

Improving modality interactions in Multimodal Systems

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science
in
Computer Science and Engineering
by Research

by

Tanmay Sachan

2018111023

`tanmay.sachan@research.iiit.ac.in`



International Institute of Information Technology

Hyderabad - 500 032, INDIA

June 2023

Copyright © Tanmay Sachan, 2023
All Rights Reserved

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled “**Improving modality interactions in Multimodal Systems**” by **Tanmay Sachan**, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Adviser: Prof. Vasudeva Varma

To my parents for their unending love and support.

Acknowledgments

I remember it like yesterday, when as a freshman I was struggling to get my code to compile. Fast forward 5 years, and I am writing a whole thesis. IIIT has helped me become the best version of myself, and I would be eternally grateful to the time I spent here.

First and foremost, I would like to thank Prof. Vasudeva Varma for his limitless support. This journey would not have been possible without his constant inputs and endless enthusiasm. Despite having been inducted into IREL at the start of the covid pandemic, I never felt disconnected from him or the lab. Discussions with him have helped elevate my ideas and allowed me to tackle problems from different perspectives, in not just areas of research, but life as well.

Secondly, I would like to thank Prof. Manish Gupta. I had the pleasure of working with him on multiple research statements at IREL and I have never met anyone who has been more of a delight to work with than Manish sir. I remember looking forward to having meetings with him to brainstorm ideas and dive deep into technical details.

As part of research collaborations with Adobe, I also had the opportunity to work with Dr. Balaji Vasan Srinivasan. Balaji sir provided valuable inputs to new fields I was getting my feet wet with, and every meeting with him resulted in an almost instant improvement to our work.

Thirdly, I would like to thank all my fellow lab mates who were always there for engendering discussions and debates on research ideas. Without co-authors and co-researchers like Anshul, Bhavyajeet, Sumanth and Anubhav this thesis would have been a lot shorter! I also cannot thank enough my seniors at IREL - Sayar, Nikhil and Himanshu, who helped me get started in the lab, as well as enabled me to clearly demarcate my goals and not waver in the long journey that led to this thesis.

This acknowledgement would be incomplete without mentioning people who were always there for me, through the thick and thin - Vedansh, Arundhati, Aniket, Anmol, Vansh and Fiza. I am also grateful to my college friend group “Daddycated” for all the laughs and memories across my years spent here.

Lastly, I would like to thank my parents. Whatever I am and whatever I will become, I owe it to them. From helping me pack my bag for school every morning, to pulling all-nighters just so I could finish my homework - they have been the most important constants in my life. It would be unfair not to include my dog Liza (both the first and the second), for the unlimited cuddles during study sessions.

Abstract

Data on the internet is growing at ever increasing rates. People share content with each other (or the world at once) over social media platforms such as the likes of Twitter, and consume content from online media outlets such as news agencies like CNN and Dailymail. Gone are the days of monolithic text-only blogs. Online content today generally encompasses multiple modes, or modalities, of communication coming together to convey information. These modalities include text along with images, audio and video. Machine learning models have long been able to capture and understand images and text as separate entities. The first neural network to be used on images dates to before the advent of internet itself. However, only recently, have machine learning researchers started to adopt the notion that multiple modalities can be understood better in a shared setting and under a common architecture, not as isolated black boxes.

In this thesis, we attempt to use data-driven approaches towards understanding and improving the interaction of modalities within neural network architectures. The first problem we tackle is that of fake news detection in tweets and posts on microblogging websites such as Weibo. Existing works on the problem focus on independently encoding the different modalities and there is a lack of emphasis on shared learning. Our model attempts to generate richer embeddings through a combination of embeddings generated from pre-trained models. We managed to achieve results that beat the state of the art architecture on the problem statement, and the work was accepted as a full paper in the ASONAM 2021 conference.

The second problem we tackle is that of image-aided summarization. While text summarization is a problem that has existed for an eternity, it is not enough to condense information in modality rich sources like news articles. Our model tries to generate textual summaries of articles that demonstrate overlap with image content present in the article, along with selecting the most relevant image from the entire article. We make use of multimodal information retrieval models such as OSCAR to aid in the intermixing of modal information.

The third problem we dive into in this thesis is that of content recommendation. Undertaken as a project at LinkedIn, in this problem we try to improve the ranking of LinkedIn Learning content for each user. Since user history is causal, we enable use of time as a modality through making use of techniques such as Time2Vec and train ranking models jointly to enable better

representation of user history and prediction of future action. Through this methodology, we were able to create a strong recommendation system.

In the fourth problem, we take a look at the availability of multimodal datasets in Indic languages. To enable and enrich research in this domain in an Indic setting, we try to create the first authentic (not translated) dataset of Image-Text pairs in 11 Indian languages. We use deep-learning based caption filtration techniques to prune down the Samanantar dataset, and then use a query simplification algorithm to create queries to download images related to those sentences. Our work enables the creation of large multimodal models such as CLIP and OSCAR within an Indian setting.

Contents

Chapter	Page
1 Introduction	1
1.1 Overview	1
1.2 Multimodality in content	1
1.2.1 Target and Supporting modalities	2
1.2.2 Mixed modalities	3
1.3 Multimodal Interactions	3
1.3.1 Fake news Detection	3
1.3.2 Image-guided summarization	4
1.3.3 Personalized content ranking	4
1.4 Expanding Multimodal Work	4
1.5 Contributions of this Thesis	5
1.6 Thesis Workflow	5
2 Related Work	7
2.1 Approaches to solving multimodal tasks	7
2.2 Fake News Detection	9
2.2.1 Knowledge based	9
2.2.2 Style/content based	9
2.2.2.1 Multimodal fake news detection	9
2.3 Summarization	10
2.3.1 Extractive summarization	10
2.3.2 Abstractive summarization	10
2.3.3 Multimodal summarization with multimodal output	11
2.4 Multimodal Datasets	12
2.4.1 Samanantar (text-only dataset)	12
3 Multimodal Fake News Detection	13
3.1 Introduction	13
3.2 Datasets	16
3.2.1 Twitter Media-Eval dataset	16
3.2.2 WeiboA dataset	17
3.2.3 WeiboB dataset	17
3.2.4 Model	17
3.2.4.1 Inputs	18
3.2.4.2 Cross-modal attention and shared Feedforward layers	19

3.2.4.3	Dot Product Scaling to aid classifier	21
3.2.4.4	Classification layers	21
3.2.4.5	Training	22
3.3	Baselines	22
3.4	Experimental Settings	24
3.4.1	Text processing	24
3.4.2	Image Processing	24
3.4.3	Hyperparameters	24
3.5	Comparison	25
3.6	Results Observation	27
3.7	Ablation	27
3.7.1	Ablation Observations	29
3.8	Case Studies	29
3.9	Conclusion	31
4	Image-aided summarization	32
4.1	Introduction	32
4.2	Multimodal Summarization	34
4.2.1	Encoder	34
4.2.1.1	Text Encoder	35
4.2.1.2	Image Encoder	35
4.2.2	Multimodal Image and Sentence Scorer	35
4.2.3	Sentence Simplification	36
4.2.4	Scorer	36
4.2.5	Decoder	37
4.2.6	Training Methodology	37
4.3	Baselines	37
4.4	Experiments	38
4.4.1	Dataset statistics	38
4.4.2	Implementation Details	38
4.4.3	Evaluation Metrics	38
4.4.4	Results	39
4.5	Discussions	40
4.6	Conclusion	40
5	Personalized content recommendation	41
5.1	Introduction	41
5.2	Data	42
5.3	Problem Description	42
5.4	Model	43
5.4.1	Interaction Embedding	43
5.4.2	Transformer Encoder	44
5.4.3	Training	44
5.5	Results	45
5.6	Conclusion	45

6	Indic Multimodal Dataset creation	46
6.1	Introduction	46
6.2	Quality of a caption	47
6.3	Dataset creation	49
6.3.1	Classifier	49
6.3.1.1	Datasets for training	49
6.3.1.2	Training splits	50
6.3.1.3	Results	50
6.3.2	Image Collection	50
6.4	Dataset Samples	51
6.5	Conclusion	51
7	Conclusion and Future Work	53
7.1	Conclusion	53
7.2	Future Work	54
7.2.1	Multimodal Fake News Detection	54
7.2.2	Image-aided Summarization	54
7.2.3	Indic Multimodal Dataset	55
	Bibliography	57

List of Figures

Figure	Page
1.1 Example with Visual target modality and Textual supporting modality.	2
1.2 Example with Textual target modality	2
1.3 Example of mixed modality content.	3
2.1 Structure of most early classification multimodal models. " \oplus " refers to some combination operator.	7
2.2 Structure of imagebert. The transformer architecture is used to take both image and text as input at the same time.	8
2.3 Examples of extractive and abstractive summarization.	11
2.4 Example of multimodal summarization with multimodal output.	11
3.1 Fake News examples from the Twitter and Weibo datasets. (Translation for the rightmost example using Google Translate: "The Amway boss is dead! Only 56 years old, eating Nutrilite for 27 years, so ironic!")	14
3.2 <i>SCATE</i> System Architecture	16
3.3 Process of multimodal compact bilinear pooling.	21
4.1 Model Overview	35
5.1 Sample of a user's activity in the dataset.	42
5.2 Transformer Encoder	44
5.3 Model Architecture	45
6.1 Classifier architecture	49

List of Tables

Table	Page
3.1 # Posts and # Images across Twitter dataset	17
3.2 Twitter Dataset Class Distribution	17
3.3 WeiboA Dataset Class Distribution	18
3.4 WeiboB Dataset Class Distribution	18
3.5 # Posts and # Images across Weibo datasets	18
3.6 Results on Twitter Dataset	25
3.7 Results on WeiboA Dataset	26
3.8 Results on WeiboB Dataset	26
3.9 Twitter ablation results	28
3.10 WeiboA ablation results	28
3.11 WeiboB ablation results	29
3.12 Case Studies. Text Contribution and Image Contribution correspond to SCATE's.	30
4.1 Text in orange shows the textual overlap between gold and predicted summaries, while text in blue shows the additional context our model gathers from the image.	34
4.2 Comparison of our models with the baselines.	39
6.1 Examples from the multimodal Indic dataset.	52

Chapter 1

Introduction

1.1 Overview

This thesis explores ways in which machine learning systems can develop a holistic understanding of modern media. Here modern media refers to the kind of content consumed by a majority of users on the internet, i.e., content which contains all sorts of modalities (text, images, audio, video). Models which understand and interpret multimodal data better have all sorts of applications - in search engines, generative art, visual question answering, image captioning - the list is endless. In this introduction, we introduce various terms and concepts that would be used throughout the thesis.

The first part of this thesis focuses on the exploration of *Multimodal interactions* within neural networks. This part forms the core of the thesis. To accomplish this, we tackle multiple research statements spread over the domain and try to beat, or improve upon in some way, the state-of-the-art. These statements include Fake News Detection, Image-aided summarization and Personalized Content Recommendation. The results we obtain from these experiments help us understand better how neural networks learn in such a setting, along with their drawbacks.

The second part of this thesis focuses on improving the state of multimodal research in Indic languages. While this part does not concern itself with neural interactions of any sort, the findings from this part help produce an authentic Image-pair parallel dataset in 11 Indian languages that can be used to train and/or enrich multimodal models such as OSCAR and CLIP with an understanding of Indian context.

1.2 Multimodality in content

Multimodality refers to the existence of distinct sub-groups, or *modalities* within content that are perceived by humans uniquely through an individual, or a combination of, bodily systems. For example, eyes enable the perception of images, while ears grant the beholder the ability to listen. The corresponding modes are hence visual and auditory respectively.

Multimodal information can generally be divided into two categories, one where there is an explicit target and supporting modality, whereas the other where such a distinction is not obvious.

1.2.1 Target and Supporting modalities

In a lot of cases, multimodal content consists of a target modality and other supporting modalities. The target is expected to be the key source of information, whereas the supporting modalities add context and supplementary data. When encountering such content, a person would be able to takeaway information by only considering the target modality as well. However, the information might end up being misleading or lacking context.

For example, in 1.1, we see an instagram post. Instagram as a social media relies on the sharing of images. In most instagram posts, we see that the target modality is Visual in nature, however it is generally accompanied with a supporting modality of text. Similarly, on a platform like twitter (shown in 1.2) generally focuses on text as the target modality.



Figure 1.1 Example with Visual target modality and Textual supporting modality.



Figure 1.2 Example with Textual target modality

Supporting modalities play a critical role in providing additional context to the target modality and have the ability to completely alter the meaning of the target.

One example of supporting modalities altering the meaning of the target is in the case of *texting* over messaging applications. A person exclaiming "I loved the movie!" has text as its target modality, however, it could have a completely different meaning if the text is augmented with the tone (auditory input) or the facial expressions (visual input) making use of voice-notes/images and incorporating phrasal techniques such as sarcasm.

1.2.2 Mixed modalities

In this section, we take a look at content where the distinction between a target and support is blurred. In this kind of content, any kind of modality taken alone would fail to convey information.

An example of this phenomenon is found in the majority of videos on video sharing platforms such as Youtube. In 1.3 we see the example of such a video, where taking the visual input alone or the audio input alone would fail to evoke any laughter from the viewer (or the listener).



Figure 1.3 Example of mixed modality content.

In this thesis, we strictly deal with problems that involve a target and a supporting modality, i.e., content with mixed modalities is out of scope of this thesis. The solutions to the problems we deal with are focused towards enriching neural representations of the target modality by exploiting the supporting modalities to improve model performance.

This thesis is divided into the following 2 major modules, where the first one deals with understanding and improving the multimodal interactions within neural networks and the second one deals with expanding work on this domain under an Indian context.

1.3 Multimodal Interactions

In this module, we develop neural network architectures for solving multiple problem statements involving multimodal content.

1.3.1 Fake news Detection

The first problem we deal with is that of Fake News Detection. Our aim for this statement is to take a piece of text along with its corresponding image and detect whether the information conveyed is potentially fake or not. The text and images in this statement are sourced through a dataset consisting of Twitter *tweets* and posts from the Chinese microblogging website Weibo.

We run multiple experiments over different models, and make use of various pre-trained architectures as submodules. The best performing model we get consists of a shared transformer encoder that encodes the image and text representations from submodules together, followed by a bilinear pooling and classification. This model achieves a state of the art accuracy over all the datasets and was published at ASONAM 2021 under the name of **SCATE**.

1.3.2 Image-guided summarization

In this problem statement, we attempt to make a summarization system which takes as input a long body of text and a stream of images, and returns a compact summary along with the most relevant image out of all provided. Moreover, the compact summary generated does not consist of content from the text alone, but also tries to pull additional context from the images to be more fulfilling to the user. For the architecture, we make use of multiple encoding techniques and pre-trained multimodal IR models. We also make use of techniques such as sentence-simplification which are discussed in more detail in chapter 4. Our model achieves close to state of the art scores on metrics like ROUGE, and at the same time is smaller and much more efficient. We label this model as **MMSumm**.

1.3.3 Personalized content ranking

This problem was undertaken as an internship project at LinkedIn. For this problem, we take as input user actions (such as querying, bookmarking, viewings) on each document, and then predict user's action on a target document. We create a model that takes as input these actions, along with time, and then tries to embed them together as a time series, followed by a multi-class classification of the predicted action. This model then results in a creation of *user specific embeddings* that are then used as part of a bigger search recommendation model.

1.4 Expanding Multimodal Work

In this second module, we explore ways to create authentic multimodal data in Indian languages, and their feasibility. We start off with understanding the Samanantar dataset from AI4Bharat, and then attempt to prune it using deep learning paired with rule based methods to result in sentences which can be possible captions to images. Our pruning results in a subset of the bigger dataset, however these sentences can then be queried onto search engines like Google Images and Bing to give high quality images. Using these image-text pairs, we can enable researchers to build high quality multimodal models that possess an understanding of diverse Indian context, spread over 11 languages.

1.5 Contributions of this Thesis

This section summarizes the key contributions of this thesis:

1. To study and understand how modalities influence each other within deep learning architectures. We accomplish this by studying model performances in different multimodal-content based experiments and settings.
2. Proposing a novel architecture for fake news detection that achieves state of the art accuracies. This is where we explore training the model jointly over intertwined modal embeddings instead of keeping them isolated.
3. Proposing an efficient and lightweight architecture for summarization of news articles using guidance provided by images.
4. Improving content ranking in search engines through personalized recommendations by modeling user actions as a time series, and treating time like a modality.
5. Creating and contributing a high quality multimodal dataset in Indic languages.

1.6 Thesis Workflow

This thesis is divided into seven chapters. While chapter 2 covers related work, chapter 3 onwards we cover the tasks discussed above. The chapters are organized in a way that readers do not require knowledge of the previous chapters and can pick and choose which to read. A brief summary of each chapter is provided here.

- Chapter 1 gives us an overview of the thesis. We discuss about what multimodality is, the kinds of multimodality, the research statements undertaken and the major contributions of this thesis.
- Chapter 2 describes the existing literature that is relevant to the work presented in this thesis.
- Chapter 3 covers *SCATE*, the fake news detection architecture. We go into depth about what fake news is, and why it is important to curb it. We then lay down the technical details of the model and examine its performance.
- Chapter 4 covers *MMSumm*, the system we developed to summarize textual content with image-aid. We start by discussing the motivation for this problem and its effect on user satisfaction by exploring some past work. We then dive deep into the technical details.

- Chapter 5 covers personalized content ranking by generation of user embeddings. We start by describing the way we model user actions which allows us to embed them in a neural network. We then describe how this model will be used inside of a bigger search system.
- Chapter 6 covers the creation of our Indic Multimodal dataset. We first analyze the Samanantar dataset, and then describe ways to prune it down into a more condensed captions-only dataset. We then go over ways to convert this into a multimodal dataset through image scraping via query formation for search engines.
- Chapter 7 concludes this thesis with a summary of our contributions along with potential ideas that can be explored further.

Chapter 2

Related Work

This chapter describes the existing literature that is relevant to this thesis. We cover topics related to solving multimodal tasks such as Fake News Detection, Image-aided summarization and creation of multimodal datasets.

2.1 Approaches to solving multimodal tasks

Multimodal content had researchers operating on it since the inception of good encoding techniques for text and images, such as Word2Vec [38] and AlexNet [22]. Papers all the way back in 2012 were performing classification over videos using text and image features, such as [28].

For a long time however, the proposed deep learning models had a similar approach.

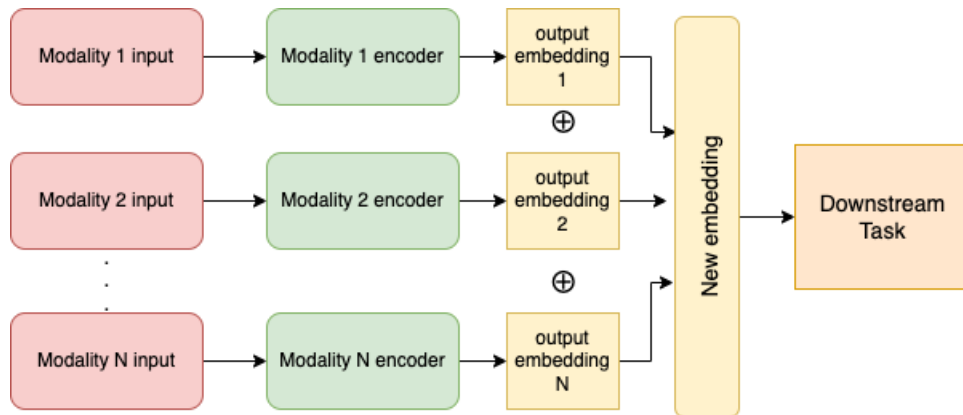


Figure 2.1 Structure of most early classification multimodal models. " \oplus " refers to some combination operator.

The models simply took individual modalities and used encoding techniques to create vectors, which they then combined using concatenation, multiplication, or pooling, and used for

downstream tasks. Even for more complicated tasks, the approach was same, albeit hidden slightly better. For example, in [36] - a model to generate image captions, we see that the authors try to predict the n -th token of the caption by trying to embed the image through a convolutional neural network, then embedding the $(n-1)$ tokens generated so far, and then combining the embeddings together.

The creation of the **transformer architecture** [56] was revolutionary for many fields including the understanding of multimodal content. A general purpose architecture like Transformer could be used to encode both image and text together and could develop a shared understanding of the context. This led to the rise of models such as ImageBERT [42].

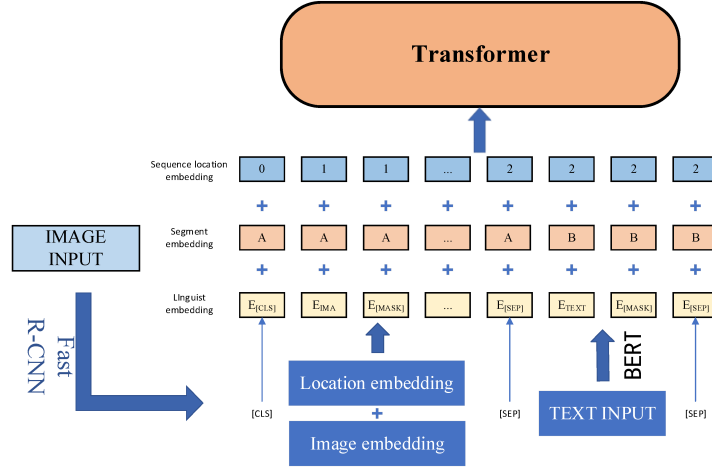


Figure 2.2 Structure of imagebert. The transformer architecture is used to take both image and text as input at the same time.

These models possess a good understanding of shared context between texts and images, and achieve high scores for downstream tasks such as visual question answering, image caption generation and retrieval. The embeddings generated from such models could be used for even more niche tasks through a process called “finetuning”, which refers to re-training pre-trained models over new data. Currently, transformer models such as OSCAR [24], which use image labels as anchor points within text, dominate the performance metrics. There is also a rise in usage of models that individually encode image and text, however use advanced methods of combination to enrich the embeddings. One such example showcasing very strong zero-shot performance is CLIP [43] - which uses independent image and text encoders, but combines them in a way which tries to guess what image-text pair belongs together.

2.2 Fake News Detection

This problem is generally discussed under two broad categories - knowledge based, and style/content based.

2.2.1 Knowledge based

This method refers to the extraction of facts from documents and comparing them to knowledge sources, also called knowledge bases. This comparison can be manual as well as automated. Websites such as Snopes, Politifact, FactCheck, HoaxSlayer, TruthOrFiction use manual fact checking to classify posts and articles through the use of domain experts. This process is highly time consuming as well as costly. There also exist platforms for crowdsourcing fact checking, however they are difficult to manage and easily influenced by internal biases within fact-checkers.

There also exists automated fact checking methods. For example, the work by Shi et al. [49] utilizes knowledge graphs and examines paths between entities for fact checking claims.

2.2.2 Style/content based

In this method, researchers try to use linguistic features extracted from documents. Work by Castillo et al. makes use of features such as special characters, use of positive/negative words, emojis, etc. to detect patterns prevalent in fake news. More recent works include modelling documents as inputs to deep learning architectures, brought forth by Ma et al. [33]. Chen et al [6] introduced the concept of using attention within RNN architectures to help in pooling of temporal-linguistic features and improve detection metrics.

2.2.2.1 Multimodal fake news detection

The task we deal with consists of fake news where articles or posts are accompanied by images. This task falls strictly under the category of style/content based fake news detection, as the proposed models do not refer to any knowledge bases when making their decision.

Studies [[62], [19]] show that usage of images for this task results in better performance of the models compared to text-only baselines. Khattar et al. [19] make use of variational autoencoders to learn rich representations of image and text. Spotfake [53] was the first model to utilize text embeddings from a pretrained transformer (BERT) along with image embeddings from VGG-19 for the task of multimodal fake news detection. CARMN [55] utilizes cross modal residual attention to attend to relevant parts of a modality and has one of the highest current scores across accuracy and F-1 metrics.

2.3 Summarization

Summarization of data refers to the process of shortening data through computational methods to get a subset of the whole information. This subset should contain the most relevant/important parts from within the original data so that the end user can consume information more efficiently.

The problem of summarization has been tackled extensively by researchers. Broadly, the method of summarizing documents can be broken down into two techniques - extractive and abstractive summarization.

2.3.1 Extractive summarization

Extractive summarization refers to picking out individual sentences from a document, and using them without modification for the final summary. More formally, if a document consists of a set S of sentences, we're interested in finding a set N such that $N \in S$ with the added condition that these N sentences are the most relevant. In figure 2.4 we can see that only the first sentence of the text is picked as the summary, as it is the most relevant. Examples of earlier work include Mihalcea et al. [37], who make use of graph based ranking algorithms to filter best ranked sentences and Shen et al. [48], who use conditional random fields by treating the document like a sequence of sentences, with each sentence requiring a 1 or a 0 label for being included, or not included. Nallapati et al. [39] modelled the problem as a sequence classification problem and used recurrent neural networks to classify sentences. More recent works like Liu et al. [30] make use of pretrained transformer models such as BERT for richer sentence embeddings and better classification.

2.3.2 Abstractive summarization

Abstractive summarization refers to a more human form of condensing relevant information. Within it, we are not bound by tokens present in the original text, rather we are free to generate tokens in a way that can aid in this condensation. In figure 2.4 we see how tokens such as “overcome” and “against” are not present in the original document, but they are in the summary as they help in the creation of a more concise statement. Nallapati et al. [40] showed how attention based recurrent neural networks can be used as encoders and decoders to create a sequence to sequence model for abstractive summary generation. See et al. [46] improved upon this work by utilizing a pointer generator network that had the ability to copy source words, thus preserving factual accuracy better in the summary along with a “coverage” module to memorize content summarized till a given point of time, thus preventing repetition. Modern approaches utilize the transformer architecture to perform summarization. Lewis et al. [23] implemented BART, a seq-2-seq model that utilizes sentence denoising as a training

objective, and results in high summarization scores. Zhang et al. [65] introduced PEGASUS, a model which masks important sentences and generates them on its own using the surrounding context to learn against the golden sentences.

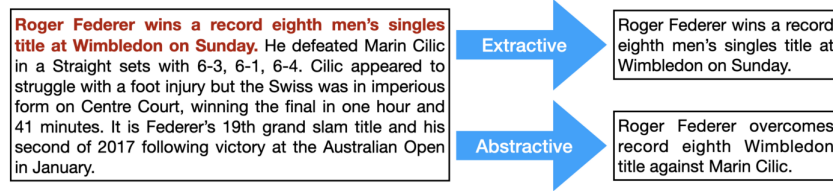


Figure 2.3 Examples of extractive and abstractive summarization.

2.3.3 Multimodal summarization with multimodal output

The task of multimodal summarization with multimodal output (or MSMO) falls under the umbrella of abstractive summarization. It refers to the use of image content in a mostly-textual document to aid in abstractive summarization. First proposed by Zhu et al. [69], they introduced a dataset of CNN and Daily Mail news articles along with their corresponding images. They used an attention based seq-2-seq model built using bidirectional LSTMs and showcased exceptional performance with respect to that time over human evaluation. They also showed that providing the user with the most relevant image boosted user satisfaction than a text-only summary. Zhu et al. [17] again improved upon their model by using a multimodal ranking method to rank images. Zhang et al. [67] obtain the highest metrics currently on this task; they make use of BART along with CLIP as a knowledge distillation module.

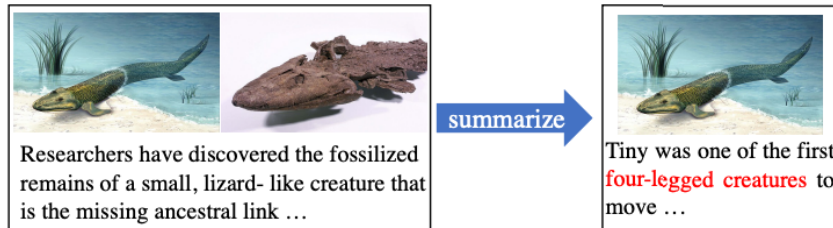


Figure 2.4 Example of multimodal summarization with multimodal output.

2.4 Multimodal Datasets

To train multimodal models, we require datasets that consist of greater than one modality. This includes datasets containing image-text pairs, image-text-speech triples, video-speech pairs, etc. Within this thesis we would primarily be dealing with datasets that contain image and text as modalities. Hodosh et al. [14] introduced the flickr8k dataset, which consisted of 8000 images from the website flickr along with 5 captions for each of those images. Microsoft released the COCO dataset [26] which stands for "Common Objects in COntext". In this dataset, the focus is on common objects in everyday surroundings. The dataset consists of over 300,000 images with accompanying tags to help in object detection. Google recently released its Conceptual Captions dataset [47] which contains over 3.5 Million image-caption pairs. They source this dataset by scraping the internet for over 1 Billion image-caption pairs and then pruning the set down through an extensive list of rules. They also try to remove named entities from captions to keep the sentences as generic as possible.

2.4.1 Samanantar (text-only dataset)

In our attempt to get an Indic multimodal dataset, we make use of the samanantar dataset from AI4Bharat [44], which is text-only. Samanantar dataset consists of 49.7 Million pairs of sentences between English and 11 Indic languages spread across 2 language families - Indo-aryan and Dravidian. These languages are - Hindi, Bengali, Tamil, Telugu, Odiya, Kannada, Assamese, Marathi, Punjabi, Gujarati and Malayali.

Chapter 3

Multimodal Fake News Detection

Our task on multimodal fake news detection aims to solve the problem of classifying posts containing text and an accompanying image as fake or not. There exist quite a few approaches currently which tackle this problem, however most of them fail to build a good crossmodal embedding and resort to concatenating or combining modalities through independent encoders. While some works like CARMN [55] attempt to use residual attention techniques from other modalities, we demonstrate a stronger way to combine embeddings by utilizing the transformer architecture. To solve this problem, we explore ways on how to handle variation of text across microblogging websites such as Twitter and Weibo, and how to encode image information together with text information effectively. We propose the SCATE (Shared Cross Attention Transformer Encoders) architecture which is able to perform this task of utilizing text and image modalities through the use of shared attention layers. Through thorough benchmarking and ablation studies, we show an improvement of approximately 3 percentage points over the three datasets used.

3.1 Introduction

The advent of social media has completely taken over all other forms of media, with over 67% of Americans ¹ reporting that they consume information from social networking websites. While social media is great for sharing information and communicating with one’s circle of acquaintances, it also allows users with ill-intent to spread misinformation. These users can be financially motivated, politically motivated, or often just be doing it for fun. There are very real consequences to fake news, and major events such as presidential elections have been affected by the spread of malicious information. For example, the Brexit referendum² as well

¹<https://www.journalism.org/2017/09/07/news-use-across-social-media-platforms-2017/>

²<https://www.independent.co.uk/news/uk/politics/final-say-brexit-referendum-lies-boris-johnson-leave-campaign-1818111.html>

as the 2016 presidential election in the US³ had been suspected of being influenced by false information. Fake news also possesses the ability to influence the opinions of major investors and can have drastic financial ramifications. Recently, an impersonator of the pharmaceutical company "Eli Lilly" on Twitter tweeted about a drug being made free of cost⁴, which caused the stock price of Eli Lilly to plunge, causing the company a loss of over 1 billion US dollars. Misinformation can also cause criminal consequences in the real world, such as the riots that happened in the US capitol on Jan 6th⁵ after Donald Trump's enticement to storm the building stating false claims of election fraud.

It has also been shown that fake news travels roughly 6 times faster⁶ than true news. This makes it hard to curb its impact once damage has been done. Therefore, it is necessary to devise ways to stop fake news at its inception.



Sharks seen roaming
in New Jersey streets
and metro stations.
#Sandy



Brilliant
photo:
believe in global
warming".



telling
安利老板死了! 才 56 岁,
吃了 27 年的纽崔莱, 好
讽刺啊!

Figure 3.1 Fake News examples from the Twitter and Weibo datasets. (Translation for the rightmost example using Google Translate: "The Amway boss is dead! Only 56 years old, eating Nutrilite for 27 years, so ironic!")

In this chapter we deal with solving the problem of fake news detection within the multimodal domain. Specifically, we deal with classifying posts on the websites Twitter and Weibo as being fake or not. Figure 3.1 shows 3 examples of fake news datapoints from Twitter and Weibo. Each example has certain textual content and an image associated with it. For the tweet on the left, both the image and text indicate that it is most likely fake. In the post on the right, the image does not add substantial information as it is just a generic store front, but the text indicates that it may have been fabricated to spread fast. In the tweet in the middle, it is difficult to reach a conclusion from the text alone, but the morphed image suggests that it might include

³<https://techcrunch.com/2018/03/13/un-says-facebook-is-accelerating-ethnic-violence-in-myanmar/>
⁴<https://www.washingtonpost.com/technology/2022/11/14/twitter-fake-eli-lilly/>
⁵<https://time.com/6133336/jan-6-capitol-riot-arrests-sentences/>
⁶<https://www.pbs.org/newshour/science/false-news-travels-6-times-faster-on-twitter-than-truthful-news>

foul play. This example reflects the hypothesis that pairs of visual and textual information can give better insights into the content and possibly improve the detection of fake news.

Detection of fake news for such posts is challenging for many reasons -

- Textual content of posts is very short and consists of slang and flexible grammatical structure unlike news articles.
- Since these posts relate to fresh news and spread extremely fast over social media platforms, it is very difficult to cross-verify their claims with credible news sources.
- It is difficult to handle text variations over Twitter and Weibo, and further identify the best way to fuse text and image information.

Previous work on this problem has mostly focused on concepts such as comparison of posts with facts from knowledge bases or textual feature engineering [5, 60]. However, textual representations prove to be insufficient for platforms such as twitter, where the amount of text (i.e. number of characters) is very limited. Additionally, they depend heavily on quickly updated knowledge bases which are difficult to maintain.

Recently, multimodal models have shown impressive performance on this task. Existing multimodal models such as those by Khattar et al.[20] and Singhal et al.[54] follow a similar structure. One part of the model encodes the text while the other is tasked with encoding the image. Following this, the encoded vectors are combined through some operation (generally concatenation) and then passed on to a classification (fully connected) layer to get the predicted labels. However, we believe that for a complex task like Fake News Detection, richer embeddings must be created using fusion of text and image modalities. Works such as Song et al.[55] try to use attention based mechanisms to create these richer representations.

In our work, we propose a more effective approach which utilizes cross modal attention scaling over BERT based text embeddings and deep convolutional neural network based image embeddings. This builds embeddings that are aware of the other modality. We also use shared feedforward layers in the attention network to further enrich the representations. We also make use of compact bilinear pooling to combine the modalities, and then finally perform the classification.

Figure 3.2 presents the architecture of our suggested model SCATE (Shared Cross Attention Transformer Encoder).

In this work, we make the following contributions:

- We propose the use of cross-modal attention at a post level, instead of at a more granular textual level for fake news detection on microblog websites, because of the noisy nature of text on such platforms.
- We show how a 3-layer transformer architecture with a shared attention layer within it creates richer representations.

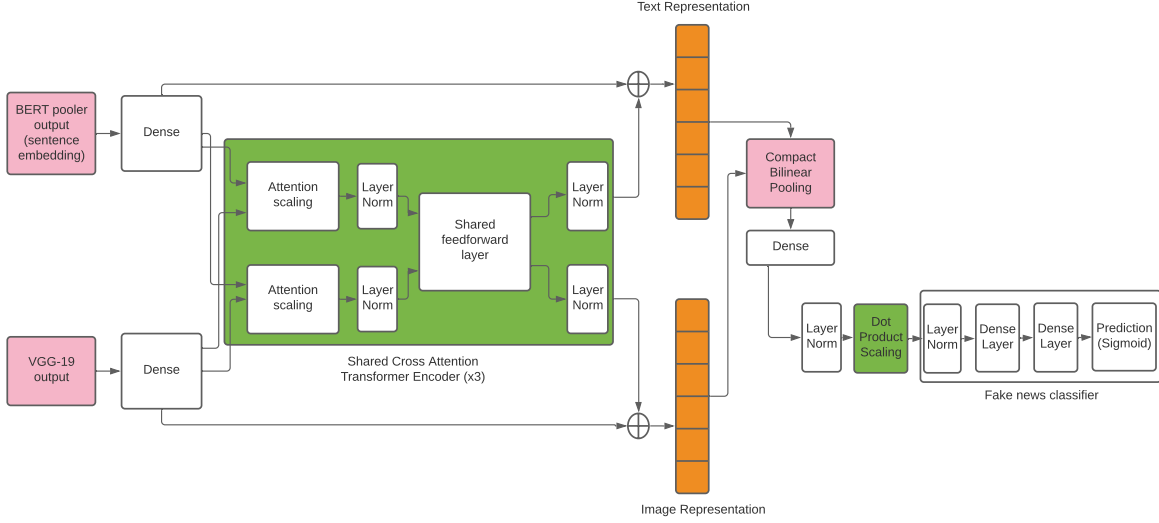


Figure 3.2 *SCATE* System Architecture

- We demonstrate the efficacy of our approach by showcasing a ~ 3 percentage point increase in accuracy over latest methods across all the datasets (Twitter, WeiboA and WeiboB) used in this work.

3.2 Datasets

For this task we make use of 3 datasets. The first dataset consists of Twitter posts, while the other 2 datasets consist of Weibo posts. The twitter dataset is called “Twitter MediaEval” while the Weibo datasets are referred to as “WeiboA” and “WeiboB”. Authors from the CARMN [55] paper made use of “WeiboA”, “WeiboB” as well as a dataset “WeiboC”. However, when we used the “WeiboC” dataset, we found that the fake images of the dataset had an obvious watermark on them. Our model was able to exploit this artifact and achieve close to 100% accuracy on this dataset, which is why we decided to drop it in our work.

3.2.1 Twitter Media-Eval dataset

The dataset can be found on the link⁷ present in the footnote. It was released for the Varying Multimedia Use Task[7]. The dataset consists of over 14k image-text pair samples. The train test-validation split for the dataset can be seen in table 3.2 and the number of posts along with number of unique images can be seen in 3.1. We can see that the number of posts far exceeds

⁷<https://github.com/MKLab-ITI/image-verification-corpus>

the number of unique images. Because of this, one major challenge while training our model is to avoid overfitting.

Dataset	# Posts	# Unique Img
Twitter	14514	480

Table 3.1 # Posts and # Images across Twitter dataset

	Real	Fake
Train	5264	6810
Valid	596	746
Test	468	630

Table 3.2 Twitter Dataset Class Distribution

3.2.2 WeiboA dataset

The dataset “WeiboA” consists of data collected from May 2012 to January 2016 from the chinese microblogging website Weibo by Jin et al.[16]. The dataset consists of 9.5k image-text pair samples. In order to label truthful news, Jin et al. adopted news verified by the Xinhua News Agency⁸ as true. The fake news is verified as fake and collected from the official fake news debunking system of the Sina Weibo, another website similar to Twitter. The full dataset can be found on the link⁹ in footnote. The training-testing splits of the dataset can be examined in table 3.3 and the number of posts along with number of unique images in the dataset can be found in table 3.5.

3.2.3 WeiboB dataset

The second dataset “WeiboB” was created as a benchmark dataset for the internet fake news detector challenger¹⁰. It was released by Cao et al.[4]. Similar to WeiboA, it consists of image-text pair samples. A lot of images in WeiboB are of a higher resolution as compared to WeiboA and hence require digital downsampling. We process the text in the same way as WeiboA. The training-testing splits of the dataset can be found present in table 3.4 and the number of posts along with number of unique images in the dataset can be found in table 3.5.

3.2.4 Model

We formally define the problem statement as follows - Given an input social media post as text t , and an associated image i , our goal is to predict whether the combination (t, i) is fake or not. Hence, our model will take (t, i) as an input and return a score s representing its confidence in whether the post is truthful or fake.

⁸https://en.wikipedia.org/wiki/Xinhua_News_Agency

⁹<https://drive.google.com/file/d/14VQ7EWPiFeGzxp3XC2DeEHl-BEisDlNn/view>

¹⁰<https://biendata.com/competition/falsenews/>

	Real	Fake
Train	2571	2998
Valid	353	442
Test	718	873

Table 3.3 WeiboA Dataset Class Distribution

	Real	Fake
Train	3229	3830
Valid	479	530
Test	958	1058

Table 3.4 WeiboB Dataset Class Distribution

Dataset	# Posts	# Unique Img
WeiboA	7955	7955
WeiboB	10084	9525

Table 3.5 # Posts and # Images across Weibo datasets

3.2.4.1 Inputs

Here, we define how we convert text and image inputs to their corresponding embeddings.

- **Text** - We make use of the BERT [9] language model to encode our text. We do not make use of the individual token embeddings but rather the pooled embedding, in order to consider the entire context at once. This gives us a vector of length d_t , corresponding to the hidden dimension of BERT. It is important to note here that the BERT weights are frozen, and for the remainder of the training, we would not be training BERT. This prevents overfitting of text, since the dataset sizes we are using are not large enough for a model with such a high parameter count.
- **Image** - To encode the image, we make use of the deep convolutional neural network VGG-19 by Simonyan et al. [50]. VGG-19 is a model trained on the ImageNet dataset for classification of 1000 object classes, and produces rich embeddings. VGG-19 consists of 19 layers, however the last 3 layers are fully connected layers used for classification alone. Because of this, we consider the the output of the 16th layer of VGG as our desired embedding. This embedding is a vector of length d_i . Due to reasons similar to text encoding, we keep our VGG model frozen, i.e., untrainable.

Since VGG and BERT models have differently sized outputs, we make use of a linear layer to transform the vectors into a common size of d . In the equations that follow, E_t and \hat{E}_t refer to the text embedding and the transformed text embedding respectively. Similarly, E_i and \hat{E}_i refer to the image embedding and the transformed image embedding. *Dense1* and *Dense2* corresponding to dense layers that have input dimension equal to that of the corresponding modality (to support multiplication) and possess the same output dimension.

$$\hat{E}_t = Dense1(E_t) \quad (3.1)$$

$$\hat{E}_i = Dense2(E_i) \quad (3.2)$$

The dimensions of \hat{E}_t and \hat{E}_i are now equal, and referred to as dim_E .

3.2.4.2 Cross-modal attention and shared Feedforward layers

This is the part of the model where we introduce cross-modal dependence. We modify scaled dot product attention that is used within the transformer architecture[57], such that it uses information from both the modalities simultaneously. This attention value is then used to scale the modalities with respect to each other by some factor.

In a standard transformer architecture, attention is used to impart a multiplicative relationship to the input by using key and query input sources. The input is calculated as a weighted sum of value vectors across positions. The weights are determined using dot product between the query vector for the current position and key vectors for other positions. Formally, defining scaled dot product attention at $DotAttn(Q, K)$,

$$DotAttn(Q, K) = \text{sigmoid} \left(\frac{Q \cdot K^T}{\sqrt{d}} \right)$$

Where d corresponds to the dimension of the query vector, and the superscript T corresponds to a transpose operation.

In our model, the attention matrices (W_Q, W_K, W_T) are calculated simultaneously for both text and image modalities. All matrices share a common shape of $d \times d$. To achieve cross-modal dependence, we use one modality as the query in the attention calculation of another.

More formally, For the text, we calculate the scaling value S_t and Q_i, K_t, V_t vectors as follows:

$$Q_i = \hat{E}_i \times W_Q^t; K_t = \hat{E}_t \times W_K^t; V_t = \hat{E}_t \times W_V^t$$

$$S_t = DotAttn(Q_i, K_t) \quad (3.3)$$

For the image, we get S_i and Q_t, K_i, V_i vectors as follows:

$$Q_t = \hat{E}_t \times W_Q^i; K_i = \hat{E}_i \times W_K^i; V_i = \hat{E}_i \times W_V^i$$

$$S_i = DotAttn(Q_t, K_i) \quad (3.4)$$

Where all W values refer to learned weights for queries, keys and values. A superscript of t corresponds to text, while a superscript of i corresponds to image. These scaling values S_t and S_i are then multiplied to the embedding vectors.

This attention model can also employ the use of multi-headed attention to gather additional context from different sub-sections of the embedding. By using H heads, the output of each

head gets a subset of the dimension, i.e. $\frac{d}{H}$. These dimensions can then be concatenated together to result in the full vector.

$$\text{Attention} = \text{concat}(\text{head}_1, \dots, \text{head}_H) \times W_{\text{output}} \quad (3.5)$$

Post scaling the embeddings, we make use of Layer Norm[2] to normalize the layer weights for smoother gradients.

After normalization, the embeddings are passed onto 2 consecutive shared feedforward layers. The job of the first layer is to transform the embedding into a higher dimensional space, while the job of the second is to bring the dimension back down to dim_E . This is done so to mimic the behaviour of a sparse autoencoder[34]. The computation that happens is as follows -

$$\hat{E}_t' = (\hat{E}_t \times W_x + b_x) \times W_y + b_y \quad (3.6)$$

$$\hat{E}_i' = (\hat{E}_i \times W_x + b_x) \times W_y + b_y \quad (3.7)$$

Where the W_x layer increases the dimension to dim_{new} and W_y layer decreases the dimension back down to dim_E . We again use LayerNorm to normalize the inputs.

For sake of readability, we haven't mentioned the use of activation functions. However, after each linear layer, we make use of a standard ReLU [12] activation.

The list of operators from equation 3.3 to 3.7 refer to one transformation of the input. We can apply this model arbitrary number of times, and resort to 3 applications in our implementation based on validation results.

The outputs generated from the main body of the model, i.e. \hat{E}_t', \hat{E}_i' are concatenated to the original outputs of the pre-trained models BERT and VGG, i.e., E_t, E_i

$$E_t' = \hat{E}_t \oplus \hat{E}_t' \quad (3.8)$$

$$E_i' = \hat{E}_i \oplus \hat{E}_i' \quad (3.9)$$

To combine these embeddings, we make use of multimodal compact bilinear pooling (MCB) [11].

Regular outer product to capture inter-element relationship between vectors suffers from high dimensionality. Fukui et al. proposed using bilinear pooling for multimodal inputs as it captures the relationship along with constraining the dimensionality of the output.

MCB works by performing fast fourier transformer on the count-sketch vector of the modality. This results in 2 new vectors for each modality, which are then combined using a convolution. Finally an inverse FFT converts the combined vector to a desired dimensionality. Figure 3.3 shows how the algorithm works.

Output from the MCB is then converted down to a lower dimensionality using another linear layer.

This process can be summarized as follows -

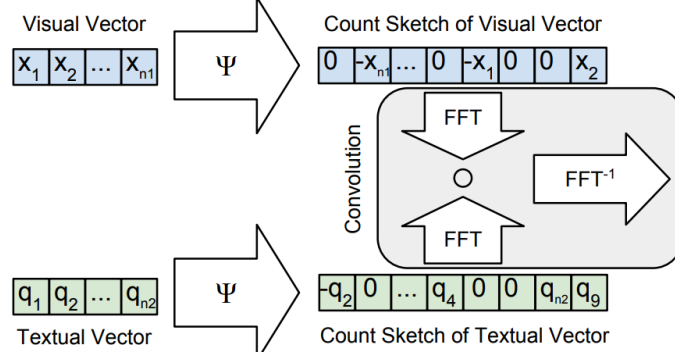


Figure 3.3 Process of multimodal compact bilinear pooling.

$$E = \text{Dense}(\text{MCB}(E'_t, E'_i)) \quad (3.10)$$

The dense layer converts the output of MCB of a higher dimensionality (8000 in our model) to $4 \times \dim_E$.

3.2.4.3 Dot Product Scaling to aid classifier

Unlike the previous attention module where we deal with inter modality interactions, here we deal only with one vector, i.e., the final output vector E .

However, we make use of a similar mechanism that we did in equation 3.3. Instead of providing query from a different modality, we use the same modality. This gives us the following Q, K, V values -

$$Q_E = E \times W'_Q; K_E = E \times W'_K; V_E = E \times W'_V.$$

and we calculate the scaling value just like before -

$$S = \text{DotAttn}(Q_E, K_E)$$

We use this S to scale our final vector E , and then use a final LayerNorm. While a regular linear layer should be able to learn this relationship, we found out in our experimentation that with this module, we were able to achieve better scores. This can likely be attributed to the fact that a direct scaling of E can make its behaviour more predictable to the classifier layer up ahead, and result in increased metrics.

3.2.4.4 Classification layers

The output of the system above results in our fully transformed vector E , which can be directly fed into linear layers for classification. We make use of 2 fully connected layers followed

by the use of a sigmoid function to convert the resulting output into a probability.

$$p(\text{fake}) = \text{sigmoid}(\text{Dense2}(\text{Dense1}(E)))$$

Where $p(\text{fake})$ refers to the probability of a post being fake and Dense1, Dense2 are dense classifier layers.

3.2.4.5 Training

For training the model, we make use of a standard binary cross entropy loss function [35], since we’re dealing with a binary classification problem and our output is a single float denoting the probability.

$$L(\theta) = -y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log (1 - \hat{y}_i)$$

L here is our loss function which takes as parameter θ that represents the parameters of our model. Here, y_i are the ground truth labels used to compare against \hat{y}_i which stands for the output of the model.

3.3 Baselines

In this section we define the baselines that we compare our model against. While we include multimodal baselines in this list, we also decided to include baselines that utilize only text or image to classify the post. We believe this comparison further legitimizes the task of multimodal fake news detection.

There exists a lengthy history of work done in this area, but we limit our comparison to the following models -

- **Text-only Convolutional Network** - This baseline consists of a single channel convolutional neural network to encode the tokens in text, while ensuring surrounding context of a token is also taken into consideration. The resultant output through this convolution is then classified through a simple dense+sigmoid classifier.
- **Image-only Deep Convolutional Network** - This baseline consists of a VGG-19 [50] network (the same one used in SCATE) from which we take output of the 16th layer (just before the classification layers). We pass this output onto a simple dense layer and sigmoid function to classify.
- **Visual Question Answering (VQA)** - The VQA [1] paper proposed the visual question answering task where the model is expected to answer a question about an image, along

with the model itself. The model consisted of LSTM cells to encode the text and a multi-class classification layer. We augment the model to support a binary output for our usecase.

- **Neural Talk** - The model for Neural Talk [59] uses Recurrent Neural Networks to encode text along with image. Since the model was built to generate captions for images, it takes as input the image along with the text generated *so far*, which is how it is able to generate text one token at a time. By providing the entire post content as text and the post image as the image, we can get a representation for classifying as fake or not. The representation is then passed through a dense layer and sigmoid function for classification.
- **att-RNN** - The original work [16] uses an attention based RNN to encode features such as text and image along with social and user profile elements together. For our usecase, we use it to encode the image and the text together.
- **Event Adversarial Neural Network (EANN)** - The author's propose the EANN [61] model which encodes text through a CNN and image through VGG-19 similar to our model. However, EANN uses an adversarial framework which improves their classifier by training parallelly alongside it. It is trained on the same datasets as used in this chapter, and has the same inputs and outputs as our model.
- **Multimodal Variational AutoEncoder (MVAE)** - Proposed by Khattar et al. [20], MVAE proposes the use of a variational autoencoder. Through the use of a variational autoencoder trained on a reconstruction loss, they are able to create a layer which outputs a rich representation that contains the most important features of the modalities. They pass text and images through this autoencoder and use a dense layer and sigmoid function to classify. Since they are trained on the same datasets used on this chapter, they also follow the same input output format and require no further transformation.
- **Memory Knowledge Network (MKN)** - Multi-modal Knowledge-aware Event Memory Network (MKEMN) [64] is an event-level multimodal fake news detection framework, which use the visual information and the external knowledge. The authors use an event memory network to learn event invariant features. Considering the differences between event-level and post-level fake news detection and the fairness of comparison, we remove the external knowledge component and event memory network from this model. We only use the remaining components and hence refer to the ablated model as MKN.
- **SpotFake** - Singhal et al. [53] proposed the first model for fake news detection which made use of a transformer architecture (BERT in this case) for text encoding. They take individual embeddings from BERT and VGG-19, and concatenate them for a larger representation. This representation is then processed through a standard dense+sigmoid classification layer to net a score.

- **Crossmodal Attention Residual and Multichannel convolutional neural Networks (CARMN)** - Song et al. [55] proposed the CARMN model which uses BERT and VGG-19 to obtain image and text features respectively, but then makes use of residual attention within each modality to attend to the other and obtain greater context. They use a method similar to SCATE, by using different modality queries while calculating scaled dot product attention. Since CARMN is trained on the same datasets and follows the same input output format, no further transformation to the model is required. CARMN achieves the best current performance on the task of multimodal Fake News Detection as of the writing of this thesis.

3.4 Experimental Settings

3.4.1 Text processing

We use BERT-base-cased and BERT-base-Chinese models from the huggingface library¹¹ for obtaining sentence embeddings for the Twitter and Weibo datasets. We pre-processed posts to remove special Unicode characters and URLs from the text. We truncate long posts at 25 tokens (sub-words) for Twitter, and 203 tokens for Weibo. These truncation values were decided based on the average number of tokens present in the corresponding datasets.

3.4.2 Image Processing

Since the images available to us are of different sizes, we convert all the images to a common size before processing, i.e., VGG-19’s desired input matrix, which has a size of 224×224 . We use OpenCV¹² to resize the image and the use the INTER_LINEAR function for interpolation.

3.4.3 Hyperparameters

The model was trained for 300 epochs with an early stopping criterion to report the results. The batch size used was 256. We used a variable learning rate in the range $(1^{-3}, 1^{-2})$ for our experiments. For all our linear layers, the ReLU activation function was used to provide non-linearity.

To avoid overfitting we also make use of dropout along with L2-regularizer on the weights. We experimented with weight-decay in the range of $(0, 1^{-4})$ and finally settled with 1^{-3} for the Twitter dataset and 1^{-4} for the Weibo dataset. An adam optimizer was used.

The feedforward layers inside the transformer have a dimension of 768. We use 3 Transformer encoder layers. For the text encoder the output dimension is 768 and for the image encoder the

¹¹<https://huggingface.co/>

¹²<https://opencv.org/>

output dimension is 4096. We use the same hidden layer dimension of 128 as the size of query, key and value for our transformer attention layers. These layers also use 4 attention heads per layer. The dimensions of the last two dense layers is fixed to (512×128) and (128×1) respectively.

All the hyper-parameters mentioned are explored by the experimentation on the validation dataset. The experiments were performed locally on 4 NVidia GeForce 2080 Ti GPUs.

3.5 Comparison

This section contains the comparison of our proposed architecture with the previously discussed baselines across the 3 datasets, i.e., Twitter, WeiboA and WeiboB.

Model	Modality	Accuracy	Prec (F)	Rec (F)	F1 (F)	Prec (T)	Rec (T)	F1 (T)
Textual	Text	0.526	0.586	0.553	0.569	0.469	0.526	0.496
Visual	Img	0.596	0.695	0.518	0.593	0.524	0.700	0.599
VQA	Text+Img	0.631	0.765	0.509	0.611	0.550	0.794	0.65
Neural Talk	Text+Img	0.610	0.728	0.504	0.595	0.534	0.752	0.625
att-RNN	Text+Img	0.664	0.749	0.615	0.676	0.589	0.728	0.651
EANN	Text+Img	0.648	0.810	0.498	0.617	0.584	0.759	0.660
MKN	Text+Img	0.664	0.753	0.537	0.627	0.611	0.805	0.695
MVAE	Text+Img	0.745	0.801	0.719	0.758	0.689	0.777	0.730
CARMN	Text+Img	0.741	0.854	0.619	0.718	0.670	0.880	0.760
SpotFake	Text+Img	0.764	0.825	0.557	0.663	0.741	0.914	0.818
SCATE	Text+Img	0.796	0.839	0.805	0.822	0.750	0.790	0.769

Table 3.6 Results on Twitter Dataset

Model	Modality	Accuracy	Prec (F)	Rec (F)	F1 (F)	Prec (T)	Rec (T)	F1 (T)
Textual	Text	0.643	0.662	0.578	0.617	0.609	0.685	0.647
Visual	Img	0.608	0.610	0.605	0.607	0.607	0.611	0.609
VQA	Text+Img	0.736	0.797	0.634	0.706	0.695	0.838	0.760
Neural Talk	Text+Img	0.726	0.794	0.713	0.692	0.684	0.840	0.754
att-RNN	Text+Img	0.772	0.854	0.656	0.742	0.720	0.889	0.795
EANN	Text+Img	0.782	0.827	0.697	0.756	0.752	0.863	0.804
MKN	Text+Img	0.792	0.805	0.788	0.796	0.778	0.796	0.787
MVAE	Text+Img	0.824	0.854	0.769	0.809	0.802	0.875	0.837
CARMN	Text+Img	0.853	0.891	0.814	0.851	0.818	0.894	0.854
Spotfake ³	Text+Img	0.842	0.870	0.837	0.852	0.812	0.847	0.828
SCATE	Text+Img	0.885	0.892	0.881	0.886	0.870	0.881	0.876

Table 3.7 Results on WeiboA Dataset

Model	Modality	Accuracy	Prec (F)	Rec (F)	F1 (F)	Prec (T)	Rec (T)	F1 (T)
Textual	Text	0.762	0.861	0.623	0.723	0.706	0.9	0.791
Visual	Img	0.702	0.734	0.630	0.678	0.678	0.773	0.722
VQA	Text+Img	0.704	0.706	0.695	0.701	0.702	0.713	0.707
Neural Talk	Text+Img	0.735	0.778	0.652	0.709	0.704	0.817	0.756
att-RNN	Text+Img	0.780	0.853	0.675	0.753	0.733	0.884	0.801
EANN	Text+Img	0.815	0.903	0.703	0.791	0.759	0.925	0.834
MKN	Text+Img	0.778	0.880	0.643	0.743	0.720	0.913	0.805
MVAE	Text+Img	0.741	0.779	0.671	0.721	0.713	0.811	0.759
CARMN	Text+Img	0.869	0.935	0.796	0.860	0.820	0.944	0.878
Spotfake ³	Text+Img	0.883	0.896	0.848	0.871	0.874	0.917	0.895
SCATE	Text+Img	0.914	0.907	0.918	0.912	0.897	0.881	0.889

Table 3.8 Results on WeiboB Dataset

As discussed in the dataset section, we do not report comparison with another Weibo dataset (“WeiboC”) despite it being used in other literature for this task. Within WeiboC, we discovered that our model was able to exploit a constant artifact (watermark) within the fake images to achieve accuracies of over 99%, which is unrealistic. We even tried downsampling to a much lower resolution of (180×180) followed by upsampling to VGG-19’s required input, however even this approach gave us no advantage.

3.6 Results Observation

From tables 3.6, 3.7 and 3.8, we can infer the following results.

- A text only baseline performs better on the Weibo datasets. However, we see an opposite trend for the Twitter dataset. Through this we can hypothesize that twitter images contain a richer signal for fake news detection.
- Across all the three datasets, we can see that multimodal methods lead to a higher accuracy compared to using a unimodal model design.
- Our proposed method, *SCATE* outperforms other baselines by approximately 3 percentage points in terms of accuracy across the three datasets. The cross modal attention along with the shared feedforward transformation in the linear layers helps *SCATE* outperform these very competitive baselines.
- CARMN computes attention per word in the post, while we compute only one attention value at the post level. Still, our method outperforms CARMN. This shows that for the fake news detection task, estimating relative importance between text and image is more important.
- By observing the disparity between CARMN scores and ours, we understand that calculation of scaling values at a post level is more capable of separating fake news from real and is easier to predict by our classifier. The attention scaling layers widen the divide between the modalities, with the shared feedforward layers understanding the relationship between the two. In the end this causes the transformer to generate embeddings that are easy to distinguish from each other using a standard classifier.

3.7 Ablation

We performed ablation analysis with respect to two important components of *SCATE*: feed-forward shared weights across Transformers, and using dot product self scaling. We put together results where the transformers share no weights or all the weights (including attention layers),

along with using and not-using self dot product scaling. The results are shown in Table ?? . We make the following observations.

To scrutinize our results, we perform an ablation analysis of our model *SCATE*. More particularly, we ablate over 2 major components of our model - the shared feedforward layers and the dot product scaling layer. We combine the results together in 4 scenarios -

- 1. Where the attention scaling layers and the feedforward layers are not shared, and dot product scaling is used.
- 2. Where the attention scaling layers as well as the feedforward layers are shared, and dot product scaling is used.
- 3. Where the attention scaling layers are not shared, but the feedforward layers are shared, and dot product scaling is not used.
- 4. Where the attention scaling layers are not shared, but feedforward layers are shared, and dot product scaling is used.

We refer to each of these scenarios by their bullet number, as 1, 2, 3 and 4 in the tables that follow.

Scenario	Dataset	Accuracy	Prec (F)	Rec (F)	F1 (F)	Prec (T)	Rec (T)	F1 (T)
1	Twitter	0.701	0.826	0.601	0.694	0.614	0.834	0.706
2	Twitter	0.730	0.760	0.764	0.762	0.689	0.685	0.686
3	Twitter	0.773	0.839	0.786	0.803	0.701	0.751	0.724
4	Twitter	0.796	0.822	0.805	0.822	0.750	0.790	0.769

Table 3.9 Twitter ablation results

Scenario	Dataset	Accuracy	Prec (F)	Rec (F)	F1 (F)	Prec (T)	Rec (T)	F1 (T)
1	WeiboA	0.858	0.860	0.888	0.873	0.857	0.820	0.838
2	WeiboA	0.856	0.915	0.817	0.863	0.796	0.905	0.847
3	WeiboA	0.861	0.923	0.815	0.865	0.803	0.916	0.855
4	WeiboA	0.885	0.892	0.881	0.886	0.870	0.881	0.876

Table 3.10 WeiboA ablation results

Scenario	Dataset	Accuracy	Prec (F)	Rec (F)	F1 (F)	Prec (T)	Rec (T)	F1 (T)
1	WeiboB	0.900	0.894	0.916	0.904	0.906	0.881	0.893
2	WeiboB	0.881	0.852	0.936	0.892	0.921	0.821	0.867
3	WeiboB	0.891	0.862	0.944	0.901	0.932	0.834	0.879
4	WeiboB	0.914	0.907	0.918	0.912	0.897	0.881	0.889

Table 3.11 WeiboB ablation results

We can see in the tables 3.9, 3.10, 3.11 that the most performant model in all the settings was the one that did not share the attention layers, but shared the feedforward layers and used dot product scaling in its representation.

3.7.1 Ablation Observations

- We see that sharing of the feedforward layers, isolation of the attention layers as well as dot product scaling are all very important for the high accuracy across all the datasets.
- One interesting observation is that removing dot product scaling on Twitter dataset leads to a higher fake precision of 0.839 but at the expense of poor fake recall. Note that this is very similar to the performance of CARMN on Twitter as seen in Table 3.6.
- *SCATE* obtains slightly higher fake recall on removal of the dot product scaling module for the WeiboB dataset. But we believe that it is not significant to warrant removal of the same.

3.8 Case Studies

In this section, we take a look at particular case studies where our proposed model *SCATE* is able to accurately predict the label but atleast one of text-only, image-only or multimodal(SpotFake) fails.

To precisely calculate the degree of contribution from the text and image parts of the post in the calculation of the prediction made by the multimodal model, we make use of a method similar to Zhou et al. [68]. Given an instance and the ground truth class label, we compute the contribution in the activation of the corresponding output neuron from text/image neurons from the previous hidden layer. Since these numbers might not be normalized, we calculate the output of the softmax function for all these elements. Lastly, we take the average of all instances belonging to one class.

Dataset	Image	Text	Visual Pred.	Textual Pred.	SpotFake Pred.	Actual Label	Text Contribution	Image Contribution
Twitter		Fuji created huge lenticular clouds and they were painted red at the sunrise.	Fake	True	True	Fake	0.4230	0.5770
Twitter		Eiffel Tower lights up in solidarity with Pakistan after #PrayForLahore.	Fake	Fake	True	Fake	0.7931	0.2069
WeiboA		通报表扬武汉官员洪水中坐轿在水中被五个男青年护送前行领导自己撑伞避雨冰哥认为在武汉遭遇特大暴雨袭击的情况下有关方面能够想官员之所想急官员之所急及时为领导准备四...	True	Fake	True	Fake	0.4002	0.5998
WeiboA		下午茶时间这只名为 Ura 的苏格兰折耳猫已经 17 岁了因肾脏问题而动作缓慢但这丝毫不妨碍她在社交网站上获得大批拥趸她慵懒的样子也成为萌点之一。	Fake	Fake	Fake	True	0.2241	0.7759
WeiboB		注意！驾车用蓝牙耳机接电话也要被扣分！近日不少商家以“不怕新交规”“新交规必备”为噱头促销蓝牙耳机有交警表示驾车时用蓝牙耳机接听电话仍属“妨碍安全驾...	Fake	True	True	Fake	0.4464	0.5536
WeiboB		房产图片高通联合创始人的”几何化”大宅-高通公司联合创始人维特比以约人民币 3.89 亿元的价格出售了位于兰乔圣菲的住宅该房产出自著名建筑设计师德雷尔之手占接...	Fake	Fake	Fake	True	0.0348	0.9652

Table 3.12 Case Studies. Text Contribution and Image Contribution correspond to SCATE’s.

As can be seen in table 6.1, many examples exhibit high image contribution from *SCATE*. This shows that using image as a modality for fake news detection improves classification.

In table 6.1 case 1, one can see that the image looks like it might have been doctored, which is why it gets a higher image contribution. In case 2, while the image looks realistic, the text sounds made up. This is also visible in the text contribution for case 2. The image in Case 3 shows pigs being ferried while a monkey holds an umbrella, which is not an everyday occurrence. This nets to a higher image contribution towards the model’s decision to label it as fake. In case 5 the image looks doctored to the naked eye as well, possessing signs of image editing. In cases such as 4 and 6, we see that *SCATE* predicts the label correctly, while all other comparison baselines fail.

3.9 Conclusion

In this chapter, we discuss our proposed model *SCATE*. We go into detail about its technical nuances and perform extensive experimentation to convey its efficacy, followed by a discussion on various case studies to understand its behaviour. We also gain an understanding of the importance of image as a modality in conveying contextual information.

In the future, we plan to extend this work to include multiple modalities such as speech and video. We also believe that work towards augmenting this architecture with knowledge-base related frameworks can improve its performance on real time data, and hope to pursue it in the future.

Chapter 4

Image-aided summarization

The need for summarization has been driven by the growing amount of content available on the internet. However, summarization focusing only on text can no longer be used to give an expansive summary as a lot of context is hidden away in the use of alternate modalities such as Image and Audio. Past efforts to solve this problem of multimodal summarization have tried to independently pool modalities together, however this does not lead to an effective representation. In our work, we propose a knowledge-distillation based approach that uses multimodal information retrieval methods to score tokens of importance higher as compared to other tokens. Our summarization model results in performance metrics just shy of the current best performing architecture, while simultaneously being much smaller in size and more efficient. Post automated metrics, we perform human evaluation over our summaries to indicate the viability of our model. We also propose an image scoring mechanism that is able to select the most relevant image to the article, and it outperforms the current best in image precision by an ~ 11 percent absolute difference.

4.1 Introduction

Summarization is a method of distilling large amounts of information into a shorter, more manageable format that captures the main ideas of a text while preserving its essential content. There are two primary types of summarization: extractive and abstractive. Extractive summarization involves selecting and reproducing key phrases, sentences, or paragraphs from the original text, while abstractive summarization involves the creation of a new, condensed body of text that captures the essence of the original while using novel phrasing and sentence structure.

In extractive summarization, the goal is to identify the most significant parts of the original text and reproduce them as accurately as possible. This method can be accomplished through the use of statistical techniques such as text clustering and ranking algorithms, which automat-

ically identify important content based on various criteria such as keyword frequency, sentence length, and semantic similarity.

On the other hand, abstractive summarization requires the machine to understand the meaning of the original text and generate a new text that conveys the same ideas but in a condensed form. This method involves natural language generation techniques to interpret the text and generate new sentences that convey the same ideas as the original text but using different words and sentence structures.

Moving further from summarization based on the content of text alone, Zhu et al. [70] introduced the task of multimodal summarization and created the Multimodal Summarization with Multimodal Output (MSMO) dataset built over CNN and Daily Mail articles. The dataset consists of text articles with associated images along with an LSTM (Long Short Term Memory) and attention based accompanying summary generation model. Liu et al. [71] improved the dataset by introducing a golden reference image for each datapoint involving ranking techniques such as rouge based overlap and order of occurrence in document. The text generation part of the problem of MSMO falls under the umbrella of **abstractive summarization**.

Zhu et al. showed that utilizing the multimodal information in the input improves the quality of the produced summaries. Further it has been shown in literature [70, 52] that multimodal outputs in such tasks yield increased human contentment since an image-text combination results in better cognition and understanding for the readers. They provide extensive metrics to prove the correlation between multimodality and human satisfaction. However, the summary generated by the model proposed by [71] is poorer when compared to the best performing text only summarization models over automated metrics such as ROUGE which show that the overlap of textual content suffers when multimodality is introduced. This can be attributed to various facts such as inefficient combination of modalities, weak decoding methods, etc.

The rise of pre-trained language models (such as the likes of BERT and GPT) has led to improved models for several tasks including summarization, such as BERTSUM [29]. Further, with the spread of the transformer architecture [58] to domains beyond Natural Language Processing, such as computer vision, large pre-trained models trained on images and text parallelly have emerged, some examples of which include OSCAR [24] and CLIP [43]. In our work, we propose a novel model which takes advantage of image data with the help of such advances and uses it as a guidance signal while decoding the summaries, and an encoder which fuses the modalities utilizing semantic image retrieval.

Table 4.1 shows the outputs from our model compared against the gold summary. We can see that our model is capable of generating novel keywords gathered from the image context that are absent in the gold summary, as well as in summary generated from a text-only summarization model.


Top Image	
Gold Summary	Photo shows grey tabby perched on a tree stump with camera 'in its paw'. Funny picture also shows three dogs lurking in the background. Has had more than one million likes and thousands of shares on instagram.
Text-only model	The bizarre photo has gone viral with more than one million likes on instagram in a couple of days, and thousands of shares. It shows the tabby perching on a tree stump as it appears to reach out to take the self-portrait safely out of reach of three dogs poised behind it in a field.
Our Model	A photo that appears to be a selfie taken by a cat has gone viral with more than one million likes on instagram in a couple of days. The bizarre snap shows the grey and white tabby perching on a tree stump as it appears to reach out to take the self-portrait safely out of reach of three dogs . In one snap the tabby has three feet squarely on the ground but in another it balances on its hind legs .

Table 4.1 Text in **orange** shows the textual overlap between gold and predicted summaries, while text in **blue** shows the additional context our model gathers from the image.

4.2 Multimodal Summarization

The problem statement we are trying to solve can be formally described as follows - Given an article containing text T and a set of images I , we need to generate a textual summary t such that $length(t) < length(T)$. Along with the summary we also need to return the most relevant image $i \in I$ that complements the generated summary. In the dataset, while we possess ground truth summaries for training the textual part of the model, we do **not** have the access to ground truth relevant images while training (however, these are present in the validation/testing iterations).

In the sections that follow, we describe the modules within our architecture in detail. Fig. 4.1 shows an overview of the model.

4.2.1 Encoder

We begin with encoding the input text and input images via 2 independent encoders.

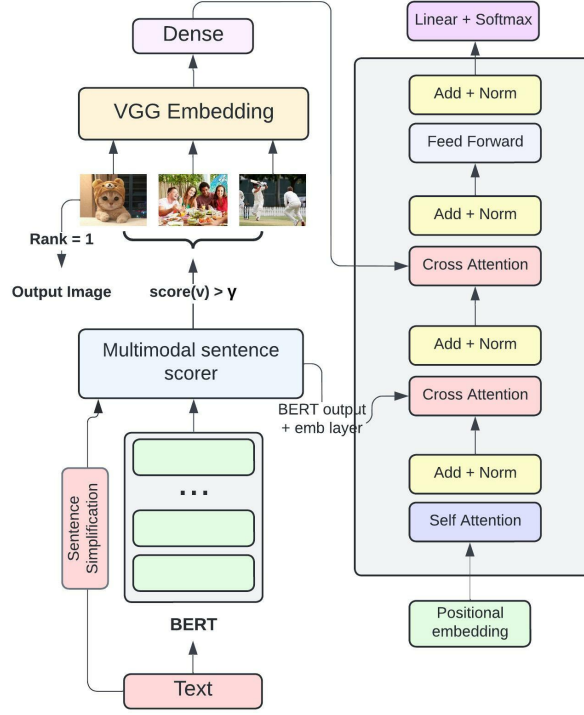


Figure 4.1 Model Overview

4.2.1.1 Text Encoder

We use BERT [8] for generating contextual embeddings of size $d \times d_h$ where d is the number of tokens and d_h is the hidden dimension of our architecture. Since BERT only allows for a maximum context size of 512 tokens, we truncate our documents at that limit.

4.2.1.2 Image Encoder

We use VGG-19 [51] embeddings generated from the 'fc2' layer of the deep convolutional net. We set a threshold η , and only consider embeddings of the top- η images returned by our multimodal scorer, explained in the next section. The size of the returned embeddings is $\eta \times d_v$. In case the document has $< \eta$ images, we pad the embeddings with zero vectors. In order to utilize these embeddings with cross attention within our architecture, we perform a linear transform with a dense layer to get embeddings of final size $\eta \times d_h$.

4.2.2 Multimodal Image and Sentence Scorer

For summarizing both textual and visual information together, we develop a scoring mechanism for each modality using our multimodal model's information retrieval setup, where we

feed the model text and image (extracted faster-rcnn [45] image features in the case of OSCAR) and it outputs a single scalar indicating the semantic similarity of the image to the text. For each article D with a set $S = \{s_1, s_2, \dots, s_a\}$ of sentences of size a and containing a set $V = \{v_1, v_2, \dots, v_b\}$ of images of size b , we define $\phi(t, v)$ as the multimodal model score for every pair of text t and image v . We use OSCAR [24] and CLIP [43] as our multimodal models. We briefly discuss these models here -

- **OSCAR** - It is a vision-language pretrained model developed by Microsoft. It is trained using a large-scale corpus of text and images. The model consists of a visual encoder based on the ResNet architecture and a textual encoder based on the Transformer architecture. During pretraining, OSCAR learns image and text alignment using “anchors” within the modalities that refer to simple objects. OSCAR has demonstrated state-of-the-art performance on a range of language+vision tasks and has been used for a variety of downstream applications.
- **CLIP** - It is a deep learning model developed by OpenAI that can understand both images and text using a unified approach. It is pre-trained on a large multimodal dataset (image + text) and utilizes contrastive learning. CLIP achieves state of the art performance in zero-shot learning, recognizing objects and concepts it has never seen before, due to its ability to generalize visual features.

4.2.3 Sentence Simplification

Before calculating ϕ , we simplify sentences to remove named entities, as OSCAR is pre-trained on the COCO [27] dataset, which contains simple objects. We replace named entities with their closest COCO class using GloVe word embeddings [41]; and remove phrases with dependencies which contain abstract objects like time and place using Spacy [15]. **Without this simplification process, the output of the model is illegible** owing to a lack of understanding of the language by our multimodal scorer.

4.2.4 Scorer

Now, we define the score for each sentence s and image im as,

$$score(s) = \sum_{j=1}^b \phi(s, v_j) \quad (4.1)$$

$$score(im) = \sum_{i=1}^a \phi(s_i, im) \quad (4.2)$$

We sort the images by $score(im)$. Those with rank $> \eta$ are discarded. The remaining top- η images are used for our decoder. The image with rank 1 is our desired **multimodal output**.

Now, we alter our text encoding, and augment it by incorporating these scores we calculated. For each token t , our text encoder BERT generates an embedding, which we multiply by the score of the sentence the token belongs to. Formally,

$$Emb(t) = Bert(t) \times Score(s)$$

Where $t \in s$, and $Bert(t)$ refers to the BERT embedding of token t . $Emb(t)$ refers to the final embedding to be passed on further in the model.

4.2.5 Decoder

The decoder for our model is inspired from the GSum architecture [10], where, we pass additional information to the model using an extra cross attention layer in the decoder. Post linear transformation of the VGG-embeddings to d_h , they can be directly passed to the transformer decoder as the hidden dimension of the transformer is the same. The remaining decoder layers are identical to that of a standard transformer layout.

4.2.6 Training Methodology

While training, we have golden reference summaries against which we compute the standard Negative log likelihood loss,

$$\mathcal{L}_{NLL} = -\frac{1}{K} \sum_{k=1}^K \log p(s_k|c),$$

where $p(s_k|c)$ represents the probability of token s_k given previous tokens and context c .

4.3 Baselines

To test the efficacy of our proposed model, we compare against three baselines. We use one unimodal baselines, i.e., BertSum, while 2 multimodal baselines.

- **BertSum** - The model [31] uses BERT fine tuning with token level alternating sentence embeddings [8] to encode text and a standard transformer decoder to generate the summaries.
- **MSMO** - The model [70] consists of a bidirectional LSTM [13] to embed text and VGG-19 [51] to embed images. Bahdanau Attention is used to combine both the modalities and the output is decoded through a unidirectional LSTM layer.

- **UniMS** - [67] proposed a multimodal summarization model that uses BART’s [23] seq-2-seq architecture to pass both text and image embeddings by concatenating them. It uses CLIP for knowledge distillation and to guide image selection.

The code for MSMO and UniMS is not publically available, so we compare our findings with their reported ROUGE scores only and Image Precision calculated over identical train-val-test splits.

4.4 Experiments

For our experiments, we make use of the MSMO dataset [70], which consists of scraped articles from CNN and daily-mail websites along with the images contained on the page.

4.4.1 Dataset statistics

The full dataset consists of 314,581 articles, with each article containing a median of 6 images. The total number of images across all articles exceeds 1.5 million. Due to space and computation constraints, we keep only the top 7 images of each article (the images are sorted in order of their occurrence in the article). While the dataset does not contain ground truth images for training, it does contain a list of reference images for the articles from the test set to be used for evaluation.

4.4.2 Implementation Details

We make use of BERT-base to encode the text, which has a hidden dimension of 768, so we set $d_h = 768$. Due to computational constraints, for our encoder, we use BERT weights from a checkpoint of BertSum (one of our baselines) and freeze the weights at the time of training. We use Adam optimizer [21] for managing gradient updation. The decoder is 6 layers deep (with 8 attention heads per layer) and uses 8000 warmup steps following a learning rate decay of 0.02. We make use of the base setup of OSCAR and ViT-B-32 version of CLIP for our multimodal scorers. Our model takes roughly 1 day to train on a Nvidia GTX 1080 Ti GPU.

4.4.3 Evaluation Metrics

For automated metrics, we report ROUGE [25], as is common practice in summarization literature. We also report the F1 score of BERTScore [66], which utilizes contextualized BERT embeddings for calculating similarity. This tests the summarization quality on a semantic level unlike ROUGE, and has been demonstrated to possess a high correlation with manual, human based annotators [66].

Evaluation of image outputs from our model is performed with respect to reference images from testing set of the MSMO dataset by computing the image precision given by,

$$IP = \frac{|\{\text{ref}_{img}\} \cap \{\text{rec}_{img}\}|}{|\{\text{rec}_{img}\}|}$$

where ref_{img} and rec_{img} refer to the reference images and our model recommended image respectively. We report this due to its high correlation with human satisfaction [70].

Apart from automated metrics, we also examine the results of manual (human) evaluation on the outputs generated through our model. We ask annotators to perform a double blind assessment by asking them to rate 50 randomly sampled predicted summaries + image on a scale of 1-5 (1 being incomprehensible; 5 being human-like). For the text-only baseline, only the text was shown.

4.4.4 Results

Scores from our experimentation are shown in Table 4.2. BERTScore or human evaluation for MSMO and UniMS are not available due to their model not being made public. Image Precision is omitted for BertSum as it is text-only.

With respect to the automated metrics, transformer based models beat LSTM based MSMO by quite a margin. We observe that through the use of images in our architecture, we are able to beat our strong text-only baseline (BertSum) with respect to overlap, semantic similarity as well as manual evaluation. Our model also shows a much higher image precision compared to the other models. We also see that OSCAR performs better than CLIP as our multimodal model with respect to both ROUGE and IP scores.

<i>Model</i>	<i>R1</i>	<i>R2</i>	<i>RL</i>	<i>BERT-F1 Scores</i>	<i>Human Scores</i>	<i>Image Precision</i>
MSMO	40.86	18.27	37.75	-	-	62.44
BertSum	42.13	19.60	39.18	0.810	2.80	-
UniMS	42.94	20.50	40.96	-	-	69.38
Ours _{CLIP}	42.28	19.37	39.81	0.852	3.01	67.41
Ours _{OSCAR}	42.51	19.97	39.28	0.882	3.17	77.99

Table 4.2 Comparison of our models with the baselines.

4.5 Discussions

The rise in BERTScore tells us that the summaries generated by our model are semantically closer to the gold and a rise in Human scores tells us that they are more satisfactory. **Due to both BERT and Human scores being based on semantics (as opposed to ROUGE being based on text overlap alone), they are a much better measure of summarization quality.** Hence, our best performing model from Table 4.2 is **Ours_{OSCAR}**, with the highest BERTscore, Human score and IP. Our model outscores a strong text-only baseline (BertSum) across all metrics. The higher image precision of our model compared to MSMO and UniMS displays the efficacy of our multimodal scorer module. While UniMS does outscore us by a tiny margin on the ROUGE scores, it does so by utilizing roughly **150 million more parameters**, showcasing our architecture’s efficiency.

4.6 Conclusion

In this work we see the potential of utilizing multimodal information for the task of summarization. We propose an architecture through which we are able to achieve high summarization qualities effectively as well as efficiently. As future work, we would like to improve this architecture further with the use of stronger text and image encoders, which require higher compute resources.

Chapter 5

Personalized content recommendation

Content ranking is one of the most important aspects of search engines and rank aggregation websites such as LinkedIn Learning¹. It is responsible for driving and boosting user interaction. Many large search systems use deep learning based scoring systems to rank the content, which take as input many features related to the content to be scored, and the user the content is being scored for. In this chapter, we take a look at modelling user interaction over a time period as a feature to be used by a search system.

5.1 Introduction

Within modern search systems used by websites that host user traffic in millions, deep learning based scorers form one component out of thousands. Some of the other components include scoring algorithms such as PageRank [3], BM25, Tf-Idf which make use of rudimentary features within text and are much more efficient than deep learning methods. There also exist scoring algorithms that improve scores of sponsored posts to drive revenue for search engines. However, the use of deep learning based modules allows search engines to learn patterns which might have been overlooked by traditional rule-based methods. These patterns are learned within the weights of a black box neural network and help in assisting search queries be more precise and relevant. The neural networks work so well in practice, that the most popular search engine, Google, has been utilizing these since 2015².

In this chapter we showcase work towards creating user embeddings for a large search system, that utilizes user's past interactions with the platform. This project was done as part of an Internship at LinkedIn, with the motivation of improving the search system of the LinkedIn Learning platform by making its recommendations user-specific. We start by modelling user interaction as a classification problem, then follow it up by extracting representations from the intermediate layers, and using the same for the search system.

¹<https://learning.linkedin.com/>

²<https://blog.google/products/search/how-ai-powers-great-search-results/>

5.2 Data

The data for this statement consists of users, along with a list of their previous interactions with the search system. We define an **interaction** as a user inputting a query into the search bar, or performing an action on a particular result from a previous search (any action such as clicking, engaging, bookmarking). Each of these actions (total 7) is converted into a label. We use a binary variable for each interaction to denote whether it was a query or not, and each interaction also includes the text associated with it (title in case of post/course, query text in case of query) and a timestamp. Figure 5.1 shows the interactions of one such user.

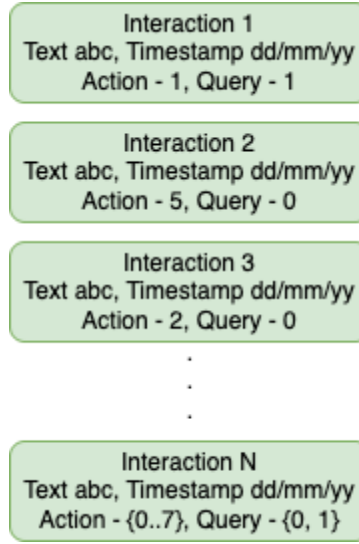


Figure 5.1 Sample of a user's activity in the dataset.

For each row in the dataset, we also have a target interaction, which has associated text, query and timestamp label, and we need to predict the action label.

5.3 Problem Description

Formally, we can define the problem statement as follows -

We define an interaction I as the set $\{T, \tau, A, Q\}$, where T refers to the text, τ refers to the timestamp, A refers to the action and Q refers to whether the interaction is a query or not. Then, given a list L of interactions I , i.e. $L = \{I_1, I_2, \dots, I_N\}$, and a target interaction I_t consisting of $\{T, \tau, Q\}$, we need to predict A for I_t .

Since we have to predict $A \in \{1, 2, \dots, 7\}$, this problem is essentially a multi-class classification problem, and in the next section we propose a model for the same.

5.4 Model

Here, we define an approach used to solve the problem. The model is distributed into various modules, and we discuss them sequentially.

5.4.1 Interaction Embedding

We start by encoding each of the entities within an Interaction I to a vector of a fixed dimension -

- **Text** - To encoder the text T , we make use of a standard BERT transformer encoder and use the pooled representation from the output. This gives us an embedding E_T .
- **Timestamp** - To encode timestamp τ , we use a technique called Time2Vec [18]. The algorithm for Time2Vec works similarly to positional embeddings, and embeds a given timestamp (with relation to other timestamps in the sequence) into a vector space. Let $T2V$ be the desired embedding for timestamp τ at index i within the interactions list. Then $T2V$ is given by -

$$T2V(\tau)[i] = \begin{cases} \omega_i \tau + \varphi_i, & \text{if } i = 0 \\ \sin(\omega_i \tau + \varphi_i), & \text{otherwise} \end{cases} \quad (5.1)$$

Here, ω and φ are learnable parameters. We use the sin function in our setup, however any other function capable of capturing periodic behaviour (such as \cos) also works. The function $T2V$ gives us an embedding E_τ .

- **Action** - We encode action A using an embedding layer that converts each of the 7 labels into a learnable vector, giving the embedding E_A .
- **Query** - We encode the query boolean variable Q using a similar embedding layer as Action embedding. It converts the boolean variable having values 1 or 0 to a learnable vector, giving the embedding E_Q .

From these embeddings, i.e. E_T, E_τ, E_A, E_Q , we get a final embedding by adding them all together.

$$E_I = E_T + E_\tau + E_A + E_Q$$

Where E_I represents the embedding of the interaction. We make sure that the output of each of our encoding layers results in vectors of equal dimension to enable addition.

Once we are able to encode an interaction, we can encode a list of interactions - $L_u = \{I_1, I_2, \dots, I_N\}$, where L_u represents the interactions for a user u . This gives us a list of embedding $E_{L_u} = \{E_{I_1}, E_{I_2}, \dots, E_{I_N}\}$ for each user.

5.4.2 Transformer Encoder

To make use of the encoded list of embeddings, we use a standard transformer encoder to transform them into a single contextualized vector. Figure 5.2 shows the architecture of the same.

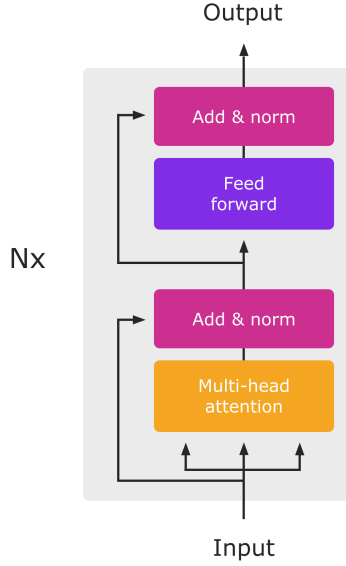


Figure 5.2 Transformer Encoder

In the transformer introduced in [56], the input provided to the model is textual in nature. We simply change the input to be the list of interaction embeddings received from the previous step.

As output, our model gives us a vector embedding E_u , which represents the embedding of a user u . **This is the embedding that is supplied to the bigger search system as a feature.** However, we still need to train this model so that the embeddings are meaningful.

5.4.3 Training

Once we have the embedding E_u , we add the embeddings of the target interaction I_t given by $\{E_{T_t}, E_{\tau_t}, E_{Q_t}\}$, and encoded using the same tools as mentioned in section 5.4.1.

$$E'_u = E_u + E_{T_t} + E_{\tau_t} + E_{Q_t}$$

This gives us E'_u . We then make use of a single classifier layer of appropriate dimensions to result in a vector of size 7, representing the set of possible actions.

$$A = \operatorname{argmax}(\operatorname{softmax}(W \times E'_u + b))$$

Where A is the final predicted action and W, b are the appropriate dimension-ed weights and biases. To train the model, we make use of a **cross entropy loss** function against the gold labels in the dataset. Figure 5.3 shows the full architecture.

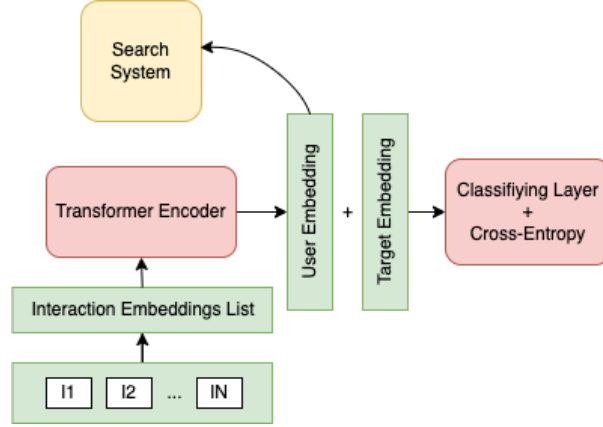


Figure 5.3 Model Architecture

5.5 Results

The model achieves an accuracy of 77% and 73% on the train and validation sets respectively. The results produced over inference on the test set and the model's affect on the search system are not available to be shared through this thesis.

5.6 Conclusion

In this chapter, we looked at one potential approach towards improving search systems using features created by deep learning architectures. We showcased a technique to embed a time series of user interactions to generate a unified contextualized embedding, by utilizing prediction on user actions to train a model.

Chapter 6

Indic Multimodal Dataset creation

With the advent of transformers and rise in the creation and access of models such as DALL-E, StableDiffusion and Imagen, it is undoubtedly clear that multimodal models possess great versatility and provide value to people beyond academia. However, training such models requires very large amounts of clean data. This clean data while plentiful in English, is not easily accessible in more obscure languages. In order to make multimodal machine learning accessible in an Indian context, in this chapter, we propose our methodology to create a large scale Image-text pair multimodal dataset in 11 Indian languages by pruning down the Samanantar dataset to produce high quality caption-like sentences.

6.1 Introduction

Vision-Language pre-trained models such as OSCAR [24] and CLIP [43] require large amounts of multimodal data. Within English, many such datasets exist. Some of the popular ones are -

- **MS-COCO** - Released by microsoft, it consists of over 300,000 images, with each image consisting of 5 captions. COCO stands for “Common Objects in Context”, and the dataset only consists of common objects within 81 categories present in some standard context. Example captions -
 - two jets fly overhead while the crowds look.
 - a large pig and small dog stand next to a truck.
- **Flickr8k** - This dataset consists of 8000 images collected from the Flickr website, with each image consisting of 5 captions, similar to COCO. The images were chosen from six different Flickr groups, and tend not to contain any well-known people or locations, but were manually selected to depict a variety of scenes and situations. Example captions -
 - A girl going into a wooden building.
 - Two constructions workers sit on a beam taking a break.

- **Conceptual Captions** - This dataset was built by Google [47] and consists of over 3 million image text pairs. The dataset was collected by scraping the internet for all possible image-text pairs followed by rigorous pruning to eliminate noise. We discuss in this chapter further about what "noise" can be defined as in this context. Example captions -

- interior design of modern living room with fireplace in a new house.
- even though agricultural conditions are not ideal for growing tobacco , there is indigenous production.

We can see that the captions of conceptual captions are of a lower quality as compared to MS-COCO and Flickr8k, however they are also much more in number, which is why it is a valuable dataset for the training of large models. OpenAI’s CLIP [43] also makes use of a filtering technique similar to Conceptual Captions, however their dataset is not made public.

One thing shared between all the big image-text datasets is their choice of language, which is English. Currently, there exists no authentic dataset which sufficiently covers Indian languages. By “authentic” we mean captions created directly in target languages and not translated from a language like English or French, for which data is abundant (such an approach already exists as well¹). In this chapter, we explore ways to start from a high volume Indic dataset like Samanantar [44] and process it over multiple pipelines to get a multimodal parallel corpus in Indic languages.

Our motivation to do this comes from 2 main reasons. Firstly, translation from English to certain Indic languages is not good enough, and captions generated authentically would not suffer from mistranslations. Secondly, large scale multimodal models do not possess information in an Indian context. Information for events such as Diwali, Janamashtami, Pongal and understanding of Indian-centric ideas is lacking within translation datasets. In our methodology of using sentences from a dataset like Samanantar, we can induce models to learn such concepts.

6.2 Quality of a caption

For the creation of a dataset, we need to have high quality sentences that correspond well with an image. However, on simply scraping the internet for image-text pairs, we come across many examples which are not fit to be included in a general purpose vision-language model’s dataset. Some groups which these sentences belong to are -

- **File Photos** - This class refers to photos that label a named entity. While they do provide information about the entity present in the picture, they are too specific and

¹<https://github.com/shantipriyap/Hindi-Visual-Genome-1.0>

hinder generalization of vision language models.

Example - a file photo might contain the photo of the president of the France with a caption “Emmanuel Macron”.

Reasoning - While the caption tells us that the face in the image is of Emmanuel Macron, labelling an individual face out of billions is an unnecessary task for a general purpose model.

- **Named Entity Rich captions** - This class refers to captions which contain too many named entities. These image-text pairs suffer from the same problem as that posed by file photos, i.e., they inhibit generalization.

Example - Photo of a concert with the caption “Dua Lipa performing in the Anfield at London”.

Reasoning - In order to successfully process a caption like this, the model needs to have an understanding that “Dua Lipa” is a person who does “performing” of some sort and “Anfield” is the name of a stadium which is present in London. It is also hard to deduce the venue from an image of a concert, which further reduces the efficacy of this data-point.

- **Information Rich captions** - This class refers to captions which contain too much information in general, that the image fails to provide. The presence of too much information in the caption while too little in the image hinders the neural networks ability to form connections between the image and text.

Example - A close-up of “Mahatma Gandhi” captioned “Mahatma Gandhi in South Africa fighting a court case”.

Reasoning - While the caption describes a generic situation that a model can learn, it is too specific and the image might not correspond well with it. From a close-up of Gandhi, it is impossible to tell whether he is in South Africa or if he is fighting a court case.

- **Out-of-Context captions** - This class consists of captions that are part of a bigger text and do not make sense when isolated out of context. This is a common occurrence while scraping text from blogging websites.

Example - Image of a tree with the caption “Mike loves to climb trees”.

Reasoning - An example like this provides no valuable information out of context since we have no way to decipher who “Mike” is and what “climbing” refers to in this context with the image of an ordinary tree.

These classes are not exhaustive and building a multimodal dataset that does not contain poor captions requires a long list of pruning rules along with machine learning based filtration methods.

6.3 Dataset creation

In order to create our multimodal Indic dataset, we start by taking all the sentences within the Samanantar dataset. Within samanantar, each datapoint consists of an english sentence, along with its corresponding Indic language interpretation. An example datapoint -

- English - However, Paes, who was partnering Australia’s Paul Hanley, could only go as far as the quarterfinals where they lost to Bhupathi and Knowles.
- Hindi - आस्ट्रेलिया के पाल हेनली के साथ जोड़ी बनाने वाले पेस मियामी में क्वार्टरफाइनल तक ही पहुंच सके क्योंकि इस दौर में उन्हें भूपति और नोल्स ने हराया था।

Since we have access to the english version of each sentence, we can use them to classify datapoints as possibly denoting a caption or not. We build a deep learning based classifier that is able to filter such sentences. We discuss architecture of the classifier, datasets used and its training procedure in the following sections.

6.3.1 Classifier

For the classifier, we make use of vanilla BERT-base-uncased from huggingface² as our text-encoder, followed by a linear layer for classification. Figure 6.1 shows the architecture.

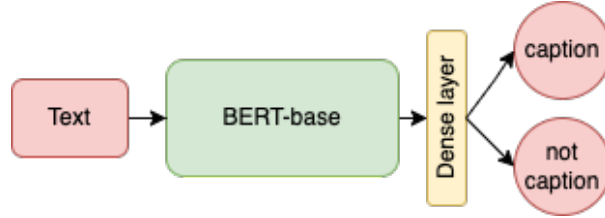


Figure 6.1 Classifier architecture

We also attempted the use of stronger text encoders such as RoBERTa [32] and XLNet [63], however for our purposes, we found vanilla BERT to perform the best.

6.3.1.1 Datasets for training

To train this classifier, we must provide it with positive and negative samples for the captions class.

- **Positive samples** - Positive samples here refers to sentences which are potential captions. To get these sentences, we make use of the multiple image-text pair datasets available publically such as MS-COCO and Conceptual captions. We take only the captions from

²<https://huggingface.co/bert-base-uncased>

these two datasets and discard the images. These sentences act as our positive samples. Total number of positive samples collected is 3,605,148.

- **Negative samples** - Negative samples here corresponds to sentences which cannot serve as captions to images. This part of the dataset consists of regular everyday english sentences, since a majority of sentences do not classify as captions. We also try to include named-entity rich sentences in this class to discourage classifying them as captions due to the reasons mentioned in section 6.2. We make use of sentences from the popular CNN-Dailymail summarization dataset [40] and the bookcorpus dataset [72]. These sentences include a good mix of the required attributes - being everyday english sentences and being named entity rich. Total negative samples collected is 6,417,256.

6.3.1.2 Training splits

We use a validation set of size 1% of the total sentences, consisting of roughly 100,000 sentences. Since this model eventually needs to be run on the Samanantar dataset, we take a subset of 200 sentences from Samanantar and manually annotate them to create a test-set. The test-set consists of 83 positive samples and 117 negative samples.

To train the model we make use of a standard binary cross-entropy loss. We use the results on the validation set to tune our hyperparameters.

6.3.1.3 Results

Through our extensive experimentation, we were able to achieve a throughput of roughly 0.17% on Samanantar. This means that for every 10,000 sentences in the bigger Samanantar dataset, we were able to classify 17 sentence as being a possible caption. For a total of 47 million sentences, we were able to capture approximately 153,000 sentences.

On our manually annotated test set, we were able to achieve an accuracy of 84%. We continue to improve the classifier to improve the throughput percentage and accuracy.

6.3.2 Image Collection

To collect images for each sample, we convert each sentence into a query. To do this, we first use spacy to perform Named Entity Recognition (NER) on the sentence. This gives us the named entities and their corresponding classes. We simplify entities such as Organizations (NASA, ISRO), Money (1 billion Dollars, 2 million Rupees) and dates by their corresponding classes, while leaving out entities such as place and name unchanged. Through our experimentation we find out that this results in less noisier output from Google images.

Using a scraper built from the Selenium library³, we scrape the top 3 image-urls from the search results and store them in the dataset. In total, we collect roughly 500,000 image-urls.

If any of the top 3 image-urls contain an image with a pixel count lower than 150×150 , we consider the next most relevant image in place of it, since we deem it to be lacking in information.

6.4 Dataset Samples

In this section, we take a look at some of the rows collected from our dataset.

While some sentences have images that correspond almost perfectly with the captions, there is a decent amount of noise in the samples as well. Looking at table 6.1, in sample 4, the caption says “rural home”, while the image cannot support that claim. In example 5, the caption says “half teaspoon”, however the image shows 2 spoons of different sizes, with both of them full.

6.5 Conclusion

We spent this chapter discussing the state of multimodal datasets, and the need for an authentically generated Indic multimodal dataset. We then looked at how we can prune the Samanantar dataset to filter sentences which can act as potential captions, along with using Google images to gather images corresponding to the captions, and be used to create the Indic dataset. In the future, we aim to release large vision-language models trained on this dataset and having the ability to be fine-tuned over several downstream multimodal tasks such as Visual Question Answering, Image Retrieval, Caption Generation, etc.

For future experimentation, we aim to improve the quality of our classifier to increase in a larger dataset size. We are still dealing with sentences that can serve as good captions that get classified as being non-captions. Through data analysis on random subsets of the dataset, we believe a throughput percentage of 1-1.5% should be achievable, essentially improving the size of our dataset manyfold.

³<https://www.selenium.dev/>





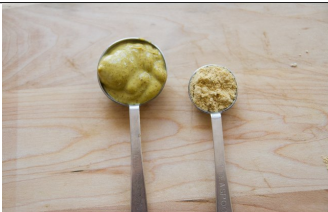
Top Image	English	Indic Language
	Ram temple in Ayodhya.	अयोध्या में राम मंदिरा (Hindi)
	Fire at a building located in masjid area.	मशीद परिसरात असलेल्या इमारतीला आग (Marathi)
	Indian women hockey team in final.	இந்திய மகளிர் ஹாக்கி அணி இறுதிப்போட்டியில் (Tamil)
	Micro solar dome lighting up a rural home.	মাইক্রো সৌর গম্বুজ একটি গ্রামীণ বাড়িতে আলোকিত (Bengali)
	half teaspoon of mustard seeds.	ਰਾਈ ਦੇ ਬੀਜ ਦਾ ਅੱਧਾ ਚਮਚ (Punjabi)

Table 6.1 Examples from the multimodal Indic dataset.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we tackled multiple problems that helped us get a better understanding of interactions within modalities in neural networks. We were able to achieve improvements over past works, and demonstrate their working in detail within this thesis. We covered problem statements regarding fake news detection, multimodal summarization, content ranking and multimodal Indic dataset creation.

In chapter 2, we introduced the reader to various concepts within the realm of multimodality, along with providing background knowledge on the tasks mentioned in the paragraph above.

Within chapter 3, we discussed ways to impede the problem of fake news within a multimodal setting by making use of deep learning models. We explored how the transformer architecture applied to embeddings generated using pre-trained models could be useful to improve modality interactions. We managed to achieve state of the art scores across metrics through our proposed architecture, and further scrutinized our model through rigorous ablation studies. We also provided various case studies to examine in detail what our model was getting right, that enabled it to outperform the others.

We follow it up with chapter 4, discussing how to aid the task of text summarization using Images as a supporting modality. We take a look at general purpose multimodal understanding models such as OSCAR, which perform well at retrieval, and utilize them with a summarization architecture that is able to score sentences and images for better. This helps us build a knowledge distillation setup that filters out sentences considered irrelevant. We also made use of techniques such as constrastive learning to widen the gap between relevant and irrelevant. Our results proved that our proposed model gets quite close to the state of the art while being much more efficient.

In chapter 5, we entered the realm of content ranking, and devised a model that can create embeddings of individual users based on their previous interactions. We demonstrated how

user interactions can be embedded efficiently using a transformer architecture, and how time can be used as a feature using techniques such as Time2Vec.

In chapter 6, we turned our focus to solving the problem of lacking datasets in the multimodal space. In order to boost the research on multimodal architectures in the Indic context, we devised a way to prune the Samanantar dataset and get caption-like sentences along with their corresponding relevant images. We built a deep-learning based method for the same, and made use of popular datasets such as Conceptual Captions, CNN-DailyMail and BookCorpus to train the classification model. To our knowledge, this is also the first attempt to create an authentic indic multimodal dataset, and we hope it serves as a benchmark for future experimentation on multimodality.

As multimodality is a relatively recent topic, there is a lot of ongoing research in the field. The solutions proposed in this thesis are by no means exhaustive, or definitive. We finally conclude this thesis by discussing possible future work in some of these areas.

7.2 Future Work

7.2.1 Multimodal Fake News Detection

Currently, the proposed model strictly works with the provided knowledge, and does not consider into account facts collected from news articles or information from knowledge bases. We believe that in the future, models with improved modality interaction can be augmented with information retrieval based models that can extract facts from existing knowledge bases. This would add a layer of fact checking to the deep learning based classifier, and would also help the model become more explainable instead of a black box.

7.2.2 Image-aided Summarization

Due to computational constraints, the submodules utilized within our proposed model are restricted in their sizes. The best performing model currently in use for the Multimodal Summarization with Multimodal Output (MSMO) task, i.e. UniMS, consists of over 500 Million parameters. We believe that stronger models such as BART when used in conjunction with our framework to rank sentences and images through a multimodal model (like OSCAR) would perform exceptionally well and exceed the performance of UniMS. We aim to achieve the same in our future work on this problem statement.

Moreover, with the rise of natural text to image generation models such as Dalle and StableDiffusion, we hypothesize that we can augment the dataset for this problem statement, which can further help in improved summarization quality. We can also use these models to generate an image based on the end result of summarization, thereby resulting in the “multimodal

output” part of the statement to not be constrained with the images already present in the document.

7.2.3 Indic Multimodal Dataset

We produced a dataset consisting of roughly 150,000 sentences, but the original dataset we started pruning from consists of over 47 Million sentences. We believe there is scope for a lot of improvement in classification that can raise the throughput percentage to 1-1.5%, and raise the size of the dataset to around 500,000 sentences. Currently, our classifier consists of a simple transformer encoder. We theorize that the classification between caption versus non-caption is based a lot more on the structure of the sentence, and in our future work we aim to utilize architectures which take into account the parts of speech tags and usage of verbs to provide more accurate results.

Related Publications

1. **Tanmay Sachan**, Nikhil Pinnaparaju, Manish Gupta, Vasudeva Varma **SCATE: shared cross attention transformer encoders for multimodal fake news detection**. Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. 2021.
2. **Tanmay Sachan**, Anshul Padhi, Balaji Vasan Srinivasan, Vasudeva Varma **MMSumm: Multimodal Summarization via Semantic Reranking and Cross-Modal Knowledge Distillation**. Publication under review.

Bibliography

- [1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.
- [2] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization, 2016.
- [3] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30:107–117, 1998.
- [4] J. Cao, J. Guo, X. Li, Z. Jin, H. Guo, and J. Li. Automatic rumor detection on microblogs: A survey. *arXiv preprint arXiv:1807.03505*, 2018.
- [5] C. Castillo, M. Mendoza, and B. Poblete. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684, 2011.
- [6] T. Chen, L. Wu, X. Li, J. Zhang, H. Yin, and Y. Wang. Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection, 2017.
- [7] Detection and visualization of misleading content on Twitter. Boididou, christina and papadopoulos, symeon and zampoglou, markos and apostolidis, lazaros and papadopoulou, olga and kompatsiaris, yiannis. *International Journal of Multimedia Information Retrieval*, 7(1):71–86, 2018.
- [8] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.
- [9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [10] Z. Dou, P. Liu, H. Hayashi, Z. Jiang, and G. Neubig. Gsum: A general framework for guided neural abstractive summarization. *CoRR*, abs/2010.08014, 2020.
- [11] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding, 2016.
- [12] K. Fukushima. Cognitron: A self-organizing multilayered neural network. *Biological cybernetics*, 20(3-4):121–136, 1975.
- [13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

- [14] M. Hodosh, P. Young, and J. Hockenmaier. Flickr8k dataset.
- [15] M. Honnibal and I. Montani. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017.
- [16] Z. Jin, J. Cao, H. Guo, Y. Zhang, and J. Luo. Multimodal fusion with recurrent neural networks for rumor detection on microblogs. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 795–816, 2017.
- [17] Z. Junnan, Y. Zhou, J. Zhang, H. Li, C. Zong, and C. Li. Multimodal summarization with guidance of multimodal reference. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:9749–9756, 04 2020.
- [18] S. M. Kazemi, R. Goel, S. Eghbali, J. Ramanan, J. Sahota, S. Thakur, S. Wu, C. Smyth, P. Poupart, and M. Brubaker. Time2vec: Learning a vector representation of time, 2019.
- [19] D. Khattar, J. S. Goud, M. Gupta, and V. Varma. Mvae: Multimodal variational autoencoder for fake news detection. In *The World Wide Web Conference, WWW '19*, page 2915–2921, New York, NY, USA, 2019. Association for Computing Machinery.
- [20] D. Khattar, J. S. Goud, M. Gupta, and V. Varma. Mvae: Multimodal variational autoencoder for fake news detection. In *The World Wide Web Conference*, pages 2915–2921, 2019.
- [21] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017.
- [22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, NIPS'12*, page 1097–1105, Red Hook, NY, USA, 2012. Curran Associates Inc.
- [23] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, 2019.
- [24] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, Y. Choi, and J. Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. *CoRR*, abs/2004.06165, 2020.
- [25] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.
- [26] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft coco: Common objects in context, 2014.
- [27] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft coco: Common objects in context, 2015.
- [28] W.-H. Lin and A. Hauptmann. News video classification using svm-based multimodal classifiers and combination strategies. In *Proceedings of the Tenth ACM International Conference on Multimedia, MULTIMEDIA '02*, page 323–326, New York, NY, USA, 2002. Association for Computing Machinery.

- [29] Y. Liu. Fine-tune BERT for extractive summarization. *CoRR*, abs/1903.10318, 2019.
- [30] Y. Liu and M. Lapata. Text summarization with pretrained encoders, 2019.
- [31] Y. Liu and M. Lapata. Text summarization with pretrained encoders. *CoRR*, abs/1908.08345, 2019.
- [32] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.
- [33] J. Ma, W. Gao, and K.-F. Wong. Rumor detection on Twitter with tree-structured recursive neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1980–1989, Melbourne, Australia, July 2018. Association for Computational Linguistics.
- [34] A. Makhzani and B. Frey. k-sparse autoencoders, 2013.
- [35] S. Mannor, D. Peleg, and R. Rubinstein. The cross entropy method for classification. In *Proceedings of the 22nd International Conference on Machine Learning, ICML '05*, page 561–568, New York, NY, USA, 2005. Association for Computing Machinery.
- [36] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). 2014.
- [37] R. Mihalcea. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the ACL interactive poster and demonstration sessions*, pages 170–173, 2004.
- [38] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space, 2013.
- [39] R. Nallapati, F. Zhai, and B. Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.
- [40] R. Nallapati, B. Zhou, C. N. d. santos, C. Gulcehre, and B. Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. 2016.
- [41] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
- [42] D. Qi, L. Su, J. Song, E. Cui, T. Bharti, and A. Sacheti. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data, 2020.
- [43] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021.

- [44] G. Ramesh, S. Doddapaneni, A. Bheemaraj, M. Jobanputra, R. AK, A. Sharma, S. Sahoo, H. Didee, M. J, D. Kakwani, N. Kumar, A. Pradeep, S. Nagaraj, K. Deepak, V. Raghavan, A. Kunchukuttan, P. Kumar, and M. S. Khapra. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages, 2021.
- [45] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.
- [46] A. See, P. J. Liu, and C. D. Manning. Get to the point: Summarization with pointer-generator networks, 2017.
- [47] P. Sharma, N. Ding, S. Goodman, and R. Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018.
- [48] D. Shen, J.-T. Sun, H. Li, Q. Yang, and Z. Chen. Document summarization using conditional random fields. In *IJCAI*, volume 7, pages 2862–2867, 2007.
- [49] B. Shi and T. Weninger. Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-Based Systems*, 104:123–133, jul 2016.
- [50] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [51] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.
- [52] H. Singh, A. Nasery, D. Mehta, A. Agarwal, J. Lamba, and B. V. Srinivasan. MIMOQA: Multimodal input multimodal output question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5317–5332, Online, June 2021. Association for Computational Linguistics.
- [53] S. Singhal, R. R. Shah, T. Chakraborty, P. Kumaraguru, and S. Satoh. Spotfake: A multi-modal framework for fake news detection. In *2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM)*, pages 39–47, 2019.
- [54] S. Singhal, R. R. Shah, T. Chakraborty, P. Kumaraguru, and S. Satoh. Spotfake: A multi-modal framework for fake news detection. In *2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM)*, pages 39–47. IEEE, 2019.
- [55] C. Song, N. Ning, Y. Zhang, and B. Wu. A multimodal fake news detection model based on cross-modal attention residual and multichannel convolutional neural networks. *Information Processing Management*, 58(1):102437, 2021.
- [56] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2017.
- [57] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

- [58] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
- [59] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015.
- [60] S. Volkova, K. Shaffer, J. Y. Jang, and N. Hodas. Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 647–653, 2017.
- [61] Y. Wang, F. Ma, Z. Jin, Y. Yuan, G. Xun, K. Jha, L. Su, and J. Gao. Eann: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining*, pages 849–857, 2018.
- [62] Y. Yang, L. Zheng, J. Zhang, Q. Cui, Z. Li, and P. S. Yu. Ti-cnn: Convolutional neural networks for fake news detection, 2018.
- [63] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding, 2019.
- [64] H. Zhang, Q. Fang, S. Qian, and C. Xu. Multi-modal knowledge-aware event memory network for social media rumor detection. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1942–1951, 2019.
- [65] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization, 2019.
- [66] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with bert, 2020.
- [67] Z. Zhang, X. Meng, Y. Wang, X. Jiang, Q. Liu, and Z. Yang. Unims: A unified framework for multimodal summarization with knowledge distillation, 2021.
- [68] B. Zhou, Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus. Simple baseline for visual question answering. *arXiv preprint arXiv:1512.02167*, 2015.
- [69] J. Zhu, H. Li, T. Liu, Y. Zhou, J. Zhang, and C. Zong. MSMO: Multimodal summarization with multimodal output. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4154–4164, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics.
- [70] J. Zhu, H. Li, T. Liu, Y. Zhou, J. Zhang, and C. Zong. MSMO: Multimodal summarization with multimodal output. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4154–4164, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics.

- [71] J. Zhu, Y. Zhou, J. Zhang, H. Li, C. Zong, and C. Li. Multimodal summarization with guidance of multimodal reference. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9749–9756, Apr. 2020.
- [72] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.