Deep Learning (COSC 2779) – Assignment 2 – 2024

Dong Manh Duc (S3972890)

1 Problem Definition and Background

a) Background

Social networks play a crucial role in modern society, with platforms like Twitter, Facebook, and Instagram being key channels for sharing ideas, thoughts, and political beliefs. While these platforms spread information quickly, it can be difficult to determine if the content shared influences users' judgment in subtle ways. Memes are powerful tools for spreading cultural ideas and are commonly used in disinformation campaigns, leveraging techniques like oversimplification, name-calling, and smears to influence users. Detecting these harmful memes automatically is a challenging but essential task for managing social networks. In recent years, advances in machine learning and deep learning have significantly improved the ability to extract and analyze multimedia content at scale from social networks.

b) Problem definition

Main	**Enhancement**
The task is to analyze a meme and determine which rhetorical or psychological techniques, out of a set of 22 possible techniques, are used within it. This involves considering both the textual content (the caption or overlayed text) and the visual elements of the image. Since a meme can employ multiple techniques simultaneously, this is a **multilabel classification** problem, meaning the model must identify and label all applicable techniques for each meme. These techniques could range from name-calling and causal oversimplification to exaggeration and appeal to fear. The challenge lies in effectively combining the analysis of both text and image to detect the subtle methods being used to influence viewers.	The task involves analysing the text within a meme and determining which rhetorical techniques, from a predefined set of 20, are being used. In addition to identifying the techniques, the model must pinpoint the exact spans of text where each technique is applied. This is like NER task, however, it is a much more complex **multilabel sequence tagging** problem, as multiple techniques can be present in different parts of the text, and the model needs to recognize each one independently. The challenge lies in not only classifying the techniques but also accurately locating the relevant text spans, making it essential to capture the nuanced ways in which disinformation and manipulation tactics are embedded within memes.

2 Evaluation Framework

Main	Enhancement
We evaluate the models using both Micro and Macro F1 scores, which are appropriate metrics for multi-class, multi-label tasks where the labels are extremely imbalanced. Specifically, I aim for 0.1 and 0.45 on the test dataset for macro and micro F1 score respectively.	Given the extreme imbalance in our dataset, we chose to focus on micro F1 score, precision, and recall as key metrics. Precision helps us understand how effectively the model avoids false positives, while recall shows how well it captures true positives, minimizing false negatives. The micro F1 score combines both precision and recall, offering a balanced perspective on performance by considering the trade-offs between them. These metrics provide a clearer, more reliable assessment of the model's ability to handle the imbalanced labels in our task. As for target, I aim for 0.25 for all metrics on the test dataset.

3 Approach & Justifications

For both tasks, in addition to Binary Cross-Entropy (BCE), Focal Loss was experimented with to address the issue of class imbalance in the dataset. The goal of using Focal Loss was to help the model focus more on the minority labels, which are underrepresented. Given the highly imbalanced nature of the dataset, this adjustment was crucial in improving the model's ability to detect less frequent classes. Additionally, label smoothing was applied across all loss functions to mitigate the issue of overconfidence—a common problem in models trained on imbalanced data. By softening the sharp probabilities typically assigned to dominant labels, label smoothing helps the model generalize better and avoid becoming overly biased toward the majority class.

Recognizing the limited size of the dataset, I also concluded that training a model from scratch to extract multimodal features was not feasible, as it would likely result in poor performance. Instead, I opted to use

pretrained models, which are trained on much larger datasets and capable of capturing more nuanced information that can be beneficial for our task. Several pretrained models were evaluated and compared to select the ones that best suited the problem. For visual feature extraction, ResNet50 was chosen due to its efficiency and strong performance in a wide range of vision tasks. For textual feature extraction, I selected ALBERT, a lightweight language model that offers competitive performance while being much smaller in size compared to other models. This choice strikes a balance between computational efficiency and accuracy.

	Main	Enhancement
Baseline	As there are 2 different modalities, a dual-branch approach was used for the baseline model. Specifically, pretrained models were used to extract information from 2 modalities. The extracted features are then concatenated, and followed by a Linear layer with sigmoid activation	Since there is only text modality present, a pretrained text model was used to extract its information, followed by a Linear layer with sigmoid activation. Since the spans are given on character-level in the dataset and the transformer models run on token-level, all spans are transformed to token-level.
Proposed	The baseline model lacked the capability to effectively combine features from two modalities, leading to limited information extraction from the data. This resulted in a stagnant validation F1 score. To address this, I implemented a transformer-based fusion mechanism, inspired by the AIMH approach, but modified to suit our specific settings (Messina et al., 2021). Recognizing that transformers can overfit with small datasets, I introduced regularization in the model implementation. The visual and textual features were first extracted using pretrained vision and text models. These extracted features were then passed through transformer layers, which provided contextualized representations of each modality. In this setup, two separate transformer layers were used—one for text features and one for image features. These contextualized features were then passed through a linear layer with a sigmoid activation to generate the output probabilities. To prevent overfitting and improve generalization, an ensemble-like approach was used: the model averaged the probabilities from both the text and image transformers to produce the final prediction. This averaging of multiple perspectives acts as a form of implicit regularization, helping the model avoid reliance on a single feature set and thus reducing the risk of overfitting. While this method significantly increased the micro F1 score and eliminated overfitting, it inadvertently led to a lower macro F1 score. To address this, the decision threshold was fine-tuned to balance both micro and macro F1 scores, optimizing the model's overall performance.	While the baseline model exceeded the target for Precision, Recall improved very slowly. This was expected due to the highly imbalanced dataset, which caused the model to focus more on the majority classes. To address this, Focal Loss was introduced to shift the model's attention toward the minority classes, and label smoothing was applied to reduce overconfidence in the dominant labels. Both Focal Loss and label smoothing were fine-tuned to find the optimal parameters that would maximize Recall. Finally, threshold tuning was applied to the fine-tuned model to strike a balance between Recall and Precision, ultimately leading to a higher F1 score.

4 Experiments & Tuning

Main	Enhancement
In all experiments, the Nadam optimizer with a learning rate of 5e-5 was used for faster convergence, with ResNet50 as the image extractor and ALBERT large for text extraction. To improve the macro F1 score, Binary Cross-Entropy (BCE) and Focal Loss were compared. Surprisingly, the model trained with BCE outperformed the one trained with Focal Loss in both macro and micro F1 scores, even though Focal Loss is typically designed for handling class imbalance (Appendix 1). This may be because Focal Loss tends to focus on hard-to-classify instances, which are not necessarily minority class samples. Based on these results, BCE was chosen for future models. The model was then fine-tuned using a stepwise approach: first freezing both extractors, then sequentially unfreezing and training the text	In all experiments, the Nadam optimizer with a learning rate of 5e-5 was used for faster convergence, with ALBERT large employed for text feature extraction. The baseline model was initially trained using Focal Loss with default parameters. Despite this, the baseline achieved high Precision but low Recall, indicating the model was primarily focusing on the majority classes due to the dataset's imbalance. The possible reason why Focal Loss and these adjustments did not fully resolve the imbalance could be due to the inherent difficulty in classifying minority classes when the dataset is highly skewed. Focal Loss focuses on hard-to-classify instances but may not always align with minority classes.

extractor, followed by the image extractor, and finally unfreezing both. Although this strategy helped prevent overfitting, it did not surpass the performance of fine-tuning only the text extractor (Appendix 2). Due to limited time and computational resources, I shifted focus to threshold tuning rather than retraining multiple models. I tested various thresholds, aiming to balance macro and micro F1 scores. Ultimately, a threshold of 0.28 provided the best micro F1 score while maintaining a reasonably high macro F1 score (Appendix 3).	To address this imbalance, I experimented with different values for label smoothing and the alpha parameter in Focal Loss to find the combination that would improve both Recall and Precision. After testing 20 models, I found that a label smoothing value of 0.1 and an alpha of 1.2 in Focal Loss provided the best balance. While this improved Recall compared to the baseline, Precision and Recall remained somewhat imbalanced. To further address this issue, I fine-tuned the decision threshold, ultimately selecting 0.4 (Appendix 5). This tuning helped achieve a more balanced Precision and Recall, both around 0.3, surpassing the target.

5 Ultimate Judgment, Analysis & Limitations

	Main	Enhancement
Final model selection	For the final model, I choose the proposed architecture with a fine-tuned text module and a threshold of 0.28. This decision is based on the model's strong performance, achieving macro and micro F1 scores of 0.15 and 0.498, respectively, on the test dataset. These results marked a significant improvement over the baseline model, despite the use of limited computational resources. The improved macro F1 score indicates that the model was better at handling minority classes, which was a key goal given the dataset's imbalance. Overall, the final model exceeded the initial targets, demonstrating a balanced approach that avoided overfitting the majority classes. However, some limitations remain, such as further enhancing the model's performance on underrepresented labels and exploring additional techniques to improve generalization. These aspects could be addressed in future iterations to further optimize the model for real-world applications.	For the final model, I selected the architecture utilizing ALBERT as its backbone, with a threshold of 0.4. This decision was based on its balanced performance across recall, precision, and F1 score, with each metric scoring around 0.3 on the test dataset. Notably, the model outperformed the baseline, particularly in recall with limited resources. Overall, the final model not only met but exceeded the initial objectives, delivering higher scores while avoiding an overemphasis on majority classes. However, there remain areas for improvement, which could be addressed in future iterations to further enhance the model's effectiveness
Real world challenges	The model performs well for common labels but struggles with rare ones, even after threshold adjustments. For example, the **Bandwagon** label, which appears fewer than 10 times in the dataset, is rarely predicted correctly. This is largely due to the dataset imbalance, making it difficult for the model to learn patterns for infrequent labels. While the model excels with frequent labels like **Smears**, it lacks sufficient examples to handle the rarer ones effectively. To improve performance on minority classes and boost the macro F1 score, a more powerful model like RoBERTa or DeBERTa could be used, but this would require more computational resources. Alternatively, collecting more data for the underrepresented labels would help, though this approach would demand additional time and resources.	Although the model met the initial targets, it is still not suitable for real-world use as the model frequently misses key spans, even for majority classes, indicating it struggles to consistently identify correct labels. This makes it unreliable in practical applications. To improve the model, particularly for minority classes and to boost the macro F1 score, one option is to use a more powerful language model like RoBERTa or DeBERTa,, which may not be practical on a local machine. Another approach is to gather more data for the underrepresented classes. This would help address the dataset imbalance, providing the model with more examples to learn from and improving its ability to generalize. However, this solution requires additional time and resources.

6 References:

Nicola Messina, Fabrizio Falchi, Claudio Gennaro, and Giuseppe Amato. 2021. AIMH at SemEval-2021 Task 6: multimodal classification using an ensemble of transformer models. In Proceedings of the International Workshop on Semantic Evaluation, SemEval '21, Bangkok, Thailand

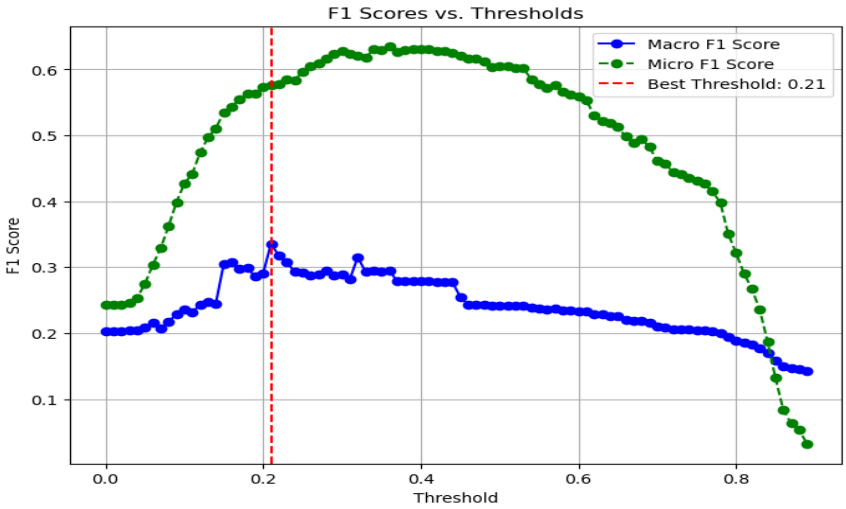
7 Appendix

Loss function approach	Macro F1 score	Micro F1 score
Focal Loss	0.0624	0.4333
BCE (Binary Cross Entropy)	0.0915	0.5022

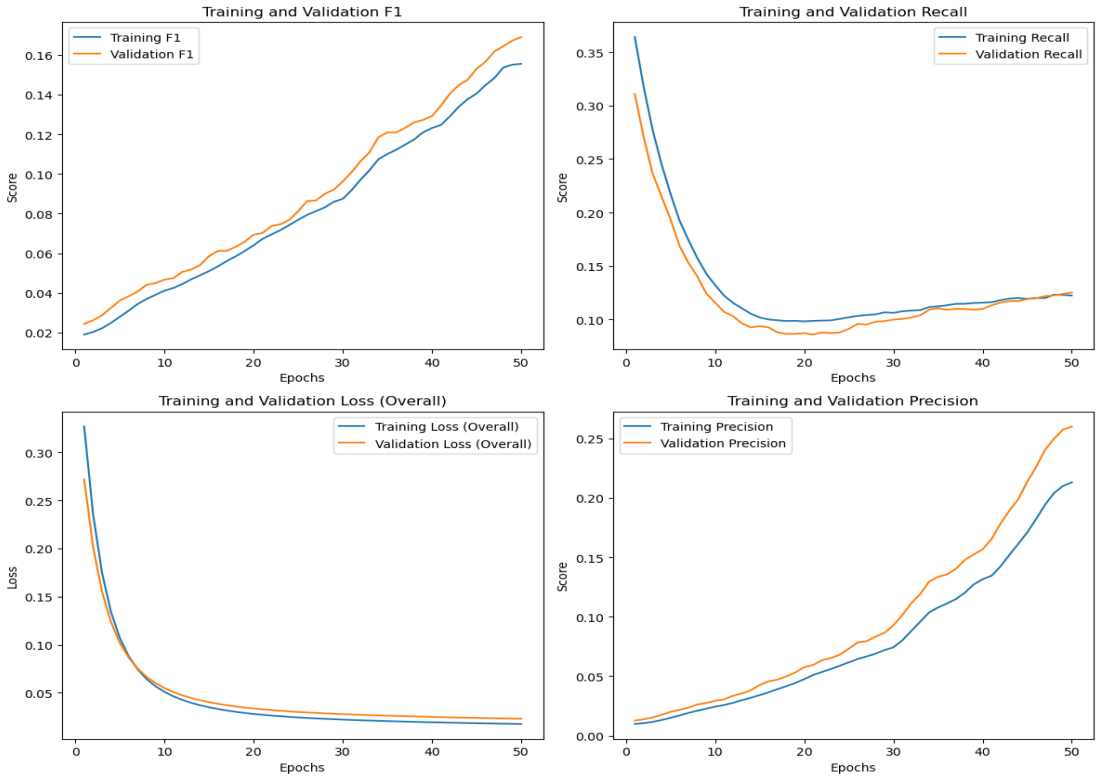
Appendix 1

Approach	Macro F1 score	Micro F1 score
Finetuning both extractors	0.095	0.57
Finetuning only text extractor	0.1	0.58

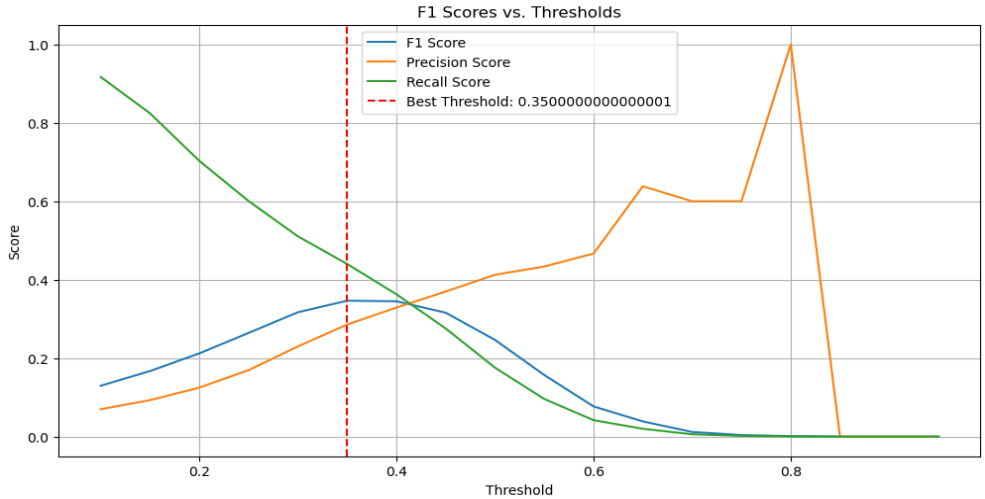
Appendix 2



Appendix 3



Appendix 4



Appendix 5