DIPARTIMENTO DI MATEMATICA PADOVA
DATA SCIENCE M.Sc A.A. 24-25
NATURAL LANGUAGE PROCESSING

NLP Project Work Plan

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

1 Introduction

For our project, we have chosen to focus on sentiment analysis, a natural language processing (NLP) task that involves determining the sentiment or emotional tone behind a piece of text. This analysis is crucial as it enables businesses to gauge customer opinions, reviews, and feedback, which can inform decision-making, product improvements, and marketing strategies. Sentiment analysis plays an essential role in understanding public perception and sentiment trends in various domains such as social media, e-commerce, and customer service.

We have decided to use the Amazon Fine Food Reviews dataset, which is a collection of reviews for fine food products sold on Amazon. This dataset includes over 500,000 reviews, categorized into three sentiment classes: Positive, Negative, and Neutral.

Initially, we expected the dataset to be imbalanced, and after performing a preliminary exploratory data analysis (EDA), our findings confirmed this expectation. The Positive sentiment class was the majority, while we observed that the Negative sentiment class outnumbered the Neutral class. Based on this, we anticipated that detecting the Neutral class would pose a greater challenge for the models we plan to implement. To maintain the realism of the problem, we have decided not to discard the Neutral class and will continue with a multi-class classification approach instead of simplifying it to a binary classification problem.

2 Project Structure

Our project is divided into two main parts:

- Testing models on the original dataset: We will begin by evaluating different models on the original dataset, starting with simple machine learning algorithms for text classification and progressing to more advanced deep learning models. Our focus will be on optimizing recall and precision, while also considering the F1-score, which balances the two. Our goal is to achieve a performance of at least 0.75 across these three metrics. Given that the dataset is skewed toward the positive class, we believe that 0.75 is a reasonable baseline.

  Accuracy is not a priority in this phase, as we anticipate it to exceed 0.90, which is not an informative metric for our study. Additionally, we will analyze the loss function values to examine convergence across models. Another crucial aspect is the training cost, where we will consider the number of parameters and training time. While increasing model complexity may lead to slight improvements in performance, it is essential to weigh these against computational costs, especially for real-world applicability. By the end of this phase, we identified a model that exceeded our target performance threshold of 0.75. However, the high training cost made it impractical given our limited computational resources, preventing further fine-tuning.

- Applying resampling techniques: In the second phase, we will employ resampling techniques, specifically oversampling, as described in [7]. Given that the dataset has a significantly larger proportion of positive samples, undersampling would result in a loss of information. Thus, we opt for oversampling to balance the classes.

  However, increasing the minority class size results in a larger dataset, which may necessitate adjustments to our models to handle the added complexity. We will evaluate the same metrics as in the previous step, but in this case, the accuracy metric will become more informative.

3 Models and Implementation details

We chose to use pre-trained GloVe embeddings, which are available in various dimensions such as 50, 100, 200, and 300. Considering computational cost and resource constraints, we opted for an embedding dimension of 100. While higher-dimensional embeddings can potentially improve model performance, it is crucial to balance this with the implementation cost.

Initially, we planned to experiment with multiple pre-trained embeddings, including Word2Vec, expecting to observe notable performance differences across models. However, after testing one model with Word2Vec, we found its performance to be nearly identical to GloVe. Given our resource limitations, we decided to exclude Word2Vec from our study, as it neither provided a significant performance difference nor allowed for extensive tuning due to computational constraints. Our project implementation was carried out using Google Colab Pro resources.

3.1 Logistic Regression with TF-IDF

Firstly, we started with a well-known machine learning algorithm for classification tasks: Logistic Regression. For feature extraction, we utilized TF-IDF with n-grams, specifically bi-grams, as we believe that considering pairs of consecutive words can capture contextual meaning better than unigrams [1]. For example, in sentiment analysis, the phrase ”not good” carries a negative sentiment, whereas analyzing the words ”not” and ”good” separately might lead to an incorrect interpretation.

Due to the expected class imbalance in our dataset, the model exhibited poor performance, particularly in detecting the minority classes. A well-known approach to mitigate this issue is using class weights to give higher importance to underrepresented classes. However, after applying this method, we observed no significant improvement in the results [7].

3.2 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are widely used deep learning architectures that have demonstrated effectiveness across various domains. After conducting a literature review, we decided to implement a CNN model for our sentiment analysis task, drawing insights from previous studies [2, 6, 13]. Our CNN was built and tested on both versions of the dataset we described earlier.

On the original dataset, the CNN model reached a performance level that was nearly in line with our target. However, when tested on the oversampled dataset, it exhibited significantly improved results, emphasizing the advantages of resampling. Additionally, we observed differences in training cost, but still computational cost of the model was reasopnable compared to other models that we have tested.

3.3 LSTM and BiLSTM

Recurrent Neural Networks (RNNs) are widely used in Natural Language Processing (NLP) tasks due to their ability to capture sequential dependencies in text, making them particularly effective for text classification. Unlike traditional feedforward networks, RNNs can maintain context across time steps, allowing them to model complex linguistic structures. Based on previous studies [4, 9], we decided to implement and evaluate both Long Short-Term Memory (LSTM) and Bidirectional LSTM (BiLSTM) architectures for our sentiment analysis task.

Our initial experiments were conducted on the original dataset, as running these models on the orignal version of dataset proved computationally expensive. Given that LSTMs and BiLSTMs are specifically designed to address long-term dependencies, we anticipated that they would excel in handling class imbalance. However, our results showed performance levels comparable to those of our CNN model, suggesting that additional fine-tuning could be necessary to fully leverage their potential. Future work could focus on optimizing hyperparameters to further enhance their performance, especially in imbalanced datasets.

3.3.1 LSTM+CNN model

As the final model in our project, we implemented a hybrid architecture combining LSTM and Convolutional Neural Networks (CNNs), drawing inspiration from the approach discussed in [2]. This hybrid model demonstrated strong performance across all evaluation metrics. We further tested it on the resampled dataset, where it continued to deliver promising results. However, our best-performing model on the resampled dataset was a deeper CNN architecture, designed to better capture the complexity of the task. The CNN model outperformed the hybrid model on all metrics while also being more computationally efficient.

4 Conclusions

To summarize, we successfully met the objectives of our project using the original dataset, achieving the desired thresholds for precision, recall, and F1 score, while keeping the model's training cost manageable. Moving forward, we aim to explore models such as BERT [3, 12] or fine-tuned versions of BERT [10] [12, 11][8], as well as VADER [5], to determine if we can identify a model that performs well on the original dataset. Although these models were initially part of our primary work plan, limited GPU resources prevented us from fully pursuing this goal as a result, we had to leave these models as future work. As for the resampled dataset, we anticipated that addressing the class imbalance would improve performance, and our CNN architecture delivered satisfactory results in both terms of cost and performance.

Gruppo 1

Name/Surname	Email address	Student ID
Eldar Eyvazov	eldar.eyvazov@studenti.unipd.it	2073221

Riferimenti bibliografici

[1] Author(s). Analysis of sentiment on amazon product reviews. In 2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC), page 697. IEEE, 2023.

[2] C. Colón-Ruiz and I. Segura-Bedmar. Comparing deep learning architectures for sentiment analysis on drug reviews. Journal of Biomedical Informatics, 2021.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2019.

[4] A. Graves, A. R. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. IEEE Transactions on Audio, Speech, and Language Processing, 22(4):586–597, 2013.

[5] Clayton J Hutto and Eric E Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the 8th International Conference on Weblogs and Social Media (ICWSM 2014), 2014.

[6] Hannah Kim and Young-Seob Jeong. Sentiment classification using convolutional neural networks. In Proceedings of the Conference, 2019.

[7] Guillaume Lemaître, Fernando Nogueira, and Christos K Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. Journal of Machine Learning Research, 18(17):1–5, 2017.

[8] Yinhan Liu, Myle Ott, Naman Goyal, Jianmo Du, Mandar Joshi, Danqi Chen, Mike Lewis, Luke Zettlemoyer, Anatoli Polozov, Yin Xu, et al. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019.

[9] U. B. Mahadevaswamy and P. Swathi. Sentiment analysis using bidirectional lstm network. Journal of Advanced Computing Research, 12(3):45–56, 2021.

[10] Chi Sun, Danqi Li, Eunsol Choi, and Ruslan Salakhutdinov. Fine-tuning bert for document classification with label-wise attention. arXiv preprint arXiv:1910.09572, 2019.

[11] V Sundararajan, V Suresh, V Sundararajan, and V Sundararajan. Transformer based contextual model for sentiment analysis of customer reviews. International Journal of Engineering and Technology, 12(11):1–5, 2020.

[12] Michelle Lu Wang. Fine-tuning bert for sentiment analysis. eScholarship, 2024.

[13] Yufei Xie and Rodolfo C. Raga Jr. Convolutional neural networks for sentiment analysis on weibo data: A natural language processing approach. In Proceedings of the Conference, 2020.