset.

This was not only due to it's slightly higher accuracy score (see figure 8), but also based on the knowledge that generally a larger amount of training data means a large language model that has better performance across all metrics (Vieira et al., 2024).

4 Results

4.1 Traditional Metrics

According to a traditional, in-domain evaluation metric of accuracy, the sentiment model was correct 94.78% (2.d.p) of the time. A break down of the results by sentiment category can be seen in Table 1. This table also displays a slightly lower accuracy score for the out-of-domain data.

Data	Data Split	Accuracy % (2.dp)
IMDB test	Total	94.78
	positive	95.47
	negative	94.09
Yelp data	Total	94.73
	positive	92.59
	negative	94.87

Table 1: Sentiment accuracy by category

4.2 CheckList Evaluation tests

Test	Accuracy	Failure Rate
INV	89.41%	7.38%
DIR	97.62%	0%

Table 2: CheckList test scores

4.2.1 Invariance

Our sentiment model achieved an accuracy score of 89.41% on the INV test set.

There was a failure rate of 7.38% under invariance. Meaning that 366 reviews had their sentiment label change after NE substitutions. An example of this is shown in table 3.

4.2.2 Directional Expectation

Our sentiment model achieved an accuracy score of 97.62% on the DIR test set.

There was a 0% failure rate under directional expectation. Failure here means that the sentiment model's prediction confidence moved by at least 10% in the opposite direction intended by the DIR perturbation. One of the reviews with the largest incorrect confidence change is shown in table 4.

[2] https://github.com/dariusk/corpora/blob/master/data/-corporations/fortune500.json

[3] https://github.com/marcotcr/checklist/tree/master-/checklist/data

Testing Review with Perturbation	Sentiment Labels	Pass/Fail
Im grateful to [Cesar Montano → Jodi Freeze] and his crew in [...]	neg → pos	Fail

Table 3: INV test example

Testing Review with Perturbation	Sentiment Labels	Confidence Change	Pass/Fail
[...] televisionperfected 12 from I despise it	neg → neg	−0.076178	Pass

Table 4: DIR test example

5 Analysis

It is suggested from traditional metrics, that this sentiment model is high performing. There is high accuracy, the between category accuracy difference is minimal, and the out-of-domain accuracy remains high.

The 7.38% INV failure rate shows that the sentiment model does undesirably utilise NEs in it's prediction of sentiment.

Therefore, INV testing can give an explanation behind why out-of-domain testing is slightly less accurate. Under the assumption that similar adverbs, adjectives, and participles are contained in both domains.

An increase in accuracy of the test set upon directional expectation perturbation, alongside a 0% failure rate, from this we can assume that the sentiment model's choices, are being affected, as intended, by the appropriate sentiment language.

6 Discussion

From the two CheckList evaluation tests conducted, we have drawn a reasonable explanation for the results in the sentiment classifier. This shows that there is benefit to the use of CheckList methods.

Considering that the two tests we performed were able to make some suggestions of the internal workings of the sentiment model we chose, we hypothesise that further inclusion of more tests from the CheckList evaluation would provide an even more accurate analysis of the sentiment model's functioning.

Regardless of the depth of knowledge gained from CheckList evaluation, it is clear that a more intricate understanding of the internal functioning of the sentiment classifier has been attained.

We expect that with a less accurate sentiment model, there would be an increase in the failure rates upon the test we performed, allowing them to provide more certain insight into a sentiment model's pitfalls. In order to confirm this, different sentiment models would have to be trained and tested.

However, CheckList has only hinted towards the potential pitfalls in our sentiment model according to the tests we chose, we have made the educated assumption that these results are then true to the inner workings of the sentiment model. If other tests had been performed, we may also say that they were the reasons for imperfect performance of the sentiment model.

There are token-level methods of analysis that may provide more intricate insights into a large language model's focus to combat this uncertainty. Shapley values (Schoch et al., 2023) and Intergrated Gradients (Sanyal and Ren, 2021) provide scores for how strongly individiual words contribute to the predicted sentiment classification. These methods would allow us to see how specific groups of words, such as adjectives or NEs, do or don't contribute to sentiment classification. More intricately they would also be able to show us which specific individual words within these groups are having the largest effect on the large language model tested. This could potentially remove the assumption step that we've had to make using only CheckList and accuracy.

There is a clear downside to only having utilised the CheckList evaluation to assess one NLP model. There would have been merit to using the same training and test data, but with different NLP models. In this way, if there was benefit to using CheckList to differentiate between the capabilities of two or more NLP models, this could give credence to the use of CheckList as a method for the development cycle of a NLP model to be intricately tracked over it's versions.

There is other work, similar to CheckList, that could be included in the evaluation method to help add nuance to the insights drawn. Work on counterfactual generation could add a new dimensional insight into the types of words that NLP models treat as important in sentiment analysis. (Yang et al., 2021)

7 Conclusion

We performed accuracy and CheckList testing on a sentiment model. Using the NER related tests from CheckList, we were able to learn that our sentiment model was sensitive to changes in NEs, and not solely intentionally sentimental language. From our perspective, CheckList was simple to comprehend and straightforward to employ. Therefore, alongside other evaluation metrics, we can see benefit to CheckList tests as NLP model analysis tools. We have also highlighted other evaluation tools that could be used in conjunction with CheckList to add complexity and improve the assessment of NLP models.

References

Khetam Al Sharou, Zhenhao Li, and Lucia Specia. 2021. Towards a better understanding of noise in natural language processing. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 53–62, Held Online. INCOMA Ltd.

Anonymous. 2023. Opedabsa: A dataset for open domain aspect-based sentiment analysis from public reviews. In *ACL Rolling Review - September 2022: A new initiative of the Association for Computational Linguistics*. 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.

Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. Certified robustness to adversarial word substitutions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4129–4142, Hong Kong, China. Association for Computational Linguistics.

Kian Kenyon-Dean, Eisha Ahmed, Scott Fujimoto, Jeremy Georges-Filteau, Christopher Glasz, Barleen Kaur, Auguste Lalande, Shruti Bhanderi, Robert Belfer, Nirmal Kanagasabai, Roman Sarrazingendron, Rohit Verma, and Derek Ruths. 2018. Sentiment analysis: It‘s complicated! In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1886–1895, New Orleans, Louisiana. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Preprint*, arXiv:1907.11692.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Barbara Plank. 2021. Cross-lingual cross-domain nested named entity evaluation on English web texts. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1808–1815, Online. Association for Computational Linguistics.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Soumya Sanyal and Xiang Ren. 2021. Discretized integrated gradients for explaining language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10285–10299, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Stephanie Schoch, Ritwick Mishra, and Yangfeng Ji. 2023. Data selection for fine-tuning large language models using transferred shapley values. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 266–275, Toronto, Canada. Association for Computational Linguistics.

Lolitha Tika Dewi Amelia and Nadira Rania Balqis. 2023. Changes in communication patterns in the digital age. *ARRUS Journal of Social Sciences and Humanities*, 3(4):544–556.

Inacio Vieira, Will Allred, Séamus Lankford, Sheila Castilho, and Andy Way. 2024. How much data is enough data? fine-tuning large language models for in-house translation: Performance evaluation across multiple dataset sizes. In *Proceedings of the 16th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 236–249, Chicago, USA. Association for Machine Translation in the Americas.

Linyi Yang, Jiazheng Li, Padraig Cunningham, Yue Zhang, Barry Smyth, and Ruihai Dong. 2021. Exploring the efficacy of automatically generated counterfactuals for sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 306–316, Online. Association for Computational Linguistics.

Group Contributions: All group members contributed equally. Use of chatbots: We acknowledge that generative AI tools were used throughout this project—for code assistance, explaining external implementations. Chatbots were not used in anyway for report writing.

Appendix A. Chart showing sentiment distribution

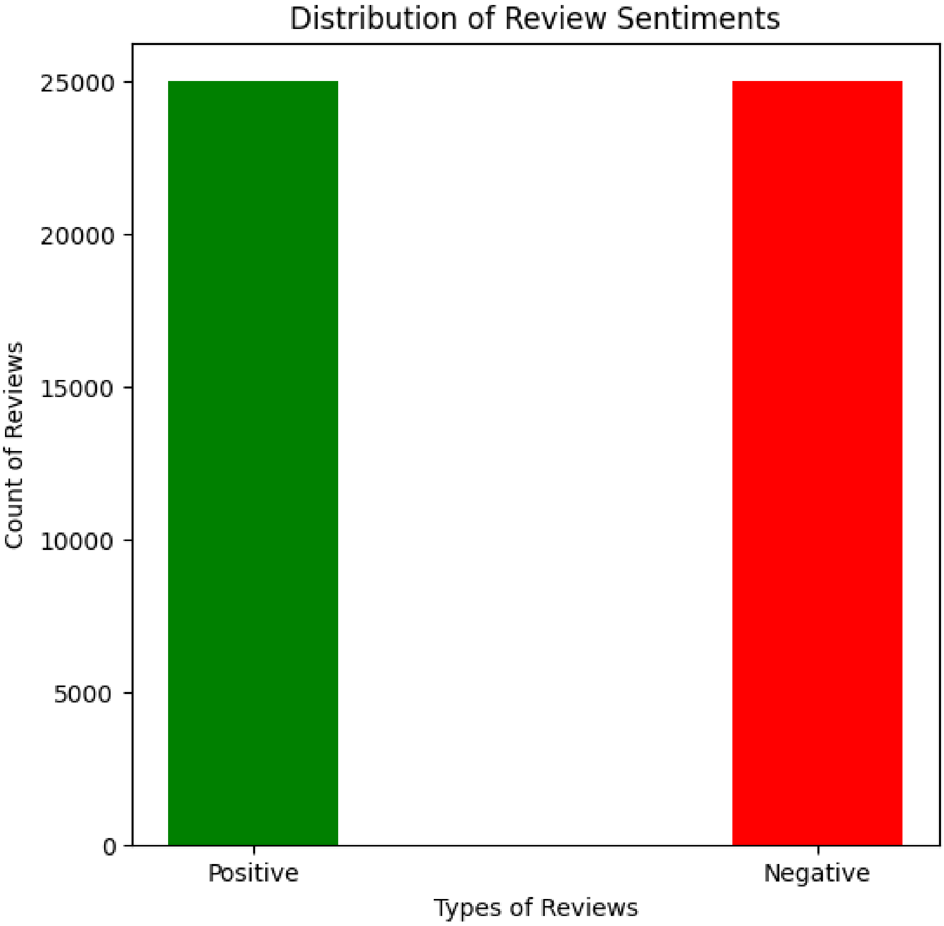


Figure 3: Distribution of Review Lengths

Appendix B. Two charts showing word count distibution by sentiment

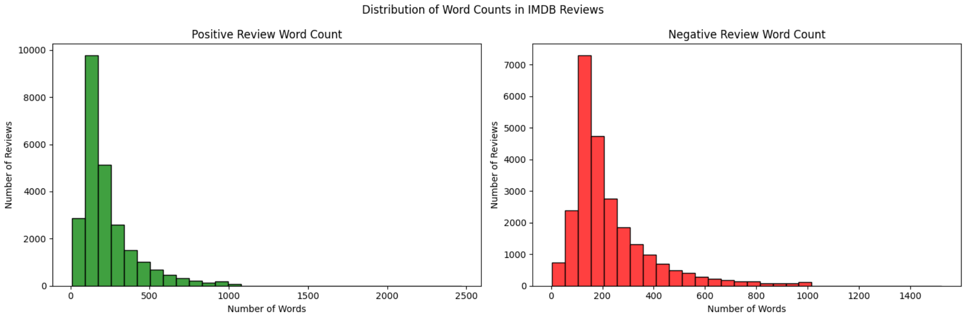


Figure 4: Sentiment Label Distribution

Appendix C. Overlayed word count distributions by sentiment, with bin sizes relative to the review of largest word count

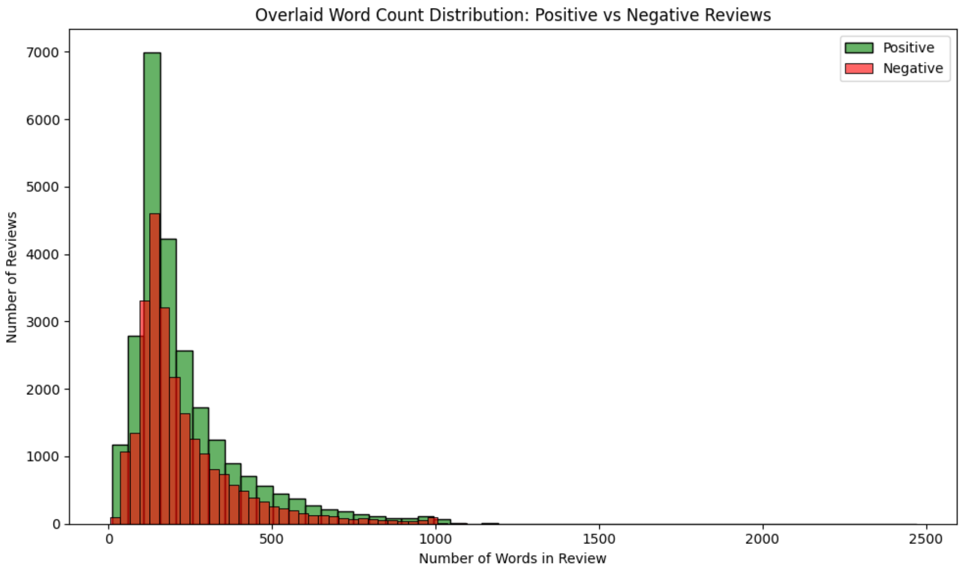


Figure 5: Word Cloud for Positive Reviews

Appendix D. Bar chart of word counts

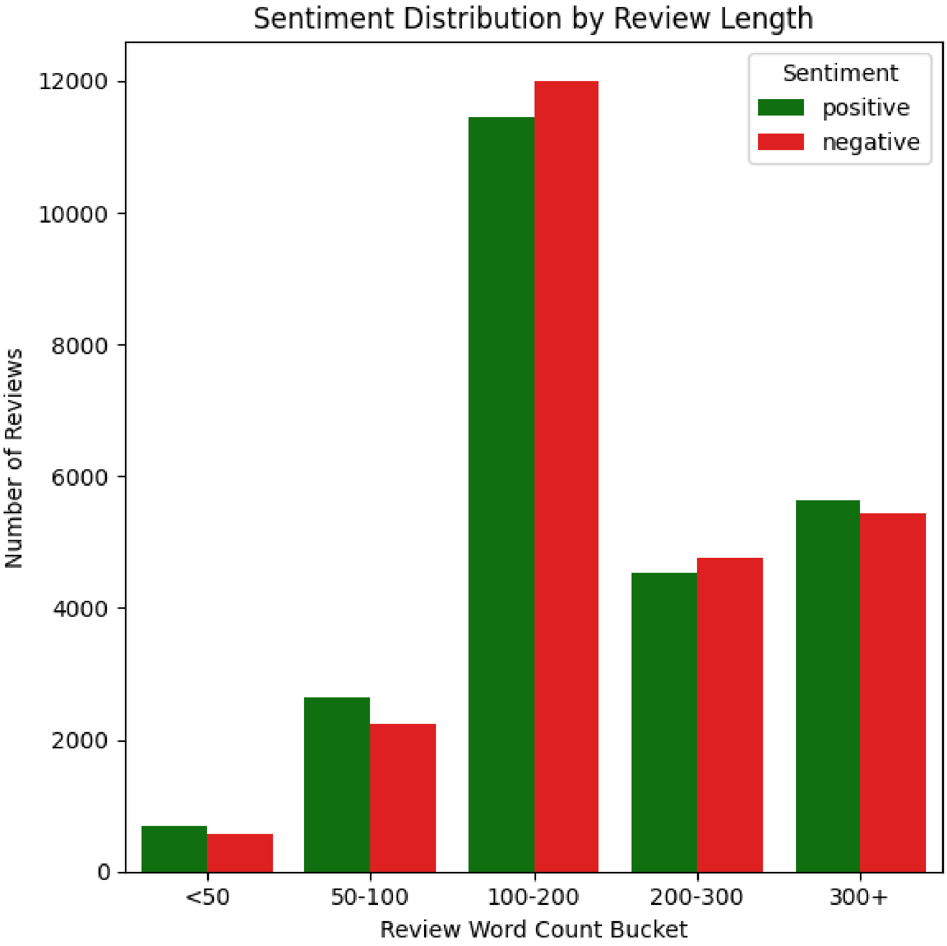


Figure 6: Word Cloud for Negative Reviews

Appendix E. Frequency chart of words by sentiment prior to data cleaning

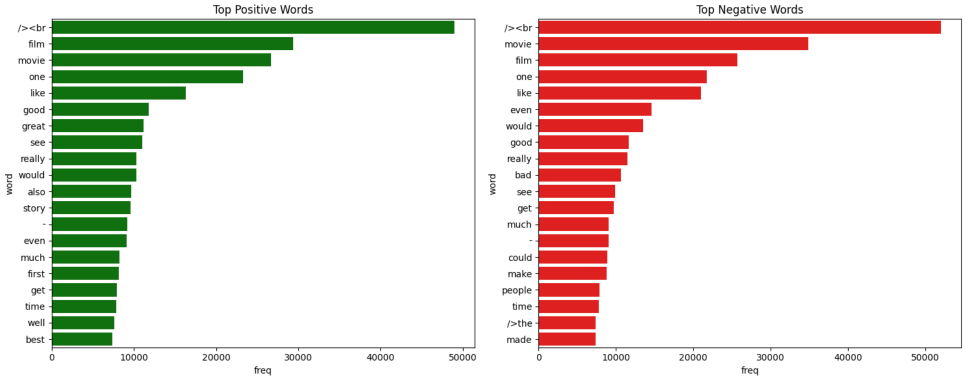
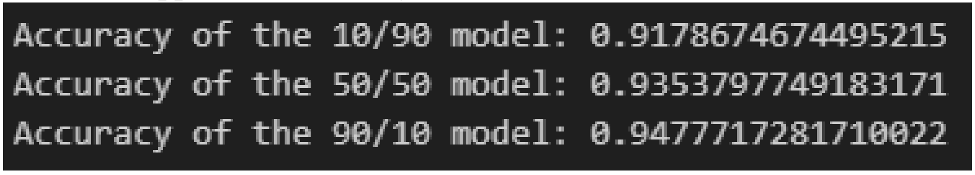


Figure 7: Stopword Frequency Distribution

Appendix F. Accuracy scores of the three model versions



```
Accuracy of the 10/90 model: 0.9178674674495215
Accuracy of the 50/50 model: 0.9353797749183171
Accuracy of the 90/10 model: 0.9477717281710022
```

Figure 8: Train-Test split accuracy scores