

Multimodal Detection of Persuasion Techniques in Memes under Computational Constraints

by

Timur Rakhishov

Submitted to the Department of Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science

at the

NAZARBAYEV UNIVERSITY

June 2026

© Nazarbayev University 2026. All rights reserved.

Author
Department of Computer Science

April 11

Certified by
Michael Lewis

Associate Professor, School of Engineering and Digital Sciences,
Nazarbayev University
Thesis Course Coordinator

Certified by
Latif Zohaib

Associate Professor, School of Engineering and Digital Sciences,
Nazarbayev University
Thesis Course Coordinator

Accepted by
Yelyzaveta Arkhangelsky

Dean, School of Engineering and Digital Sciences

Multimodal Detection of Persuasion Techniques in Memes under Computational Constraints

by

Timur Rakhishov

Submitted to the Department of Computer Science
on April 11, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science

Abstract

This thesis investigated the multimodal architecture for detection of persuasion techniques in memes in computational environments based on the SemEval-2024 Task 4 shared task on multilingual persuasion detection. The research addressed three main challenges: the extreme class imbalance among the 22 persuasion techniques; the complex fusion of multimodal input; and the real-world constraint of hardware power of consumer-grade GPUs.

Multiple architectural methods were systematically compared: beginning with late fusion using an attention mechanism, followed by multimodal fusion using a gated attention mechanism, and concluding with a low rank adaptation of large language models through the use of LoRa.

The combination of TinyLLaMa (1.1 B parameters), LoRa adaptation, CLIP-Large for visual representation, and a gated attention network used selective samples where multimodal was required achieved a Hierarchical F1 Score of 0.704 on the English Test Set. A key finding was that training exclusively on samples requiring both image and text for correct interpretation outperformed training on all available data. This demonstrates that, under resource constraints, curating data for multimodal relevance is more valuable than maximizing data volume. These results further demonstrated that small modular architectures may have greater potential than larger theoretical models such as VisualBERT if training and/or testing batch sizes are restricted due to hardware limitations.

These results provide practical guidance for the development of multimodal systems and indicate that there is a significant lack of research into how computational resources affect the efficacy of different architectures.

Thesis Course Coordinator: Michael Lewis

Title: Associate Professor, School of Engineering and Digital Sciences, Nazarbayev University

Thesis Course Coordinator: Latif Zohaib

Title: Associate Professor, School of Engineering and Digital Sciences, Nazarbayev University

Contents

1	Introduction	13
1.1	Background and Motivation	13
1.2	Problem Statement	15
1.3	Research Objectives	17
1.4	Significance of Thesis	18
1.4.1	Methodological Contributions	18
1.4.2	Empirical Insights	19
1.4.3	Practical Impact	19
2	Related works	21
2.1	Literature Review	21
2.1.1	The Multimodal Challenge of Meme Analysis	21
2.1.2	Technical Approaches to Multimodal Meme Classification . . .	22
2.1.3	Propaganda and Persuasion Detection	22
2.1.4	Synthesizing Insights: Toward Comprehensive Harmful Content Detection	23
3	Dataset	25
3.1	Dataset Overview	25
3.2	Task Definitions and Subtasks	25
3.3	Dataset Statistics and Composition	26
3.4	Persuasion Techniques Distribution	27
3.5	Label Distribution and Multimodal Complexity	28

3.6	Dataset Challenges and Implications	28
4	Methodology	31
4.1	Overview	31
4.2	Experimentation Setup	32
4.3	Data Augmentation	33
4.3.1	Reasoning behind choosing data augmentation techniques . . .	33
4.3.2	Selecting Labels to Augment	34
4.3.3	Augmentation Using NLPAug	35
4.3.4	Pseudolabeling	35
4.4	Replication of the baseline models	36
4.4.1	Subtask 1: Ensemble Learning with Chain-of-Thought Data Augmentation	36
4.4.2	Subtask 2a: Hierarchical Multilabel Classification Turned into Hierarchy-Aware Prompt-Tuning Task	42
4.5	Novel Architectures and Experimental Extensions	44
4.5.1	Attention-based Fusion for Multi-modal Tasks	44
4.5.2	Gated Fusion Architecture and Training	49
4.5.3	Low-Rank Adaptation (<i>LoRA</i>) with Gated Fusion and Larger Text Model Backbone	50
4.5.4	Pretraining with Image-Text Contrastive Learning and Tar- geted Prediction	57
4.5.5	Cross-Model Validation of the Best-Performing Method	59
4.5.6	Early Fusion: VisualBERT with LoRA and Contrastive Fine- Tuning	61
4.5.7	Improving Class Balance with Weighted Losses and Balanced Pretraining	70
5	Results	75
5.1	Detailed Description of the Best Model	75
5.1.1	Architecture Overview	75

5.1.2	Training Configuration	77
5.1.3	Performance Summary	79
5.1.4	Implementation Details	79
6	Discussion	81
6.1	Hardware Constraints and Their Impact on Model Performance . . .	81
6.1.1	Batch Size Limitations	82
6.1.2	Memory Constraints and Model Architecture	82
6.2	Key Findings and the Superiority of TinyLLaMA-based Approaches .	83
6.3	Architectural Analysis: Why Early Fusion Failed to Outperform Late Fusion	83
6.4	Ensemble Methods and Their Limitations	84
6.5	Resource-driven Implications	85
7	Conclusion	87
7.1	Main Contributions	87
7.2	Lessons from Failed Approaches	88
7.3	Future Directions	89
7.4	Final Remarks	89

List of Figures

1-1	Example of a meme from the training dataset.	15
4-1	Textual Data Augmentation by nlpaug	34

List of Tables

3.1	Number of memes for every language on each subtask and associated data splits. Note that only the test split contains all four languages. EN=English, BG=Bulgarian, MK=North Macedonian, AR=Arabic [7]	27
3.2	Subtask 2b label distribution [7]	28
4.1	GPU resources available on Google Colab free tier	33
4.2	Comparison of ensemble model performance with various pretraining datasets (augmentation techniques)	39
4.3	Comparison of metrics for HPT for persuasion technique detection.	43
4.4	Hyperparameters selected by Optuna for models	48
4.5	Performance of individual models and ensemble on the test set	48
4.6	Gated Fusion Metrics	49
4.7	LoRA-enhanced gated fusion on TinyLLaMA	53
4.8	Comparison of CLIP-Base and CLIP-Large as image encoders in gated fusion with LoRA	54
4.9	Performance under different training strategies based on multimodal dependency	56
4.10	Results from contrastive pretraining followed by supervised fine-tuning	58
4.11	Performance using model selection based on multimodal dependency	58
4.12	Performance comparison across different model architectures using the multimodal-only strategy without image contrastive learning pretraining	60
4.13	Results of two types of losses: Margin-Based Contrastive Loss and Binary Cross-Entropy Loss	68

4.14	Fine-tuning with Margin-Based Contrastive Loss for 10 and 20 epochs	69
4.15	Performance with balanced contrastive pretraining and BCE loss with two ways of setting weights	73

Chapter 1

Introduction

1.1 Background and Motivation

With the rise of social media and immediate access to the Internet people were able to share their thoughts, beliefs, and ideas with a global audience quickly and without too much effort. This kind of freedom of expression has provided a powerful platform for political activism, public awareness campaigns, and social movements. However, this same freedom has also opened the door to the manipulation of public opinion through the spread of disinformation, political propaganda, and persuasive content designed to influence audiences.

A meme is commonly defined as a digital artifact that combines a visual element—usually an image or video—with a textual component, creating a message that is easily shared, modified, and spread across the Internet. Nowadays, memes are particularly powerful because of their simplicity, wide reach, ability to quickly capture attention and to get people’s reaction. With the spread of infinite feeds and advances in recommendation systems, people are consuming content like that on a daily basis. While they can be used for humor and social bonding, memes are also powerful when used for persuasion. Given their growing influence in the digital space, it is crucial to understand how persuasive techniques are embedded within memes. This understanding is essential for recognizing their influence on the audience’s world perception — particularly when they are used to spread misinformation, ignite intercultural hatred,

or deepen societal divisions [25]. Persuasive memes can contribute to polarization and tensions within a country, making their identification and critical evaluation an urgent matter of public concern [26]. Therefore, the identification of such persuasive techniques could be an important tool that ensures digital and societal security in the future.

This thesis focuses on the detection of persuasion techniques within memes. SemEval (Semantic Evaluation) is a series of international shared tasks that provide standardized benchmarks for evaluating natural language processing systems across a wide range of semantic challenges. There are 22 persuasion techniques identified in the context of memes that were originally presented in the SemEval-2021 Task 6 (Detection of Persuasion Techniques in Texts and Images) [6]. Building on the work presented in the SemEval-2024 Task 4 [7], which addressed the multilingual and multimodal detection of persuasion strategies in memes, this research aims to provide a critical analysis of current approaches in meme analysis and conduct further experimentation of multimodal persuasion technique detection under computational constraints. The SemEval-2024 shared task is significant in that it extends previous work on persuasion detection in digital content by introducing multilingual datasets and a novel hierarchical evaluation framework. The task is structured into three subtasks:

- Subtask 1 (ST1) identifying persuasion techniques used in a meme only considering the textual content of a meme,
- Subtask 2a (ST2a) detecting specific persuasion techniques within the meme’s textual and image contents,
- Subtask 2b (ST2b) identifying if the image contains persuasion techniques and is propagandistic or not.

This thesis builds upon these tasks, firstly critically analyzing the models submitted to the SemEval-2024 shared task [7], mostly focusing on Subtask 2a, which combines both the multilabel nature of Subtask 1 and multimodal nature of Subtasks 2a and 2b.

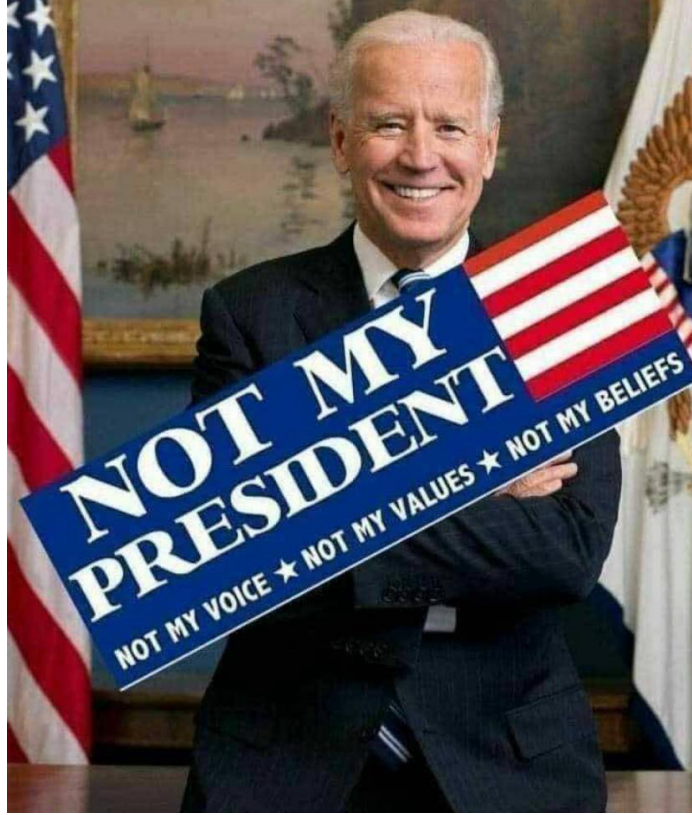


Figure 1-1: Example of a meme from the training dataset.

While the multimodality is already challenging, as it requires understanding both textual content and visual elements, as well as their interaction, the task also involves multilabel classification where multiple techniques can both be integrated within a single meme, which further increases the complexity of this task.

Then, the work explores three main directions after careful consideration: improving model performance through data augmentation, replicating strong existing methods, and re-evaluating underperforming but theoretically sound approaches. The aim is not only to build a competitive model, but also to gain insights into what makes certain strategies effective or not.

1.2 Problem Statement

The purpose of this thesis is to address the challenges of developing effective multimodal architectures for detecting persuasion techniques using computationally con-

strained systems. In multimodal learning, fusion strategies refer to the methods used to combine visual and textual representations into a unified feature space for downstream classification. Previous work has shown many methods for addressing the problem, but little has been done to investigate how different fusion strategies function at levels of computing resources available to most academics.

The primary research question can thus be stated as follows: **How can multimodal fusion architectures be effectively developed and optimized for detection of persuasion techniques in memes and what variables affect their performance with respect to their computational constraints?**

These questions involve the resolution of several other problems which are important:

- P1. Selection of Multimodal Fusion Strategy:** What criteria should guide the decision to fuse visual and textual representations, when to use an “early” vs. “late” fusion strategy and what are the tradeoffs of each?
- P2. Management of Computational Constraints:** How do researchers develop architectures that will run on standard equipment including GPUs with limited memory, limit batch sizes and complete training quickly enough to meet deadlines imposed by funding agencies or the need to get results published?
- P3. Optimization of Parameter Efficiency:** How can researchers take advantage of pre-trained large language models and adapt them to their particular domains without needing either large amounts of additional computation to fine tune the models or access to specialized hardware such as TPUs?
- P4. Class Imbalance and Dataset Characteristics:** How can researchers address issues related to class imbalances in the frequency of appearance of persuasion techniques as well as limited availability of labeled training data (i.e., some classes have less than 100 labeled examples)?
- P5. Assessment of Modality Relevance:** When does the addition of visual representations contribute meaningfully to the detection of persuasion and when

would a purely textual based approach suffice? What design strategies can we employ to take advantage of these distinctions?

Through investigation of relationships between architectures, training procedures, and techniques for augmenting datasets, this thesis hopes to increase knowledge about detection of persuasive messages while shedding light on how hardware limits influence the quality of the architectures used.

1.3 Research Objectives

To better address those challenges, the thesis is centered on the objectives listed below:

1. Compare/evaluate the performance of early fusion architectures as well as late fusion methods using several different types of attention.
2. Analyze how hardware constraints (e.g., GPU memory, batch size limits, training time), limit the performance differences among various architectures; and to gain insight into what architecture may be best suited for use in real-world applications.
3. Investigate parameter efficient approaches to adapt large pre-trained language models to detect multi-modal persuasion. Examine how the changes will influence the trade-off between model performance and computational resource usage.
4. Identify and apply the most effective class imbalanced methodologies (data augmentation, re-weighted loss functions, sampling strategies) to improve the ability of a system to identify both common and uncommon persuasion techniques within an un-balanced dataset.
5. Challenge common assumptions regarding the optimal amount of training data required to train a system capable of identifying persuasion in multi-modal

input; investigate whether or not it is possible to achieve acceptable results by training solely on examples which include multi-modal input.

6. Improve contextual understanding within the prediction generated by the model through integration of pre-trained language models, hierarchical label dependency relationships and modality fusion techniques to generate a unified representation of the images/text within a given example.
7. Implement/compare the most successful models used to perform multi-label classification of persuasion techniques for both text-only and multi-modal content; particularly focus on memes containing both textual and visual representations; and provide a comparison to validate the effectiveness of some of the methodology employed.
8. Design and evaluate ensemble methodologies that aggregate predictions produced by several different models, each utilizing different text encoders; determine under what conditions ensemble-based methodologies provide improved performance compared to individual models.

These objectives are intended to create a complete picture of the computational challenges associated with detecting persuasion techniques in multimodal content.

1.4 Significance of Thesis

This thesis proposes a novel approach that combines in-domain pre-training, gated attention late fusion of multimodal features, and a parameter-efficient fine-tuning method to address these challenges, and makes several significant contributions to the field of multimodal learning and persuasion detection:

1.4.1 Methodological Contributions

- Development of a selective training strategy that improves performance by focusing on truly multimodal samples

- Adaptation of Gated Attention Fusion Networks for persuasion technique detection
- Implementation of a new Technique-Aware sampling for contrastive learning
- Comprehensive evaluation of parameter-efficient adaptation techniques (LoRA) in multimodal settings
- Analysis of how computational constraints affect architectural choice and performance

1.4.2 Empirical Insights

- Demonstration that late fusion approaches can outperform early fusion under resource constraints
- Evidence that smaller, parameter-efficient models can achieve superior performance to larger architectures when properly optimized
- Analysis of the relationship between batch size limitations and model performance across different architectures
- Identification of factors that determine when visual information contributes to persuasion detection

1.4.3 Practical Impact

- Provision of guidelines for multimodal architecture selection under computational constraints
- Development of a modular framework that allows independent optimization of text and visual components
- Creation of replicable methods for parameter-efficient multimodal adaptation

Chapter 2

Related works

2.1 Literature Review

2.1.1 The Multimodal Challenge of Meme Analysis

The fundamental challenge in analyzing memes is its multimodal nature, which includes combination of image and text information. Kiela et al. [17] in their work identified a benchmark for this problem through the Hateful Memes Challenge. The main discovery of this competition is that the models that try to rely only on one of the components (image or text) perform worse rather than models that integrate both modalities. It means that the memes nature relies on the combination of both modalities, rather than on one of the isolated components.

Based on the work of Kiela et al., Lippe et al. [22] developed a multimodal approach that focuses on the understanding of connection between text and image. They found out that memes frequently use irony, sarcasm, and cultural references that require contextual interpretation beyond surface-level content analysis. This complexity is further studied by Ntouvli and Geenen [29], who explore how “ironic memes” function as multimodal texts requiring additional literally analysis to interpret. Their work main point is that memes provide a unique communicative form where meaning is constructed through the play of different semiotic and rhetorical modes.

2.1.2 Technical Approaches to Multimodal Meme Classification

Based on the recent research, some authors used various strategies for solving multimodal classification challenge. One of that strategies was introduced in Radford et al. [31] work, where authors used the CLIP (Contrastive Language-Image Pre-training) technique. This technique is a model that learns transferable visual representations from natural language supervision. This approach is considered as one of the influential ones. Thanks to this work, there is an opportunity to use a shared embedding space for text and images, so there is better reasoning between those two modalities.

Burbi et al. [2] suggested an innovative approach that makes a connection map between memes and words. Their method addresses the semantic gap between visual and textual modalities by creating intermediate representations that facilitate better integration of multimodal information. This technique demonstrates improved performance in identifying hateful content by capturing subtle relationships between image elements and accompanying text.

Hamza et al. [15] emphasized a need to do image-text analysis based on the specific domain of religiously hateful memes. As a result, they developed classification systems based on text-image data. From their work, it is possible to see that it is important to also identify and use domain-specific approaches, because religious hate speech uses similar particular symbols, imagery, and linguistic patterns, so there is a need in detection of those specialized patterns.

2.1.3 Propaganda and Persuasion Detection

While the majority of research papers mainly focused on hate speech, recent work has expanded previous research to rhetorically analyze propaganda and persuasion techniques in memes. Da San Martino et al. [4] provided foundational work on fine-grained propaganda analysis in news articles, developing taxonomies and detection methods for various propaganda techniques. Although the authors main focus was

on texts from the news, their work has also succeeded and provided a valuable start for meme analysis.

Ranjan et al. [33] applied previously done propaganda detection methods to memes. It was done for the dataset consisting of different multilingual contexts. Based on that work, it is possible to see that propaganda in memes often uses a combination of visual with textual information. Another considerable feature is that propaganda inside memes uses language-specific patterns, requiring models that can process universal persuasive mechanisms with cultural understanding.

Recent development of large language models was another significant improvement in detection of propaganda in memes. Hasanain et al. [16] experimented with the usage of GPT-4 model. They tried to effectively detect propaganda in news articles. As a result, their conclusion is that there is a promise in advanced language models. However, there are still obstacles in using those LLMs, since they require careful prompting and validation. Without these two variables it is hard to achieve reliable results in propaganda detection.

2.1.4 Synthesizing Insights: Toward Comprehensive Harmful Content Detection

In the work of Sharma et al. [35], the authors conducted and united approaches to detect and understand harmful memes. This research was done on experimenting with different types of problematic content, such as hate speech, propaganda, and other forms. The main challenges and problems found in this survey are the need for cultural and contextual understanding. Another problem that was found in that research is the difficulty of detecting irony messages and the importance of explaining model decisions to get more effective cultural understanding.

Based on this literature review, it is possible to see a significant improvement in detection of harmful content. The start was from baseline model like binary classification classical methods, developing into more nuanced systems that can understand context between multiple dimensions like image and text. Based on current research,

it is possible to see that only multimodal processing is not enough to get proper results in that challenge. There is also a need in proper understanding of social, cultural, and linguistic context. As a result, more research is needed to focus on sophisticated multimodal approaches that can integrate and understand textual and visual information while taking into account irony and cultural context. Recent advances in that field, especially CLIP and cross-modal attention mechanisms, have significantly improved results of detection capabilities. However, there are still challenges in achieving proper results and performances across diverse cultural contexts, languages, and evolving meme formats. There is still a room for future research that should include more generalized models that can understand and address multilingual and cross-cultural differences. Also, there is a need to create systems that can identify new developed forms of harmful content as meme cultures continue to grow.

Chapter 3

Dataset

3.1 Dataset Overview

For this research, the dataset from the "SemEval-2024 Task 4: Multilingual Detection of Persuasion Techniques in Memes" [7] is used. This dataset consists of memes on different languages and is specifically collected to solve the problem of identifying propaganda and persuasion techniques in image-text modal context. Inside the dataset are four languages: English, Bulgarian, North Macedonian, and Arabic, with majority of data is being English as the most popular one.

The memes were collected from personal Facebook profiles by using scraping services. The majority of memes address controversial subjects, including political issues, vaccination, COVID-19, and so on. Due to lack of the data, some of the dataset was scraped from Instagram. This information is related to the Macedonian and Arabic languages. To ease the task, organizers decided to remove the data and leave has only image based memes with text. Organizers decided not to include memes in cartoon style, and memes without any text on it.

3.2 Task Definitions and Subtasks

To count this task as successful, participants must implement and develop models that can detect persuasion techniques and identify the type used. The task itself

consists of three subtasks.

Subtask 1 – Textual Analysis: Based only on the text of a meme, the model must detect which persuasion techniques are present. This subtask focuses exclusively on linguistic features of a meme.

Subtask 2a – Multimodal Analysis: Participants are provided with a complete meme (both image and text), the model must identify which persuasion techniques was used in both dimensions. This is the representation of multimodal task, so model requires analysis and integration of information from both modalities [17, 22].

Subtask 2b (ST 2b) – Binary Classification: Given a complete meme, the model must determine whether it contains any persuasion technique. This subtask was added to address understanding of the initial presence of propaganda in memes.

3.3 Dataset Statistics and Composition

Table 3.1 presents the statistics for all subtasks across the four languages. It can be seen that the majority of data is in English language since it is easier to obtain that data comparing to othe languages. For Subtask 1, the English subset contains 7,000 training examples, 500 validation examples, 1,000 development examples, and 1,500 test examples, resulting in 10,000 memes. Bulgarian, North Macedonian, and Arabic were introduced as surprise languages in the test set only, with 436, 259, and 100 test examples respectively.

The distribution of Subtask 2a is quite similar to Subtask 1. Training, validation, and development sets have identical test set sizes. On the other hand, Subtask 2b has a smaller subset with 1,200 training examples, 150 validation examples, 300 development examples, and 600 test examples for English. Other languages has 100, 100, and 160 test examples.

Table 3.1: Number of memes for every language on each subtask and associated data splits. Note that only the test split contains all four languages. EN=English, BG=Bulgarian, MK=North Macedonian, AR=Arabic [7]

	EN					BG	MK	AR
Subtask	Train	Val	Dev	Test	Total	Test	Test	Test
ST 1	7,000	500	1,000	1,500	10,000	436	259	100
ST 2a	7,000	500	1,000	1,500	10,000	436	259	120
ST 2b	1,200	150	300	600	2,250	100	100	160

3.4 Persuasion Techniques Distribution

The total dataset has 22 distinct persuasion techniques that are organized in a hierarchical taxonomy. The distribution of such techniques are quite imbalanced, and some techniques were presented more than others. The most popular techniques across Subtask 1 and Subtask 2a is Smears, Loaded Language and Name Calling. As a result more than half of all dataset uses those techniques.

On the other hand, several techniques are very rare and used in less than 1% of the memes. Based on this information, we can see a significant imbalance and challenge for model development. It means that the model should be implemented properly, so it can adequately identify those techniques trained on a very little information. Also, distribution of techniques are different across languages, because some of the languages use some techniques more than other reflecting cultural and linguistic differences.

A notable observation in Subtask 2a is the presence of multimodal-specific techniques such as Transfer and Appeal to (Strong) Emotions, which do not appear in Subtask 1. Transfer, which involves projecting positive or negative qualities from one entity to another, appears 2,286 times in English ST2a but is absent in ST1, indicating that this technique relies heavily on visual elements [29]. Similarly, Appeal to (Strong) Emotions shows 537 instances in English ST2a compared to zero in ST1, demonstrating how emotional manipulation in memes often depends on the interplay between image and text.

3.5 Label Distribution and Multimodal Complexity

For Subtask 2b, the binary classification task, Table 3.2 presents the label distribution across all languages. The dataset contains 1,500 propagandistic memes and 750 non-propagandistic memes in English, resulting in a 2:1 ratio. The other languages have similar ratio: Bulgarian 80 propagandistic to 20 non-propagandistic, North Macedonian with 90 to 10, and Arabic with 113 to 47 respectively.

Table 3.2: Subtask 2b label distribution [7]

Label	EN	BG	MK	AR
propagandistic	1,500	80	90	113
non propagandistic	750	20	10	47
Total	2,250	100	100	160

A notable observation of this dataset is that these memes are not considered as only one persuasion technique only, but combination of them. It can be said that the nature of propagandastic memes lies in the combination of those persuasion techniques [35].

3.6 Dataset Challenges and Implications

The dataset presents several key challenges for model development. First of all, the imbalances of the dataset leads to a need of proper training phase. Without this the model will perform poorly on the data that was not presented a lot in the dataset. Another challenge presented in this dataset is a need of proper multimodal connection. It means that model would require a correct identification of the meaning between both image and text modalities [31, 2]. Another major problem that is present in this dataset is that some of the techniques can only be found in-between those two modalities. For example, Appeal to Emotion technique can only be properly understood when proper analysis of both modalities done. The final challenge is the aspect of different languages, as they introduce own complexity based on different cultural and linguistic differences.

These challenges align with the broader research landscape discussed in the literature review, where effective meme analysis requires not only technical sophistication in multimodal processing but also deep understanding of cultural context, irony, and the interplay between visual and textual rhetorical strategies [35].

Chapter 4

Methodology

4.1 Overview

This methodology section is built around three latter goals.

1. As was discussed previously, the original dataset has an imbalance label distribution and a small size problem. To diminish and solve this problem several data augmentation techniques were introduced. The goal was to improve the end result and explore whether augmentation would improve model performance or not. To make a comparison between the augmented dataset and not, the models were trained on different variations of the augmented datasets for Subtask1, which includes only text-classification task and Subtask2a, which in his turn related to multimodal classification.
2. This research also tries to play and replicate best models from SemEval Task 2024 [7]. The reason behind this is to validate previous research and determine if it is possible to receive similar results under exact or slightly varied conditions. Almost all works from [7] with high F1 score used some variation of data augmentation techniques and enlarged the dataset. So the first and second goal is connected in some way.
3. In comparison with goal number two, this goal is focused on building custom models. The main approach for these models is based on theoretical reasoning,

works submitted and recent updates in multimodal field and sentiment analysis, discussed in works by Jing et al. [12], Yuan et al [41]. This research tries to evaluate failed models which in theory should show higher results. The goal here is to understand reasons behind fail of certain models and whether different changes in the implementation can improve the model and unlock its potential.

It is possible to combine these three subgoals into the larger one - to build a competitive model that performs at the same level as existing models or even better. If this large goal will fail, it still possible to receive helpful insights from working approaches and understand why failing models fail is still valuable. In the end, this research leads to greater insights into the challenges of multimodal classification.

4.2 Experimentation Setup

Before proceeding with the description of all the experimentation conducted in this thesis, it is important to give some background on the environment. All experiments in this thesis were conducted using the free tier of Google Colaboratory (Colab). Colab is a cloud-based Jupyter solution for running Jupyter notebook environment online. All experiments were used with the free version of Colab. This free subscription offers access to GPU resources on a session-based basis, with GPU type and availability determined dynamically at the time of connection. Since the free tier operates on a shared resource model, access to GPUs is not guaranteed and session durations are limited. As a result, some experiments had to be restarted or split across multiple sessions due to runtime disconnections or GPU unavailability during peak usage times. To cope with these problems and constraints, experiments were required adjusting batch size, training duration, and other parameters like weight decay and gradient accumulation to fit within the available session time and memory limits. Also, after every step of the experiments, all models, weight and graphs were saved to not lose the progress due to free tier subscription. Despite the limited flexibility in scheduling and the time-consuming nature of running jobs on a free cloud

platform, the use of Google Colab was necessary due to hardware constraints. Local experimentation was not feasible, as the available personal device lacked the necessary GPU capabilities for efficiently training large-scale multimodal models. As a result, all substantial model development and evaluation had to rely on Colab’s resources, which, while useful, were subject to session time limits, runtime disconnections, and limited availability during high-demand periods. The most commonly available GPU type during experimentation was the NVIDIA Tesla T4 (see more details in Table 3.1). The T4 belongs to the Turing architecture and supports CUDA 12. It was accessed through Colab’s runtime allocation system, typically under moderate to high demand.

Environment	GPU Model	GPU Memory	Total System RAM
Google Colab (free)	NVIDIA Tesla T4	15 GB	~12.7 GB

Table 4.1: GPU resources available on Google Colab free tier

4.3 Data Augmentation

One of the first and most important challenges that should be processed accordingly is imbalance in label distribution. Some persuasion techniques occur more frequently (Smears, Loaded Language), while others have a few examples (Confusion). As a result this label distribution imbalance leads to poor model performance, especially at detecting labels that were rarely seen during the training part. To address and solve this problem the next four augmentation strategies were introduced: augmentation using the nlpaug library for creating new samples for underrepresented labels, pseudo-labeling, GPT-4-o-mini paraphrasing and creating captions for images with GPT-4-o-mini.

4.3.1 Reasoning behind choosing data augmentation techniques

nlpaug is a python library that has inside it 16 different text augmentation techniques with different bases. The nlpaug example of some techniques could be seen in the

figure below.

	Sentence
Original	The quick brown fox jumps over the lazy dog
Synonym (PPDB)	The quick brown fox climbs over the lazy dog
Word Embeddings (word2vec)	The easy brown fox jumps over the lazy dog
Contextual Word Embeddings (BERT)	Little quick brown fox jumps over the lazy dog
PPDB + word2vec + BERT	Little easy brown fox climbs over the lazy dog

Figure 4-1: Textual Data Augmentation by nlpaug

To reserve the meaning of the text and reduce the risk of adding layered meanings and increased complexity the augmentation technique was chosen with caution, only incorporating Synonym Augmentation for Subtask1. nlpaug library’s main concept is to substitute words to its synonyms using WordNet/PPDB databases. The main reason why this library was chosen because it is commonly used as a reliable dataset augmentation technique for addressing label imbalance [1] [37] [39] [32] [27] [11]. Another approach to increase lable diversity without additional noise is using LLM GPT-4o-mini model. This LLM was used for parahprasing and caption generation. Before text paraphrasing, language model was provided with the needed prompts which will be discovered in the next subsection. The main idea was to preserve meaning and rhetorical framing, while introducing some variations of the same meaning. For Subtask2a, which includes multimodality, caption generation will help with adding additional context for the model, because some images carried strong meaning, while associated text for this image was ambiguous. The main reason of choosing GPT-4o-mini is its strong language capabilities for low cost. Pseudo-labeling was also used because it allowed expansion of the training set, especially for under-represented persuasion labels, while keeping the pipeline lightweight and automated.

4.3.2 Selecting Labels to Augment

Starting from the nlpaug augmentation, it was decided that if the persuasion technique was presented less than 1000 times in the dataset, it was considered as under-represented. This number seems reasonable since less than 1000 occurrences refers

to more than 75% of labels in Subtask1. Samples that contain at least one of underrepresented labels were selected for being augmented to improve class imbalance. In contrast, GPT-4o-mini paraphrasing used another threshold for augmentation. In that case label frequency of 16% was used. The reason behind that decision is that all labels except "Smears" fall for this category. This allowed for a broader range of rare classes to be considered during augmentation, particularly benefiting multi-label scenarios where some labels co-occurred with more frequent ones.

4.3.3 Augmentation Using NLPAug

As it was discussed before nlpaug [24] library was used on samples with less than 1000 label occurrences. To be more specific, masked language model (MLM) was used to preserve the context and meaning with proper alternatives. It helps to expand the dataset. From the total of 7000 entries in the train set, 4029 entries were augmented because they were from the labels low than threshold value. From the total of 500 samples in the validation set, 412 samples were augmented since they were from below thresholds labels. Combining these two values, 4441 samples were added by the result of augmentation.

4.3.4 Pseudolabeling

Pseudo-labeling technique was used to additionally enrich the dataset. A total of 1,000 samples from the development set and 1,500 from the test set were classified using the trained model, and their predicted labels were added as a new pseudo-labeled dataset. This technique is considered as an effective way to use in semi-supervised learning, especially in the limited or unbalanced dataset. This method introduces additional training signals that help refine decision boundaries and improve generalization across low-resource labels [18].

In overall, 19,890 samples were generated, of which 18,642 for the train set and 1,248 for the validation set. Additionally, to prevent memorization of augmentation patterns, the final dataset was shuffled. As a result, final dataset has more balanced

label representation and has improved model’s robustness by providing a wider variation of different techniques.

4.4 Replication of the baseline models

To establish a reliable baseline and better understand what makes a high performing system, the experiments focused on replicating several existing solutions - which happened to be among the top-ranking solutions from the SemEval-2024 Task 4 competition.

4.4.1 Subtask 1: Ensemble Learning with Chain-of-Thought Data Augmentation

The first experiment was replicating the work of the team "914isthebest" [14] with slight variations in accordance to the model updates, which will be explained later. Their approach was chosen not only because it ranked first in Subtask 1 (text-only classification), but also because it outperformed the second-best submission by a significant margin, and it encompassed many of the methods that were concluded to be used by many teams. Their work was clearly described, well-structured, and parts of it were publicly available, making it a solid starting point for deeper exploration.

The original system treated the problem as a multi-label classification task and ignored the hierarchical structure of the labels, since all training samples were annotated with final (leaf) nodes. The authors developed an ensemble-based architecture that combined one DeBERTa-v3-large model and four RoBERTa-base models, each trained with different random seeds. Their ensemble used soft voting, averaging the prediction probabilities from all models and applying a fixed threshold (0.25) for label selection. A key part of their success was the integration of three strategies: in-domain pretraining, extensive data augmentation, and ensemble learning. Each component was tested and shown to improve performance. The in-domain pretraining involved masked language modeling (MLM) using a combination of official data

and additional related datasets, such as those from previous SemEval tasks. This allowed the models to adapt more closely to the domain-specific language used in meme texts and persuasive discourse.

Data augmentation played a critical role in the training pipeline. Two methods were used: (1) synonym replacement using the `nlpaug` library to increase data diversity for underrepresented labels, and (2) a more advanced augmentation using Chain-of-Thought (CoT) prompting with GPT-3.5. The CoT approach was applied to samples associated with rare labels, using an LLM to generate paraphrased versions of the input text. This helped maintain semantic consistency while introducing structural variation in the inputs. In total, over 10,000 augmented samples were generated using this technique.

In addition to augmentation, pseudo-labeling was used to extract extra training samples. The team applied a trained model to the development and test splits, selecting high-confidence predictions as pseudo-labels. These new samples were then merged with the augmented dataset to form richer training inputs.

To replicate this system, the same base models (RoBERTa-base and DeBERTa-v3-large) were used, and ensemble predictions were averaged across multiple seeds. RoBERTa-base was trained on four different seeds: 42, 3407, 114514, 4096, and DeBERTa-v3-large on seed 1234. The training process followed the original setup, with a batch size of 16, a learning rate of $3e-5$, and a cosine learning rate scheduler with warmup steps. The AdamW optimizer was used, and models were trained for up to 7 epochs. Tokenization and maximum sequence length settings were aligned with the original implementation.

Following a similar logic, six custom datasets were constructed:

- *GPTDataset*: in this case, texts were augmented with CoT prompting using GPT4o-mini instead of GPT-3.5. According to OpenAI [30], GPT4o-mini surpasses all other small models in the market (at the time of release of the article) in terms of reasoning, and despite GPT-3.5 being larger (17B parameters), since GPT4o-mini is part of a new 40 architecture, it is often more optimized and better performing, in addition to being cheaper to run. Currently, it is one of

the models that best balances performance with cost-efficiency.

- *CaptionDataset*: images were fed into a GPT-4.1mini, which was released during the work on the thesis, and with better context understanding, generated new samples with texts that include the visual context of the meme, as well as the textual part.
- *NLPAugDataset*: similar synonym-based paraphrase technique was used for augmentation.
- *PseudoDataset*: unlabeled dev/test data labeled by a trained model were added into the training pipeline.
- *NGPTDataset*: union of *nlpaug* and GPT-paraphrased augmented samples.
- *NGPTCaptDataset*: union of *nlpaug* and GPT-based augmented samples.
- *PNGPTCaptDataset*: union of the *PseudoDataset* and *NGPTCaptDataset*.

According to the original pipeline, the first step was in-domain pretraining using MLM (development set was used for this), fine-tuning, and finally training with augmented datasets. By carefully reproducing each part of this pipeline, including architecture, data processing, training strategy, and augmentation methods, it was possible to build a system comparable in structure to the original. The replication was used to serve as an initial point of reference from which we could experiment with other model alternatives and new data augmentation techniques

The results of the ensemble learning can be found in Table 3.2.

Discussion of Data Augmentation Techniques

Replication of "914isthebest" demonstrates how data augmentations can significantly enhance the performance of multi-label classification tasks. Figure 3.1 illustrates how each type of augmentation strategy, when paired together, has an effect on performance level.

Model	Hierarchical F1	Hier. Precision	Hier. Recall
<i>Original Dataset</i>	0.63420	0.66210	0.60856
<i>CaptionDataset</i>	0.62866	0.56040	0.71586
<i>NGPTDataset</i>	0.65614	0.65255	0.65976
<i>NGPTCaptDataset</i>	0.70382	0.70186	0.70579
<i>PNGPTCaptDataset</i>	0.70423	0.70197	0.70649
914isthebest	0.752	0.684	0.836

Table 4.2: Comparison of ensemble model performance with various pretraining datasets (augmentation techniques)

Data Augmentations lead to an incremental increase in model performance. The *CaptionDataset* uses generated descriptions via GPT-4.1-mini to add context to the images, yielding a Hierarchical F1 Score of 0.62866. Although, it is worth noting that the *CaptionDataset* yields a poorer result than the original dataset for two reasons; First, there is an increased amount of sentiment that may arise from images that were not considered during the labeling/annotation process as described by SemEval Task 6 [6]; Second, the *NGPTDataset*, created using a combination of the two datasets because of the paraphrasing characteristics of both datasets, utilized the newer GPT-4o-mini model to perform semantic paraphrasing ("Chain-of-Thought") instead of utilizing the same model originally used in "914is-thebest", thus increasing the performance to 0.65614. This increase supports the use of the GPT-4o-mini over the GPT-3.5 model, which although had less parameters, utilizes its optimized 40 architecture and better reasoning abilities.

The greatest increases in performance are seen when using multiple approaches for augmenting data. The *NGPTCaptDataset*, utilizing both nlpaug based on synonyms replacement, and GPT based on paraphrasing, along with caption based augmentation is able to generate an average F1 score of 0.70382. A 4.7 percent increase above the 0.65614 *NGPTdataset* shows how important it is to include the image as a part of the data being used for training the classification model.

The inclusion of pseudo-labeled examples within the *PNGPTCaptDataset* resulted in a somewhat, yet consistently, improved performance for each of the three measures. Specifically, the F1 score increased by approximately 0.04 from 0.7012 to 0.70423. The modest but measurable improvement suggests that when the ensemble model

generates confident predictions about unlabeled instances, those may be used as supplementary signals to improve its training.

In addition to this, the best performing configurations (*NGPTCaptDataset* and *PNGPTCaptDataset*) produce well-balanced values of precision and recall, each of which are approximately 0.7. These balanced values suggest that the augmentation methods effectively addressed both false positives and false negatives, as opposed to focusing on improving one of these metrics at the expense of the other.

As such, the results show that combining GPT-4o mini (as an advanced language model for augmentation), visual captions (multimodality), and using two classic techniques (synonym substitution and pseudo labeling), creates a synergistic effect that is significantly better than each method alone. The results therefore indicate that augmentation techniques were successful in addressing the problem of too little training data for this thesis.

Discussion of Results

The original “914isthebest” team achieved first place on the English test set with a hierarchical F1 score of 0.752, while the replication achieved a maximum F1 score of 0.704 with the *PNGPTCaptDataset* configuration. This represents a performance gap of approximately 4.8 percentage points, which, taking into account access to the same competition data and evaluation procedures, points to specific implementation differences rather than fundamental methodological issues.

A primary factor contributing to the performance difference appears to be the scale of CoT-based data augmentation. The original team generated 10,000 samples using GPT-3.5 with Chain-of-Thought prompting, while in this work 5,113 samples were augmented, and hence, generated with GPT-4o-mini. This 50% reduction in the volume of augmented data probably had a substantial impact on the performance of the model, particularly given the multi-label nature of the task and the inherent class imbalance.

In multi-label classification scenarios, where some persuasion techniques are significantly underrepresented in the training data, the volume of augmented samples

can be crucial for enabling the model to learn robust representations of rare classes. Although the filtering condition to choose which labels to augment using CoT was similar (the original paper specifically noted that they filtered data corresponding to labels occupying less than 0.16 of the entire label distribution), the reason for such a large difference in augmented samples remains difficult to explain. The original paper provides insufficient detail about their specific augmentation procedures to account for this discrepancy, leaving the question for such sample difference open. By generating only half the number of CoT samples, this replication may have left several minority classes with insufficient training examples, directly impacting the hierarchical F1 score.

Ensemble seed selection may also play a role. Although the same random seeds (42, 3407, 114514, 4096 for RoBERTa-base and 1234 for DeBERTa-v3-large) were used, the specific combination and weighting of these seeds in the final ensemble could influence results. The original team’s iterative optimization process during competition may have identified optimal seed combinations that wasn’t replicated exactly.

Although there are some constraints on this replication, it does show that the key ideas of the original approach were valid. As stated in the previous subsection (3.4.1), demonstrating a progressive relationship between the size and complexity of the dataset used for testing shows that the individual contributions to performance from all the methods described previously (i.e., augmentations) was significant; further, when compared to the use of only the *NGPTDataset*, the multimodal caption based method provided an especially large increase (approximately 4.7%) in model performance.

The insight into sensitivity to volume of semantic augmentation from this replication is very useful for next steps. That the highest ROI will be obtained through large amounts of high quality augmentations as opposed to many low quality augmentations highlights a trade-off of using fewer high quality augmentations versus many lower quality augmentations. Also, the results show how important it is to have enough diverse training data when performing multi-label classifications with an underlying

hierarchy or where there are imbalances in the amount of data available per class.

In conclusion, while the replication achieved strong performance (0.70423 F1), the 5-point gap from the original results appears primarily because of the reduced number of samples augmented by CoT, with a potential secondary reason coming from model substitutions and implementation details. The core methodological insights from the original work remain valid and transferable to this work, but achieving state-of-the-art performance requires careful attention to volume of the augmentations and implementation with exactly the same input variables (in terms of environment).

4.4.2 Subtask 2a: Hierarchical Multilabel Classification Turned into Hierarchy-Aware Prompt-Tuning Task

Another, completely different, direction explored in this thesis was based on the method proposed by team “HierarchyEverywhere” at SemEval-2024 Task 4, which introduced a Hierarchical Prompt Tuning (HPT) strategy for textual persuasion technique classification (Ghahroodi and Asgari [9]). Although their full system implementation was not released, the underlying method was described in sufficient detail and was mostly based on an existing open-source implementation of HPT, which is accepted to be the last state-of-the-art (SOTA) method of handling Hierarchical Text Classification by Wang et al. (2022) [40]. That version served as the foundation, while the rest of the pipeline—prompt design, dataset integration, and training logic—was implemented independently.

In the case of persuasion technique detection, the 22 fine-grained techniques are organized under three high-level rhetorical categories: ethos, pathos, and logos. Rather than treating these labels as independent, HPT aims to incorporate this structure into the learning process. This can potentially improve performance and interpretability, as it encourages the model to reason through the hierarchy rather than making isolated predictions.

A critical preprocessing step involved converting the original Directed Acyclic Graph (DAG) structure of persuasion techniques into a hierarchical tree format re-

quired by the HPT model. This transformation addressed nodes with multiple parents (like “Whataboutism” with parents “Distraction” and “Ad hominem”) by creating duplicate nodes with combined labels (“Distraction_Whataboutism” and “Ad hominem_Whataboutism”). The team found that organizing the hierarchy with “propagandistic” and “non-propagandistic” nodes at the root level yielded superior performance.

The system incorporated additional training data from two supplementary datasets: persuasion detection in news articles (Da San Martino et al., 2019) [4] and previous SemEval 2021 meme persuasion detection task 6. While the news article data proved less effective due to domain differences, the meme dataset from SemEval-2021 (Dimitrov et al., 2021) [6] enhanced model accuracy. The team also implemented a sentence boundary handling approach using [CLS] tokens between sentences in multi-sentence memes, which improved performance.

It is worth mentioning that for the multimodal subtasks (2a and 2b), the system intentionally ignored image content and focused exclusively on textual features. This counterintuitive decision led to first-place performance in English Subtask 2a, suggesting that textual content alone was sufficient for effective persuasion detection in memes.

The model was trained on the PNGPTCapt dataset, which includes both paraphrase-augmented and pseudo-labeled data, providing it with both high-coverage and diverse examples. Despite not having access to the exact architecture used in the original submission, this reimplementation achieved strong results, which can be seen in Table 3.3. These results demonstrate the potential of hierarchy-aware prompting strategies and suggest that incorporating task structure explicitly can lead to meaningful improvements over baseline flat classification.

Team	Hier. F1	Hier. Precision	Hier. Recall
HierarchyEverywhere	0.74592	0.86682	0.65461
Thesis (w/o pretrain)	0.73183	0.77631	0.69217

Table 4.3: Comparison of metrics for HPT for persuasion technique detection.

The slight difference in the result is most probably due to the absence of pre-

training on the previous SemEval 2021 dataset, as the goal in this replication was to test the effectiveness of the hierarchical text classification approach and validate whether ignoring the image component in multimodal persuasion detection truly gives superior performance.

By deliberately excluding visual information in the multimodal task, the replication aimed to empirically validate the HierarchyEverywhere team’s counterintuitive finding that textual content alone could outperform multimodal approaches in meme persuasion detection. The results support this hypothesis, achieving competitive performance (**0.73183 F1**) compared to the original work (**0.74592 F1**), suggesting that the hierarchical text classification paradigm can effectively capture the essential persuasion techniques without requiring complex image-text fusion techniques. This finding reinforces the notion that persuasion techniques in memes are primarily conveyed through textual elements, with the hierarchical structure providing sufficient semantic relationships to distinguish between different persuasive strategies.

4.5 Novel Architectures and Experimental Extensions

Building upon the comparative analysis of existing approaches, the replication of some of the important submissions, analysis of several theoretically promising approaches that failed to achieve their expected performance levels, and being informed by recent developments in multimodal content analysis, a series of custom model architectures and training strategies are presented in this section. These experimental approaches were designed to address limitations observed in competition submissions while leveraging theoretical insights from the broader literature.

4.5.1 Attention-based Fusion for Multi-modal Tasks

The larger idea behind the first several architectures lies within the idea of various ways of using image-text features as input. Specifically, this part of the work explores how it is possible to combine visual and textual representations for persuasion technique detection, focusing on how both feature-level late fusion and early fusion

impact downstream classification performance. Inspired by the success of attention-based architectures and modular multimodal models like CLIP, the experiments in this section test how image and text embeddings can be fused using various attention mechanisms. The goal is to assess how different strategies for encoding and integrating modalities affect the model’s ability to understand and classify complex multimodal content.

The attention mechanism had improved the performance of multi-modal models by a significant degree [42, 19, 3], since it is more flexible, precise, and better in capturing the image-text correlations, when dealing with image and text processing. This mechanism allows models to be focused on certain important parts of the images, and generate highly relevant text descriptions [13, 23, 28].

The proposal of the Transformer model (2017 [38]) allowed the multi-modal models to develop significantly faster. Unlike traditional RNNs or CNNs, which rely on loop or convolution operators, Transformers mainly use self-attention mechanisms. This enables more efficient processing of long sequences. Therefore, Transformers offer better scalability through parallel computation. As a result, Transformer-based models have become a dominant choice in both language understanding and cross-modal tasks, showing substantial improvements in the performance of multimodal systems that combine image and text data.

Transformers are also often used to integrate image and text inputs in various ways. In single-stream architectures like VisualBERT [21], VL-BERT [36], or Unicoder-VL [20], visual features are inserted into the token sequence processed by a single transformer. This allows early fusion of the modalities at the input level. On the other hand, dual-stream vision-language models — like CLIP [31] — process images and text independently using two separate encoders. Each encoder maps its respective modality into a shared embedding space, and the final interaction typically occurs through a similarity measure like cosine similarity.

Building on this context, the approach adopted in this thesis focuses on exploring two primary strategies for fusing visual and textual features, motivated by coming to a conclusion about the effectiveness of attention-based architectures and the perfor-

mance of different feature-level integrations (early and late fusion). The first strategy implements a late fusion approach: pre-extracted image features from CLIP are combined with text features from four different large-scale pretrained language models: CLIP-text, DeBERTa-v3-large, PoliticalBiasBERT, and XLM-Twitter-Politics. This setup allows for a diverse representation of the textual content, capturing both general and domain-specific linguistic patterns.

Late fusion was chosen for two reasons: first, it is computationally more efficient than early fusion, allowing independent feature extraction and reuse; and second, it reflects common design choices observed in several high-performing systems submitted to the SemEval-2024 task. Once features are extracted, they are fused using self-attention and cross-modal attention layers. Self-attention allows each modality to refine its own internal representations, while cross-modal attention enables interaction between the visual and textual streams. This dual attention setup provides a mechanism for the model to focus on relevant information both within and across modalities, improving its ability to make fine-grained, label-sensitive predictions.

The goal of this experimental setup is to evaluate how well attention-based late fusion — combined with diverse text encoders and CLIP-based visual embeddings — can support hierarchical multilabel classification in the persuasion technique detection task.

Late Fusion with Self and Cross-Modal Attention: Architecture and Training Workflow

This system’s architecture centers around a late fusion approach to flexibly integrate multimodal features from both text and images. Rather than feeding raw data directly into a single model, the pipeline starts with separately encoding the text and image data using pretrained backbone models, since this is computationally and memory-wise very efficient. Image features are extracted using CLIP and CLIP-Large, while text features are generated by four different language models: CLIP-text, DeBERTa-v3-large, PoliticalBiasBERT, and XLM-Twitter-Politics. These encoded features are precomputed and stored, which allows for efficient downstream processing and easy

experimentation with different fusion strategies.

Once feature extraction is complete, the stored embeddings are fed into a late fusion module. The model begins by projecting the input image and text features into a shared embedding space of dimension (768) using two linear layers. Each modality is then refined through self-attention modules. A cross-modal attention module further integrates the modalities, producing fused representations. Additionally, a persuasion-guided attention mechanism is employed to extract both unified and technique-specific features relevant to classification.

This attention mechanism employs learnable query vectors, one for each persuasion technique, which attend to different regions of the input feature space through scaled dot-product attention. Each technique query computes attention scores across the input features, generating representations that capture the most relevant patterns for detecting that particular persuasion technique. This approach allows different persuasion techniques to focus on distinct aspects of the input data. The architecture returns both the unified representation — which is obtained by transforming individual technique-specific features through a linear projection with layer normalization and ReLU activation — and the individual technique-specific features.

The fusion layer concatenates the cross-modal features, text self-attention output, and unified persuasion features, then processes them through a linear layer followed by layer normalization, ReLU activation, and dropout. This structure enables the model to learn interactions not only within each modality but also across them, giving it the capacity to align image cues with textual context in a flexible way. By decoupling feature extraction from fusion, the system remains modular and easy to extend or adapt for other encoder choices.

Model training follows a two-phase process: pretraining and fine-tuning. For pretraining, the late-fusion model is trained on both the original train and validation data. Hyperparameter optimization is performed for each base model using Optuna, which is an efficient framework for automated search, to select the best learning rates, dropout rates, and hidden dimensions. This ensures that each fusion variant is well-adapted to the data and maximizes its initial learning capacity. Through a series of

experiments it was found that the best seed for CLIP-text is 1234, for DeBERTa-v3-large and XLM-Twitter-Politics it is 42, and for PoliticalBiasBERT it is 4096. The hyperparameters chosen for these seeds during training can be seen in Table 4.4. After pretraining, each model undergoes a fine-tuning step, where it is further trained using labeled task data with carefully selected hyperparameters. Fine-tuning is performed separately for each fusion model, enabling them to specialize based on the strengths of their respective text encoders and the multimodal context.

Table 4.4: Hyperparameters selected by Optuna for models

Model	Hidden Dims	Dropout	Learning Rate	Weight Decay
CLIP-text	[512, 512]	0.25448	5.6e-4	0.338
XLM-Twitter-Politics	[1024, 512]	0.46571	4.1e-4	0.381
PoliticalBiasBERT	[1024, 512]	0.06944	7.6e-4	0.369
DeBERTa-v3-large	[512, 512]	0.06336	3.7e-4	0.480

Table 4.5: Performance of individual models and ensemble on the test set

Model	F1	Precision	Recall
CLIP-text	0.34512	0.29122	0.46442
XLM-Twitter-Politics	0.35023	0.31418	0.49602
PoliticalBiasBERT	0.35907	0.35309	0.47443
DeBERTa-v3-large	0.37332	0.34010	0.48856
Ensemble (Base Fusion)	0.39627	0.32233	0.51423
Baseline	0.447	0.688	0.331

For the final predictions, an ensemble method is employed. This choice was reasoned from the previous success of the ensemble method during the replication of models and the ability to get the best out of all models. The outputs from each fine-tuned fusion model are combined — typically by averaging their predicted probabilities — to generate a consensus prediction for each sample. This ensemble strategy helps mitigate individual model biases and generally leads to a more robust overall performance.

This multi-step workflow — late feature fusion, Optuna-driven pretraining, fine-tuning, and ensemble prediction — forms the backbone of the multimodal classification system developed in this thesis. The architecture was chosen for its flexibility, efficiency, and proven effectiveness in handling complex multimodal data.

However, the results (see Table 4.5) were unable to compete with any other results, even the baseline; the ensemble fusion achieved a Hierarchical F1 score of 0.396, which is considerably lower than 0.447 (baseline). To build on this idea, it was decided to find a better-suited fusion technique and evaluate if it improves the performance.

4.5.2 Gated Fusion Architecture and Training

Building on the late fusion setup described earlier, the next logical step was to address its limitations by introducing a more structured fusion mechanism. The basic late fusion through self-attention, cross-modal and persuasion attention showed poorer results than expected, since it lacked a way to control which image features were truly useful and how they should influence the text; the number of layers in the classification head was also very small, including the fusion layers (15). This became especially important when some images were irrelevant or gave less information than the text, which is possible (or even common) in the dataset. A solution to this was found in the Gated Attention Fusion Network (GAFN), introduced by Du et al. (2022) [8] for multimodal sentiment classification. Their method integrates fine-grained image features and enhances the text representation using both attention and a gating mechanism, offering a more refined and noise-resilient fusion strategy (see more details about the structure of Gated Attention Fusion in Subsection 4.5.2).

The rest of the pipeline (pretraining and finetuning) remained the same. Hyperparameters were again chosen by Optuna. This time the main difference was that CLIP-text had hidden dimensions of (1024, 512), while the rest of the three models had (512, 512).

This fusion technique improved the scores, as can be seen from Table 4.6.

Table 4.6: Gated Fusion Metrics			
Model	F1	Precision	Recall
Base Fusion	0.39627	0.32233	0.51423
Gated Fusion	0.44352	0.42545	0.46319
Baseline	0.447	0.688	0.331

Discussion and Summary

The results in Table 4.6 demonstrate that the gated attention fusion technique offers a clear improvement over the initial base fusion approach. Specifically, it improves the F1 score from 0.39627 to 0.44352, reflecting a better balance between precision and recall. Unlike the baseline model, which shows high precision (0.688) but low recall (0.331), the gated fusion model maintains a more stable and reliable trade-off (precision: 0.42545, recall: 0.46319). This suggests that the gated attention mechanism helps the model become more consistent across classes, avoiding overconfidence in fewer predictions and instead capturing a broader set of correct labels. Overall, gated fusion enhances robustness and represents a meaningful step forward in effective multimodal feature integration.

However, despite these improvements, the overall performance remains modest. One of the key limitations in this stage of experimentation was the relatively shallow architecture used during training, which may have restricted the model’s capacity to capture deeper cross-modal interactions. To address this, the next step involved using a pretrained language model backbone that would allow building upon the current late fusion pipeline.

4.5.3 Low-Rank Adaptation (*LoRA*) with Gated Fusion and Larger Text Model Backbone

While the previous gated fusion architecture demonstrated significant gains over the base ensemble fusion approach, its overall performance was still worse than the baseline and it was limited by the depth and capacity of the models used. The gated attention mechanism clearly showed potential in improving the quality of multimodal feature integration, but a larger model was necessary to further evaluate its capabilities. To address this, the next stage of experimentation focused on scaling up the architecture in a parameter-efficient way by integrating TinyLLaMA with Low-Rank Adaptation (LoRA).

Architecture

To test out this direction, a larger language model backbone was integrated with the gated fusion architecture. Although TinyLLaMA was initially selected due to its efficiency, it turned out to be a practical and justifiable choice. Given its smaller size, TinyLLaMA allowed for rapid prototyping and experimentation, especially when combined with LoRA (Low-Rank Adaptation). LoRA enables parameter-efficient fine-tuning by injecting trainable rank-decomposed matrices into existing attention layers, reducing the number of updated parameters (by 10,000 times) while preserving performance. This combination provided a flexible way to scale experiments without requiring extensive compute, making it a rational intermediate step before moving to heavier multimodal models.

Before integrating TinyLLaMA into the gated fusion framework, the small language model undergoes in-domain pretraining to adapt it to the patterns and vocabulary characteristic of the input data. This pretraining phase uses the PNGPT-CaptDataset — the comprehensive augmented dataset that combines NLPAug-based synonym replacement, GPT-4o-mini generated paraphrases, caption-based augmentation, and pseudo-labeled samples; more details about the dataset and augmentation techniques are available in Section 4.3 and Subsection 4.4.1. The model is pretrained using masked language modeling objectives with LoRA adaptation, allowing it to develop robust representations of persuasive language patterns while maintaining parameter efficiency. This domain adaptation step ensures that the language backbone is well-suited to the specific vocabulary and rhetorical structures present in the persuasion detection task before being integrated into the multimodal architecture.

To integrate the LoRA-tuned backbone into the gated fusion framework, the training process was configured for stable and reproducible experimentation. The training loop used a batch size of 16 and a learning rate of 1×10^{-4} , which proved effective for convergence without overfitting. The model was trained for up to 20 epochs using early stopping based on validation F1 score. This configuration provided a balance between training time and model stability, especially considering the constraints of

shared GPU resources on the cluster.

In the gated fusion architecture, a small but impactful modification was introduced to the final classification component. The original classifier was replaced with a deeper feedforward head, designed to better leverage the fused multimodal representations. The new classification head can be seen in Listing 4.1.

Listing 4.1: New classification head for LoRA

```
self.multimodal_classifier = nn.Sequential(
    nn.Dropout(0.3),
    nn.Linear(fusion_dim, fusion_dim // 2),
    nn.ReLU(),
    nn.Dropout(0.2),
    nn.Linear(fusion_dim // 2, num_labels)
)
```

This architecture introduces an additional intermediate layer with ReLU activation and dropout regularization before producing the final logits. The added depth allows the model to better learn nonlinear interactions between the gated image-text fusion features. The dropout layers (with probabilities of 0.3 and 0.2) help mitigate overfitting, particularly in the low-data regime characteristic of the persuasion classification task.

Combining LoRA fine-tuning with this adjusted classifier and gated attention fusion enabled the model to capture both domain-specific language patterns and nuanced visual-textual alignments. It also helped scale the model capacity without significantly increasing the number of trainable parameters, aligning well with the constraints of the available hardware.

In the first experiments the following LoRA hyperparameters were used:

- `lora_r = 16`: This sets the rank of the low-rank decomposition. A moderate value of 16 was chosen to balance expressive power and training efficiency.
- `lora_alpha = 32`: This scaling factor controls how much influence the low-rank

updates have. A value of 32 provides enough capacity for adaptation without overwhelming the base model.

- `lora_dropout = 0.1`: A dropout rate of 0.1 was applied to the LoRA layers to reduce overfitting and improve generalization.
- LoRA was applied to the following `target_modules`: `["q_proj", "k_proj", "v_proj", "o_proj", "gate_proj", "up_proj", "down_proj"]`, which cover the attention and feedforward projection layers of the transformer. These modules were selected because they account for the bulk of computation and learning capacity in transformer-based architectures.
- The `bias` parameter was set to `"none"` to avoid learning biases separately from the LoRA parameters, and the `task_type` was set to `TaskType.SEQ_CLS`, due to the sequence classification nature of the persuasion technique detection task.

Summary

Table 4.7 shows the results obtained by training the LoRA-enhanced gated fusion model with textual encodings from four models (CLIP-text, DeBERTa-v3-large, XLM-Twitter-Politics, and PoliticalBiasBERT) and CLIP-Base image encodings for 5 and 20 epochs.

Table 4.7: LoRA-enhanced gated fusion on TinyLLaMA

Training Epochs	Hier. F1	Hier. Precision	Hier. Recall
5	0.66692	0.81094	0.56634
20	0.68977	0.76458	0.62829

The results indicate that while precision was highest after 5 epochs (0.811), the model continued to improve in F1 (from 0.667 to 0.690) and recall (from 0.566 to 0.628) with extended training. This suggests that LoRA allowed the model to gradually refine its understanding of complex multimodal features over time without overfitting, validating its suitability for this task.

Importance of Image Embeddings and Multimodal Nature of Samples

After validating the effectiveness of LoRA with the gated fusion model using CLIP-Base image features, the next step was to investigate whether upgrading to CLIP-Large could further improve performance. CLIP-Large is a larger variant of the original CLIP model, with more transformer layers and parameters in both the image and text encoders. More specifically:

- **CLIP-Base (ViT-B/32)** consists of a 12-layer Transformer with a hidden size of 512 for text and 768 for vision. It produces 512-dimensional embeddings and contains roughly 63 million parameters in its visual encoder.
- **CLIP-Large (ViT-L/14)** uses a 24-layer Transformer with a hidden size of 768 for text and 1024 for vision. It produces 768-dimensional embeddings and its visual encoder alone has over 300 million parameters.

This increase in representational capacity enables CLIP-Large to model more fine-grained semantic correspondences between images and text. Given its stronger performance on zero-shot vision-language tasks, it was hypothesized that using CLIP-Large for image encoding in the gated fusion setup could yield richer fused representations and boost downstream classification performance.

The experimental setup remained identical to the previous runs: the same gated attention mechanism, classification head, and LoRA-enhanced language model were used. Only the image features were changed, replacing CLIP-Base embeddings with those from CLIP-Large. The model was again trained for 20 epochs with LoRA applied to the same transformer modules and achieved the highest F1 score of 0.704.

The performance results are summarized in Table 4.8.

Table 4.8: Comparison of CLIP-Base and CLIP-Large as image encoders in gated fusion with LoRA

Image Encoder	Hier. F1	Hier. Precision	Hier. Recall
CLIP-Base + LoRA (Epoch 20)	0.68977	0.76458	0.62829
CLIP-Large + LoRA (Epoch 5)	0.68762	0.77844	0.61577
CLIP-Large + LoRA (Epoch 20)	0.70393	0.77396	0.64552

Additionally, to better understand the role of visual information in persuasion technique detection, an additional set of experiments was conducted to test how the model performs when trained on different subsets of the dataset — specifically, based on whether a meme requires multimodal analysis or not. A curated list of sample IDs was available to identify memes that genuinely depend on both text and image for correct interpretation. This enabled an investigation into how much visual information contributes when selectively included or excluded from training.

Four distinct configurations were tested:

1. **Only use samples requiring multimodal analysis:** The model was trained solely on examples where the image plays a critical role in understanding the persuasive technique. This setup was expected to benefit the most from image-text fusion.
2. **Only use samples that do not require image information:** These samples were effectively text-only, allowing the model to rely entirely on the textual encoder and gating with minimal impact from visual features.
3. **Use all samples, regardless of modality relevance:** This was the default approach in earlier experiments and served as a baseline for comparison.
4. **Ignore image features entirely during training:** Here, the model operated as a text-only classifier, with the image branch of the gated fusion module disabled.

Table 4.9 summarizes the results of these training strategies, using CLIP-Large as the image encoder and the LoRA-enhanced gated fusion model, all trained for 20 epochs with hyperparameters given in detail earlier.

Discussion and Summary

In conclusion of this investigation, it is worth stating some of the findings. CLIP-Large offers a deeper architecture, higher embedding dimensionality, and greater parameter

Table 4.9: Performance under different training strategies based on multimodal dependency

Training Strategy	Hier. F1	Hier. Precision	Hier. Recall
Multimodal-only samples	0.70393	0.77396	0.64552
Non-multimodal-only samples	0.69185	0.76417	0.63204
All samples	0.68451	0.76870	0.61695
Text-only (no images)	0.67016	0.74606	0.60828

capacity, all of which contribute to more expressive and semantically rich visual representations. As can be seen in Table 4.8, models using CLIP-Large achieved higher F1 and recall scores, which is why it was chosen as the default visual encoder in subsequent experiments.

A closer look at the experiments summarized in Table 4.9 reveals that the model’s performance is closely tied to how and when visual information is used during training. Training only on samples that explicitly require multimodal analysis yielded the best overall F1, precision, and recall scores, indicating that the model learns more effectively when the image-text relationship is central to the task. Interestingly, training only on text-dominant examples — those not reliant on visual context — resulted in second-best results in F1 and recall, suggesting the model finds it easy to generalize across simpler, linguistically grounded patterns. However, training on the full dataset without differentiating between multimodal and text-only samples introduced a modest performance drop, likely due to the model being forced to process visual features even when they add noise or are irrelevant.

The weakest performance came from the configuration where image features were excluded entirely, reinforcing the importance of visual input in cases where textual context alone is insufficient. These results collectively suggest that models benefit not only from high-capacity visual encoders like CLIP-Large, but also from modality-aware training strategies that prioritize image-text interaction only when it contributes meaningfully to the task. This insight supports the idea that multimodal classification is not just about fusing features, but about knowing when and how to use them.

4.5.4 Pretraining with Image-Text Contrastive Learning and Targeted Prediction

Following the initial experiments using LoRA-enhanced gated fusion models with CLIP-based encodings, a discussion with the supervising professor raised an important point: instead of incorporating images only during training, it may be more effective to expose the model to visual information earlier — during pretraining. This approach is based on the intuition that learning generic cross-modal alignment first may help the model generalize better before adapting to task-specific labels.

It is worth reminding that the earlier fusion and LoRA-related experiments in this thesis were effectively supervised training rather than true unsupervised or contrastive pretraining. The gated fusion model was trained directly on labeled data, with LoRA applied to a frozen language backbone (TinyLLaMA), and used as-is, since finetuning in the initial experiments with cross-modal attention fusion and gated fusion introduced major drops in performance.

Pretraining Strategy and Performance of Models

To follow the new strategy, a contrastive learning objective was introduced during pretraining. CLIP (Radford et al., 2021 [31]) first proposed contrastive learning for multi-modal learning, and CLIP embeddings were already used during pretraining and finetuning in this thesis. However, in this part of the experiments the idea of contrastive learning was directly applied. In this phase, the model was trained to align text and image embeddings using unlabeled image-text pairs from Subtask 2. The model learns to minimize the distance between matching text-image pairs while pushing apart mismatched ones. This allows it to capture a general-purpose joint embedding space, useful for downstream tasks.

Two configurations were tested for the pretraining phase: 10 epochs and 100 epochs of contrastive learning, both using a batch size of 8. Following contrastive pretraining, the model was fine-tuned using task-specific labels for 10 and 50 epochs respectively.

The shorter 10+10 epoch configuration performed better (F1: 0.590) than the longer 100+50 configuration (0.572), suggesting that overtraining on the contrastive objective may hurt generalization; however, considering small batch sizes this is rather an expected behaviour. Results for each setup are shown in Table 4.10.

Table 4.10: Results from contrastive pretraining followed by supervised fine-tuning

Configuration	Hier. F1	Precision	Recall
10 ep contrast + 10 ep fine-tune (batch 8)	0.58957	0.78640	0.47154
100 ep contrast + 50 ep fine-tune (batch 8)	0.51272	0.77206	0.38380
100 ep contrast + 50 ep fine-tune (batch 12)	0.57245	0.75513	0.46095

Conditional Prediction Based on Multi-Modality of Samples

As the next experiment, a hybrid prediction-time routing strategy was tested. The idea was to select the prediction model dynamically based on whether a meme requires multimodal analysis or not:

- For memes that did **not** require image information, predictions were made using a TinyLLaMA model trained only on text.
- For memes that **did** require image-text fusion, predictions were made using the best contrastively pretrained and fine-tuned model (10 + 10 epochs).

This targeted routing approach significantly improved performance (see Table 4.11) by reducing unnecessary reliance on image features for text-only memes, and selectively leveraging visual features when needed.

Table 4.11: Performance using model selection based on multimodal dependency

Prediction Strategy	Hier. F1	Precision	Recall
Text-only model (all samples)	0.67016	0.74606	0.60828
Multimodal model (all samples)	0.58957	0.78640	0.47154
Hybrid: route by multimodal need	0.62437	0.74877	0.53542

Summary

These experiments confirm that early image-text alignment through contrastive pre-training can improve the model’s ability to integrate multimodal features. However, over-pretraining can lead to diminishing returns. Additionally, a hybrid prediction strategy that selectively routes examples through the appropriate model depending on their modality needs appears to be a promising practical approach, although it currently loses in performance in comparison with earlier models.

4.5.5 Cross-Model Validation of the Best-Performing Method

To further validate the working approach, the highest-performing strategy from the experiments — the configuration without image contrastive learning that only learned multimodal-only samples during training — was selected and its generalizability across different transformer architectures was evaluated. As a reminder, this strategy achieved scores of 0.70393, 0.77396, and 0.64552 for the F1 score, Precision, and Recall, respectively, with the TinyLLaMA model.

This approach was tested with four additional models to assess whether performance gains were architecture-dependent or generalizable between different language models. Table 4.12 presents the comparative results.

The results demonstrate that while the baseline approach maintains the strongest performance, the strategy shows reasonable transferability across different architectures. The DeBERTa v3 Large model achieved the closest performance to our baseline, while the domain-specific models (Political Bias BERT and XLM-Twitter-Politics-Sentiment) showed moderate performance. Notably, CLIP ViT-Large performed poorly and was hence discarded. Such poor performance is likely due to its text input limitations that force truncation from 512 to 77 tokens, which may have severely impacted the model’s ability to process longer textual content effectively.

Table 4.12: Performance comparison across different model architectures using the multimodal-only strategy without image contrastive learning pretraining

Model	Hier. F1	Hier. Precision	Hier. Recall
Baseline (TinyLLaMA)	0.70393	0.77396	0.64552
DeBERTa v3 Large	0.65917	0.75282	0.58624
Political Bias BERT	0.61270	0.74826	0.51872
XLM-Twitter-Politics-Sentiment	0.61196	0.75903	0.51263
CLIP ViT-Large ¹	0.46408	0.39766	0.55714
Ensemble Hard Voting	0.59431	0.81270	0.46844
Ensemble Soft Voting	0.46643	0.85437	0.32078

¹Text input truncated from 512 to 77 tokens.

Discussion of Ensemble Results

The ensemble results reveal interesting performance trade-offs across different metrics. While the ensemble methods significantly improve Precision — with soft voting achieving the highest score of 0.85437 — they demonstrate substantially lower performance on Hierarchical F1 score and Recall compared to the baseline approach. The hard voting ensemble provides a middle ground, achieving 0.81270 on Precision, while maintaining moderate performance on the other metrics.

This pattern suggests that different approaches may be optimal depending on the specific evaluation criteria prioritized. The multimodal-only baseline without image contrastive learning pretraining maintains the most balanced performance across all three metrics, while ensemble methods appear to specialize in improving specific aspects of the task at the expense of others.

The relatively poor performance of domain-specific models (Political Bias BERT and XLM-Twitter-Politics-Sentiment) compared to the general-purpose DeBERTa v3 Large indicates that task-specific pretraining does not necessarily translate to better performance in this multimodal setting.

Conclusion

The experiments demonstrate that the multimodal-only approach without image contrastive learning pretraining provides the most robust and balanced performance across evaluation metrics. While ensemble methods can achieve superior performance

on specific metrics, they introduce trade-offs that may not be desirable for applications requiring consistent performance across multiple evaluation criteria.

The generalizability testing across different transformer architectures shows that the approach is reasonably transferable, with DeBERTa v3 Large providing the closest alternative performance to the baseline. Future work could explore hybrid approaches that combine the balanced performance of the baseline method with the specialized strengths of ensemble techniques.

4.5.6 Early Fusion: VisualBERT with LoRA and Contrastive Fine-Tuning

After extensive experimentation with late fusion architectures, the next step, as mentioned at the beginning, in exploring multimodal persuasion detection was to evaluate early fusion models. While late fusion offers flexibility and modularity, it inherently limits the depth of interaction between modalities, since they are encoded separately. In contrast, early fusion models integrate image and text inputs at an early stage, allowing for fine-grained, joint reasoning across modalities. This architectural advantage may be particularly valuable in tasks like persuasion technique detection, where subtle visual-textual relationships — such as implied tone, visual framing, or other details — may require deeper cross-modal alignment to be effectively captured and used later.

Among the early fusion models available, VisualBERT [21] (pretrained on the COCO dataset) was selected for this phase of the project. VisualBERT is a BERT-based transformer that accepts both textual input and region-based visual embeddings, embedding visual features directly into the token stream and processing them jointly from the very first layer. This architecture aligns well with the multimodal nature of persuasion memes, where image and text often form a cohesive message. VisualBERT was chosen for its relatively mature open-source implementation and its superior performance on the same task (detection of persuasion techniques in memes) in the work by Dimitrov et al. (2021, [5]).

Preparing the Encodings

Since VisualBERT is designed to process region-based visual inputs with tokenized text, it was necessary to extract precomputed image region features before training. Although allowing the model to learn to extract better features each time during pretraining would have been a better idea, computational constraints made this hard to achieve in the available timeframe. VisualBERT’s architecture expects visual input in the form of object-level embeddings, typically obtained using Region-based Convolutional Neural Networks (RCNNs) in research papers, including the original VisualBERT setup [21]. For this purpose, a Faster R-CNN model was used — one of the most widely adopted object detection architectures in vision-language research.

The use of Faster R-CNN was motivated by its established role and extensive use in vision-language pretraining pipelines. While newer image encoders such as ViT and Swin Transformer operate on full images or patches, region-based encodings are often more interpretable and are better suited in tasks where specific visual elements are semantically meaningful.

The Faster R-CNN model was applied to each image in the dataset to extract a fixed number of object-level features. Each input image was resized and processed by a convolutional backbone of ResNet-152 (pretrained on ImageNet), followed by a region proposal network (RPN) that identifies bounding boxes corresponding to salient objects.

For each image:

- The model proposed up to 36 bounding boxes per image, selected based on objectness confidence scores.
- Each region was represented by a 2048-dimensional feature vector, which corresponds to the output of the Region of Interest (ROI) pooling layer after convolution.

Thus, the final visual representation per image had the shape $[36, 2048]$, where 36 corresponds to the number of object regions and 2048 is the feature dimensionality. These raw features were not projected manually — VisualBERT internally includes a

linear projection layer that maps each region feature from 2048 to the model’s hidden size (768), so that the image tokens can be embedded directly alongside textual tokens in the transformer’s input stream.

Architecture

The model was trained with a learning rate of 2×10^{-5} during pretraining and 2×10^{-6} during finetuning, AdamW optimizer with weight decay of 0.01, batch size 6 (due to environment memory constraints) and OneCycleLR scheduler with a linear warm-up from a low learning rate to the maximum learning rate over the first 10% of training and cosine annealing learning rate decay from the maximum back to a lower learning rate over the remaining 90% of training.

To align and compare with previous stages of experimentation and maintain efficient training under hardware constraints, LoRA was again employed. During pre-training, LoRA was configured to target all attention projection layers (values match previous experiments):

- `lora_r = 16`: rank of the low-rank decomposition.
- `lora_alpha = 32`: influence of the low-rank updates.
- `lora_dropout = 0.1`: dropout rate.
- `target_modules: ["query", "key", "value", "dense"]`.
- `bias: "none"`.
- `task_type: TaskType.FEATURE_EXTRACTION`.

This configuration ensured that cross-modal alignment could be learned effectively without modifying the full model weights. For the supervised fine-tuning stage, a more conservative configuration was used to avoid overfitting and focus on the most influential attention components:

- `lora_r = 8`: rank of the low-rank decomposition.

- `lora_alpha = 16`: influence of the low-rank updates.
- `lora_dropout = 0.05`: dropout rate.
- `target_modules: ["query", "value"]`.
- `bias: "none"`.
- `task_type: TaskType.SEQ_CLS`.

The selection of `target_modules` in the LoRA configurations was made to align with the different goals of pretraining and fine-tuning. During pretraining, the goal was to learn meaningful joint representations across image and text, so LoRA adapters were applied to a broader set of transformer components: `["query", "value", "key", "dense"]`. These modules correspond to the key projection layers in the self-attention mechanism and the feedforward network, which are critical for learning contextual and cross-modal interactions. The `task_type` was set to `TaskType.FEATURE_EXTRACTION`, reflecting the fact that the model was not being optimized for a specific downstream label prediction task, but rather for learning representations through a contrastive loss.

In contrast, during fine-tuning, the model was adapted for a classification objective. To reduce the number of trainable parameters and avoid overfitting, LoRA adapters were only applied to the `["query", "value"]` modules — the core components of the attention mechanism — while omitting key and dense layers. This more conservative configuration was sufficient to adapt the attention focus to task-specific distinctions while preserving the integrity of the pretrained encoder. The `task_type` was set to `TaskType.SEQ_CLS` to reflect the goal of multilabel persuasion technique classification. This distinction in LoRA configurations allowed the model to remain efficient while remaining flexible across the representation learning and supervised phases.

Cross-Modal Alignment with Technique-aware Negatives

The previously used standard contrastive learning method relied on random negative sampling, which may not have captured the intricate semantic relationships between image and text pairs. The challenge in persuasion technique detection lies in the fact that techniques are not mutually exclusive and often co-occur within the same meme [5]. For instance, a meme might simultaneously employ “Loaded Language” and “Appeal to Fear” techniques, creating complex inter-technique relationships that standard contrastive learning methods may not capture effectively.

Building upon these foundations, a novel technique-aware contrastive sampling strategy is proposed, specifically designed for persuasive meme analysis. This approach leverages the inherent structure of persuasion technique labels to construct semantically meaningful hard negatives, thereby improving the model’s ability to distinguish between different persuasive techniques in a multimodal environment.

Methodology This cross-modal alignment strategy extends the standard contrastive learning framework by incorporating domain-specific knowledge about persuasion techniques into a sophisticated pair sampling process. Given a dataset of memes with associated text and persuasion technique labels, training pairs are created as follows:

Positive Pairs. An exact technique matching strategy that creates semantically meaningful positive examples is used. Given an anchor sample with technique set T_a , positive examples are sampled from the pool of samples that have identical technique combinations. The positive sampling process works as follows:

1. *Exact Matching:* Identify all samples in the dataset that contain exactly the same set of persuasion techniques as the anchor ($T_{\text{candidate}} = T_a$).
2. *Candidate Filtering:* Exclude the anchor sample from the candidate pool.
3. *Random Selection:* If candidates exist, uniformly sample from them to form the positive pair; otherwise, use the anchor sample itself as a fallback.

This approach ensures that positive pairs share identical persuasion strategy profiles, enabling the model to learn robust representations for specific technique combinations.

Negative Pairs. A sophisticated within-batch sampling strategy is used: it creates a hierarchy of negative examples based on technique similarity. This approach uses the current batch composition to generate informative negative pairs while maintaining computational efficiency. There are three categories of potential negative samples:

1. *Hard Negatives (Technique Overlap)*: Samples that share some but not all techniques with the anchor ($T_{\text{neg}} \cap T_a \neq \emptyset$ and $T_{\text{neg}} \neq T_a$). These represent the most challenging negatives as they share semantic similarities while employing different persuasive strategies.
2. *Easy Negatives (Technique Disjoint)*: Samples with completely different persuasion techniques ($T_{\text{neg}} \cap T_a = \emptyset$). These help the model learn to distinguish between fundamentally different persuasive approaches.
3. *Fallback Negatives*: Any remaining samples in the batch when no hard or easy negatives are available.

The negative sampling algorithm prioritizes hard negatives over easy negatives, as the former provide more informative gradients for learning fine-grained technique distinctions.

Balanced Technique Groups To implement the technique-aware sampling, balanced technique groups are pre-computed during dataset initialization. For each persuasion technique t_i in the label space the following is stored:

- *Technique Combination Groups*: A mapping from each unique technique combination (represented as a `frozenset`) to all samples containing exactly that combination.

- *Sample-to-Techniques Mapping*: A mapping from each sample to its set of active techniques.
- *Technique Similarity Function*: A function that categorizes the relationship between any two samples as ‘identical’, ‘overlap’, or ‘different’ based on their technique sets.

The technique similarity computation follows a simple yet effective categorization:

- *Identical*: $T_1 = T_2$ (exact same technique set).
- *Overlap*: $T_1 \cap T_2 \neq \emptyset$ and $T_1 \neq T_2$ (some shared techniques).
- *Different*: $T_1 \cap T_2 = \emptyset$ (no shared techniques).

This preprocessing enables the collation function to quickly identify appropriate positive and negative candidates for each anchor sample during batch construction.

Contrastive Loss Functions Two contrastive loss formulas are used during fine-tuning to optimize the technique-aware representation learning:

1. **Margin-Based Contrastive Loss**: Based on the original contrastive loss formula by Hadsell et al. [10] and adapted to maximize similarity for similar pairs, enforcing a maximum similarity threshold for dissimilar pairs:

$$\mathcal{L}_{\text{positive}} = \max(0, 1 - \text{sim}(z_i^{(1)}, z_i^{(2)}))^2 \quad (4.1)$$

$$\mathcal{L}_{\text{negative}} = \max(0, \text{sim}(z_i^{(1)}, z_i^{(2)}) - m)^2 \quad (4.2)$$

The complete margin-based loss is then:

$$\mathcal{L}_{\text{margin}} = \frac{1}{N} \sum_{i=1}^N [y_i \cdot \mathcal{L}_{\text{positive}} + (1 - y_i) \cdot \mathcal{L}_{\text{negative}}] \quad (4.3)$$

where $z_i^{(1)}$ and $z_i^{(2)}$ are L2-normalized embeddings of the pair, $y_i \in \{0, 1\}$ indicates positive/negative pairs, $\text{sim}(\cdot, \cdot)$ is cosine similarity, and m is the margin hyperparameter (set to 0.5). The positive loss term encourages similar technique pairs to have high similarity (approaching 1), while the negative loss term pushes dissimilar pairs to have similarity below the margin threshold.

2. Binary Cross-Entropy Loss with temperature scaling.

Implementation The technique-aware contrastive learning is implemented as a preprocessing step before fine-tuning on the persuasion detection task. The framework consists of:

- **TechniqueAwareContrastiveDataset**: Extends the base VisualBERT dataset class to implement technique combination grouping and similarity computation.
- **TechniqueAwareCollater**: Custom collation function that creates technique-aware positive/negative pairs during batch construction using within-batch sampling.
- *Preprocessing Pipeline*: Computes technique statistics and sampling weights beforehand to ensure efficient training.

Results

Initially, two different types of losses were evaluated to be chosen as the primary loss for further experiments and trained for 10 epochs.

Table 4.13: Results of two types of losses: Margin-Based Contrastive Loss and Binary Cross-Entropy Loss

Loss Type	Hier. F1	Precision	Recall
Margin-Based Loss	0.44706	0.68778	0.33116
Binary Cross-Entropy Loss	0.26774	0.70387	0.16531

As can be seen from Table 4.13, the Binary Cross-Entropy loss achieved the highest precision (0.70387), but resulted in significantly lower recall (0.16531) and overall

hierarchical F1 score (0.26774). This indicates that the model trained with Binary Cross-Entropy was overly conservative, correctly identifying a small subset of techniques but failing to generalize across the full label space. In contrast, the margin-based (technique-aware) contrastive loss achieved a much higher F1 score (0.44706) and better recall (0.33116), suggesting a more balanced and comprehensive understanding of the task. Since the ultimate goal of the model is to recognize multiple persuasion techniques per sample with both precision and coverage, F1 score served as the primary metric for evaluation. Therefore, despite Binary Cross-Entropy Loss yielding slightly higher precision, the margin-based loss was chosen for all subsequent VisualBERT experiments due to its more robust and generalizable performance.

To prove continual growth of performance, this model with Margin-Based Loss was trained one more time with 20 epochs, proving the initial hypothesis (see Table 4.14). This resulted in a higher F1 score (0.49328) and recall (0.40028), with slightly lower precision (0.64256), yet still overall more balanced results. Despite this slight increase in the F1 score and Recall, this is due to the model learning a second dominant class for this task.

Both the model trained for 10 epochs and the one trained for 20 epochs demonstrated a classic mode collapse problem, and predicted two dominant classes for all samples. For instance, the first model predicted “Smears” for all samples, and then after learning for 20 epochs it predicted “Smears” for all samples and “Transfer” for about 7% of the test set, which increased some performance metrics. Interestingly, it did not predict “Loaded Language” or “Name Calling/Labeling”, which are both more dominant classes than Transfer, but the problem still stands.

Table 4.14: Fine-tuning with Margin-Based Contrastive Loss for 10 and 20 epochs

Number of Epochs	Hier. F1	Precision	Recall
10	0.44706	0.68778	0.33116
20	0.49328	0.64256	0.40028

4.5.7 Improving Class Balance with Weighted Losses and Balanced Pretraining

Following the previous VisualBERT experiments, the next step was to target label (class) imbalance, which was clearly limiting the model’s ability to generalize and predict underrepresented persuasion techniques. This could be seen from the predictions made for the test set, where dominant classes were predicted for all samples.

To address this, two changes were implemented, with changes in both the pre-training and fine-tuning phases of the pipeline:

1. Modify the dataset to ensure balanced exposure to all persuasion techniques. For this, inverse frequency weighting was applied to ensure that rare techniques received proportionally more attention during contrastive pair generation.
2. Use a custom Weighted Binary Cross-Entropy Loss during the finetuning stage.

Architecture

In the pretraining stage the previously described balanced technique-aware contrastive learning approach was used, but this time using Inverse Frequency Weights stored in the dataset for each persuasion technique, computed as $w_i = \frac{1}{\text{freq}(t_i)}$ and normalized across all techniques, as well as during positive and negative pair generation.

The same LoRA configurations were used during 20 epochs of contrastive pre-training with a learning rate of 1×10^{-5} , and using the margin-based contrastive loss described in Subsection 4.5.6.

During the classification fine-tuning process, class-specific weights were incorporated into the loss function to further mitigate the effects of class imbalance.

Weighted Binary Cross-Entropy Loss

A custom weighted binary cross-entropy loss function was implemented to penalize misclassifications of rare classes more heavily than those of common classes:

$$\mathcal{L}_{\text{weighted}} = \frac{1}{N} \sum_{i=1}^N \mathcal{L}_i \quad (4.4)$$

$$\mathcal{L}_i = \sum_{j=1}^C \mathcal{L}_{ij} \quad (4.5)$$

$$\mathcal{L}_{ij} = -w_j [y_{ij} \log(\sigma(z_{ij})) + (1 - y_{ij}) \log(1 - \sigma(z_{ij}))] \quad (4.6)$$

where w_j represents the positive weight for class j , y_{ij} is the ground truth label, z_{ij} is the logit output, and σ denotes the sigmoid function. The positive weights w_j were computed as the inverse of class frequencies, normalized across all classes.

Class Weight Computation Strategies Two different approaches to class weight computation were evaluated:

1. **Standard Inverse Frequency Weighting:** The initial approach computed weights using simple inverse frequency:

$$w_j = \frac{N_{\text{total}}}{N_j} \quad (4.7)$$

where N_{total} is the total number of samples and N_j is the number of samples containing technique j . This standard approach yielded moderate improvements in handling class imbalance.

2. **Extreme Class Weighting:** A more aggressive weighting scheme was subsequently developed to provide stronger emphasis on underrepresented classes:

$$w_j = \begin{cases} \min\left(100.0, \frac{N_{\text{total}}}{N_j}\right) & \text{if } N_j < 100 \text{ (very rare)} \\ \min\left(20.0, \frac{N_{\text{total}}}{N_j}\right) & \text{if } 100 \leq N_j < 500 \text{ (rare)} \\ \min\left(5.0, \frac{N_{\text{total}}}{N_j}\right) & \text{if } 500 \leq N_j < 1000 \text{ (medium)} \\ 1.0 & \text{if } N_j \geq 1000 \text{ (common)} \end{cases} \quad (4.8)$$

This tiered weighting approach applied progressively stronger penalties for rarer classes while preventing extreme weight values that could destabilize training. Very rare techniques (fewer than 100 occurrences) received weights up to 100 times larger than common techniques, while rare and medium-frequency techniques received proportionally smaller but still significant weight boosts.

Training Configuration

The complete training pipeline consisted of:

1. *Pretraining phase*: 20 epochs using the balanced technique-aware contrastive dataset with learning rate 1×10^{-5} .
2. *Fine-tuning phase*: 20 epochs using weighted BCE loss with learning rate 1×10^{-6} .

Results

The results show that although overall precision decreased, F1 increased slightly and recall increased substantially to 0.792 (Table 4.15). This trade-off is acceptable in multi-label settings where coverage (recall) is often more important than precision alone, particularly for exploratory or diagnostic use cases. However, this increase was not the result of predicting very rare classes, but only extended the prediction set from two dominant classes (“Smears” and “Transfer”) to five: “Loaded Language”, “Name Calling/Labeling”, “Slogans”, “Smears”, and “Thought-terminating cliché”.

The technique-aware contrastive pretraining combined with extreme class weighting successfully improved model diversity beyond the most dominant classes, achieving better recall and hierarchical F1 scores. However, the approach was still insufficient for predicting the rarest persuasion techniques (fewer than 100 examples), showing that severe class imbalance requires alternative solutions to loss reweighting.

The balanced contrastive pretraining provided a solid foundation for representation learning, but future work should explore data augmentation strategies, few-shot learning approaches, or much larger batch sizes where computationally possible, to

Table 4.15: Performance with balanced contrastive pretraining and BCE loss with two ways of setting weights

Setting	Hier. F1	Precision	Recall
Balanced Contrastive + Inverse Frequency Weights	0.53093	0.59627	0.47849
Balanced Contrastive + Extreme Class Weights	0.54158	0.41139	0.79232

address the fundamental challenge of learning from extremely limited examples in persuasion technique detection.

Chapter 5

Results

Based on the extensive experimental evaluation described in the previous chapter, this chapter first presents the final system architecture that achieved the best performance, followed by a summary of all experimental results.

5.1 Detailed Description of the Best Model

The winning architecture uses TinyLLaMA (1.1B parameters) as the base language model, which is adapted using Low-Rank Adaptation (LoRA) for parameter-efficient fine-tuning, with in-domain pretraining and gated multimodal fusion, trained specifically on samples that require multimodal analysis.

5.1.1 Architecture Overview

The final system follows a late fusion approach that processes textual and visual information through separate encoder pipelines before combining them using a gated attention mechanism. The system consists of four main components: getting text encodings, a visual encoder, a gated fusion module, and a classification head.

Text Encoding Pipeline

The textual component consists of a LoRA-enhanced small language model backbone with additional processing components for improved text representation:

- **Base Language Model:** TinyLLaMA processes input text through its transformer layers, producing contextualized hidden states for each token in the sequence.
- **Token-level Self-Attention:** An additional multi-head attention layer (8 heads, 0.1 dropout) is applied to the hidden states to enhance token representations by allowing each token to attend to all other tokens in the sequence.
- **Attention-Aware Pooling:** The attended token representations are pooled using mean pooling with attention masking, ensuring that only valid tokens contribute to the final representation.
- **Output Projection:** A two-layer neural network with Tanh activation projects the pooled representation to the final embedding space.

Visual Encoding Pipeline

For visual feature extraction, the system uses CLIP-Large (ViT-L/14) as the image encoder. This choice was made after comparing CLIP-Base and CLIP-Large variants, with the latter demonstrating superior performance. CLIP-Large produces 768-dimensional image embeddings that capture fine-grained visual semantics and their correspondence to textual concepts.

Gated Attention Fusion Mechanism

The multimodal fusion is handled by a sophisticated Gated Attention Fusion Network, introduced by Du et al. [8], that implements a two-stage attention mechanism with gating control.

Classification Head

The final classification component uses a deeper feedforward network compared to earlier experiments:

Listing 5.1: Classification Head

```
classification_head = nn.Sequential(  
    nn.Dropout(0.3),  
    nn.Linear(fusion_dim, fusion_dim // 2),  
    nn.ReLU(),  
    nn.Dropout(0.2),  
    nn.Linear(fusion_dim // 2, num_labels)  
)
```

This architecture introduces an intermediate layer with ReLU activation and dropout regularization at two stages (0.3 and 0.2 dropout rates). The additional depth allows the model to learn more complex nonlinear relationships between the fused multimodal features, while the dropout layers help prevent overfitting in the low-data regime typical of specialized classification tasks.

5.1.2 Training Configuration

Before the main multimodal training phase, the TinyLLaMA backbone undergoes in-domain pretraining on the augmented text-only dataset created during the data augmentation phase. This pretraining step serves to adapt the language model to the specific linguistic patterns and vocabulary used in persuasive memes. The pretraining process uses the PNGPTCaptDataset, which combines multiple augmentation strategies:

- NLPAug-based synonym replacement for underrepresented labels.
- GPT-4o-mini generated paraphrases using Chain-of-Thought prompting.
- Caption-based augmentation that incorporates visual context into textual descriptions.

- Pseudo-labeled samples from high-confidence model predictions.

This comprehensive augmented dataset ensures that the language model develops robust representations of persuasive language patterns before being integrated into the multimodal framework. The pretraining phase uses standard masked language modeling objectives with LoRA adaptation, allowing the model to adapt to the domain-specific vocabulary while maintaining parameter efficiency.

LoRA Hyperparameters

The LoRA fine-tuning is configured with the following parameters:

- **Rank** ($r = 16$): Balances expressive power with training efficiency.
- **Alpha** ($\alpha = 32$): Controls the influence of low-rank updates on the base model.
- **Dropout** (0.1): Reduces overfitting in LoRA layers.
- **Target modules**: Applied to attention projections (q_proj, k_proj, v_proj, o_proj) and feedforward layers (gate_proj, up_proj, down_proj).
- **Task type**: Sequence classification for hierarchical persuasion technique detection.

Training Strategy

A critical finding from the experimental phase was the importance of training data selection. The winning configuration uses only samples that explicitly require multimodal analysis for correct classification.

The training process consists of two phases:

1. *In-Domain Pretraining*: TinyLLaMA is pretrained on the augmented text-only dataset to develop domain-specific language understanding.
2. *Multimodal Training*: The full system is trained end-to-end using selective multimodal samples with a batch size of 16 and a learning rate of 1×10^{-4} .

The training process uses early stopping based on validation F1 score. Models are trained for up to 20 epochs (both phases), which allows sufficient time for the LoRA adaptation to converge without overfitting.

5.1.3 Performance Summary

This final architecture achieves the following performance on the hierarchical persuasion technique detection task:

- Hierarchical F1: 0.70393
- Hierarchical Precision: 0.77396
- Hierarchical Recall: 0.64552

These results represent a significant improvement over baseline approaches and demonstrate the effectiveness of combining parameter-efficient fine-tuning with selective multimodal fusion. The high precision indicates that the model makes confident and accurate predictions, while the F1 score shows a good balance between precision and recall for this challenging multi-label classification task.

5.1.4 Implementation Details

The system is implemented using Python version 3.11.11, PyTorch version 2.6.0, and the Transformers version 4.51.3 library, with LoRA integration through the PEFT (Parameter-Efficient Fine-Tuning) version 0.9.2 library. The modular design allows for easy experimentation with different encoder combinations and fusion strategies. All models are trained on GPU clusters described in Section 4.2 and Table 4.1 to optimize memory usage and training speed.

Chapter 6

Discussion

The work of this thesis has looked at multiple ways that can be used to detect persuasion techniques in memes through the use of multimodal approaches. These were examined sequentially starting with basic late-fusion based methods to advanced gated-attention type mechanisms that have been augmented by using parameter efficient fine tuning. Along the way, a number of important findings regarding the relationships between the design decisions made during architecture selection, the characteristics of the hardware environment available and the resulting performance in terms of classification accuracy were realized. As such, the goal of this section is to discuss how the environmental context impacted upon the performance of the models; identify key results; examine the application and limitations of these findings.

6.1 Hardware Constraints and Their Impact on Model Performance

The Google Colaboratory free tier environment imposed significant constraints that fundamentally shaped the experimental outcomes. Working with an NVIDIA Tesla T4 (15GB VRAM) GPU, accessed through Colab’s shared resource allocation system with limited session times and frequent runtime disconnections, created several bottlenecks that likely influenced the relative performance of different approaches.

6.1.1 Batch Size Limitations

The most critical constraint was the restricted batch sizes across all experiments. VisualBERT experiments were limited to batch sizes of 6–8, while TinyLLaMA-based approaches could train with batch sizes of 16. This had several serious consequences:

- **Noisier gradient estimates**, which are particularly problematic for contrastive learning objectives used in VisualBERT pretraining. In original papers with contrastive learning by Robinson et al. [34], a small batch size for contrastive learning is considered to be up to 512 samples per batch. So, the technique-aware contrastive learning strategy required diverse negative samples within each much larger batch, but with only 6–8 samples per batch, the model had very limited exposure to hard negatives during training.
- **Worse Label Coverage**. In a multi-label classification task with 22 persuasion techniques, small batch sizes meant that many labels were absent from individual batches, hindering the model’s ability to learn balanced representations across all classes.
- **Training instability**, particularly evident in the VisualBERT mode collapse problem where the model converged to predicting only dominant classes, due to pretraining with small batch sizes.

6.1.2 Memory Constraints and Model Architecture

The GPU memory limitations forced architectural compromises that may have inadvertently favored certain approaches. VisualBERT’s early fusion architecture used precomputed R-CNN features; however, for contrastive learning purposes (Radford et al. [31]) the best scenario would be to allow the model to learn to retrieve better representations during pretraining. However, due to memory limitations it was not possible. In contrast, the late fusion approach could process modalities independently and store features, enabling more efficient memory utilization.

6.2 Key Findings and the Superiority of TinyLLaMA-based Approaches

With these constraints in mind, examining the results of this work we identify that the most significant finding is the unexpected superiority of the TinyLLaMA-based late fusion approach over more sophisticated early fusion architectures. The winning configuration — TinyLLaMA (1.1B parameters) with LoRA adaptation, gated fusion, and selective multimodal training — achieved an F1 score of 0.70393, substantially outperforming VisualBERT-based early fusion approaches which had the highest F1 score at 0.54158.

This performance gap reveals several fundamental insights about multimodal learning in resource-constrained environments. The TinyLLaMA approach benefited from its modular architecture, allowing independent optimization of text and visual encoders. More importantly, the parameter-efficient LoRA adaptation enabled effective domain-specific fine-tuning while avoiding the overfitting problem that affected the results of the more complex VisualBERT model.

The importance of selective multimodal training emerged as a critical factor. Training exclusively on samples requiring multimodal analysis (F1: 0.70393) outperformed training on all available data (F1: 0.68451), challenging the assumption that more training data always improves performance. This suggests that data quality and relevance are more critical than quantity, particularly when working with imbalanced datasets and limited computational resources.

6.3 Architectural Analysis: Why Early Fusion Failed to Outperform Late Fusion

Although the initial hypothesis was that originally multimodal models like VisualBERT with early fusion would outperform modular late fusion models, this was not the case. The underperformance of VisualBERT compared to TinyLLaMA-based

approaches reveals important insights about multimodal architecture design in constrained environments:

- VisualBERT showed extreme sensitivity to the severe label imbalance in the dataset, consistently falling into mode collapse where only 2–5 dominant classes were predicted. Despite various mitigation strategies — extreme class weighting, balanced contrastive pretraining, and architectural enhancements — the early fusion architecture struggled to maintain diverse predictions across all 22 persuasion techniques. The TinyLLaMA approach, processing modalities separately before fusion, proved to be more robust to this imbalance.
- The VisualBERT architecture’s higher parameter count, small batch size and frozen early fusion design, combined with class imbalance, made it more susceptible to overfitting. The model needed contrastive samples, positive and negative pairs to learn meaningful representations, but since every batch consisted of mostly dominant techniques, the predictions were in accordance with what the model saw.

6.4 Ensemble Methods and Their Limitations

Another expectation, this time regarding the effectiveness of ensemble methods, was not met. Contrary to the idea based on successful model replications (see Section 3.4), ensemble methods at the model level failed to result in substantial improvements over the best individual model. The ensemble of different transformer architectures achieved lower F1 scores (0.59431 for hard voting) compared to the single TinyLLaMA model (0.70393).

This underperformance can be due to several factors:

- All ensemble components suffered from similar limitations imposed by hardware constraints. Instead of having different abilities, all of the individual models were biased towards major classes and failed similarly when trying to identify rare ones.

- The soft and hard voting strategies may have amplified the conservative prediction tendencies observed in individual models, further reducing recall for underrepresented techniques.

6.5 Resource-driven Implications

The results show how hardware limitations will impact the comparative ranking of various methods’ effectiveness as a function of their relative ability to take advantage of abundant resources. With an abundance of computing resources available for use, the VisualBERT early-fusion architecture may be able to demonstrate some or all of its theoretically greater potential due to being able to train with larger batch sizes, and/or longer training times; and/or the possibility of performing additional hyperparameter optimizations. Under normal conditions of academic research, smaller, modular designs such as that based on the TinyLLaMA design are typically more successful. Also notable is that the TinyLLaMA model (with 1.1 billion parameters) outperformed larger models; this suggests that constraint on parameters will have a positive effect on constrained computing environments. As shown by LoRA adaptation, it was possible to improve the model’s domain-specific performance without changing more than 0.01% of the total number of parameters. These improvements were realized practically as well: they produced faster training times, reduced memory requirements, and improved stability of optimization under small batch size conditions.

Chapter 7

Conclusion

This thesis investigated multiple multimodal architectures for detecting the persuasion techniques used in memes to better understand how the physical limitations (e.g., memory size, processing speed) of computers affect overall architecture success. A primary outcome from this study is that an architecturally successful multimodal system will be effective as both theoretically optimal and practically feasible given the available resources and properties of the data.

7.1 Main Contributions

The key contributions of this research are:

1. Shown that although larger, more complex models (e.g., VisualBERT) may have theoretical advantages over smaller, more modular models (i.e., TinyLLaMA + late fusion), if constrained by limited batch sizes and/or limited GPU memory, smaller, more modular models will often perform at least as well as larger models.
2. Demonstrated that training with only those examples which require multimodal analysis results in better performance than training on all available data, and provided an approach to data selection based on multimodality in cases where resources are limited.

3. Illustrated that LoRA enhancements to language models enable high-quality multimodal models with minimal updates to model weights (only 0.01% of weight space is updated).
4. Showed that large differences exist in the ability of various multimodal architectures to perform given batch-size limits, especially when contrastive learning or class imbalance are involved.

7.2 Lessons from Failed Approaches

Similar, yet different, conclusions were drawn from the results of failed strategies as well:

- VisualBERT’s consistent underperformance was attributed to the following: (1) Memory limitations forced small batch sizes which impeded the effectiveness of contrastive learning; (2) Optimization challenges arose when trying to learn both cross-modal alignment and task-specific features simultaneously under limited resources; and (3) Class balance sensitivity resulted in mode collapse across numerous attempts at mitigation.
- Poor ensemble strategy performances demonstrated how computational constraints could also limit the diversity among the components of an ensemble. Individual models were biased similarly with identical failure modes and a large amount of computational overhead made it impossible to optimize each component adequately.
- Enhanced architecture (e.g., deeper classifiers, attention pooling) always resulted in decreased performance for visual question answering tasks. This suggested that as model complexity increases so does the required training data and computational power. Under these types of constraints, the simpler architecture performed better.

7.3 Future Directions

In addition to this work’s limitations, there are a number of ways to improve upon it in the future. Dynamic batch size scheduling and gradient accumulation techniques may enable larger effective batch sizes on memory-constrained devices for early-fusion models as well as provide the potential to utilize the theoretically advantageous architecture designs such as those used by VisualBERT and other single-stream models. The development of automatic sample selection mechanisms will reduce the burden associated with manual annotation and make selective training methodologies (selective-training) much more feasible. Finally, additional research into few-shot learning or data generated synthetically specific to multimodal application domains may also be beneficial.

7.4 Final Remarks

The transition from simple concat based fusion to complex gated attention with parameter-efficient adaptations clearly highlights both the opportunities and difficulties associated with multimodal learning within constrained computing environments. Resource limitations provided a challenge but also highlighted aspects of multimodality which would otherwise likely remain unexplored. The success of TinyLLaMA demonstrates more than just a technological breakthrough, it shows that architecturally designed systems, in consideration of resource limitations, are capable of producing competitive results while being applicable and deployable. Constraint aware design is an exciting concept for use as a starting point for multimodal research going forward. As research and application environments will be characterized by wide variations in available computational resources, the study of how constraints influence architectural efficacy will become increasingly important. Ultimately, this work supports the hypothesis that there exists evidence suggesting that less complex, modularity-based architectures provide more resilient performance over varying levels of available computational resources; thus directly challenging the continued

increase in architectural complexity. The results of this research – related to selective training, parameter efficiency and the relationship between computational constraints and performance – add to our overall understanding of practical multimodal system design.

Bibliography

- [1] Bartoli, A., & Fusiello, A. (2020). Computer vision–ECCV 2020 workshops. In *European Conference on Computer Vision Workshops (ECCV)*, volume 12536 of *Lecture Notes in Computer Science*. Springer.
- [2] Burbi, G., Baldrati, A., Agnolucci, L., Bertini, M., & Del Bimbo, A. (2023). Mapping memes to words for multimodal hateful meme classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 2832–2836).
- [3] Kan Chen, Trung Bui, Fang Chen, Zhaowen Wang, and Ram Nevatia. AMC: Attention guided multi-modal correlation learning for image search. *CoRR*, abs/1704.00763, 2017. URL <http://arxiv.org/abs/1704.00763>.
- [4] Da San Martino, G., Seunghak, Y., Barrón-Cedeno, A., Petrov, R., & Nakov, P. (2019). Fine-grained analysis of propaganda in news article. 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)* (pp. 5636–5646). Association for Computational Linguistics.
- [5] Dimitar Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. Detecting propaganda techniques in memes. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *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 6603–6617, Online, Au-

gust 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.516. URL <https://aclanthology.org/2021.acl-long.516/>.

- [6] Dimitrov, D., Bin Ali, B., Shaar, S., Alam, F., Silvestri, F., Firooz, H., Nakov, P., & Da San Martino, G. (2021). SemEval-2021 task 6: Detection of persuasion techniques in texts and images. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)* (pp. 70–98). Association for Computational Linguistics.
- [7] Dimitrov, D., Alam, F., Hasanain, M., Hasnat, A., Silvestri, F., Nakov, P., & Da San Martino, G. (2024). SemEval-2024 Task 4: Multilingual Detection of Persuasion Techniques in Memes. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*.
- [8] Yongping Du, Yang Liu, Zhi Peng, and Xingnan Jin. Gated attention fusion network for multimodal sentiment classification. *Knowledge-Based Systems*, 240:108107, 2022. doi: 10.1016/j.knosys.2021.108107. URL <https://doi.org/10.1016/j.knosys.2021.108107>.
- [9] Omid Ghahroodi and Ehsaneddin Asgari. (2024). HierarchyEverywhere at SemEval-2024 Task 4: Detection of persuasion techniques in memes using hierarchical text classifier. In Atul Kr. Ojha, A. Seza Doğruöz, Harish Tayyar Madabushi, Giovanni Da San Martino, Sara Rosenthal, and Aiala Rosá, editors, *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 1727–1732, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.semeval-1.247. <https://aclanthology.org/2024.semeval-1.247/>
- [10] Raia Hadsell, Sumit Chopra, and Yann Lecun. Dimensionality reduction by learning an invariant mapping. *Institute of Electrical and Electronics Engineers*, pages 1735–1742, 02 2006. doi: 10.1109/CVPR.2006.100.
- [11] Jing, E., Schneck, K., Egan, D., & Waterman, S. A. (2021). Identifying introductions in podcast episodes from automatically generated tran-

scripts. White Paper. Retrieved from <https://www.siriusxm.com/research/podcast-intro-identification>. *arXiv preprint arXiv:2108.10454*.

- [12] Jing, J., Wu, H., Sun, J., Fang, X., & Zhang, H. (2023). Multimodal fake news detection via progressive fusion networks. *Information Processing & Management*, 60(1), 103120.
- [13] John Boaz Lee, Ryan A. Rossi, Sungchul Kim, Nesreen K. Ahmed, and Eunye Koh. Attention models in graphs: A survey. *CoRR*, abs/1807.07984, 2018. URL <http://arxiv.org/abs/1807.07984>.
- [14] Dailin Li, Chuhan Wang, Xin Zou, Junlong Wang, Peng Chen, Jian Wang, Liang Yang, and Hongfei Lin. (2024). 914isthebest at SemEval-2024 Task 4: Leveraging In-Domain Pretraining and Ensemble Learning for Multimodal Propaganda Detection. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*. <https://aclanthology.org/2024.semeval-1.190>
- [15] Hamza, A., Javed, A. R., Iqbal, F., Yasin, A., Srivastava, G., Polap, D., ... & Jalil, Z. (2024). Multimodal religiously hateful social media memes classification based on textual and image data. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 23(8), 1–19.
- [16] Hasanain, M., Ahmed, F., & Alam, F. (2024). Can GPT-4 identify propaganda? Annotation and detection of propaganda spans in news articles. *arXiv preprint arXiv:2402.17478*.
- [17] Kiela, D., Firooz, H., Mohan, A., Goswami, V., Singh, A., Fitzpatrick, C. A., ... & Parikh, D. (2021, August). The hateful memes challenge: Competition report. In *NeurIPS 2020 Competition and Demonstration Track* (pp. 344–360). PMLR.
- [18] Lee, D.-H. (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*.

- [19] Haoran Li, Junnan Zhu, Tianshang Liu, Jiajun Zhang, and Chengqing Zong. Multi-modal sentence summarization with modality attention and image filtering. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, IJCAI-18, pages 4152–4158. International Joint Conferences on Artificial Intelligence Organization, 7 2018. doi: 10.24963/ijcai.2018/577. URL <https://doi.org/10.24963/ijcai.2018/577>.
- [20] Gen Li, Nan Duan, Yuejian Fang, Daxin Jiang, and Ming Zhou. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. *CoRR*, abs/1908.06066, 2019. URL <http://arxiv.org/abs/1908.06066>.
- [21] Liunian Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language, 08 2019.
- [22] Lippe, P., Holla, N., Chandra, S., Rajamanickam, S., Antoniou, G., Shutova, E., & Yannakoudakis, H. (2020). A multimodal framework for the detection of hateful memes. *arXiv preprint arXiv:2012.12871*.
- [23] Siyu Lu, Mingzhe Liu, Lirong Yin, Zhengtong Yin, Xuan Liu, and Wenfeng Zheng. The multi-modal fusion in visual question answering: a review of attention mechanisms. *PeerJ Computer Science*, 9, 05 2023. doi: 10.7717/peerj-cs.1400.
- [24] Ma, E. (2019). NLP Augmentation. <https://github.com/makcedward/nlpaug>.
- [25] National Counterterrorism Center (NCTC) & Office of the Director of National Intelligence (DNI). (2021). Use of memes by violent extremists. https://www.dni.gov/files/NCTC/documents/jcat/firstresponderstoolbox/128S_-_First_Responders_Toolbox_-_Use_of_Memes_by_Violent_Extremists.pdf
- [26] National Coordinator for Security and Counterterrorism (NCTV). (2024). Memes as an online weapon: Undermining and far from recognizable. https://english.nctv.nl/binaries/nctv-en/documenten/publications/2024/08/01/far-right-memes-undermining-and-far-from-recognizable/NCTV_Memes+as+an+online+weapon_English+version+May+2024.pdf

- [27] Newman, B., Choubey, P. K., & Rajani, N. F. (2021). P-adapters: Robustly extracting factual information from language models with diverse prompts.
- [28] Zhaoyang Niu, Guoqiang Zhong, and Hui Yu. A review on the attention mechanism of deep learning. *Neurocomputing*, 452:48–62, 2021. ISSN 0925-2312. doi: <https://doi.org/10.1016/j.neucom.2021.03.091>. URL <https://www.sciencedirect.com/science/article/pii/S092523122100477X>.
- [29] Ntouvli, V., & Geenen, J. (2025). “Ironic memes” and digital literacies: Exploring identity through multimodal texts. *New Media & Society*, 27(2), 1193–1211.
- [30] OpenAI. (2024). GPT-4o mini: Advancing cost-efficient intelligence. <https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/> Accessed: 2026-02-02.
- [31] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021, July). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning* (pp. 8748–8763). PMLR.
- [32] Raghu, M., & Schmidt, E. (2020). A survey of deep learning for scientific discovery. *arXiv preprint arXiv:2003.11755*.
- [33] Ranjan, A., Papnai, I., Gupta, J., Aggarwal, M., & Saroj, A. (2025, June). Multilingual detection of persuasion techniques in memes. In *6th International Conference on Deep Learning, Artificial Intelligence and Robotics (ICDLAIR 2024)* (pp. 251–260). Atlantis Press.
- [34] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. *CoRR*, abs/2010.04592, 2020. URL <https://arxiv.org/abs/2010.04592>.

- [35] Sharma, S., Alam, F., Akhtar, M. S., Dimitrov, D., Martino, G. D. S., Firooz, H., ... & Chakraborty, T. (2022). Detecting and understanding harmful memes: A survey. *arXiv preprint arXiv:2205.04274*.
- [36] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. VL-BERT: Pre-training of generic visual-linguistic representations. *CoRR*, abs/1908.08530, 2019. URL <http://arxiv.org/abs/1908.08530>.
- [37] Vajjala, S., Majumder, B., Gupta, A., & Surana, H. (2020). *Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems*. O’Reilly Media.
- [38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL <http://arxiv.org/abs/1706.03762>.
- [39] von Werra, L., Tunstall, L., & Wolf, T. (2022). *Natural Language Processing with Transformers*. O’Reilly Media.
- [40] Zihan Wang, Peiyi Wang, Tianyu Liu, Binghuai Lin, Yunbo Cao, Zhifang Sui, and Houfeng Wang. (2022). HPT: Hierarchy-aware prompt tuning for hierarchical text classification. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3740–3751, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.246. <https://aclanthology.org/2022.emnlp-main.246/>
- [41] Yuan, X., Lin, Z., Kuen, J., Zhang, J., Wang, Y., Maire, M., Kale, A., & Faieta, B. (2021). Multimodal contrastive training for visual representation learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 6991–7000).

- [42] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. *CoRR*, abs/1708.01471, 2017. URL <http://arxiv.org/abs/1708.01471>.