Master of Science in
Electronics and Telecommunication Engineering

Investigating Multimodal Features to Identify Multilingual Offense from Social Media Memes

by

Eftekhar Hossain

ID: 19METE020P

This thesis is submitted in partial fulfillment of the requirement for the degree of Master of Science in Electronics and Telecommunication Engineering

Department of Electronics and Telecommunication Engineering
Chittagong University of Engineering and Technology
Chattogram-4349, Bangladesh.
April, 2023

CERTIFICATION

The thesis titled **Investigating Multimodal Features to Identify Multilingual Offense from Social Media Memes** submitted by **Eftekhar Hossain**, Roll No. **19METE020P**, Session **2019-2020** has been accepted as satisfactory in partial fulfillment of the requirement for the degree of **Master of Science in Electronics and Telecommunication Engineering** on **07/06/2023**.

BOARD OF EXAMINERS

1. _____

Dr. Md Azad Hossain — Chairman (Supervisor)
Professor
Department of Electronics and Telecommunication Engineering
Chittagong University of Engineering & Technology (CUET)
Chittagong-4349

2. _____

Dr. Mohammed Moshiul Hoque — (Co-Supervisor)
Professor
Department of Computer Science and Engineering
Chittagong University of Engineering & Technology (CUET)
Chittagong-4349

3. _____

Dr. Md Saiful Islam — Member
Associate Professor
Department of Electronics and Telecommunication Engineering
Chittagong University of Engineering & Technology (CUET)
Chittagong-4349

4. _____

Dr. Mirza A.F.M. Rashidul Hasan — Member (External)
Professor
Department of Information and Communication Engineering
University of Rajshahi
Rajshahi-6205

CANDIDATE'S DECLARATION

It is hereby declared that this thesis or any part of it has not been submitted elsewhere for the award of any degree or diploma.

Signature of the Candidate

Eftekhar Hossain
ID: 19METE020P

Author's Declaration for Electronic Submission of MRP

I hereby declare that I am the sole author of the MRP. This is a true copy of the MRP, including any final revisions, as accepted by my Examiners.

I hereby also declare that the work contained in this Thesis could be uploaded to the repository of the library, Chittagong University of Engineering and Technology after 1 year from the day of submission.

I authorize Chittagong University of Engineering and Technology to land this MRP to other institutions or individuals for the purpose of scholarly research.

I further authorize the Chittagong University of Engineering and Technology to reproduce this MRP by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my MRP may be made electronically available to the public.

Signature of the Candidate

Eftekhar Hossain
ID: 19METE020P
Department of Electronics and Telecommunication Engineering (ETE),
Chittagong University of Engineering and Technology (CUET).

APPROVAL by the SUPERVISOR

This is to certify that EFTEKHAR HOSSAIN has carried out this research work under our supervision and that he has fulfilled the relevant Academic Ordinance of the Chittagong University of Engineering and Technology so that he is qualified to submit the following Thesis in the application for the degree of MASTER of SCIENCE in ELECTRONICS AND TELECOMMUNICATION ENGINEERING. Furthermore, the Thesis compiles with the PLAGIARISM and ACADEMIC INTEGRITY regulations of CUET.

Signature of the Co-Supervisor

Dr. Mohammed Moshiul Hoque
Professor
Department of Computer Science and Engineering
Chittagong University of Engineering and Technology
Chittagong-4349

Signature of the Supervisor

Dr. Md. Azad Hossain
Professor
Department of Electronics and Telecommunication Engineering
Chittagong University of Engineering and Technology
Chittagong-4349

List of Publications

The following publication is a direct consequence of the research carried out during the elaboration of the thesis and gives an idea of the progression that has been achieved.

1. **Hossain, E.**, Sharif, O. and Hoque, M.M. and Dewan, M.A.A and Siddique, N. & Hossain, Md Azad., "Identification of Multilingual Offense and Troll from Social Media Memes Using Weighted Ensemble of Multimodal Features", Journal of King Saud University-Computer and Information Sciences, Elsevier (Q1, IF = 8.834), 2022.

2. **Hossain, E.**, Hoque, M.M., & Hossain, Md Azad., "An Inter-modal Attention Framework for Multimodal Offense Detection", Proceedings of the 5th International Conference on Intelligent Computing and Optimization 2022 (ICO2022), Springer.

Dedicated to,
my beloved parents,

Amir Hossain and Tahmina Akter

for their love, endless support, encouragement, and sacrifices.

Acknowledgment

First and foremost, I would like to thank Almighty Allah for giving me the strength to complete this thesis. The satisfaction that accompanies the successful completion of this thesis would be incomplete without the mention of people whose ceaseless cooperation made it possible, and whose constant guidance and encouragement crown all efforts with success. I owe thanks to a number of people without whom this thesis would not have been possible. I want to thank my supervisor Professor Dr. Md Azad Hossain and co-supervisor Professor Dr. Mohammed Moshiul Hoque, for their continuous support and assistance. Their inspiration for doing research on the realm of Natural Language Processing (NLP), infinite patience, understanding, and willingness to let me find my own wings have all contributed to my humble development as a practitioner in this field. I am grateful to them for their constant encouragement and stimulating ideas.

Finally, I am thankful to all the members of CUET NLP Lab, faculty members, and staff of the department of ETE, CUET, for their support. My family deserves special thanks for their continuing love, support, encouragement, and sacrifices.

Abstract

In recent years, memes have become a common medium for promulgating offensive views by content polluters in social media. Due to their multimodal nature, memes can easily evade the content regulators' eyes. The proliferation of these undesired or harmful memes can cause a detrimental impact on social harmony. Therefore, restraining offensive memes on social media is of utmost importance. However, analyzing memes is very complicated as they implicitly express human emotions. However, most previous literature did not investigate the joint modeling of various multimodal features and their counteractive single modality features (i.e., Image, Text) for classifying the undesired memes. Rather they focus only on either visual, textual, or particular multimodal features. In addition to that, to the best of our knowledge, no studies have been made on developing a unified framework considering the multilingual context in multimodal offensive meme classification. Our current work argues that combined learning of visual, textual, and multimodal features can compose potential influences for offensive meme detection from the multilingual scenario. This work presents a framework that utilizes the weighted ensemble technique to assign weights to the participating visual, textual, and multimodal models. The state-of-the-art visual (i.e., VGG19, VGG16, ResNet50) and textual (i.e., multilingual-BERT, multilingual-DistilBERT, XLM-R) models are employed to make the constituent modules of the framework. Moreover, two fusion approaches (i.e., early fusion and late fusion) are used to combine the visual and textual features for developing the multimodal models. The evaluations have demonstrated that the proposed weighted ensemble technique improves the performance over the investigated unimodal, multimodal, and ensemble models. The result shows that the proposed approach achieves superior outcomes in two multilingual benchmark datasets (MultiOFF and TamilMemes), with 66.73% and 58.59% weighted f_1 scores, respectively. Furthermore, the comparative analysis reveals that the proposed approach outdoes other existing works by improving approximately 13% and 2% weighted f_1-score gain.

Keywords: Natural Language Processing, Multimodal Learning, Multilingual offense detection, Memes Classification, Cross-Lingual Transfer, Zero-Shot Classification.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the phenomenal rise of social media platforms, the world is witnessing a growing epidemic of online offensive and abusive behavior. A significant portion of social media users has either experienced or witnessed some form of online offense [1]. In these platforms, the users have the freedom to post, comment or share content without modification or intervention of any legal authority [2]. This freedom allows some malign users to dispense offensive content, spread rumors/-fake news, harass communities or individuals, and damage communal harmony. This proliferation of objectionable content in public spaces has detrimental impacts on society [3]. Therefore, to maintain social harmony and ensure the quality of the social network ecosystem it is important to expel such content.

1.1 Offensive Content Detection

Offensive content detection is the task of assigning potential data (i.e., text, image, or both) into predefined offensive categories such as offense or troll. Previous research can be described with two categories.

- **Supervised Offensive Content Classification:** in this approach categories are predefined. It works on the training and testing principles. The algorithms are trained on labeled data sets and give the desired output. During the testing phase, unseen data are fed into the algorithm and it classifies them based on the knowledge gained during the training phase.
- **Unsupervised Offensive Content Classification:** in unsupervised approach categories are not defined. Here learning algorithms try to discover some patterns in data. The algorithm looks for similar patterns and structures in the data points and groups them into clusters. The classification of

the data is done based on the clusters formed. Since offense is a very subjective phenomenon this work employed supervised classification techniques to accomplish the task.

To date, many works have been conducted to detect and mitigate the spread of objectionable content (i.e., Offense, troll, hate, etc) on online platforms. The majority of the works [4, 5, 6] focused on only textual modality to identify troll and offensive content. The SemEval offensive language identification task provides a multilingual dataset to detect the type and target of offensive texts [7]. Kumar et al. [8] summarizes the system's outcome developed on the multilingual troll and aggression dataset.

1.2 Problem Statement

In recent years, the proliferation of offensive memes on social media has become a significant concern, as they can promulgate harmful views and negatively impact social harmony. Existing literature has focused on analyzing memes using single-modality approaches or investigating specific multimodal features, without considering the joint modeling of diverse multimodal features and their counteractive single-modality counterparts (i.e., Image, Text). Furthermore, there is a lack of studies addressing offensive meme detection within a multilingual context. To address these limitations, this thesis aims to develop a unified framework for offensive meme classification that integrates visual, textual, and multimodal features. The proposed framework leverages state-of-the-art visual (e.g., VGG19, VGG16, ResNet50) and textual (e.g., multilingual-BERT, multilingual-DistilBERT, XLM-R) models as constituent modules. Two fusion approaches, early fusion, and late fusion, will be employed to combine the visual and textual features for developing the multimodal models.

The main objective of this research is to investigate the effectiveness of joint modeling of visual, textual, and multimodal features in classifying offensive memes. Additionally, the study will evaluate the performance of the proposed weighted ensemble technique, which assigns weights to the participating models, to improve classification accuracy. The evaluation will be conducted on two multilingual benchmark datasets, namely MultiOFF and TamilMemes. The outcome of this research is a novel framework that demonstrates superior performance compared to existing approaches, as evidenced by substantial improvements in weighted f1 scores. By developing an effective solution for restraining offensive memes on social media, this research contributes to maintaining social harmony and mitigating the detrimental impact of harmful content.

1.3 Motivation

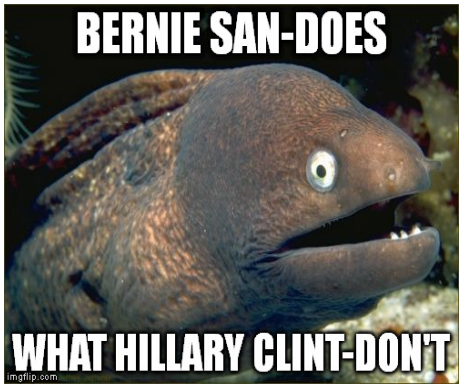
Previous studies reported that social media platforms are utilized to publicize offense, and incite political and religious violence that jeopardize communal harmony and social stability [9]. The viciousness of offensive content is strong enough to trigger massive violence, create mental health problems or even instigate suicide [10, 11]. Moreover, the contents are multilingual in nature as people from diverse regions share their thoughts in their mother tongue. Therefore, it is monumental to develop methods that can flag such multilingual content for reducing unlawful activities and keep the information ecosystem clean from polluted content.

Our major motivations to work in this area is,

- To develop a system that will flag contents that conveys any offensiveness that might try to break communal harmony and publicize distorted propaganda.
- If we are able to detect offensive content it will help our law enforcement agencies to find the perpetrator and stop the undesired events.
- Such a system will ensure the quality of conversations in social spaces.

1.4 Importance

The mode of communication in social media platforms is dramatically transforming day by day. To deceive the existing NLP system for offensive language detection, content polluters adopt new strategies to the changing system. In this regard, posting and sharing *memes* has recently become a popular form of modality to disseminate information on social media since *memes* can propagate information humorously or sarcastically. A *meme* is an image or screenshot with some text embedded into it. Offensive content creators combine images and text in such a way that can attract and mislead viewers. They often misstate or fabricate the fact with highly sentimental content to facilitate rapid dissemination. Consider the example of Figure 1.1 (c), the image is benign, which shows the photographs of two South Indian actors. However, together with the caption, insults their marriage by indicating their age gap. It is cumbersome to correctly infer the meaning of a *meme* considering only visual or textual modality. This multimodal nature of the memes makes it very challenging to differentiate between benign and malign contents. It also aids the propagation of abusive content. Such memes are increasingly used as a way to abuse individuals or attack communities



(a) Offensive

(b) Troll

(c) Troll

Figure 1.1: Few examples where textual content does not convey any exaggerated views, however, when combine with the visual information, it eventually becomes an offensive/troll meme.

based on their race, gender, religion, sexual orientation, or physical appearance [12, 13]. The pervasiveness of these contents poses a direct threat to social peace and communal harmony.

Therefore, an automatic offensive meme detection system should be developed. Responsible agencies are demanding some smart tool/ system that can detect offensive memes automatically. Besides, law and enforcement authorities can take appropriate measures immediately, which in turn helps to reduce virtual harassment, mediated online. An important real-world implication of this thesis will be

- an automated offensive content detection system that will eliminate the process of manually checking the memes, which is tiresome as well as time-consuming. Most important this system will surely help our security agencies to find the perpetrator and his/her derogatory content on social media within a short period.

- An application can be developed along with a controlled environment where we can analyze the memes. By analyzing the memes, the proposed system will be able to predict whether a meme conveys any offensiveness which might demean or derogate any entity such as a person, community, or organization, and tries to break communal harmony, publicize distorted propaganda and excite any specific group of people.

1.5 Challenges

Developing a system that can automatically flag offensive memes is still an arduous problem due to the implicit nature, multi-modality, and complicated structure of the content. The inherent ambiguity of language, computational complexity to audit a large amount of content, the issue of low-resource language, and the contextual understanding of natural language are the major obstacles [14, 15]. Therefore, developing a multimodal offense detection system is intrinsically tricky and complex because

- it requires a holistic understanding of visual and textual information in order to infer the class of a particular meme.
- The implicit meaning of the memes, the presence of ambiguous, humorous, sarcastic terms, and the usage of attractive, comical, theatrical images have made meme classification even more complicated.
- Moreover, the absence of baseline methods to capture features from multiple modalities and the prevalence of multilingual texts have further increased the complexity.

1.6 Contributions

This work proposes a multimodal architecture to learn joint representation simultaneously from visual and textual modalities to identify the offense in social media. The proposed architecture comprises four constituent modules: (i) Visual feature extraction module, (ii) Textual feature extraction module, (iii) Multimodal decision fusion module, and (iv) Multimodal feature fusion module. Each of the modules is trained independently. To extract image features, pre-trained visual (i.e., VGG16, ResNet50, Inception, Xecption) models are used. Extensive investigation is carried out with deep neural networks (i.e., CNN, BiLSTM, Attention) and transformers (i.e., m-BERT, Distil-BERT, XLM-R) to extract

the textual features. Decision and feature fusion modules are responsible for performing aggregation of the extracted features. We perform extensive experimentation on the English offensive meme [16], and Tamil troll meme [17] dataset using the modules mentioned above. After investigating models' predictions, this work proposes a weighted ensemble technique that exploits the strength of individual visual, textual, and multimodal modules. The proposed method (Section 3.5.4) can readdress the softmax probabilities of the partaking models depending on their prior results. Moreover, the effectiveness of the proposed model is empirically validated on multilingual datasets. The key contributions of this work illustrate in the following:

- Present the detailed statistics of the dataset that facilitate the preparation of the models providing useful insights.
- Propose a model that exploits visual, textual, and multimodal features of the memes. Moreover, we investigate the multimodal decision fusion, and feature fusion approaches with contemporary visual and textual models. Finally, we employ an ensemble technique that automatically assigns appropriate weight to the participating modules based on their prior performance on the dataset.
- Empirically evaluates the proposed model on multilingual (English & Tamil) datasets and demonstrates how ensemble technique can enhance the classifier's performance.
- Perform extensive experimentation and compare the performance with a set of visual, textual, and multimodal models. The proposed model outperforms all other techniques by a significant margin, thus setting up a benchmark to compare with in the future.
- Critically analyze the results and errors of the proposed model. Present a quantitative, qualitative analysis of the model's miss-classifications and point out a few directions to resolve these issues.

To the best of our knowledge, the research outcomes presented in the thesis are one of the pioneering works that leverage multimodal features to classify multilingual offenses and trolls from memes. It expects that the resources and system presented in this paper will facilitate further research in this domain.

1.7 Organization of Thesis

The remaining thesis is structured as follows. Section 2 provides Various terminologies related to the work and gives a summary of a few existing works on undesired content detection concerning unimodal and multimodal approaches. Section 3 discusses the task definition, techniques, hyperparameters, and architectures of the constituent modules of the proposed system. Section 4 reports the experimental findings and extensive error analysis of the models. Section 5 points out the prospects of future development with concluding remarks.

Chapter 2

Literature Review

This section will provide a description of some of the important terminologies that will be frequently mentioned in several subsequent sections. Following this, a summary of related literature on offensive content detection will also be discussed.

2.1 Important Terminologies

This section will give a brief summary of some important terminologies such as social media memes, offensive content, troll, abusive content, Multilingual data, etc.

- **Social Media Memes:** A meme is a cultural piece of media that is shared online, often with the intention of inciting certain emotions, such as being humorous. A social media meme is an image, video, or text format that captures the typically humorous thoughts, feelings, or experiences of a specific audience. Basically, an image meme consists of two parts, one is the visual part and another is the embedded text part where each part plays a complementary role to another.

- **Offensive Content:** Offensive content is content that reasonably causes another to experience extreme anger, insult, or disrespect.

- **Troll:** Trolling is when someone posts or comments online to 'bait' people, which means deliberately provoking an argument or emotional reaction. In some cases they say things they don't even believe, just to cause drama.

- **Memes Classification:** A task of categorizing a meme into a predefined category based on the task. For instance, in case of the sentiment analysis, a meme can be classified as positive or negative. Similarly, a meme could be classified as humorous, offensive, or motivating based on its contents.

- **Multilingual Data:** Multilingual represents the contents in multiple languages. When two different language data are analyzed for a particular or different task then it is referred to as multilingual data analysis.

2.2 Related Work

Although a considerable body of work has been conducted to identify troll [18, 19], aggression [20, 4], hate speech [21, 22] and abusive [23, 24] contents from a single modality (i.e. image, text), it is often cumbersome to understand and categorize the contents of a meme considering only one modality. Therefore, it is important to investigate both visual and textual modalities to detect offensive memes. However, researches focusing on detecting such contents from multiple modalities is still in its infancy. This section briefly summarizes previous works on undesired content (i.e., offense, abuse, hate, aggression, troll) detection considering unimodal and multiple modalities.

2.2.1 Text Based Undesired Contents Detection

In the past few years, a series of tasks have been organized to identify offense [7, 32], abuse [33, 34], hate speech [35, 36] and troll [37, 38] from social media. These tasks aimed to detect and categorize abusiveness from multilingual (*English, Arabic, Greek, Tamil, Hindi, and Bengali*) texts. Zampieri et al. [39] develop an English offensive language text dataset. Baseline experimentation is performed with CNN, BiLSTM, and SVM techniques where CNN obtained the maximum macro-f_1 score of 0.80 for the detection task. Wang et al. [40] applied a knowledge distillation method on soft labels to categorize multilingual offensive texts. Tulkens et al. [26] trained multiple SVMs with handcrafted dictionary-based features to identify racist texts. Their system achieved a f_1-score of 0.46, although it does not care about the context of the texts. Zhou et al. [41] employed the deep learning-based fusion approach to identify hate in the SemEval-2019 dataset [21]. Their work applied CNN, BERT, and ELMo to extract the textual features. Fusion of BERT and CNN achieved the highest weighted f_1-score of 0.947. Sharif et al. [42] built an aggressive text identification corpus in Bengali using hierarchical annotation schema. They applied a wide range of machine and deep learning techniques. The combined CNN and BiLSTM acquired the best f_1-score of 0.87 and 0.80 in coarse and fine-grained classification. Debjoy et al. [43] employed a genetic algorithm-based ensemble strategy to identify offense from multilingual texts. Transformers (BERT,

mBERT, DistilBERT) have been used as the ensemble base and achieved 0.78, 0.74, and 0.97 weighted f_1-score in Tamil, Malayalam, and Kannada languages, respectively. A recent work [44] showed that transformer-based models outdo ML and DL-based methods to detect multilingual offensive texts. Statistical features (number of comments, replies, positive, negative votes) are utilized to find trolls in news community forums by Mihaylov et al. [45]. SVM technique with RBF kernel obtained 82-95% accuracy for various feature combinations. Andrew et al. [46] performed experimentation with SVM, LR, RF, and KNN to detect offensive code-mixed YouTube comments. Their work did not consider any semantics and contextual features for the classification. Davidson et al. [15] offered a multiclass hate speech dataset of 25K English tweets. Logistic regression with l2 regularizer and term frequency-inverse document frequency (tf-idf) feature achieve 0.90 macro f_1-score. Bhardwaj et al. [27] applied SVM, LR, RF, and MLP techniques with m-BERT embedding to detect multi-label hostile Hindi posts where SVM achieved the highest f_1-score of 0.84 in coarse-grained classification. Their work did not adopt any deep learning methods to extract sequential features. Gamback et al. [25] tried CNN to classify tweets into four (*racism, sexism, racism & sexism, non-hate*) classes. Experimentation is carried out with random vectors, Word2Vec, and character n-grams where the model acquired 0.78 f_1-score with Word2Vec features. Sadiq et al. [47] developed a combined CNN-BiLSTM-based method over a cyber-troll dataset of 20k tweets. This system can identify cyber-aggressive texts with 92% accuracy, but its performance is inferior for short texts.

2.2.2 Image Based Undesired Contents Detection

Very few researches have been conducted focusing on image-based features to detect offense and trolling since existing models largely depend on textual features. Gandhi et al. [48] developed a system to detect and remove offensive content from an e-commerce catalog. Pre-trained visual models are employed that achieved f_1-score of 0.62. Suryawanshi et al. [17] released a dataset containing troll and not-troll memes in Tamil. They used pre-trained (ResNet, MobileNet) image classification methods to differentiate between meme classes. Although the system achieved a 0.52 macro f_1-score, it performed poorly in the troll class with a recall value of 0.37. This system is failed when the same image with different texts has a heterogeneous interpretation. Balaji and Chinmaya [49] developed a visual feature-based meme classification model. They directly employed the ResNet50 model without any modifications in the layers, resulting in a very poor

weighted f_1-score of 0.48. A CNN-based system is proposed to identify aggression from symbolic images [50] which achieved a weighted f_1-score of 0.89 on a holdout validation set. Connie et al. [51] developed a CNN-based adult content recognition system. Their system used a weighted sum of multiple CNNs, which outperformed the single and average weighted CNN.

2.2.3 Multimodal Based Undesired Contents Detection

Recently, multimodal learning has gained much attention due to its ability to efficiently combine information from multiple modalities into a single learning framework [52]. This method already showed good performance on tasks that involve both visual and linguistic understanding, such as Visual Question Answering [53] and Visual Reasoning [54]. Therefore, researchers are adopting the multimodal technique to detect offensive content from memes since such contents have a detrimental impact on society [55]. To advance research in this domain, Facebook launched a challenge to detect hate speech from multimodal memes [56]. To address this challenge, Lippe et al. [57] developed a multimodal framework using an ensemble of UNITER (UNiversal Image-TExt Representation) [58] which received 0.8053 AUROC scores. Velioglu and Rose [59] proposed a solution with VisualBERT which is a "BERT variant of vision and language" [60]. They adopted an ensemble strategy that helps to achieve an accuracy of 0.765. Few other works have also aggregated linguistic and visual information to detect hateful memes and gained promising performance [61, 62, 63]. Gomez et al. [30] offered a multimodal hate speech dataset containing images and corresponding tweets. Exploration was carried out with unimodal and multimodal architectures, but results revealed that multimodal methods could not outdo the unimodal counterparts. Perifanos et al. [28] developed a multimodal dataset considering *hateful, xenophobic, and racist* tweets. They applied pre-trained Resnet and BERT models for extracting visual and textual features that achieved a weighted f_1-score of 0.947. Rather than BERT, the authors were not employed other variants such as mBERT, and XLM-R which might improve the performance. Nakamura et al. [64] introduced a benchmark dataset for multimodal fake news detection. The authors developed a hybrid (text + image) model to perform fine-grained classification. Maximum accuracy on different classes is achieved with pre-trained BERT (text) and ResNet50 (image) models. Xue et al. [65] proposed a novel multimodal consistency network leveraging the multimodal fusion technique. This method is validated in four widely used multimodal datasets. In another similar work, crossmodal attention residual and multichan-

nel convolutional neural networks were adopted by Song et al. [66]. Kumari et al. [29] proposed a hybrid model where pre-trained VGG-16 is employed to pick out the image features while layered CNN extracted the textual features. These features are optimized by binary particle swarm optimization technique that helps to achieve 0.74 weighted f_1-score. The authors do not experiment with any transformer-based models to comprehend the textual features. Hosseinmardi et al. [67] showed that user metadata and visual features are useful to predict cyberbullying incidents. A variety of textual, visual and multimodal features are analyzed to detect cyberbullying events by Singh et al. [68]. Their results showed that aggregation of both features helps to improve the model's performance. In a similar work, the authors presented a CNN-based unified representation of text and image to detect cyberbullying [69]. In the extended work, they optimized the features using Genetic Algorithm [70]. Results indicate that model's performance has been improved by about 4% using the updated set of features.

Suryawanshi et al. [16] built a multimodal dataset of 743 offensive and non-offensive memes related to the 2016 U.S. presidential election. They adopted the early fusion approach to combine the multimodal features. Although the combined model obtained a 0.50 f_1-score, the text-based CNN model outperformed this by achieving a f_1-score of 0.54. A shared task is organized in EACL-2021 to classify multimodal troll memes [71]. The dataset contains images and associated transcripted texts of the memes in Tamil. Li [72] developed a multimodal model leveraging the pre-trained BERT and ResNet152 architectures. The multimodal attention layer is applied to map text and image features in the same semantic space in this work. The developed model won the shared task by achieving the weighted f_1-score of 0.55. Hossain et al. [31] put together image and text features using the late fusion approach. In the multimodal approach, BiLSTM is employed to extract the textual features while it can be done with transformers. Results revealed that the textual model with XLNet outdoes others by obtaining the f_1-score of 0.52. Hegde et al. [73] experimented with a state-of-the-art vision transformer to extract the image features. However, the system does not perform well and achieved only 0.46 f_1-score. Mishra and Saumya [74] combined features from image and text modalities using a hybrid approach. They used CNN and BiLSTM to obtain the image and text features. The system performed very poorly and attained only a f_1-score of 0.30. Table 2.1 presents a summary of a few works concerning the modality of the dataset, methods, results, and their limitations.

2.3 Research Gap

Despite the growing body of research in meme analysis, these issues are not addressed to date. Suryawanshi et al. [16] applied the late fusion technique to combine multimodal features. Their work employed stacked LSTM and VGG16 to extract textual and visual features. In another work, authors classify troll memes using image features without considering the textual features [17]. Sharma et al. [75] organized a SemEval task to analyze the sentiment and humor of the memes. Their study revealed that multimodal fusion techniques are effective in combining visual and textual features. Few works train textual and visual models independently and combine the model's outcome rather than training a joint multimodal network [76, 77]. Most past studies considered only a single modality (image or text) for offense or troll detection, but they did not exploit the advanced techniques to extract the multimodal features. To utilize the multimodal features concerning modalities should be simultaneously processed. Therefore, we need to design a system from scratch to perform the offensive meme classification tasks in a multilingual scenario. Therefore, the key research questions explored in this work are

'**RQ1:** how to develop a framework leveraging features from visual and textual modalities to identify offense and trolling from memes?

'**RQ2:** how to develop a unified architecture that can effectively work on multilingual multimodal memes?

'**RQ3:** Can conventional ensemble methods be appropriate enough in offensive memes detection?

'**RQ4:** Can zero-shot learning be applied in cross-domain settings for multimodal offense detection?

Table 2.1: Brief literature summary concerning undesired text classification using unimodal and multimodal methods. Here A, F, MF, and WF denote accuracy, f_1-score, macro, and weighted f_1-score respectively.

Article	**Modality of Dataset**	**Approach**	**Results**	**Limitation/Gap**
Gamback et al. [25]	Text [English tweets]	CNN with word embedding and character n-grams	0.78 (F)	Incapable of capturing sequential features as recurrent networks are not used
Tulkens et al. [26]	Text [Dutch posts]	SVM with dictionary-based features	0.46 (F)	Failed to capture the context
Bhardwaj et al. [27]	Text [Hindi comments]	mBERT embedding employed on a set of ML classifiers	0.84 (F)	Ignored the sequential information and limited number of training texts in fine-grained classes
Suryawanshi et al. [16]	Multimodal meme	Late fusion of stacked LSTM and VGG-16	0.50 (F)	Performance can be improved by pre-trained language models
Suryawanshi et al. [17]	Image	Variations of ResNet and MobileNet	0.52 (MF)	Embedded texts in the images are ignored
Perifanos et al. [28]	Multimodal Greek tweets	Combine pre-trained BERT and ResNet models	0.94 (F)	Other variants of transformers are not considered rather than BERT
Kumari et al. [29]	Multimodal posts	VGG-16 and layered CNN with binary particle swarm optimization	0.74 (WF)	Unable to capture the semantic information of the textual modality
Gomez et al. [30]	Multimodal tweets	Employ feature concatenation, spatial concatenation and text kernel models with CNN+RNN	0.68 (A)	Unimodal models achieve better results than the multimodal ones
Hossain et al. [31]	Multimodal meme	Late fusion of textual (BiLSTM) and visual (ResNet50, CNN) features	0.52 (WF)	Textual features can be extracted with transformers

Chapter 3

Offensive Memes Detection Framework

The primary concern of this work is to classify offense and troll from memes on social media. Usually, memes contained multimodal content such as visual and textual. In order to accomplish the task, we investigate several computational models considering only visual, only textual, and a combination of both modalities. State-of-the-art pre-trained convolutional neural networks (i.e., VGG19, VGG16, Xception, InceptionV3, and ResNet50) architectures are employed for visual feature extraction. On the other hand, to obtain textual features, deep recurrent neural networks (i.e., BiLSTM, Attention) and pre-trained transformers (i.e., m-BERT, XLM-R) are applied. This section briefly describes the methods and strategies employed to classify offensive and troll memes. Furthermore, to acquire more robust inferences about the content, both visual and textual features are exploited, and several models are developed by employing multimodal fusion approaches. Figure 3.1 shows the abstract view of the overall system.

3.1 Task Definition

The research objective of this work is to develop a framework **(F)** to identify offense and trolling from memes. The **F** analyzes a set of memes $M = \{m_1, m_2, ..., m_n\}$ and categorize them as offense/troll ($\mathbf{c} = \mathbf{1}$) or not ($\mathbf{c} = \mathbf{0}$). Each meme $m_i \epsilon M$ consists of visual **(v)** and textual **(t)** information and the **F** utilize this information to classify m_i. The task is represented as a mapping, $\mathbf{F} : \mathbf{M}(\mathbf{v}, \mathbf{t}) \rightarrow \mathbf{c}\ \epsilon\ (\mathbf{0}, \mathbf{1})$. Following subsections provides the definition of various meme classes and a brief analysis of datasets.

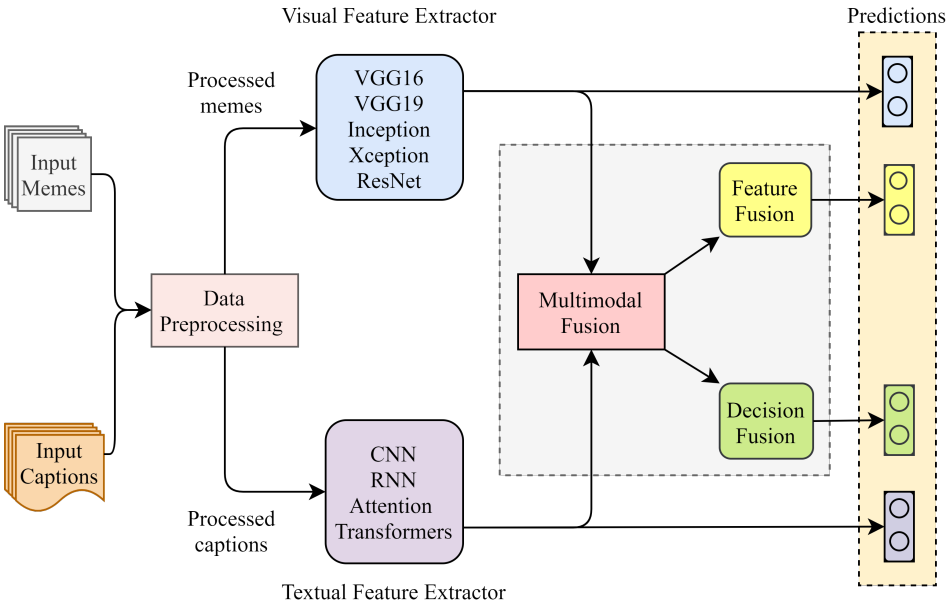


Figure 3.1: Abstract view of multimodal offense and troll detection system.

3.2 Dataset Description

Two benchmark datasets have been utilized to attain the goal: (i) English offensive meme or MultiOFF [16], and (ii) Tamil troll meme or TamilMemes [17]. For ease of understanding, MultiOFF and TamilMemes datasets are denoted as dataset-1 (D1) and dataset-2 (D2), respectively. The first dataset contains offensive and non-offensive memes related to U.S. presidential election. The second consists of troll and not-troll memes where captions are written in Tamil-English code mixed language. Previous studies [16, 17] have manually accumulated these memes from various social media platforms such as Facebook, Whatsapp, Instagram, Twitter, and Pinterest. It is crucial to have a clear understanding of the class labels to develop a successful computational model. The authors [16, 17] defined offense and troll as the following:

- **Offense:** memes that spread an idea/emotion with the intention to demean social identity, harass targeted individuals, community or a minority group.
- **Not-offense:** memes without any offensive content.
- **Troll:** memes which contain offensive texts or images and intend to provoke, offend, abuse or insult individuals, group, or a race.
- **Not-troll:** memes not having any trolling content.

3.3 Dataset Analysis

Each dataset consists of two parts: an image with embedded text and an associated caption. In dataset-1, all the captions are written in English. Most of the captions of dataset-2 are written in Tamil, and a few are in Tamil-English code mixed language. Dataset-1 has 743 memes, of which 303 are offensive, and the remaining are not offensive. Dataset-2 is four-time as large as dataset-1. Out of 2967 instances, 1677 memes are labeled trolls, while the remaining 1290 memes belong to the not-troll class. For model building and evaluation, datasets are partitioned into three mutually exclusive sets: train, validation, and test. Both datasets are summarized in Table 3.2.

Table 3.1: Number of instances in train, validation, and test set for each dataset

	Dataset-1		**Dataset-2**	
	Offensive	Not-Offensive	Troll	Not-Troll
Train	187	258	1026	814
Validation	58	91	256	204
Test	58	91	395	272
Total	303	440	1677	1290

Table 3.2: Training set statistics for textual content

	Dataset-1		**Dataset-2**	
	Offensive	Not-Offensive	Troll	Not-Troll
Total words	4064	5428	12652	4402
Unique words	2065	2569	6200	2487
Max text length	148	139	61	29
Avg. no. of words per text	21.73	21.03	12.33	9.39

The training set is analyzed to get more insights into the data. Table **??** shows the training set statistics, which exhibits both datasets are imbalanced. Not-offensive and troll classes have a higher number of total words and unique words compared to their counterparts. On average, each category on the offensive dataset has 21 words per caption. On the other hand, the captions of the troll dataset are much shorter. The troll class has approximately 12 words per caption, while the not-troll type has only 9 words long. It may be a challenging task to classify trolls due to their shorter text length accurately. Figure 3.2 depicts the number of captions that fall into various length ranges for each of the classes. It is observed that approximately 55% of the captions have less than 20 words.

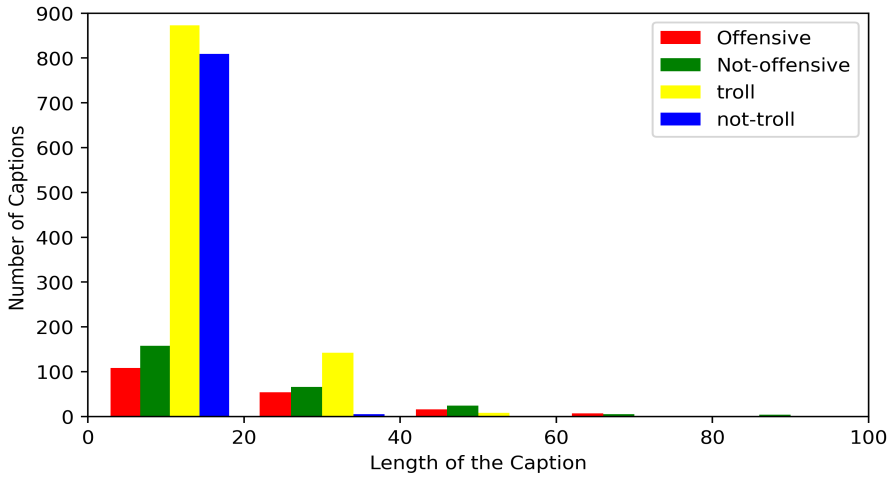


Figure 3.2: Distribution of captions with various lengths in each class

(a) Offensive (b) Not-Offensive

(c) Troll (d) Not-Troll

Figure 3.3: Sample memes of each class: dataset-1 (a,b) and dataset-2 (c,d)

Only a fraction of instances have higher than 40 words. This distribution gives an idea of selecting the input text (based on caption length) during the training phase. Finally, Figure 3.3 presents few sample memes in each class.

3.4 Data Preprocessing

Deep learning techniques are not effective at learning from unprocessed images and texts. Thus, preprocessing is required before feeding them into the networks. For the visual modality, images are transformed into equal sizes of $150 \times 150 \times 3$. The normalization is performed over the pixel matrix of the images to map the pixel intensity values between 0 and 1. Moreover, the Keras[1] image preprocessing function is used to make the input images suitable before driving them into the CNN models.

For textual modality, deep neural networks (DNN) and transformer-based models are utilized. Both architectures take input in a specific format. For DNN, the input texts are converted into a vector of unique numbers. The mapping of this word to the index is obtained using the Keras tokenizer function. Post padding technique is adopted to get equal-length vectors. The maximum text length is determined by analyzing the text length-frequency distribution for each dataset. We choose 50 and 30 as the maximum length for dataset-1 (D1) and dataset-2 (D2), respectively. Similarly, for transformers, we follow the transformer tokenization method for the respective models. After instantiating the tokenizer[2] object, the 'encode_plus' method is used to encode the inputs texts. This method adds a special [CLS] and [SEP] token at the start and end of an input text. It also converts the texts into a vector of unique ids and pad 0's to shorter vectors than the maximum length. Besides, an attention mask is enabled so that the model emphasizes the tokens having unique ids. These 'ids' and 'attention masks' are given as input to the transformer models.

3.5 Methodology

Figure 3.1 comprises three constituent modules namely the visual feature extraction module, textual feature extraction module, and multimodal fusion module. The architectures and parameters of the different modules are discussed in the subsequent subsections.

3.5.1 Visual Feature Extraction Module

Visual features are extracted by using convolutional neural networks. Rather than developing a custom network, the transfer learning approach is employed in this work. In this approach, the parameters of a neural network are trained

[1] https://keras.io/
[2] https://huggingface.co/transformers/main_classes/tokenizer.html

with a large dataset to solve the problem with a smaller dataset for a different task. Several pre-trained CNN architectures such as VGG16, VGG19, ResNet50, InceptionV3, and Xception are considered here. We explain their structure in our system in the following subsections. VGG16 and VGG19 are the variants of the VGG [78] model. VGG16 is a 16-layer whereas VGG19 is a 19-layer deep neural network and a relatively extensive network with a total of 138 million parameters. Both architectures use a fixed kernel size (3×3) in every convolution layer. However, VGG16 and VGG19 models are expensive to evaluate as they use much memory and parameters. InceptionV3 [79] is an extended version of GoogLeNet [80], having several inception modules. The modules consist of a series of stacked convolutional filters (1×1, 3×3, and 5×5) that make the Inception more powerful in learning higher representations with fewer parameters. The standard Inception modules are replaced by 'depthwise separable convolutions' in Xception [81] architecture. It slightly outperforms the Inception model in several large image classification tasks. ResNet50 [82] is another deep CNN network consisting of 50 weight layers. It utilizes the skip connection between layers to resolve overfitting problems largely present in the existing deep neural networks.

To accomplish this purpose, the upper layers of all the models keep non-trainable and only use the weights already pre-trained on the ImageNet [83] dataset for 1000 classes. The top two layers of the models are excluded; instead, a fully connected (FC) layer of 50 neurons is added, accompanied by a softmax layer for prediction. Finally, the models are fine-tuned on dataset-1 and dataset-2. Hyperband [84] optimization technique is adopted to maximize the performance and find the appropriate hyperparameters (i.e., optimizer, learning rate, and so on). The Keras tuner [85] is utilized to implement the optimization process. Several values have been experimented with for each hyperparameter, where the optimum value is selected based on the maximum validation accuracy. Table 3.3 shows the list of hyperparameters chosen for each dataset. All the visual models have been trained with the 'adam' optimizer. Learning rate settled to $1e^{-3}$ for

Table 3.3: Optimum hyperparameters value used for visual models. Here, D1, D2 denote dataset-1 and dataset-2.

Hyperparameters	**Optimum Value**
Number of neurons	50
Optimizer	'adam'
Learning Rate	$1e^{-3}$ (D1), $1e^{-4}$ (D2)
Batch Size	16 (D1), 32 (D2)
Epochs	30

D1 and $1e^{-4}$ for D2. Furthermore, the models are compiled using the *categorical cross-entropy* loss function and training for 30 epochs with a batch size of 16 (for D1) and 32 (for D2). Keras checkpoint is utilized to stop further training when validation accuracy remains unchanged up to five consecutive epochs.

3.5.2 Textual Feature Extraction Modules

Various deep-learning architectures are implemented to obtain features from the textual content. The primary investigation is carried out using RNN and CNN architectures, namely BiLSTM, BiLSTM with CNN, and BiLSTM with attention. Word embedding [86] features are used to train these models. Embeddings are generated through the Keras embedding layer that transformed each word into a 64-element vector. These vectors convey the semantic meaning of the words, which makes learning more accessible, especially for deep neural networks. Pre-trained transformers are also exploited to develop cutting-edge models. The implementation of various textual models is described in the following:

3.5.2.1 BiLSTM

BiLSTM architecture is considered due to its ability to capture long-term dependencies by utilizing both past and future information of a text [87]. The constructed network consists of two BiLSTM layers with 64 and 32 units, respectively. The outputs of the second BiLSTM layer are passed to a fully connected layer of 20 neurons. Afterward, a softmax layer is used for performing the classification. Before the softmax operation, a dropout layer is added with a 10% dropout rate.

3.5.2.2 CNN

Embedding features are propagated into a two-layer CNN architecture. Convolutional layers are equipped with 64 and 16 filters of kernel size (1×2). The extracted features are downsampled by a pooling window of (1×2). An FC layer having 20 neurons takes the pooling features and creates the final hidden representations. Finally, the softmax layer uses this representation for classification.

3.5.2.3 BiLSTM + CNN

This combined network is constructed by slightly modifying the BiLSTM described earlier and the CNN architecture. The embedding features are passed to the BiLSTM layer of 32 units. This layer's last-time step output vectors are

propagated to a convolutional layer having 16 filters of kernel size (1×2). CNN features are further downsampled by a window of size (1×2). The last three layers (i.e., FC, dropout, softmax layer) and their parameters remain unaltered.

3.5.2.4 BiLSTM + Attention

Though BiLSTM effectively captures long-range dependencies, it cannot emphasize the words that are significant for classification. Architecture is defined by adopting the attention mechanism [88] with a BiLSTM network consisting of 32 units to reconcile the weakness of BiLSTM. The forward and backward hidden representations of each word are concatenated and then passed into an attention layer with 20 neurons. Attention weights are assigned to the words through this layer. The higher the significance of a word, the more the weight. Finally, the obtained attention vector of weights is propagated to the softmax layer for the prediction.

3.5.2.5 Transformers

In recent years, models like transformers [89] trained on multilingual and cross-lingual settings achieved the state of the art performance in solving several NLP problems [90, 91, 92]. As we deal with datasets of two different languages, only multilingual and cross-lingual pre-trained transformer models are considered for the investigation to avoid ambiguity in experiments. This work employs three transformer models, namely Multilingual Bidirectional Encoder Representations for transformers (m-BERT), a lighter version of BERT (m-DistilBERT), and a cross-lingual version of robustly optimized BERT (XLM-R). The models culled from the huggingface[3] transformers library and fine-tuned on our datasets with varying hyperparameters. Multilingual-BERT [93] is a large model trained on over 104 languages. We use the 'bert-base-multilingual-cased' model with 12 transformer blocks and 110 million parameters. The distilled version of m-BERT (i.e., m-DistilBERT [94]) with 6 transformer blocks is also considered. This model alleviated the computational cost and maintained the overall system performance up to 95%. The 'distillery-base-multilingual-cased' version is downloaded for the implementation. XLM-Roberta [95] is a transformer model trained in cross-lingual fashion over 100 languages having 125 million parameters. It outperformed BERT in several multilingual benchmark problems [96, 97]. To accomplish our purpose 'xlm-roberta-base' version is utilized. Transformer models take 'token ids' and 'attention masks' as input and provide a contextualized

[3]https://huggingface.co/

embedding vector as output. The obtained vector dimension is 768, and it is taken from the first output of the last hidden state of the transformer models. The embedding vector is then passed to a fully connected layer with 32 neurons, followed by a softmax layer for prediction. The dropout technique is used with a 10% rate before the softmax classification. Similar construction and parameters are used in the last three layers (FC, dropout, and softmax layer) for all the models.

All the textual models are trained with different hyperparameter combinations. The value of the hyperparameters are listed in Table 3.4. A Hyperband tuner is used to find the optimum hyperparameter values. In this implementation, the BiLSTM, CNN, and BiLSTM + CNN models are compiled using the 'adam' optimizer with a learning rate of $1e^{-5}$ and $1e^{-4}$ respectively for dataset-1 (D1) and dataset-2 (D2). Similarly, in the case of D1, a learning rate of $1e^{-5}$, $2e^{-5}$, and $3e^{-5}$ are chosen for m-BERT, XLM-R, and m-distilBERT models, respectively. On the other hand, $1e^{-4}$ (m-BERT), $1e^{-5}$ (XLM-R), and $3e^{-4}$ (m-distilBERT) are selected as the learning rate for D2. A batch size of 16 and 32 is chosen for D1 and D2. All the models trained for 30 epochs with Keras checkpoint to stop the over-training.

Table 3.4: Optimum hyperparameters value utilized for training the textual models. Here, D1, and D2 represents the dataset-1 and dataset-2

Hyperparameters	**CNN (C)**	**BiLSTM (B)**	**B+C**	**B+A**	**m-BERT**	**m-DBERT**	**XLM-R**
Input Length	50 (D1), 30 (D2)						
Embedding Dimension	64				-	-	-
Filters (layer-1)	64	-	16	-	-	-	-
Filters (layer-2)	16	-	-	-	-	-	-
Pooling type	'max'	-	'max'	-	-	-	-
Kernel Size	2	-	2	-	-	-	-
LSTM units (layer-1)	-	64	32	32	-	-	-
LSTM units (layer-2)	-	32	-	-	-	-	-
Neurons (last FC layer)	20			-	32		
Dropout	0.1			-	0.1		
Optimizer	'adam'			'RMSprop'	'adam'		
Learning rate (D1)	$1e^{-5}$			$4e^{-7}$	$1e^{-5}$	$3e^{-5}$	$2e^{-5}$
Learning rate (D2)	$1e^{-4}$			$1e^{-5}$	$1e^{-4}$	$3e^{-4}$	$1e^{-5}$
Epochs	30						
Batch Size	16(D1) , 32(D2)						

3.5.3 Multimodal Fusion Module

Learning from multiple modalities (i.e., image, text, speech, etc.) has become a prominent research issue in recent years. Multimodal learning is widely used for various NLP problems, including image captioning [98] and visual question answering [99]. The joint feature representation of more than one modality is utilized in multimodal tasks [100, 101]. However, classification problems can also be tackled using the same idea [102, 103]. Two approaches used mainly in multimodal problems are decision fusion [104] and feature fusion [105]. In the decision fusion, the softmax outputs of the visual and textual models are combined while an arbitrary hidden layer from multiple modalities is aggregated in the feature fusion technique. After the fusion operation, a single layer neural network or FC layer is trained in both approaches by feeding the combined decision outcomes or hidden feature representations as input. In this approach, the neural network works as a meta learner. For final classification, the softmax operation is performed over the learned features obtained from the meta learner.

Algorithm 1: Process of selecting best 3 visual and textual models

Input: Weighted f_1-scores
Output: Best visual and textual models
$V_f \leftarrow [vf_1, vf_2, ..., vf_N]$ (Weighted f_1 scores of visual models);
$T_f \leftarrow [tf_1, tf_2, ..., tf_M]$ (Weighted f_1 scores of textual models);
$V_m \leftarrow []$;
$T_m \leftarrow []$;
sort$(V_f, V_f + N)$;
sort$(T_f, T_f + M)$;
//choosing best 3 visual and textual models
for $i \epsilon (1, 3)$ **do**
    $V_m.append(V_f[i])$;
    $T_m.append(T_f[i])$;
    $i = i + 1$;
end

This work applies both fusion approaches to develop computational models by utilizing multimodal features. A set of visual $VN = \{v_1, v_2, ..., v_N\}$ and textual $TM = \{t_1, t_2, ..., t_M\}$ models have already been developed (in Sections 3.5.1 and 3.5.2) to classify offense and troll memes. Here, N = 5 and M = 7, which denotes the total number of visual and textual models, respectively. The splicing of each visual model with each textual model for decision and feature fusion approach results in a total of $((N \times M) \times 2) = ((5 \times 7) \times 2) = 70$ different multimodal models. However, the training of these abundant amounts of models is computationally

expensive. It also requires a lot of memory and time. Therefore, this work considers only the best three models from each modality for ease of analysis to develop the multimodal models. The best models are chosen based on their weighted f_1-score on the validation set. The selection procedure of these models is illustrated in algorithm 1. Empirical observations revealed that VGG16, VGG19, and ResNet50 are the best visual models, whereas m-BERT, m-DistilBERT, and XLM-R are the best textual models. Thus considering these six models, we obtain a total of $((3 \times 3) \times 2) = 18$ multimodal models where each fusion approach (i.e., decision, feature) contributed 9 different models.

3.5.3.1 Decision fusion based models

The architectures of the visual (VGG16, VGG19, and ResNet50) and textual (m-BERT, DistilBERT, and XLM-R) models have remained the same as described in Sections 3.5.1 and 3.5.2. Instead of acquiring decisions from the softmax layer of visual and textual models, the softmax outputs of individual models are combined in this approach. Consider, d_{ip}^{V} and d_{jp}^{T} are the softmax outputs for p^{th} sample provided by the visual model $v_i \in VN$ and textual model $t_j \in TM$. Then the decision fusion output can be obtained by equation (3.1).

$$DF_{ij} = d_{ip}^{V} \oplus d_{jp}^{T} \tag{3.1}$$

where $\oplus$ denotes the concatenation operation, and $DF_{ij} \in \mathbb{R}^{1\times 2C}$ represents the decision fusion vector containing softmax probabilities of visual and text modalities. C indicates the number of classes in the dataset.

The vector DF_{ij} is passed to a fully connected layer with 10 neurons. Eventually, the predictions are obtained from a softmax layer. By utilizing this construction, nine multimodal decision fusion-based models namely VGG19 + m-BERT, VGG16 + m-BERT, ResNet50 + m-BERT, VGG19 + DistilBERT, VGG16 + DistilBERT, ResNet50 + DistilBERT, VGG19 + XLM-R, VGG16 + XLM-R, and ResNet50 + XLM-R are developed. The models take the pre-processed image, token ids, and attention masks as input. Due to the language and parametric diversity, we did not find any common hyperparameters for all the models. In case of D1, 'RMSprop' optimizer with learning rate of $2e^{-3}$, and $2e^{-4}$ is used for VGG19 + m-BERT and ResNet50 + m-BERT. Contrarily, VGG16 + m-BERT models have utilized 'adam' with a learning rate of $1e^{-5}$. 'Adam' and 'RMSprop' are chosen respectively for VGG16 + DistilBERT, and ResNet50 + DistilBERT where the learning rate is settled at $7e^{-4}$. Meanwhile, VGG16 + XLM-R, VGG19 + XLM-R, and ResNet50 + XLM-R are compiled using 'RM-

Sprop' optimizer with a learning rate of $1e^{-5}$, $1e^{-4}$, and $5e^{-5}$ respectively. On the other hand, all the models with D2 were compiled using 'RMSprop'. Moreover, the learning rate is settled at $3e^{-5}$ for all of them except the ones having XLM-R ($2e^{-5}$).

3.5.3.2 Feature fusion based models

The feature fusion technique takes advantage of the hidden features extracted by visual and textual models. At first, the softmax layers are excluded from the single modality models. Following this, an FC layer with 20 neurons is added at each modality side. Let, for p^{th} sample, h_{ip}^{V} and h_{jp}^{T} are the hidden or FC layers output provided by the visual model $v_i \in VN$ and textual model $t_j \in TM$. A combined representation of visual and textual features are attained through equation (3.2).

$$FF_{ij} = h_{ip}^{V} \oplus h_{jp}^{T} \tag{3.2}$$

where $FF_{ij} \in \mathbb{R}^{1\times 2h_n}$ represents the feature fusion vector containing features of both modalities and h_n denotes the number of hidden neurons. Subsequently, this unified feature vector (FF_{ij}) is fed into a fully connected layer (with 10 neurons) which is followed by a softmax layer. The number of neurons in the last FC layer is kept unaltered for all the constructed feature fusion models. The model names are similar as described in the earlier paragraph. However, different values of hyperparameters are utilized here. For D1, the visual models (VGG16, VGG19, and ResNet50) with DistilBERT combination are compiled using 'RMSProp' where the learning rate is settled at $2e^{-4}$. Likewise, VGG16 + m-BERT used a learning rate of $1e-4$, while the other two models (VGG19 + m-BERT, ResNet50 + m-BERT) used a rate of $2e-4$. However, in the case of visual models with XLM-R, 'adam' is utilized with a learning rate of $5e{-}4$ (for ResNet50 + XLM-R) and $2e^{-5}$ (for VGG16 + XLM-R, and VGG19 + XLM-R). On the other hand, for D2, all the models used 'RMSprop' (lr = $2e{-}5$) except ResNet50 + m-BERT ('adam', lr = $1e{-}4$) model. All the models are trained for 30 epochs with a batch size of 8 (for D1) and 16 (for D2). Other hyperparameter values have remained the same as described earlier.

3.5.4 Proposed Ensemble Based Framework

The aforementioned developed models can provide acceptable performance in classifying offense and troll memes. Nevertheless, language variation and dataset size largely influence the models' outcomes. Owing to these, distinct models achieved the highest performance for the two datasets. Therefore, to develop a

standard method that can acquire superior outcomes on both datasets, this work proposes a weighted ensemble technique. This approach exploits the strength of multiple models and tries to increase the overall system predictive accuracy. Figure 3.4 shows the overall architecture of the proposed method. It comprises four different models, namely, VGG19, DistilBERT, VGG19 + DistilBERT with decision fusion, and VGG19 + DistilBERT with feature fusion approach. Models are chosen based on their performance (i.e., highest weighted f_1 score) on the validation set.

Model-1 (VGG19) accepts preprocessed memes (m) as input and provides the semantic expression of the visual part by extracting suitable features f^v. The features are then encoded by a 50-dimensional FC layer and passed to a softmax function. The process can be expressed by equation (3.3)-(3.5).

$$f^v = VGG19\ (m) \tag{3.3}$$

$$h_1^v = [NN\ (f^v)]^{50\times 1} \tag{3.4}$$

$$CP^1 = [Softmax\ (h_1^v)]^{2\times 1} \tag{3.5}$$

here, h_1^v, and CP^1 represent the visual features obtained from the neural network (NN) layer, and the class probabilities predicted by model-1.

In the case of model-2 (m-DistilBERT), we utilized the textual features extracted by the pre-trained multilingual DistilBERT model. The initial features are transformed into a 32 dimensional vector. Then class probabilities are calculated by a softmax function (Eqs. (3.6)-(3.8)).

$$f^t = [mDistilBERT\ (c)]^{768\times 1} \tag{3.6}$$

$$h_2^t = \left[NN\ (f^t)\right]^{32\times 1} \tag{3.7}$$

$$CP^2 = \left[Softmax\ (h_2^t)\right]^{2\times 1} \tag{3.8}$$

where c denotes the processed caption, f^t represents the embedding vector provided by DistilBERT, h_2^t indicates the text feature representation done by the neural network, and CP^2 denotes the predicted class probabilities.

Afterwards, using decision fusion approach, model-3 is constructed simply by aggregating the class probabilities CP^1 and CP^2 respectively obtained from

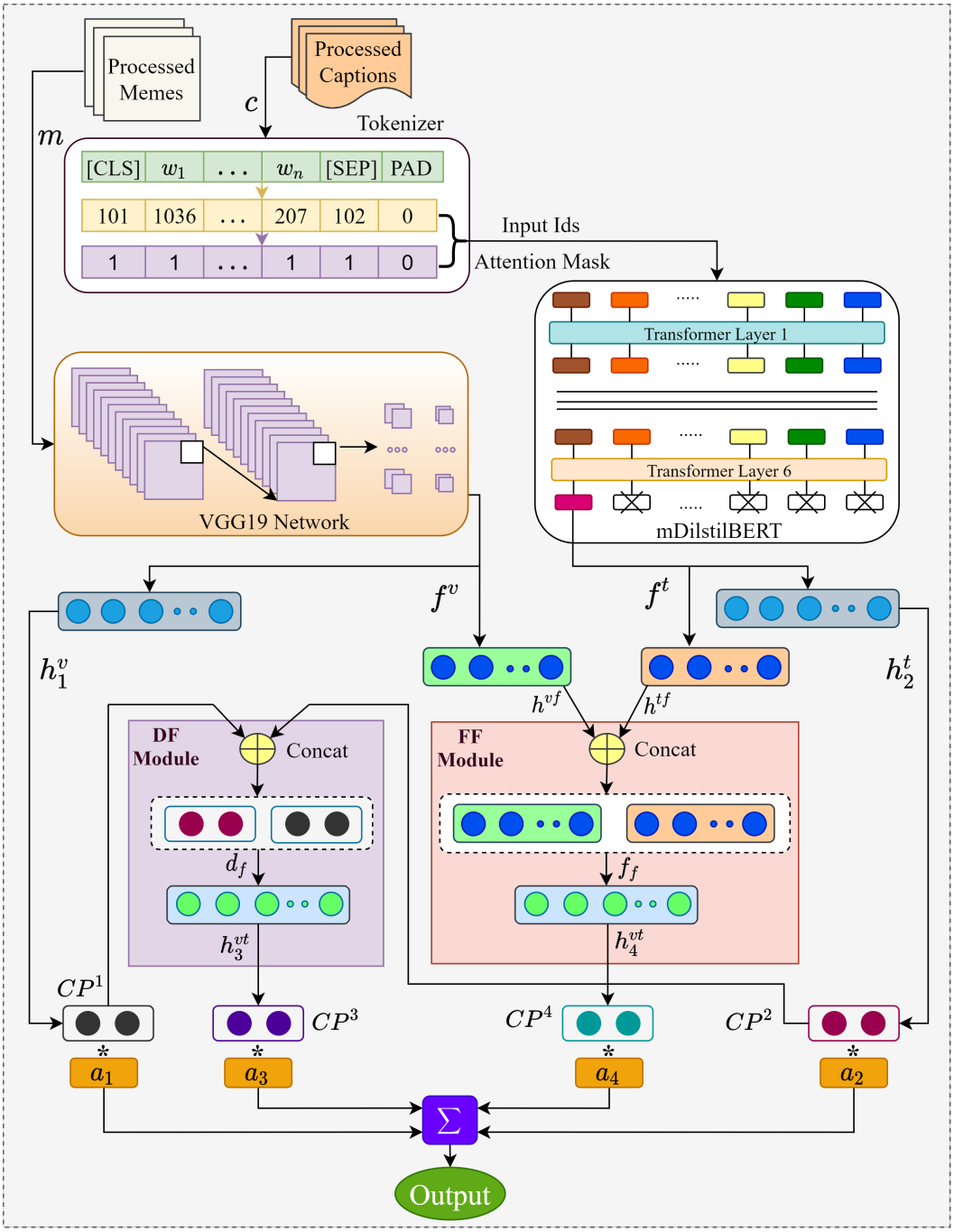


Figure 3.4: Overall architecture of the proposed framework for offensive/troll meme identification.

model-1 and model-2. These combined probabilities are then propagated to a NN resulted in a 10-dimensional feature vector. Eqs. (3.9)-(3.11) described the process of computation.

$$d_f = \left[Concat\left(CP^1, CP^2\right)\right]^{4\times 1} \tag{3.9}$$

$$h_3^{vt} = [NN\ (d_f)]^{10\times 1} \tag{3.10}$$

$$CP^3 = \left[Softmax\ (h_3^{vt})\right]^{2\times 1} \tag{3.11}$$

where d_f denotes the concatenated class probabilities, h_3^{vt} resembles the feature vector containing both visual and textual part, and CP^3 indicates the class probabilities predicted by model-3.

For developing model-4, each visual and textual feature are represented by a 20-dimensional vector. By employing the feature fusion approach, these two vectors are combined and passed to a neural network with 10 neurons, as conferred in Eqs. (3.12)-(3.16).

$$h^{vf} = [NN\ (f^v)]^{20\times 1} \tag{3.12}$$

$$h^{tf} = \left[NN\ (f^t)\right]^{20\times 1} \tag{3.13}$$

$$f_f = \left[Concat\ (h^{vf}, h^{tf})\right]^{40\times 1} \tag{3.14}$$

$$h_4^{vt} = [NN\ (f_f)]^{10\times 1} \tag{3.15}$$

$$CP^4 = \left[Softmax\ (h_4^{vt})\right]^{2\times 1} \tag{3.16}$$

where f_f denotes the feature fusion vector, h_4^{vt} resembles the feature vector containing both visual and textual information, and CP^4 indicates the class probabilities.

To sum up, a set of models $U = \{M_1, M_2, .., M_l\}$ is obtained (where $l = 4$) from the all aforementioned models. From 'm' validation set of samples, a model classifies each instances m_i into one of n predefined classes. For each m_i, model U_j provides a class probability vector of size 'n', $CP_i^j[n]$. Thus, the output of the models become: $\langle CP_1^1[], CP_2^1[], ..., CP_m^1[]\rangle$, $\langle CP_1^2[], CP_2^2[], ..., CP_m^2[]\rangle$, and $\left\langle CP_1^l[], CP_2^l[], ..., CP_m^l[]\right\rangle$. Prior that, the accuracy of the individual models on validation set also measured which can be represented as $a_1, a_2, ..., a_l$. Utilizing these values as weights, the proposed technique compute the final output as described in equation (3.17).

$$E_p = argmax\left(\frac{\forall_{i\epsilon(1,m)} \sum_{j=1}^{l} CP_i^j * a_j}{\sum_{j=1}^{l} a_j}\right) \tag{3.17}$$

here, E_p denotes the vector of $m \times 1$, which contains the ensemble method predictions. The process of calculating ensemble prediction is described in Algorithm 2. Class probabilities of the models are summed after multiplying with the accuracy. Probability values are normalized by dividing with the sum of accuracy. Finally, the output predictions are computed by taking the maximum from the probabilities.

Algorithm 2: Process of the proposed weighted ensemble technique

Input: Class probabilities and Accuracy
Output: Predictions of the W-ensemble
$cp \leftarrow []$ (class probabilities);
$a \leftarrow []$ (accuracy);
$sum = []$ (weighted sum);
for $i \epsilon (1, m)$ **do**
    for $j \epsilon (1, l)$ **do**
        $sum[i] = sum[i] + (cp_i^j[] * a_j)$;
        $j = j + 1$;
    end
    $i = i + 1$;
end
$n\_sum = 0$;
for $j \epsilon (1, l)$ **do**
    $n\_sum = n\_sum + a_j$;
    $j = j + 1$;
end
$P = (sum/n\_sum)$ //normalized probabilities;
$E_p = \arg\max(P)$ // set of predictions;

3.5.5 Process of Weight Calculation

Figure 3.5 exhibits the process of the average ensemble technique. Figure 3.6 shows how readdressed values are calculated using weights and the process of the weighted ensemble technique. Let's consider, we have two models with 80% and 76% accuracy respectively. These classifiers are trying to classify a sample meme into two classes $\{c_1, c_2\}$. The average ensemble technique simply takes the probability score of each class for the classifiers and averages them. Here, the average score for c_1 and c_2 is 0.385 and 0.380 respectively. Then the class with the maximum probability is considered the output class. Hence the output class is c_1. The prior validation accuracy of the models has no impact on the ensemble. The same priority is given to each of the model's softmax predictions.

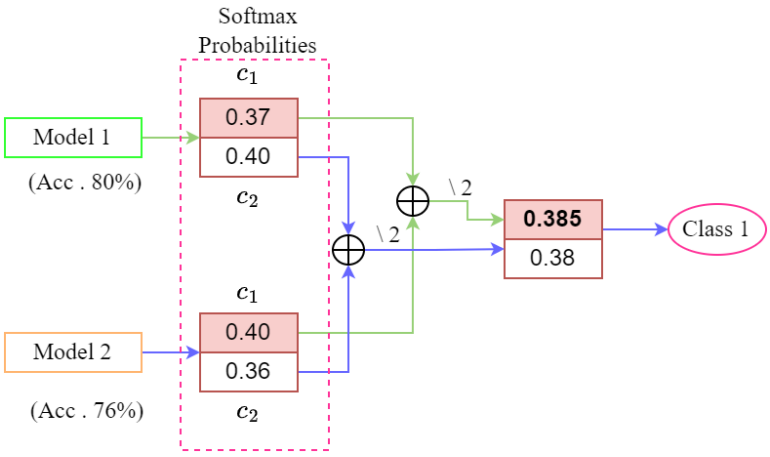


Figure 3.5: Process of average ensemble method

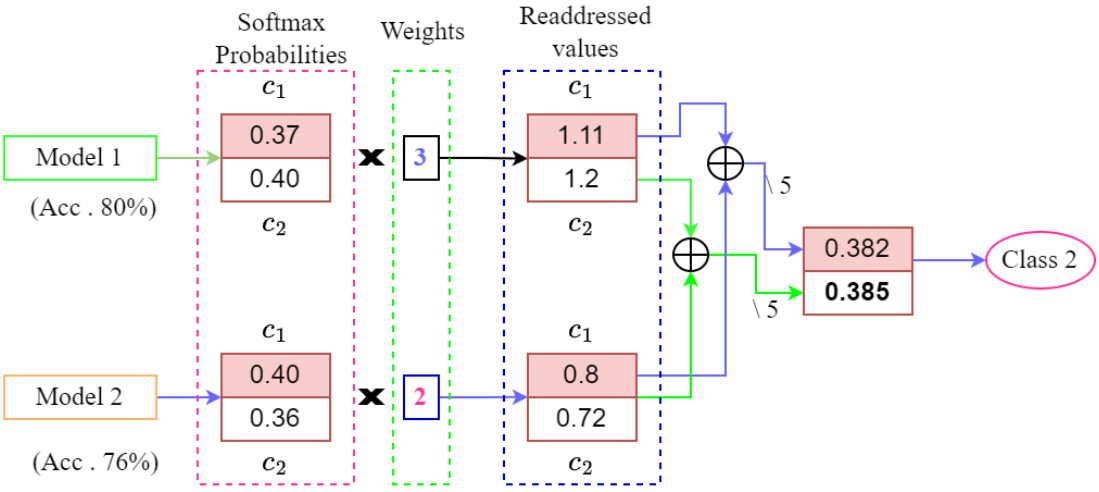


Figure 3.6: Process of proposed weighted ensemble technique

On the other hand, our proposed weighted ensemble technique does not simply take the average of the probabilities. It aggregates the probabilities of the model's after multiplying them with weights. These weights help to put emphasis on the models. The model with higher validation accuracy will get higher weights. This work considers the validation accuracy of the models as the weighting factor. For calculation simplicity, the weights are set to 3 and 2 for the two models respectively. Higher weight is given to classifier 1 since it has higher validation accuracy. After multiplying the initial softmax probabilities with the assigned weights, a set of readdressed values are obtained. These readdressed values are aggregated and divided with the sum of weights. Finally, the output is the one with the maximum probability score. For this example, the final prediction is flipped. With the average ensemble technique class 1 was the output but in the weighted ensemble technique, class 2 is the predicted class.

Chapter 4

Results and Discussions

This chapter provides a brief discussion of experimental settings and evaluation measures. Also, provide a comprehensive performance analysis of the methods employed to identify the offense and troll from social media memes. A detailed error analysis quantitatively and quantitively is also discussed here. Subsequently, several observations will be made based on which the future direction can be identified. Finally, cross-domain transfer outcomes will be presented.

4.1 Experiments

A GPU-facilitated platform, Google colaboratory, is used for conducting the experiments. Data processing and preparation are performed using pandas (1.1.4) and NumPy (1.18.5) libraries. Transformers are accumulated from the Huggingface library, and all the models are implemented with Keras (2.4.0) and TensorFlow (2.3.0). For model evaluation, Scikit-learn (0.22.2) packages are utilized. The models are developed using the train, validation, and test set instances. Train set instances are utilized for model learning, while hyperparameter tweaking and selection are performed based on the validation set. Finally, the trained models are evaluated using the test set instances.

4.2 Evaluation Measures

Various statistical measures are considered for evaluating and comparing the performance of the systems, such as accuracy (A), precision (P), recall (R), misclassification rate (MR), and weighted f_1 score (WE).

- Accuracy (A): is the proportion of correctly predicted observations to the

total number of observations (m).

$$A = \frac{True\ Positive + True\ Negative}{Number\ of\ Samples} \tag{4.1}$$

- Precision (P): calculates the proportion of correctly identified positive observations (c) among the total number of predicted observations as class (c).

$$P = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{4.2}$$

- Recall (R): calculates the proportion of correctly identified positive observations (c) among the total number of actual observations of class (c).

$$R = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{4.3}$$

- Misclassification Rate (MR): calculates how many samples are wrongly classified among the total number of test samples of class (c).

$$MR = \frac{No.\ of\ Incorrect\ Classification\ in\ Class\ (c)}{Number\ of\ Samples\ in\ Class\ (c)} \tag{4.4}$$

- f_1-score: calculated by averaging precision and recall ($F = \frac{2PR}{P+R}$). However, considering the data imbalance problem, we calculate the weighted f_1-score (WF) which is defined as,

$$WF = \frac{1}{m}\sum_{j=1}^{c} F_j N_j,\ m = \sum_{j=1}^{c} n_i \tag{4.5}$$

  here, m, F_j and n_j denotes total samples in test set, f_1-score and number of samples in class (j) respectively.

  The weighted f_1-score metric is considered to determine the superiority of the models. On the other hand, the accuracy metric is utilized as weights in the weighted ensemble method. Other scores such as P, R, and MR are also reported to get more insights about the model's performance in the individual classes.

4.3 Results

The results section is divided into three parts. Initially, we will provide outcomes of unimodal and multimodal models followed by ensemble models' performance.

4.3.1 Unimodal Models Performance Comparison

Table 4.1 presents the performance comparison of the various models developed considering only image and text modality. Concerning visual models, the results exhibited that VGG19 achieved the highest f_1-score of 0.614 and 0.514 respectively for D1 and D2. However, ResNet50 also shows good outcomes of 0.606 (D1) and 0.503 (D2), which is slightly less than the VGG19 f_1-score. Other visual models such as InceptionV3 and Xception perform poorly on both datasets. On the other hand, in the case of the textual approach, transformer models obtained outstanding performance whereas other model's (CNN, BiLSTM, BiLSTM + CNN, BiLSTM + Attention) performance vacillating between $50-56\%$ (D1) and $50-53\%$ (D2). Among the transformer models, XLM-R achieved f_1-score of 0.576 (D1) and 0.556 (D2) while m-BERT score increased $\approx 5\%$ ($f_1 = 0.626$) for D1 and $\approx 1\%$ ($f_1 = 0.561$) for D2. However, m-distilBERT outdoes all the models by achieving the highest f_1-score of 0.654 (for D1) and 0.573 (for D2), respectively. The obtained result is approximately $4-6\%$ higher (in both datasets) than the best visual model (i.e., VGG19) outcomes.

Table 4.1: Performance comparison of visual and textual models on test set where A, P, R, f_1-score denotes accuracy, precision, recall, and weighted f_1-score.

Approach	Models	Dataset-1 (D1)				Dataset-2 (D2)			
		A	P	R	f_1-score	A	P	R	f_1-score
Visual	VGG16	0.577	0.581	0.577	0.579	0.596	0.572	0.596	0.502
	VGG19	0.610	0.621	0.610	**0.614**	0.575	0.536	0.575	**0.516**
	ResNet50	0.624	0.607	0.624	0.606	0.592	0.560	0.592	0.503
	InceptionV3	0.604	0.562	0.604	0.532	0.509	0.456	0.509	0.464
	Xception	0.503	0.493	0.503	0.497	0.572	0.506	0.572	0.478
Textual	CNN	0.510	0.502	0.510	0.506	0.559	0.523	0.559	0.518
	BiLSTM	0.530	0.487	0.530	0.496	0.595	0.568	0.595	0.530
	BiLSTM + CNN	0.590	0.556	0.590	0.550	0.595	0.569	0.595	0.536
	BiLSTM + Attention	0.597	0.568	0.597	0.564	0.548	0.509	0.548	0.507
	m-BERT	0.638	0.625	0.638	0.626	0.608	0.591	0.608	0.561
	m-DistilBERT	0.671	0.662	0.671	**0.654**	0.601	0.583	0.601	**0.573**
	XLM-R	0.591	0.573	0.591	0.576	0.601	0.578	0.601	0.556

4.3.2 Multimodal Models Performance Comparison

The investigation further continued where we utilized both visual and textual information and developed several unified models using two different approaches (i.e., decision fusion, and feature fusion). The three best visual and textual models are chosen for developing the multimodal models. The outcome of different multimodal models is reported in Table 4.2. It is observed that, in the case of decision fusion-based models, ResNet50 + m-BERT obtained an f_1-score of 0.562 (D1) and 0.517 (D2) while other visual models (VGG16, VGG19) with m-BERT do not perform well. Similarly, XLM-R with visual models got the lowest f_1-score ranging between $\approx 50-53\%$ (D1) whereas only VGG16 and VGG19 with XLM-

Table 4.2: Performance comparison of multimodal models on test set. Here, (+) sign denoted the aggregation of visual and textual models. m-DBERT represents the multilingual DistilBERT model.

Approach	Models		Dataset-1 (D1)				Dataset-2 (D2)			
			A	P	R	f_1-score	A	P	R	f_1-score
Decision Fusion	m-BERT +	VGG16	0.483	0.488	0.483	0.485	0.583	0.539	0.583	0.499
		VGG19	0.544	0.541	0.544	0.542	0.589	0.555	0.589	0.513
		ResNet50	0.577	0.558	0.577	0.562	0.513	0.532	0.513	0.517
	m-DBERT +	VGG16	0.537	0.523	0.537	0.528	0.601	0.579	0.601	0.547
		VGG19	0.591	0.628	0.591	**0.595**	0.582	0.583	0.582	**0.583**
		ResNet50	0.570	0.576	0.570	0.573	0.574	0.556	0.574	0.556
	XLM-R +	VGG16	0.497	0.523	0.497	0.503	0.592	0.579	0.592	0.579
		VGG19	0.497	0.528	0.497	0.502	0.567	0.559	0.567	0.567
		ResNet50	0.604	0.563	0.604	0.532	0.574	0.551	0.574	0.548
Feature Fusion	m-BERT +	VGG16	0.584	0.564	0.584	0.567	0.580	0.556	0.580	0.549
		VGG19	0.577	0.547	0.577	0.549	0.604	0.588	0.604	0.529
		ResNet50	0.584	0.567	0.584	0.570	0.568	0.511	0.568	0.489
	m-DBERT +	VGG16	0.604	0.592	0.604	0.595	0.589	0.563	0.589	0.546
		VGG19	0.685	0.681	0.685	**0.660**	0.591	0.568	0.591	**0.557**
		ResNet50	0.611	0.598	0.611	0.600	0.597	0.571	0.597	0.528
	XLM-R +	VGG16	0.570	0.582	0.570	0.574	0.586	0.539	0.586	0.487
		VGG19	0.530	0.524	0.527	0.502	0.568	0.518	0.568	0.499
		ResNet50	0.577	0.589	0.577	0.581	0.608	0.618	0.609	0.508

R obtained acceptable outcome (f_1-score $\approx$ 57%) for D2. However, VGG19 + m-distilBERT model achieved the highest f_1-score of 0.595 and 0.583 for D1 and D2, respectively. Meanwhile, among feature fusion based models, VGG19 + m-distilBERT also got highest performance with both D1 (f_1-score = 0.660) and D2 (f_1-score = 0.557). Other models performance vacillating between $\approx 50-60\%$ (D1) and $\approx 48-54$ (D2) and thus lags almost $6-16\%$ (for D1) and $1-7\%$ (for D2) compared to the best feature fusion model. Thus, the results confirmed that the best feature fusion and decision fusion model outperformed all the unimodal and multimodal models on both datasets. It is not surprising that multimodal approaches have proven superior in identifying offense and troll memes, as the aggregation of both modals' information surely provides significant insights about a meme's overall expression. The best multimodal model obtained an f_1-score of 0.660 (D1) and 0.583 (D2), which is slightly higher than the best unimodal model (i.e., m-DistilBERT) f_1-score 0.654 (D1), and 0.573 (D2), respectively.

4.3.3 Ensemble Models Performance Comparison

The results, as mentioned earlier, confirmed that VGG19, m-distilBERT, VGG19 + m-distilBERT (DF), and VGG19 + m-distilBERT (FF) is the best-performing model in visual, textual, and multimodal contents. Finally, average and weighted ensemble techniques are applied to the various combination of these four models. Table 4.3 presents the outcomes of both ensemble approaches. Results indicate that averaging visual, textual, and feature fusion models improves the performance with f_1-score of 0.665 on the test set of D1. Conversely, different behavior

was observed in D2, where the combination of textual, decision fusion, and feature fusion models provides the highest f_1-score (0.573). Unfortunately, the obtained outcome falls behind almost 1% than the best f_1-score (0.5859) on D2. On the contrary, we used the respective best model's validation accuracy as their weights for the weighted ensemble method. The outcomes exhibited that the proposed weighted ensemble method with the visual, textual, decision, and feature fusion models acquired the highest f_1-score of 0.6673 (D1) and 0.5859 (D2). These results are the highest attained performance that outperformed all the previous outcomes.

Table 4.3: Performance comparison of various models on test set utilizing the *average and weighted ensemble* method. Here, V, T, DF, and FF represent the best visual (VGG19), textual (m-distilBERT), decision fusion (VGG19 + m-distilBERT), and feature fusion (VGG19 + m-distilBERT) models respectively.

Approach	Models	Dataset-1 (D1)				Dataset-2 (D2)			
		A	P	R	f_1-score	A	P	R	f_1-score
Average Ensemble	V + T	0.617	0.609	0.617	0.612	0.588	0.555	0.588	0.522
	V + DF	0.597	0.614	0.597	0.602	0.574	0.535	0.574	0.516
	V + FF	0.638	0.625	0.638	0.626	0.586	0.548	0.586	0.509
	T + DF	0.678	0.669	0.678	0.663	0.594	0.574	0.594	0.566
	T + FF	0.678	0.678	0.678	0.644	0.603	0.584	0.603	0.571
	DF + FF	0.678	0.673	0.678	0.651	0.594	0.573	0.594	0.563
	V + T + DF	0.570	0.565	0.570	0.567	0.585	0.556	0.585	0.540
	V + T + FF	0.678	0.669	0.678	0.665	0.592	0.566	0.592	0.546
	V + DF + FF	0.604	0.592	0.604	0.594	0.588	0.557	0.588	0.532
	T + DF + FF	0.655	0.656	0.655	0.654	0.601	0.583	0.601	0.573
	V + T + DF + FF	0.671	0.662	0.671	0.659	0.592	0.567	0.592	0.548
Weighted Ensemble	V + T	0.637	0.624	0.637	0.6232	0.583	0.551	0.583	0.5314
	V + DF	0.597	0.614	0.597	0.6019	0.574	0.535	0.574	0.5164
	V + FF	0.644	0.630	0.644	0.6133	0.593	0.564	0.592	0.5292
	T + DF	0.677	0.669	0.677	0.6627	0.594	0.573	0.593	0.5658
	T + FF	0.678	0.678	0.677	0.6444	0.597	0.576	0.596	0.5632
	DF + FF	0.671	0.663	0.671	0.6458	0.594	0.572	0.594	0.5625
	V + T + DF	0.597	0.590	0.597	0.5927	0.587	0.561	0.588	0.5457
	V + T + FF	0.677	0.669	0.677	0.6650	0.592	0.566	0.592	0.5460
	V + DF + FF	0.617	0.602	0.617	0.6041	0.592	0.565	0.592	0.5415
	T + DF + FF	0.685	0.686	0.685	0.6536	0.601	0.583	0.575	0.5734
	V + T + DF + FF	0.677	0.669	0.684	**0.6673**	0.583	0.587	0.585	**0.5859**

4.3.4 Insights

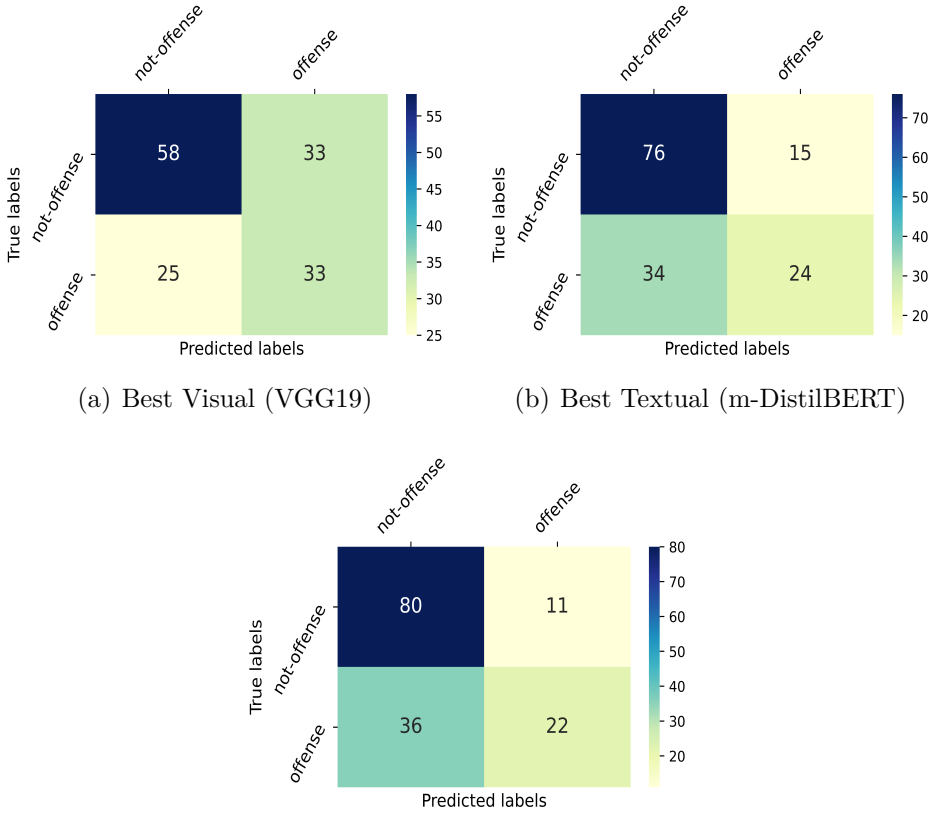
Performance analysis of various models revealed that VGG19 achieved the highest weighted f_1-score among the visual models, whereas m-distilBERT attained maximum performance in textual models. A substantial increase in performance is observed when the visual and textual information is combined. Two distinct fusion approaches with similar models combination (VGG19 + m-distilBERT) outdo all the unimodal approaches in both datasets. Apart from this, in the case of the average ensemble, the combination of textual and decision fusion models shows outstanding performance with D1 whereas the other models' combinations did not provide any consistent outcomes. The inferior performance of one or two models might be the reason for deteriorating the overall performance of different average ensemble models. However, the proposed weighted ensemble method

outperformed all the unimodal and multimodal models in both datasets (D1 and D2). The proposed method's ability to emphasize the model's softmax predictions based on their prior results might be the reason behind the amelioration of performance to a lesser extent.

4.4 Error Analysis

The results confirmed that the proposed weighted ensemble is the best-performing model in classifying offensive and troll memes (Table 4.3). However, to attain more in-depth insights, we performed a thorough analysis of the individual model's error both quantitatively and qualitatively. In order to illustrate the proposed model's preeminence, two other models (i.e., the best visual model, and the best textual model) are considered for the comparison.

4.4.1 Quantitative Analysis



(a) Best Visual (VGG19)

(b) Best Textual (m-DistilBERT)

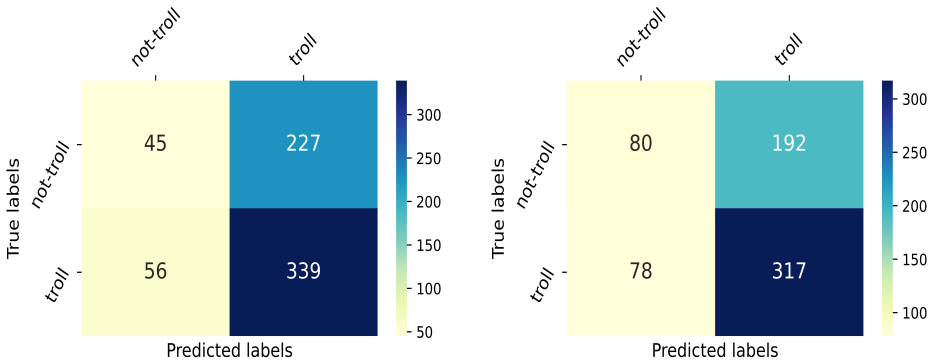
(c) Proposed method

Figure 4.1: Confusion matrices of different models developed for dataset-1 (D1)

Quantitative analysis of models' performance is performed in D1 and D2 by

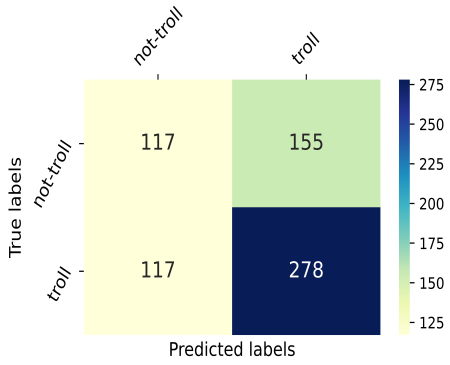
inspecting their confusion matrices. Figure 4.1 shows confusion matrices of three models for the D1 (i.e., offense/not offense). The confusion matrices (a, b, and c) exhibit that the best visual and best textual model misclassified 33 and 15 samples, respectively, whereas the proposed model incorrectly identified only 11 instances. These are the samples where models infer "Offense"; however, the actual labels say "not-offense" (known as false negatives). The textual model showed a significant boost over the visual model, whereas when multimodal features are incorporated along with unimodal features in the proposed method, the misclassification rate falls significantly from 33 to 11.

On the other hand, in the case of the *offense* class, a slight increase is observed in the misclassification of *offense* as *not-offense* (known as false positives) across the models. Figure 4.1 shows these mistakes as 25 by the visual model, 34 by the textual model, and 36 instances by the proposed model. Unfortunately, no improvements were observed from the proposed approach as noticed in the "not-offense" class. Meanwhile, an almost similar scenario is observed with D2, which can be visualized from the confusion matrices shown in Figure 4.2. It observed that the number of misclassified instances (*not-troll* predicted as *troll*)



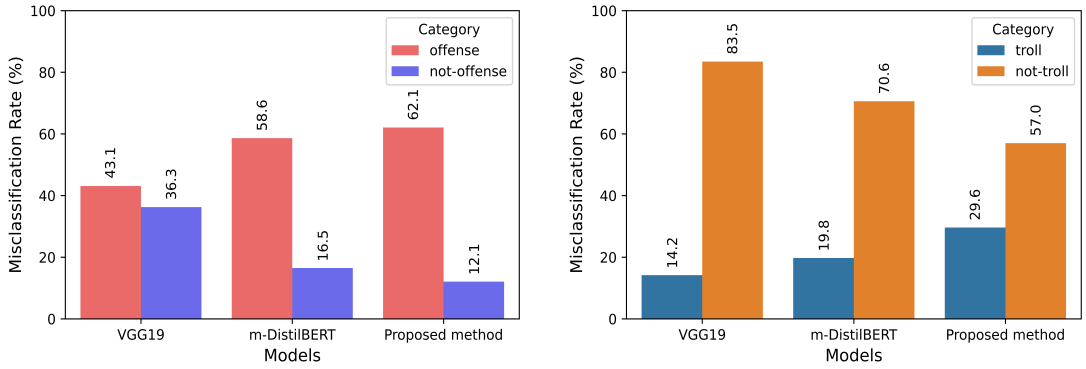
(a) Best Visual (VGG19)

(b) Best Textual (m-DistilBERT)



(c) Proposed method

Figure 4.2: Confusion matrices of different models developed for dataset-2 (D2)



(a) Dataset-1 (D1) (b) Dataset-2 (D2)

Figure 4.3: Proportion of misclassification among the classes of dataset-1 (D1) and dataset-2 (D2)

significantly dropped (227 to 155) from the visual to the proposed model. Though the textual model showed an improvement compared to the visual model, the proposed model reduces the error most for the *not-troll* class. Unfortunately, the error rate dramatically increased in the case of the *troll* class. The mistakes are observed in Figure 4.2 where visual and textual models misclassified 56 and 78 instances. On the contrary, the proposed model incorrectly identified 117 instances as the 'not-troll'. In this case, an experienced of undesirable rise in the false-positive rate is observed.

Figure 4.3 depicts the rate of misclassification (MR) across different classes attained by three models (i.e., best visual model, best textual model, and proposed method) on D1 and D2. From Figure 4.3 (a), it is observed that the MR significantly falls from 36.3% to 12.1% for *offense* class, while in *not-offense* MR rose up to 62.1% from 43.1%. Likewise, concerning D2, the MR gradually increases for the *troll* class, whereas it substantially reduced to 57% from 83.5% (*not-troll* class). The error also indicates that the best visual model (i.e., VGG19) is more appropriate in identifying the *offense* and *troll* classes, providing lower predictions. Furthermore, there is a trade-off between individual class performance as when the error of one class decreased, the other's class is increased. Although the proposed method lessens the error in the *not-offense* and *not-troll* classes, it minimized the combined errors for both datasets, which acquired the highest outcomes (found in Section 4.3).

4.4.2 Qualitative Analysis

Figures 4.4 and 4.5 show some example memes from D1 and D2 that elucidate how the proposed weighted ensemble model can capture information effectively, and

hence, lead to better predictions over the visual and textual models. Besides, to illustrate the mistakes made by the proposed method, some misclassified memes are also presented. Figure 4.4 (a) illustrates the correctly detected sample by the visual model as an *offense* meme, whereas Figure 4.4 (b) depicts the correctly identified sample as *not-offense* by the textual model. Both samples are also correctly classified by the proposed method, which further signifies the capability of acquiring the information by the model when at least one modality can identify the precise class. However, a more profound advantage of incorporating multimodal features is observed explicitly in Figure 4.4 (c), where both visual and textual models reckon the meme as *not-offensive*. On the other hand, the proposed model correctly identified this sample as the *offensive* meme. Concerning D2, the visual model did not find any trolling information from Figure 4.5 (b), whereas, the textual model labeled it as a troll meme. It is probably due to the presence of words like *expectation*, and *reality* in the textual content. Similarly, in Figure 4.5 (c), evaluating visual alone or textual alone yields incorrect predictions; however, when both modalities are jointly evaluated, they provided firm evidence for the proposed model to identify it as a *troll* meme. Furthermore, an interesting case is observed in Figures 4.4 (d) and 4.5 (d), where none of the models detects the actual label of the memes.

4.4.3 Findings

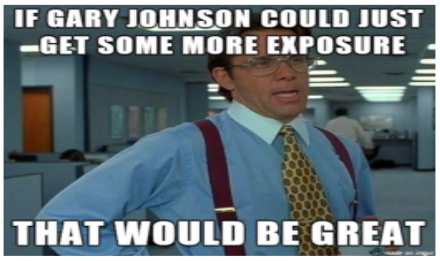
To sum up, quantitative analysis revealed that the model's performance becomes biased towards a particular class (i.e., not-offense/not-troll) for both datasets. The possible reason for this incongruity might be due to the extensive appearance of some strong words such as "Trump", "Hilary", "Bernie", "Communist", "Amala", "Sayessha", "boys", "girls", and "Anna" respectively in the textual content of the offense/not-offense and troll/not-troll classes of memes. In addition to that, dataset-1 (i.e., offense/not-offense) is developed using the memes posted during the presidential election period; thus, some world-famous person faces frequently appeared in the memes of both classes. Likewise, dataset-2 also has plenty of memes with common person faces (i.e., south Indian actors) in troll and not-troll classes. The presence of these consistent visual and textual features among the classes of each dataset made it arduous for the models to differentiate the appropriate class. Indeed, these are the major factors that resulted in one modality approach performing well in one class, and another modality approach yielding better outcomes in other classes. Frenda et al. [106] investigate how the implicit humor of textual content can reveal the aggressive intention towards

a particular entity. Furthermore, analysis of the incorrect predictions shown in Figure 4.4 (d) and Figure 4.5 (d) rendered some other reasons that lead to performance degradation across the classes. To shed light on that, we go through the memes of both datasets and found several disparities regarding contextual complexity and annotation. Among them, one reason is that memes contain very short captions (shown in Figure 4.6 (a)), specifically having less than two words. Moreover, many memes even do not have any captions at all (shown in Figure 4.6 (b)), and their visual content does not provide any meaningful information regarding the class. In particular, out of 743 memes, 65 have a very short caption consisting of less than two words, and 21 memes have no caption (dataset-1). Concerning dataset-2, among 2967 memes, 355 have a short caption (less than two words), whereas 122 memes do not have any caption. Apart from this, it observed that some memes seem offensive and troll; however, the annotated label showed that the memes are from the *not-offensive* and *not-troll* classes. For instance, in Figures 4.6 (c) and 4.6 (d), by examining both visual and textual content, it can be unequivocally said that the memes are from offensive and troll classes, respectively. Mistakes in class labeling during annotation are another prime reason for the models failing to yield improved results. The reasons men-

(a) **Visual modality:** Offense (✓)
Text modality: not-Offense (✗)
Proposed method: Offense(✓)

(b) **Visual modality:** Offense (✗)
Text modality: not-Offense (✓)
Proposed method: not-Offense(✓)

(c) **Unimodal:** Not-offense (✗)
Proposed method: Offense (✓)



(d) **Unimodal:** not-offense (✗)
Proposed method: not-offense (✗)

Figure 4.4: Few correctly and misclassified examples predicted by the proposed and other approaches on the dataset-1 (D1). The symbol (✗) indicates an incorrect classification and the symbol (✓) indicates a correct classification.

(a) **Visual modality:** Not-Troll(✓)
Text modality: Troll (✗)
Proposed method: Not-Troll(✓)

(b) **Visual modality:** Not-Troll (✗)
Text modality: Troll (✓)
Proposed method: Troll (✓)

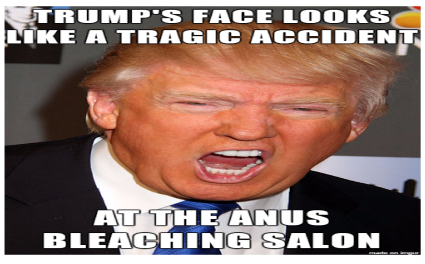
(c) **Unimodal:** Not-Troll (✗)
Proposed method: Troll (✓)

(d) **Unimodal:** Not-Troll (✗)
Proposed method: Not-Troll (✗)

Figure 4.5: Few correctly and misclassified examples predicted by the proposed and other approaches on the dataset-2 (D2).

(a) Very short caption

(b) No caption



(c) Incorrectly annotated as "not-offense"

(d) Incorrectly annotated as "not-troll"

Figure 4.6: Few ambiguous and complicated memes from D1 and D2 illustrating why models failed to detect the actual label of memes.

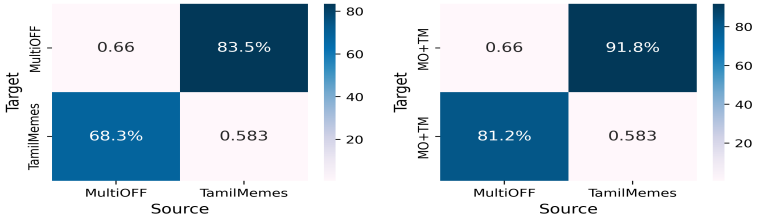
tioned above bring forward new challenges in the direction of undesired meme classification that should need to be handled to develop a more efficient model.

4.5 Cross Domain Transfer

Cross-domain transfer [107] aims at leveraging knowledge obtained from a source domain to train a high-performance learner for offense detection on a target domain. Cross-domain transfer tries to investigate that what extent one dataset performance can benefit from another dataset [108]. Cross-domain transfer can be done in both zero-shot and few-shot settings. In zero-shot settings, a model is trained with one dataset and then the inference is done on the test set of another dataset. On the other hand, the training set of the multiple datasets is combined in a few shot settings and tested on a single dataset. To the best of our knowledge, we first employ the cross-domain transfer approach in multimodal classification We examine cross-domain transfer by fine-tuning the proposed model on a source domain (D1 or D2), and evaluating on a target domain. The performance can be measured by relative zero-shot transfer ability [109]. We refer to it as recovery ratio, since it represents the ratio of how much performance is recovered by changing source domain, given as follows:

$$R(S,T) = \frac{F(S,T)}{F(T,T)} \tag{4.6}$$

here F(S, T) is a model performance for the source domain S on the target domain T. For the recovery ratio, we set a dataset as the target and the remaining ones as the source. When the source and target datasets are the same, recovery would be 1.0. Figure 4.7 shows the cross-domain transfer outcome in zero-shot and few-shot settings. In the case of the zero-shot transfer, 68.3% performance of 0.66 was recovered in the MultiOFF dataset when TamilMemes was the source domain. The recovery percentage is higher at 83.5% in the case of



(a) Zero-Shot (b) Few-Shot

Figure 4.7: Cross domain transfer performance on the two datasets.

the TamilMems dataset when the MultiOFF is the source dataset. Meanwhile, in few-shot settings, the recovery rate is improved with both datasets. For example, 91.8% performance is recovered in Tamil Memes which is approximately 8% higher than zero shot settings. A large amount of boosting around 15% is observed in the MultiOFF dataset when both datasets are used as the training set.

4.6 Comparison With Existing Methods

We have developed several multimodal models by combining the existing state-of-the-art visual and textual models (such as BERT + VGG19, DistilBERT + ResNet50, XLM-R + VGG16, etc.). The performance of the proposed model compared with the existing state-of-the-art techniques [16, 110, 111, 73, 49, 112, 113, 72, 17]. Table 4.4 shows the results of the comparison. The results revealed that the proposed method (weighted ensemble) achieved the best weighted f_1 score of 66.73% ($\approx$ 13% $\uparrow$) as compared to the weighted f_1 score of 54% of the baseline model (i.e., Suryawanshi et al. [16]) for "MultiOFF" dataset (D1). Similarly, for the "TamilMemes" dataset (D2), the proposed model gained the highest weighted f_1 score of 58.59%(1.59% $\uparrow$) as compared to the outcome of the model developed by Suryawanshi et al. [17]). Analysis of the comparison confirmed that the proposed technique outperformed other contemporary works on both datasets. In recent years, a few algorithms have been introduced for

Table 4.4: Comparative analysis of the proposed method with the existing state-of-the-art techniques. *MultiOFF* and *TamilMemes* indicates the dataset-1 (D1) and dataset-2 (D2).

Techniques	**Datasets**	**WF (%)**
Suryawanshi et al. [16]	MultiOFF	54
Mishra et al. [110]	TamilMemes	30
Huang et al. [111]	TamilMemes	40
Hegde et al. [73]	TamilMemes	47
Manoj et al. [49]	TamilMemes	48
Que et al. [112]	TamilMemes	49
Bharathi et al. [113]	TamilMemes	50
Zichao et al. [72]	TamilMemes	55
Suryawanshi et al. [17]	TamilMemes	57
Proposed (weighted ensemble)	MultiOFF	66.73
	TamilMemes	58.59

multimodal learning, such as Visual-BERT [114], VL-BERT [115], CLIP [116]. As far as we know from the most recent literature, these algorithms have not been applied to the offense and troll memes detection problems. However, we aim to investigate these models in the future.

Chapter 5

Conclusion

In this chapter, we briefly summarized the major outcomes of this research and points out some future directions to work on. Finally, publications that are related to this thesis are listed. This work proposes a weighted ensemble-based technique that can effectively learn from all types of features, including visual, textual, and multimodal, for classifying social media memes. Two benchmark multimodal meme datasets viz. *MultiOFF (D1)* and *TamilMemes (D2)* are utilized to evaluate the models. This work investigated various state-of-the-art visual (i.e., VGG19, VGG16, InceptionV3, Xception, ResNet50) and textual (i.e., LSTM, CNN, Attention, m-BERT, m-DistilBERT, XLMR) models. In addition, two different fusion approaches (i.e., decision fusion, and feature fusion) are also used to construct several multimodal models utilizing the image and text features. After analyzing all models' performance on the two datasets, this work proposed a weighted ensemble technique for classifying memes. The proposed technique can readdress the softmax probabilities of the participating models based on their previous outcomes on the datasets. The experimented results revealed that the proposed technique outdoes the unimodal (i.e., image, text), multimodal, and average ensemble models by obtaining the highest weighted f_1 score of 66.73% (MultiOFF dataset) and 58.59% (TamilMemes dataset), respectively. Moreover, the comparative analysis indicated that the proposed technique outcomes are approximately 13% (in 'MultiOFF') and 1.69% (in 'TamilMemes') ahead compared to the current state-of-the-art systems. Thus, results ensured the effectiveness of the proposed technique in detecting offensive and troll memes based on multimodal information. Quantitative and qualitative error analysis shows that it is arduous to identify offenses/trolls expressed implicitly or sarcastically. Moreover, the disparity between visual and textual information and the lack of appropriate methods to analyze the multimodal data made the problem more challenging.

5.1 Limitations

Research in natural language processing can have different types of core contributions. The most common are *dataset-centric* contributions, i.e. new datasets, potentially for new tasks.; *methodology-centric* contributions which are new methods published for existing tasks or datasets. In this thesis, we tried to contribute only from the methodological side. We proposed a dynamic weighting technique that helps to automatically readdress the softmax probabilities of the classifiers. Since it is the first attempt to classify offensive memes from the multilingual scenario, our work has some limitations.

i A meme can simultaneously express multiple types of offense, this work did not consider the domain-specific (i.e., gender, religious) classes.

ii The proposed model is not well generalized across offense classes in both datasets which are prevalent as this is our main goal.

iii Have limited data samples in the dataset

5.2 Future Recommendations

The main purpose of our work was to develop a system for detecting offensive memes using supervised learning techniques. Here we give a meme as input and it will give us feedback on whether the meme is offensive or not in the bilingual scenario. To address the limitations and improve the performance of the system, in the future, we plan to work in the following areas,

- Aim to explore visual attention and transformer architectures (i.e., VisualBERT, VL-BERT, CLIP) to capture strong visual and textual features
- It will be interesting to investigate how multimodal offense or troll detection can be tackled utilizing the multitask learning approach.
- Investigate how the models perform if we transfer knowledge from resource-rich languages using cross-lingual and multilingual transferring techniques.
- The proposed model performance can be investigated with different language datasets such as Bengali, Portuguese, and Greek.
- To develop a web-based system that can filter different online posted memes which are offensive in nature.

5.3 Implications

Several possible implications of this study could include:

- Improved Content Regulation: The proposed framework for offensive meme classification, which incorporates joint modeling of multimodal features and considers the multilingual context, can contribute to the development of more effective content regulation strategies. By accurately identifying offensive memes, social media platforms can take appropriate actions to restrict their dissemination, thereby promoting a safer online environment.

- Enhancing Social Harmony: Offensive memes have the potential to disrupt social harmony by propagating harmful views and creating divisions among users. By effectively restraining offensive memes, this study can contribute to fostering a more inclusive and respectful online community, promoting dialogue, and mitigating conflicts arising from divisive content.

- Multilingual Application: The consideration of the multilingual context in offensive meme classification expands the applicability of the proposed framework. As social media platforms continue to cater to diverse linguistic communities, the ability to detect offensive memes across multiple languages becomes crucial. The findings of this study can help develop more comprehensive and inclusive approaches to content moderation that transcend language barriers.

- Comparative Analysis: The comparative analysis of the proposed approach with existing works provides valuable insights into the strengths and weaknesses of different offensive meme classification methods. This can guide future research in identifying the most effective models and fusion techniques, leading to advancements in the field of multimodal analysis and classification n.

- Weighted Ensemble Technique: The proposed weighted ensemble technique, which assigns weights to participate models, can have broader applications beyond offensive meme classification. It may inspire researchers to explore its utility in other multimodal tasks, such as image recognition, text sentiment analysis, or video classification, where combining diverse modalities can lead to improved performance.

- Benchmark Datasets: The creation and utilization of benchmark datasets, such as MultiOFF and TamilMemes, provide valuable resources for the research community working on offensive meme classification. These datasets

can serve as standardized evaluation benchmarks for comparing and validating future models and algorithms, fostering collaboration and advancing research in the field.

- Methodological Contributions: The development of the proposed framework and the evaluation of different fusion approaches contribute to the methodology of multimodal analysis. The insights gained from this study can inform the design and implementation of future research projects in the domain of multimodal data analysis and classification, benefiting a wide range of applications beyond offensive meme detection.

5.4 List of Publications

The following publication is a direct consequence of the research carried out during the elaboration of the thesis, and gives an idea of the progression that has been achieved.

1. **Hossain, E.**, Sharif, O. and Hoque, M.M. and Dewan, M.A.A and Siddique, N. & Hossain, Md Azad., "Identification of Multilingual Offense and Troll from Social Media Memes Using Weighted Ensemble of Multimodal Features", Journal of King Saud University-Computer and Information Sciences, Elsevier (Q1, IF = 8.834), 2022.

2. **Hossain, E.**, Hoque, M.M., & Hossain, Md Azad., "An Inter-modal Attention Framework for Multimodal Offense Detection", Proceedings of the 5th International Conference on Intelligent Computing and Optimization 2022 (ICO2022), Springer.

Bibliography

[1] M. Duggan, "Men, women experience and view online harassment differently. pew research center. published july 14, 2017."

[2] R. F. Jørgensen and L. Zuleta, "Private governance of freedom of expression on social media platforms: Eu content regulation through the lens of human rights standards," *Nordicom Review*, vol. 41, no. 1, pp. 51–67, 2020.

[3] R. Bannink, S. Broeren, P. M. van de Looij – Jansen, F. G. de Waart, and H. Raat, "Cyber and traditional bullying victimization as a risk factor for mental health problems and suicidal ideation in adolescents," *PLOS ONE*, vol. 9, pp. 1–7, 04 2014.

[4] S. T. Aroyehun and A. Gelbukh, "Aggression detection in social media: Using deep neural networks, data augmentation, and pseudo labeling," in *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, (Santa Fe, New Mexico, USA), pp. 90–97, Association for Computational Linguistics, Aug. 2018.

[5] J. Pavlopoulos, N. Thain, L. Dixon, and I. Androutsopoulos, "ConvAI at SemEval-2019 task 6: Offensive language identification and categorization with perspective and BERT," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, (Minneapolis, Minnesota, USA), pp. 571–576, Association for Computational Linguistics, June 2019.

[6] O. Sharif, E. Hossain, and M. M. Hoque, "Combating hostility: Covid-19 fake news and hostile post detection in social media," 2021.

[7] M. Zampieri, P. Nakov, S. Rosenthal, P. Atanasova, G. Karadzhov, H. Mubarak, L. Derczynski, Z. Pitenis, and Ç. Çöltekin, "SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020)," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, (Barcelona (online)), pp. 1425–1447, International Committee for Computational Linguistics, Dec. 2020.

[8] R. Kumar, A. K. Ojha, S. Malmasi, and M. Zampieri, "Evaluating aggression identification in social media," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, (Marseille, France), pp. 1–5, European Language Resources Association (ELRA), May 2020.

[9] D. U. Patton, J. S. Hong, M. Ranney, S. Patel, C. Kelley, R. Eschmann, and T. Washington, "Social media as a vector for youth violence: A review of the literature," *Computers in Human Behavior*, vol. 35, pp. 548–553, 2014.

[10] R. Bannink, S. Broeren, P. M. van de Looij-Jansen, F. G. de Waart, and H. Raat, "Cyber and traditional bullying victimization as a risk factor for mental health problems and suicidal ideation in adolescents," *PloS one*, vol. 9, no. 4, p. e94026, 2014.

[11] R. A. Bonanno and S. Hymel, "Cyber bullying and internalizing difficulties: Above and beyond the impact of traditional forms of bullying," *Journal of youth and adolescence*, vol. 42, no. 5, pp. 685–697, 2013.

[12] A. Williams, C. Oliver, K. Aumer, and C. Meyers, "Racial microaggressions and perceptions of internet memes," *Computers in Human Behavior*, vol. 63, pp. 424–432, 2016.

[13] J. Drakett, B. Rickett, K. Day, and K. Milnes, "Old jokes, new media–online sexism and constructions of gender in internet memes," *Feminism & Psychology*, vol. 28, no. 1, pp. 109–127, 2018.

[14] X. Zhou, M. Sap, S. Swayamdipta, N. A. Smith, and Y. Choi, "Challenges in automated debiasing for toxic language detection," 2021.

[15] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 11, May 2017.

[16] S. Suryawanshi, B. R. Chakravarthi, M. Arcan, and P. Buitelaar, "Multimodal meme dataset (MultiOFF) for identifying offensive content in image and text," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, (Marseille, France), pp. 32–41, European Language Resources Association (ELRA), May 2020.

[17] S. Suryawanshi, B. R. Chakravarthi, P. Verma, M. Arcan, J. P. McCrae, and P. Buitelaar, "A dataset for troll classification of TamilMemes," in *Proceedings of the WILDRE5– 5th Workshop on Indian Language Data: Resources and Evaluation*, (Marseille, France), pp. 7–13, European Language Resources Association (ELRA), May 2020.

[18] L. G. Mojica de la Vega and V. Ng, "Modeling trolling in social media conversations," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, (Miyazaki, Japan), European Language Resources Association (ELRA), May 2018.

[19] L. S. Mut Altın, A. Bravo, and H. Saggion, "LaSTUS/TALN at TRAC - 2020 trolling, aggression and cyberbullying," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, (Marseille, France), pp. 83–86, European Language Resources Association (ELRA), May 2020.

[20] N. Safi Samghabadi, P. Patwa, S. PYKL, P. Mukherjee, A. Das, and T. Solorio, "Aggression and misogyny detection using BERT: A multi-task approach," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, (Marseille, France), pp. 126–131, European Language Resources Association (ELRA), May 2020.

[21] V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. Rangel Pardo, P. Rosso, and M. Sanguinetti, "SemEval-2019 task 5: Multilingual detection of hate speech

against immigrants and women in Twitter," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, (Minneapolis, Minnesota, USA), pp. 54–63, Association for Computational Linguistics, June 2019.

[22] P. Fortuna and S. Nunes, "A survey on automatic detection of hate speech in text," *ACM Comput. Surv.*, vol. 51, July 2018.

[23] E. W. Pamungkas and V. Patti, "Cross-domain and cross-lingual abusive language detection: A hybrid approach with deep learning and a multilingual lexicon," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, (Florence, Italy), pp. 363–370, Association for Computational Linguistics, July 2019.

[24] B. Vidgen, A. Harris, D. Nguyen, R. Tromble, S. Hale, and H. Margetts, "Challenges and frontiers in abusive content detection," in *Proceedings of the Third Workshop on Abusive Language Online*, (Florence, Italy), pp. 80–93, Association for Computational Linguistics, Aug. 2019.

[25] B. Gambäck and U. K. Sikdar, "Using convolutional neural networks to classify hate-speech," in *Proceedings of the First Workshop on Abusive Language Online*, (Vancouver, BC, Canada), pp. 85–90, Association for Computational Linguistics, Aug. 2017.

[26] S. Tulkens, L. Hilte, E. Lodewyckx, B. Verhoeven, and W. Daelemans, "A dictionary-based approach to racism detection in dutch social media," *CoRR*, vol. abs/1608.08738, 2016.

[27] M. Bhardwaj, M. S. Akhtar, A. Ekbal, A. Das, and T. Chakraborty, "Hostility detection dataset in hindi," 2020.

[28] K. Perifanos and D. Goutsos, "Multimodal hate speech detection in greek social media," in *Preprints*, 2021.

[29] K. Kumari, J. P. Singh, Y. K. Dwivedi, and N. P. Rana, "Multi-modal aggression identification using convolutional neural network and binary particle swarm optimization," *Future Generation Computer Systems*, vol. 118, pp. 187–197, 2021.

[30] R. Gomez, J. Gibert, L. Gomez, and D. Karatzas, "Exploring hate speech detection in multimodal publications," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[31] E. Hossain, O. Sharif, and M. M. Hoque, "NLP-CUET@DravidianLangTech-EACL2021: Investigating visual and textual features to identify trolls from multimodal social media memes," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 300–306, Association for Computational Linguistics, Apr. 2021.

[32] B. R. Chakravarthi, R. Priyadharshini, N. Jose, A. Kumar M, T. Mandl, P. K. Kumaresan, R. Ponnusamy, Hariharan, J. McCrae, and E. Sherly, "Findings of the shared task on offensive language identification in Tamil, Malayalam, and Kannada," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 133–145, Association for Computational Linguistics, Apr. 2021.

[33] S. T. Roberts, J. Tetreault, V. Prabhakaran, and Z. Waseem, eds., *Proceedings of the Third Workshop on Abusive Language Online*, (Florence, Italy), Association for Computational Linguistics, Aug. 2019.

[34] S. Akiwowo, B. Vidgen, V. Prabhakaran, and Z. Waseem, eds., *Proceedings of the Fourth Workshop on Online Abuse and Harms*, (Online), Association for Computational Linguistics, Nov. 2020.

[35] T. Mandl, S. Modha, A. Kumar M, and B. R. Chakravarthi, "Overview of the hasoc track at fire 2020: Hate speech and offensive language identification in tamil, malayalam, hindi, english and german," in *Forum for Information Retrieval Evaluation*, FIRE 2020, (New York, NY, USA), p. 29–32, Association for Computing Machinery, 2020.

[36] C. Bosco, D. Felice, F. Poletto, M. Sanguinetti, and T. Maurizio, "Overview of the evalita 2018 hate speech detection task," in *EVALITA 2018-Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*, vol. 2263, pp. 1–9, CEUR, 2018.

[37] R. Kumar, A. K. Ojha, B. Lahiri, M. Zampieri, S. Malmasi, V. Murdock, and D. Kadar, eds., *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, (Marseille, France), European Language Resources Association (ELRA), May 2020.

[38] R. Kumar, A. K. Ojha, M. Zampieri, and S. Malmasi, eds., *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, (Santa Fe, New Mexico, USA), Association for Computational Linguistics, Aug. 2018.

[39] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the type and target of offensive posts in social media," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, (Minneapolis, Minnesota), pp. 1415–1420, Association for Computational Linguistics, June 2019.

[40] S. Wang, J. Liu, X. Ouyang, and Y. Sun, "Galileo at SemEval-2020 task 12: Multi-lingual learning for offensive language identification using pre-trained language models," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, (Barcelona (online)), pp. 1448–1455, International Committee for Computational Linguistics, Dec. 2020.

[41] Y. Zhou, Y. Yang, H. Liu, X. Liu, and N. Savage, "Deep learning based fusion approach for hate speech detection," *IEEE Access*, vol. 8, pp. 128923–128929, 2020.

[42] O. Sharif and M. M. Hoque, "Identification and classification of textual aggression in social media: Resource creation and evaluation," in *Combating Online Hostile Posts in Regional Languages during Emergency Situation* (T. Chakraborty and et al., eds.), pp. 1–12, Springer Nature Switzerland AG, 2021.

[43] D. Saha, N. Paharia, D. Chakraborty, P. Saha, and A. Mukherjee, "Hatealert@DravidianLangTech-EACL2021: Ensembling strategies for transformer-based offensive language detection," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 270–276, Association for Computational Linguistics, Apr. 2021.

[44] O. Sharif, E. Hossain, and M. M. Hoque, "NLP-CUET@DravidianLangTech-EACL2021: Offensive language detection from multilingual code-mixed text using transformers," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 255–261, Association for Computational Linguistics, Apr. 2021.

[45] T. Mihaylov, G. Georgiev, and P. Nakov, "Finding opinion manipulation trolls in news community forums," in *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, (Beijing, China), pp. 310–314, Association for Computational Linguistics, July 2015.

[46] J. J. Andrew, "JudithJeyafreedaAndrew@DravidianLangTech-EACL2021:offensive language detection for Dravidian code-mixed YouTube comments," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 169–174, Association for Computational Linguistics, Apr. 2021.

[47] S. Sadiq, A. Mehmood, S. Ullah, M. Ahmad, G. S. Choi, and B.-W. On, "Aggression detection through deep neural model on twitter," *Future Generation Computer Systems*, vol. 114, pp. 120–129, 2021.

[48] S. Gandhi, S. Kokkula, A. Chaudhuri, A. Magnani, T. Stanley, B. Ahmadi, V. Kandaswamy, O. Ovenc, and S. Mannor, "Image matters: Detecting offensive and non-compliant content / logo in product images," *CoRR*, vol. abs/1905.02234, 2019.

[49] B. Manoj and Chinmaya, "TrollMeta@DravidianLangTech-EACL2021: Meme classification using deep learning," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 277–280, Association for Computational Linguistics, Apr. 2021.

[50] K. Kumari, J. P. Singh, Y. K. Dwivedi, and N. P. Rana, "Aggressive social media post detection system containing symbolic images," in *Digital Transformation for a Sustainable Society in the 21st Century* (I. O. Pappas, P. Mikalef, Y. K. Dwivedi, L. Jaccheri, J. Krogstie, and M. Mäntymäki, eds.), (Cham), pp. 415–424, Springer International Publishing, 2019.

[51] T. Connie, M. Al-Shabi, and M. Goh, "Smart content recognition from images using a mixture of convolutional neural networks," *Lecture Notes in Electrical Engineering*, p. 11–18, Aug 2017.

[52] L.-P. Morency and T. Baltrušaitis, "Multimodal machine learning: Integrating language, vision and speech," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, (Vancouver, Canada), pp. 3–5, Association for Computational Linguistics, July 2017.

[53] D. A. Hudson and C. D. Manning, "Gqa: A new dataset for real-world visual reasoning and compositional question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[54] A. Suhr, S. Zhou, I. Zhang, H. Bai, and Y. Artzi, "A corpus for reasoning about natural language grounded in photographs," *CoRR*, vol. abs/1811.00491, 2018.

[55] P. Mishra, H. Yannakoudakis, and E. Shutova, "Tackling online abuse: A survey of automated abuse detection methods," *CoRR*, vol. abs/1908.06024, 2019.

[56] D. Kiela, H. Firooz, A. Mohan, V. Goswami, A. Singh, P. Ringshia, and D. Testuggine, "The hateful memes challenge: Detecting hate speech in multimodal memes," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, eds.), vol. 33, pp. 2611–2624, Curran Associates, Inc., 2020.

[57] P. Lippe, N. Holla, S. Chandra, S. Rajamanickam, G. Antoniou, E. Shutova, and H. Yannakoudakis, "A multimodal framework for the detection of hateful memes," *CoRR*, vol. abs/2012.12871, 2020.

[58] Y.-C. Chen, L. Li, L. Yu, A. El Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu, "Uniter: Universal image-text representation learning," in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 104–120, Springer International Publishing, 2020.

[59] R. Velioglu and J. Rose, "Detecting hate speech in memes using multimodal deep learning approaches: Prize-winning solution to hateful memes challenge," *CoRR*, vol. abs/2012.12975, 2020.

[60] L. H. Li, M. Yatskar, D. Yin, C. Hsieh, and K. Chang, "Visualbert: A simple and performant baseline for vision and language," *CoRR*, vol. abs/1908.03557, 2019.

[61] W. Zhang, G. Liu, Z. Li, and F. Zhu, "Hateful memes detection via complementary visual and linguistic networks," *CoRR*, vol. abs/2012.04977, 2020.

[62] A. Das, J. S. Wahi, and S. Li, "Detecting hate speech in multi-modal memes," *CoRR*, vol. abs/2012.14891, 2020.

[63] V. Sandulescu, "Detecting hateful memes using a multimodal deep ensemble," *CoRR*, vol. abs/2012.13235, 2020.

[64] K. Nakamura, S. Levy, and W. Y. Wang, "Fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection," in *Proceedings of the 12th Language Resources and Evaluation Conference*, (Marseille, France), pp. 6149–6157, European Language Resources Association, May 2020.

[65] J. Xue, Y. Wang, Y. Tian, Y. Li, L. Shi, and L. Wei, "Detecting fake news by exploring the consistency of multimodal data," *Information Processing & Management*, vol. 58, no. 5, p. 102610, 2021.

[66] C. Song, N. Ning, Y. Zhang, and B. Wu, "A multimodal fake news detection model based on crossmodal attention residual and multichannel convolutional neural networks," *Information Processing & Management*, vol. 58, no. 1, p. 102437, 2021.

[67] H. Hosseinmardi, R. I. Rafiq, R. Han, Q. Lv, and S. Mishra, "Prediction of cyberbullying incidents in a media-based social network," in *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 186–192, 2016.

[68] V. K. Singh, S. Ghosh, and C. Jose, "Toward multimodal cyberbullying detection," in *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, (New York, NY, USA), p. 2090–2099, Association for Computing Machinery, 2017.

[69] K. Kumari, J. P. Singh, Y. K. Dwivedi, and N. P. Rana, "Towards cyberbullying-free social media in smart cities: a unified multi-modal approach," *Soft Computing*, vol. 24, no. 15, pp. 11059–11070, 2020.

[70] K. Kumari and J. P. Singh, "Identification of cyberbullying on multi-modal social media posts using genetic algorithm," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 2, p. e3907, 2021.

[71] S. Suryawanshi and B. R. Chakravarthi, "Findings of the shared task on troll meme classification in Tamil," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 126–132, Association for Computational Linguistics, Apr. 2021.

[72] Z. Li, "Codewithzichao@DravidianLangTech-EACL2021: Exploring multimodal transformers for meme classification in Tamil language," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 352–356, Association for Computational Linguistics, Apr. 2021.

[73] S. U Hegde, A. Hande, R. Priyadharshini, S. Thavareesan, and B. R. Chakravarthi, "UVCE-IIITT@DravidianLangTech-EACL2021: Tamil troll meme classification: You need to pay more attention," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 180–186, Association for Computational Linguistics, Apr. 2021.

[74] A. K. Mishra and S. Saumya, "IIIT_DWD@EACL2021: Identifying troll meme in Tamil using a hybrid deep learning approach," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 243–248, Association for Computational Linguistics, Apr. 2021.

[75] C. Sharma, D. Bhageria, W. Scott, S. PYKL, A. Das, T. Chakraborty, V. Pulabaigari, and B. Gambäck, "SemEval-2020 task 8: Memotion analysis- the visuo-lingual metaphor!," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, (Barcelona (online)), pp. 759–773, International Committee for Computational Linguistics, Dec. 2020.

[76] T. Morishita, G. Morio, S. Horiguchi, H. Ozaki, and T. Miyoshi, "Hitachi at SemEval-2020 task 8: Simple but effective modality ensemble for meme emotion recognition," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, (Barcelona (online)), pp. 1126–1134, International Committee for Computational Linguistics, Dec. 2020.

[77] L. Bonheme and M. Grzes, "SESAM at SemEval-2020 task 8: Investigating the relationship between image and text in sentiment analysis of memes," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, (Barcelona (online)), pp. 804–816, International Committee for Computational Linguistics, Dec. 2020.

[78] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2015.

[79] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *CoRR*, vol. abs/1512.00567, 2015.

[80] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015.

[81] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, 2017.

[82] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[83] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, pp. 211–252, 2015.

[84] L. Li, K. G. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Efficient hyperparameter optimization and infinitely many armed bandits," *CoRR*, vol. abs/1603.06560, 2016.

[85] T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, *et al.*, "Keras tuner." https://github.com/keras-team/keras-tuner, 2019.

[86] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *ICLR*, 2013.

[87] E. Hossain, O. Sharif, M. Hoque, and I. H. Sarker, "Sentilstm: A deep learning approach for sentiment analysis of restaurant reviews," in *HIS*, 2020.

[88] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," Jan. 2015. 3rd International Conference on Learning Representations, ICLR 2015.

[89] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, (Red Hook, NY, USA), p. 6000–6010, Curran Associates Inc., 2017.

[90] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune BERT for text classification?," *CoRR*, vol. abs/1905.05583, 2019.

[91] X. Liu, K. Duh, L. Liu, and J. Gao, "Very deep transformers for neural machine translation," *ArXiv*, vol. abs/2008.07772, 2020.

[92] D. Lukovnikov, A. Fischer, and J. Lehmann, "Pretrained transformers for simple question answering over knowledge graphs," in *International Semantic Web Conference*, pp. 470–486, Springer, 2019.

[93] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, (Minneapolis, Minnesota), pp. 4171–4186, Association for Computational Linguistics, June 2019.

[94] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *ArXiv*, vol. abs/1910.01108, 2019.

[95] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in *ACL*, 2020.

[96] E. Hossain, O. Sharif, and M. M. Hoque, "NLP-CUET@LT-EDI-EACL2021: Multilingual code-mixed hope speech detection using cross-lingual representation learner," in *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*, (Kyiv), pp. 168–174, Association for Computational Linguistics, Apr. 2021.

[97] X. Ou and H. Li, "Ynu@dravidian-codemix-fire2020: Xlm-roberta for multi-language sentiment analysis," in *FIRE*, 2020.

[98] L. Huang, W. Wang, J. Chen, and X.-Y. Wei, "Attention on attention for image captioning," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4633–4642, 2019.

[99] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Parikh, and D. Batra, "Vqa: Visual question answering," *International Journal of Computer Vision*, vol. 123, pp. 4–31, 2015.

[100] A. Illendula and A. Sheth, "Multimodal emotion classification," *Companion Proceedings of The 2019 World Wide Web Conference*, 2019.

[101] H. Solieman and E. Pustozerov, "The detection of depression using multimodal models based on text and voice quality features," *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, pp. 1843–1848, 2021.

[102] P. Zou and S. Yang, "Multimodal tweet sentiment classification algorithm based on attention mechanism," in *DMLE/IOTSTREAMING@PKDD/ECML*, 2018.

[103] H. Mouzannar, Y. Rizk, and M. Awad, "Damage identification in social media posts using multimodal deep learning," in *ISCRAM*, 2018.

[104] Z.-H. Zhou, "Ensemble learning.," *Encyclopedia of biometrics*, vol. 1, pp. 270–273, 2009.

[105] B. Nojavanasghari, D. Gopinath, J. Koushik, T. Baltrušaitis, and L.-P. Morency, "Deep multimodal fusion for persuasiveness prediction," in *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pp. 284–288, 2016.

[106] S. Frenda, A. T. Cignarella, V. Basile, C. Bosco, V. Patti, and P. Rosso, "The unbearable hurtfulness of sarcasm," *Expert Systems with Applications*, vol. 193, p. 116398, 2022.

[107] B. Myagmar, J. Li, and S. Kimura, "Cross-domain sentiment classification with bidirectional contextualized transformer language models," *IEEE Access*, vol. 7, pp. 163219–163230, 2019.

[108] M. Karan and J. Šnajder, "Cross-domain detection of abusive language online," in *Proceedings of the 2nd workshop on abusive language online (ALW2)*, pp. 132–137, 2018.

[109] I. Turc, K. Lee, J. Eisenstein, M.-W. Chang, and K. Toutanova, "Revisiting the primacy of english in zero-shot cross-lingual transfer," *arXiv preprint arXiv:2106.16171*, 2021.

[110] S. Mishra, S. Prasad, and S. Mishra, "Multilingual joint fine-tuning of transformer models for identifying trolling, aggression and cyberbullying at TRAC 2020," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, (Marseille, France), pp. 120–125, European Language Resources Association (ELRA), May 2020.

[111] B. Huang and Y. Bai, "HUB@DravidianLangTech-EACL2021: Meme classification for Tamil text-image fusion," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 210–215, Association for Computational Linguistics, Apr. 2021.

[112] Q. Que, "Simon @ DravidianLangTech-EACL2021: Meme classification for Tamil with BERT," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 287–290, Association for Computational Linguistics, Apr. 2021.

[113] Bharathi and S. Agnusimmaculate, "SSNCSE_NLP@DravidianLangTech-EACL2021: Meme classification for Tamil using machine learning approach," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, (Kyiv), pp. 336–339, Association for Computational Linguistics, Apr. 2021.

[114] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "Visualbert: A simple and performant baseline for vision and language," *arXiv preprint arXiv:1908.03557*, 2019.

[115] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "Vl-bert: Pre-training of generic visual-linguistic representations," *arXiv preprint arXiv:1908.08530*, 2019.

[116] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, pp. 8748–8763, PMLR, 2021.