ATTENTION MECHANISM

by

MIHAIL ENEV
URN: 6635123

A dissertation submitted in partial fulfilment of the
requirements for the award of

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

May 2023

Department of Computer Science
University of Surrey
Guildford GU2 7XH

Supervised by: Prof. Ferrante Neri

I declare that this dissertation is my own work and that the work of others is acknowledged and indicated by explicit references.

Mihail Enev
May 2023

© Copyright Mihail Enev, May 2023

Abstract

Attention mechanism is a powerful technique for modeling dependencies between different elements in a sequence. It has been widely used in natural language processing, computer vision, and other fields, and has led to significant improvements in various tasks, such as machine translation and image captioning. In this paper, I provide an overview of different approaches to attention mechanism, including multiplicative attention, additive attention, multi-head attention, and self-attention. We also analyze one of the most successful models that incorporates attention mechanism: the transformer. Finally, we highlight some of the limitations of attention mechanism and suggest directions for future research.

Acknowledgments

I am grateful to Professor Ferrante Neri for providing me with the opportunity to explore and analyze this problem. I am also thankful to my family, particularly my brother, who has been by my side throughout my educational journey.

Contents

List of Figures

Abbreviations

AG	Attention Gate
BERT	Bidirectional Encoder Representations from Transformers
BRNN	Bidirectional Recurrent Neural Networks
CNN	Convolutional Neural Network
DETR	Detection Transformer
DTP	Dilated Temporal Pyramid
GLTR	Global-Local Temporal Representation
GPT	Generative Pre-trained Transformer
GPU	Graphics Processing Unit
CV	Computer Vision
LSTM	Long Short-Term Memory
MLM	Masked Language Modeling
NLP	Natural Language Processing
NSP	Next Sentence Prediction
RNN	Recurrent Neural Network
SE	Squeeze-and-Excitation (Block)
SK	Selective Kernel (convolution)
SKNet	Selective Kernel Networks
TPU	Tensor Processing Unit
TSA	Temporal Self-Attention (Module)
ViT	Vision Transformer

Chapter 1

Introduction

1.1 Background

The attention mechanism is inspired by humans' and animals' ability to focus on important information while ignoring the noise, information that is going to distract them. For example, when hunting, cheetahs are able to quickly compute a wealth of information about their prey and the obstacles in their path, allowing them to focus on the key elements that will help them achieve their goal. This ability to efficiently process and prioritize information in a limited amount of time is analogous to the attention mechanism in machine learning.

This idea of allowing models to selectively attend to different elements of a sequence solves a major problem of information overload, which affects all types of neural network models. The result of this ability to improve the accuracy and efficiency of model predictions is the reason why attention mechanism has become increasingly popular in deep learning. There are several categories of attention mechanism approaches, such as the softness of the attention, forms of input feature, input representations, and output representations, which depend on the model task, the input data type and size, and the desired output format.

1.2 Project Description

This project attempts to provide a comprehensive and critical review of attention mechanisms in deep learning, focusing on their mathematical foundations, practical applications, and limitations. By examining the strengths and weaknesses of attention mechanisms, the project seeks to clarify their potential to enhance the performance of deep learning models and identify future research directions in this field.

1.3 Aim and Objectives

The aim of this paper is to provide a comprehensive and critical review of attention mechanisms in deep learning, covering their mathematical foundations, practical applications, and limitations. By examining the strengths and weaknesses of attention mechanisms, this paper aims to provide insights into the potential for these techniques to improve the performance of deep learning models and to identify future research directions in this area.
The objectives of this project are:

1. To provide an overview of the different types of attention mechanisms in deep learning, including their mathematical formulations and their role in neural network architectures.
2. To critically evaluate the performance of attention mechanisms in different applications of deep learning, such as natural language processing and computer vision.
3. To analyze the strengths and limitations of attention mechanisms, including their impact on model performance, their computational efficiency, and their sensitivity to different hyperparameters and data distributions.
4. To provide practical guidance for researchers and practitioners on the appropriate use of attention mechanisms in deep learning, including recommendations for choosing the right type of attention mechanism, tuning hyperparameters, and avoiding common pitfalls.
5. To identify potential future research directions in attention mechanisms, such as the development of new types of attention mechanisms, the integration of attention mechanisms with other types of neural network architectures, and the application of attention mechanisms to new domains and applications.

1.4 Report Structure

Chapter 1 - Introduction

This chapter details the project background, aims, objectives and problem to solve.

Chapter 2 - Literature Review

This chapter provides an overview of the existing research in directly affected areas for this project and highlights the key elements.

Chapter 3 - Attention Mechanism

This chapter focuses on the core idea behind attention mechanism, including its types, parameters and principles.

Chapter 4 - Attention Mechanism in Computer Vision

This chapter presents the use of attention mechanisms in CV tasks, including image classification and object detection. It analyses the main approaches and architectures used in attention-based CV models, and discusses the benefits and challenges of these methods.

Chapter 5 - Attention Mechanism in Natural Language Processing

This chapter presents the use of attention mechanisms in NLP tasks, including language modeling and text classification. It analyses the main approaches and architectures used in attention-based NLP models, and it discusses the benefits and challenges of these methods.

Chapter 6 - Computer Vision tasks with Transformer

This chapter focuses on applying an attention-based NLP architecture, Transformer, to CV tasks such as image classification and object detection to achieve better performance compared to other CV architectures.

Chapter 7 - Choosing the Right Attention Mechanism

This chapter focuses on the importance of selecting the appropriate attention mechanism for a given NLP or CV model and discusses the various types of attention mechanisms.

Chapter 8 - Attention mechanism: Benefits and Limitations

This chapter provides summarise on the benefits and the limitations of the attention mechanism in the different architectures.

Chapter 9 - Further Researches and Conclusion

This chapter highlights potential future research directions and offers concluding remarks on attention mechanisms in NLP and CV.

Chapter 10 - LSEP

This chapter focuses on the legal, social, ethical and professional issues.

Chapter 2

Literature Review

This chapter provides a brief review of the literature that is related to the project. Specifically, the literature review will focus on developments in the field of attention mechanisms in deep learning, with a focus on their applications in computer vision and natural language processing tasks. The review will begin by discussing the fundamental concepts and principles of attention mechanisms before exploring some of the most important architectures that have been developed using this mechanism.

2.1 Attention mechanism overview

Attention mechanisms have become one of the most important concepts in the field of deep learning. In their paper, Niu et al. (Niu, Zhong & Yu 2021) closely review the evolution of attention. It starts with the Bahdanau et al. (Bahdanau, Cho & Bengio 2014) idea to use attention mechanism for a translation task and continues with the recent Transformer architecture introduced by Vaswani et al. The authors explain the idea behind the attention mechanism and categorise it according to four criteria, including the softness of attention, forms of input feature, input representation, and output representation. This categorization provides a valuable framework for understanding the different ways in which attention mechanisms can be applied in deep learning models.

2.2 Attention mechanism in Computer Vision tasks

The attention mechanism has emerged as a highly valuable tool in the domain of computer vision, empowering models to effectively concentrate on relevant image regions and amplify their overall performance. It has become widely adopted in the field, revolutionising the way visual information is processed. The papers that are introduced in this section have not only introduced the concept of attention in the context of computer vision but have also showcased its immense potential in improving various tasks within the field.

The "Attention U-Net: Learning Where to Look for the Pancreas" paper (Oktay, Schlemper, Folgoc, Lee, Heinrich, Misawa, Mori, McDonagh, Hammerla, Kainz et al. 2018) introduces a novel attention mechanism known as the attention gate (AG), which serves a crucial role in guiding a model's focus towards specific regions of interest within an image Figure 2.1. These attention gates learn to assign importance weights to different image regions, allowing the model to selectively emphasize informative regions while suppressing irrelevant areas. By incorporating attention gates, the model can effectively localize and extract relevant features, leading to improved accuracy in tasks such as object detection and segmentation. Furthermore, the flexibility of attention gates makes them suitable for integration into various types of CNNs, enabling their widespread applicability across a range of computer vision tasks.

The Squeeze-and-Excitation (SE) network (Hu, Shen & Sun 2017) is a new type of attention mechanism widely used in computer vision called channel attention. In this approach, the SE block is embedded within CNNs to collect global information and enhance feature representations. The SE block performs channel-wise attention by adaptively recalibrating feature maps based on their importance. By explicitly modelling interdependencies among channels, the SE block allows the model to allocate more attention to informative channels while reducing the emphasis on less relevant ones. This helps improve the discriminative power of the features and boost the performance of CNNs in various tasks, including image classification, object detection, and semantic segmentation.

The "Global-Local Temporal Representations For Video Person Re-Identification" paper (Li, Wang, Tian, Gao & Zhang 2019) is an important contribution to the field of temporal attention mechanisms in the context of video person re-identification. This paper addresses the challenge of capturing and leveraging temporal information in video sequences for the task of person re-identification, which involves identifying individuals across different video frames or sequences.

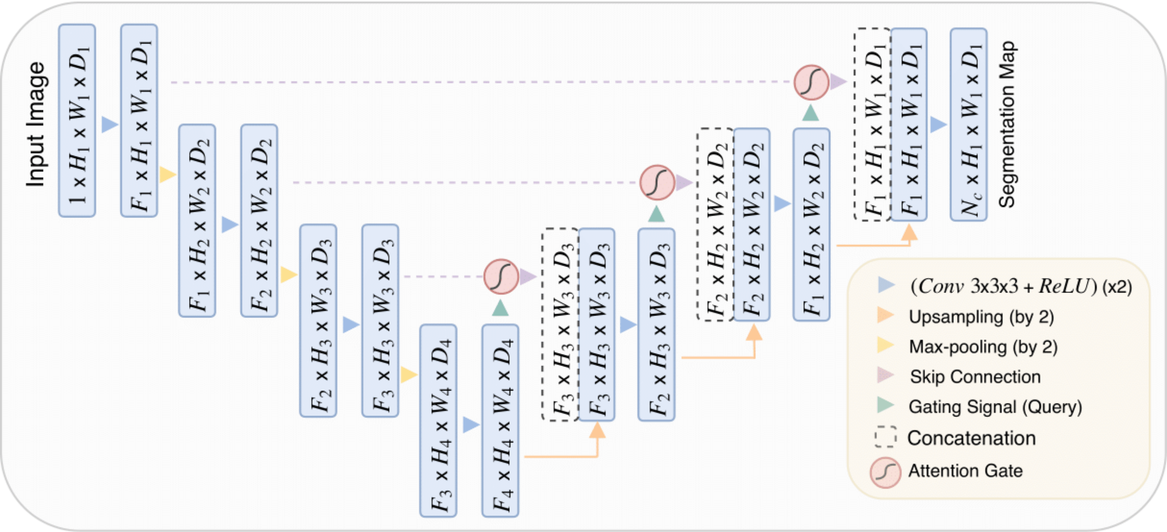

Input Image
$1 \times H_1 \times W_1 \times D_1$
$F_1 \times H_1 \times W_1 \times D_1$
$F_1 \times H_2 \times W_2 \times D_2$
$F_2 \times H_2 \times W_2 \times D_2$
$F_2 \times H_3 \times W_3 \times D_3$
$F_3 \times H_3 \times W_3 \times D_3$
$F_3 \times H_4 \times W_4 \times D_4$
$F_4 \times H_4 \times W_4 \times D_4$
$F_3 \times H_3 \times W_3 \times D_3$
$F_3 \times H_3 \times W_3 \times D_3$
$F_2 \times H_3 \times W_3 \times D_3$
$F_2 \times H_2 \times W_2 \times D_2$
$F_2 \times H_2 \times W_2 \times D_2$
$F_1 \times H_2 \times W_2 \times D_2$
$F_1 \times H_1 \times W_1 \times D_1$
$F_1 \times H_1 \times W_1 \times D_1$
$N_c \times H_1 \times W_1 \times D_1$
Segmentation Map
(*Conv* 3x3x3 + *ReLU*) (x2)
Upsampling (by 2)
Max-pooling (by 2)
Skip Connection
Gating Signal (Query)
Concatenation
Attention Gate

Figure 2.1: Attention gate in U-Net architecture

"Selective Kernel Networks" (Li, Wang, Hu & Yang 2019) is an influential paper that introduces a branch attention mechanism called selective kernel convolution. This paper addresses the challenge of capturing diverse spatial information at different scales by proposing a method that adaptively selects and combines kernels with varying receptive fields. The use of selective kernel convolution in SKNet has shown improved performance in various computer vision tasks, including image classification and object detection, by effectively integrating multi-scale spatial information.

2.3 Attention mechanism in Natural Language Processing tasks

In the field of natural language processing, before attention mechanisms, it relied on recurrent neural networks (Hopfield 1982) for sequence data. This approach has some limitations, such as the need to memorise a long sequence of data and the computational complexity. Then new architectures appeared, including Long short-term memory (Hochreiter & Schmidhuber 1997), Bidirectional recurrent neural networks(Schuster & Paliwal 1997), and Encoder-Decoder (Sutskever, Vinyals & Le 2014) which partly fix the memorization problem but the computing remains sequential.

The solution to these two problems is a new Encoder-Decoder structured model called Transformer (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser & Polosukhin 2017), introduced by Vaswani et al. Figure 2.2. The key innovation of the architecture is the use of attention mechanism in the encoder and decoder and the connections between them. This approach allows the model to selectively focus on different parts of the input sequence when generating each output token. The Transformer uses multi-head attention, which involves computing multiple attention scores in parallel for each input token and then concatenating the results Figure 2.3. This allows the model to attend to different aspects of the input sequence in a more fine-grained way.

Bidirectional Encoder Representations from Transformers (BERT) (Devlin, Chang, Lee & Toutanova 2018) and Generative Pre-trained Transformers (GPT) (Radford, Narasimhan, Salimans, Sutskever et al. 2018) are two notable applications of the Transformer architecture in the field of natural language processing (NLP). While both models utilise the Transformer architecture, their designs and training techniques differ significantly.

BERT primarily focuses on the encoder part of the Transformer Figure 2.4. It is trained using a masked language modelling (MLM) task, where a portion of the input text is randomly masked, and the model is trained to predict the original words based on the context provided by the surrounding words. This approach allows BERT to learn deep bidirectional representations, capturing contextual dependencies in both directions. BERT has achieved remarkable results in a wide range of NLP tasks, such as sentiment analysis, named entity recognition, and question answering.

On the other hand, GPT predominantly employs the decoder component of the Transformer Figure 2.5. GPT is trained using an autoregressive language modelling task, where the model is trained to generate the next word in a sequence given the preceding words. By training on a large corpus of text, GPT learns to generate coherent and contextually appropriate responses. GPT has demonstrated impressive performance in tasks like text generation, language translation, and dialogue systems.

2.4 Computer Vision tasks with Transformer

Convolutional neural networks (CNNs) have been the most widely used architecture after the success of AlexNet (Krizhevsky, Sutskever & Hinton 2017) in computer vision tasks for many

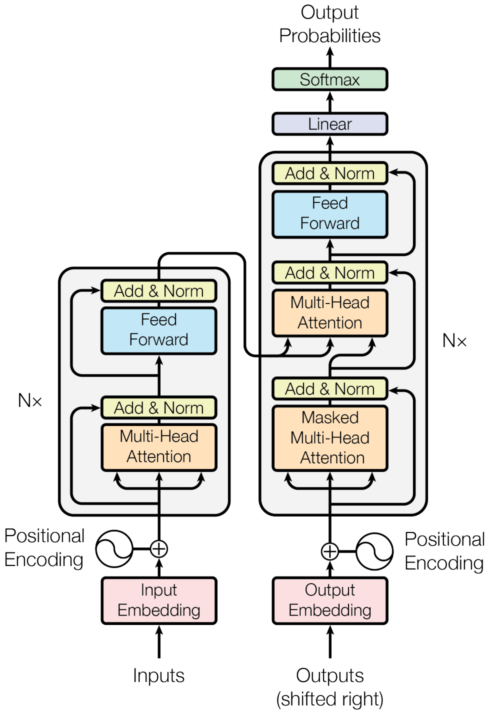


Figure 2.2: Transformer architecture

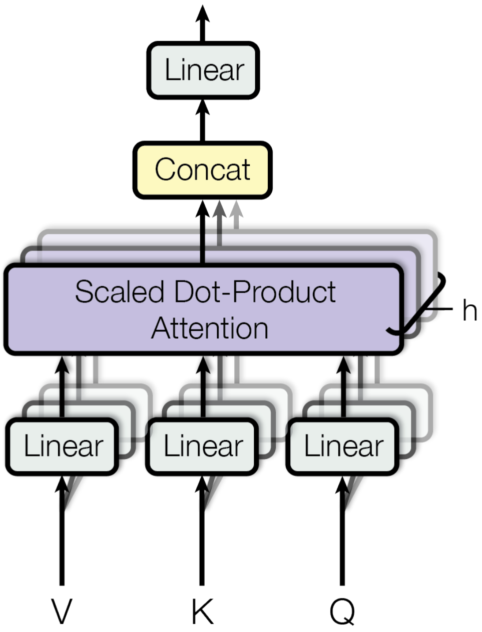


Figure 2.3: Multi-head attention

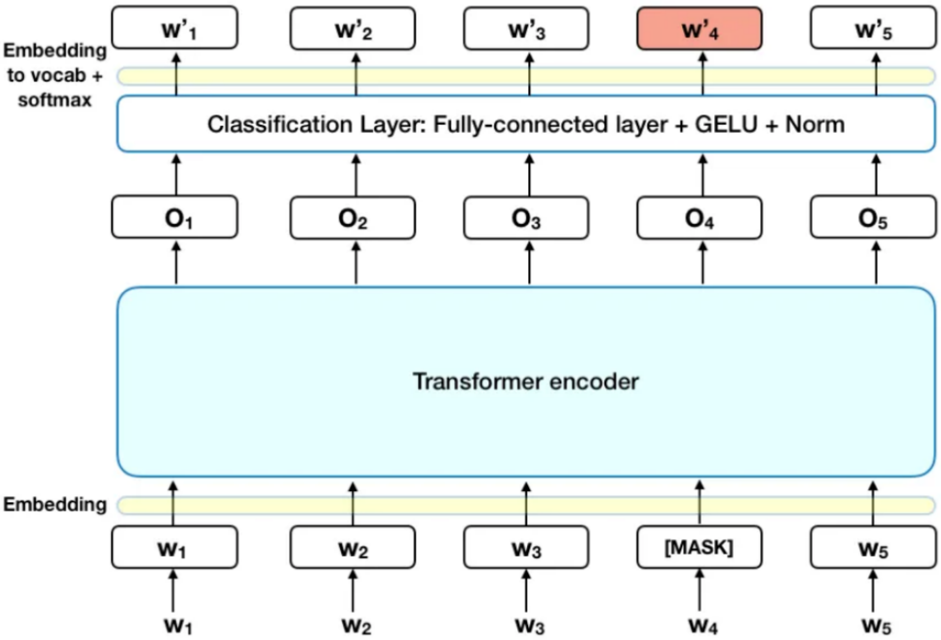


Figure 2.4: BERT architecture

years and have achieved great success in tasks such as image classification, object detection, and segmentation. However, even though the Transormer architecture was originally developed for natural language processing tasks, recent research has demonstrated that it can also be used effectively for computer vision tasks.

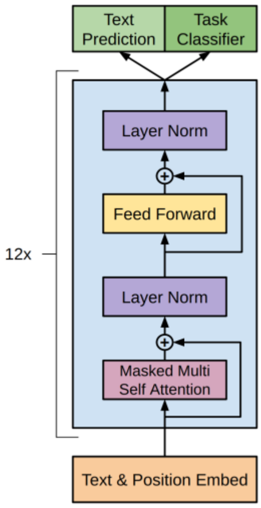


Figure 2.5: GPT architecture

End-to-End Object Detection with Transformers by Carion et al. (Carion, Massa, Synnaeve, Usunier, Kirillov & Zagoruyko 2020) provides us with an accurate and efficient architecture for object detection. Detection Transformer (DETR) architecture contains three main components: a CNN backbone to extract lower-resolution activation map from image data, encoder-decoder transformer and feed forward network (FFN) that makes the final detection prediction Figure 2.6. DETR achieves comparable results to an optimised Faster R-CNN and significantly better performance on large objects. Because of its flexible architecture, only by adding a mask head on top of the decoder outputs, DETR can do panoptic segmentation tasks with comprehensive results. In a 2021 paper titled "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", Dosovitskiy et al. (Dosovitskiy, Beyer, Kolesnikov, Weissenborn, Zhai, Unterthiner, Dehghani, Minderer, Heigold, Gelly et al. 2020) introduced a novel architecture for image recognition tasks that represents a departure from the traditional convolutional neural networks (CNNs) that have dominated before. Unlike previous approaches, where attention mechanisms were applied in conjunction with convolutional networks or certain components of convolutional networks were replaced while keeping their overall structure. Dosovitskiy's architecture, called the Vision Transformer (ViT), is based only on Transformer architecture. The ViT replaces convolutional layers with a purely self-attention mechanism that operates on patches of the input image, enabling the model to capture both local and global spatial information in a highly efficient and effective way Figure 2.7.

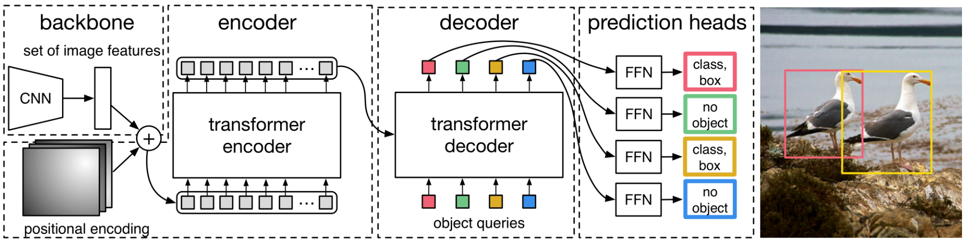


Figure 2.6: DETR architecture

By decoupling image recognition from traditional CNNs, the ViT has opened up new avenues of research in computer vision and demonstrated that self-attention mechanisms can be a highly promising alternative for a wide range of image recognition tasks. The ViT has achieved state-of-the-art results on several benchmarks. Liu et al. addressed ViT computational efficiency and scalability limitations and introduced a new hierarchical architecture called Shifted Windows Transformer (Swin Transformer) (Liu, Lin, Cao, Hu, Wei, Zhang, Lin & Guo 2021) Figure 2.8. By using the shifted window scheme, the architecture processing becomes more efficient on large images by limiting self-attention computation to non-overlapping local windows. The Swin Transformer outperforms the ViT on various image recognition benchmarks while using fewer computational resources. (Niu et al. 2021)

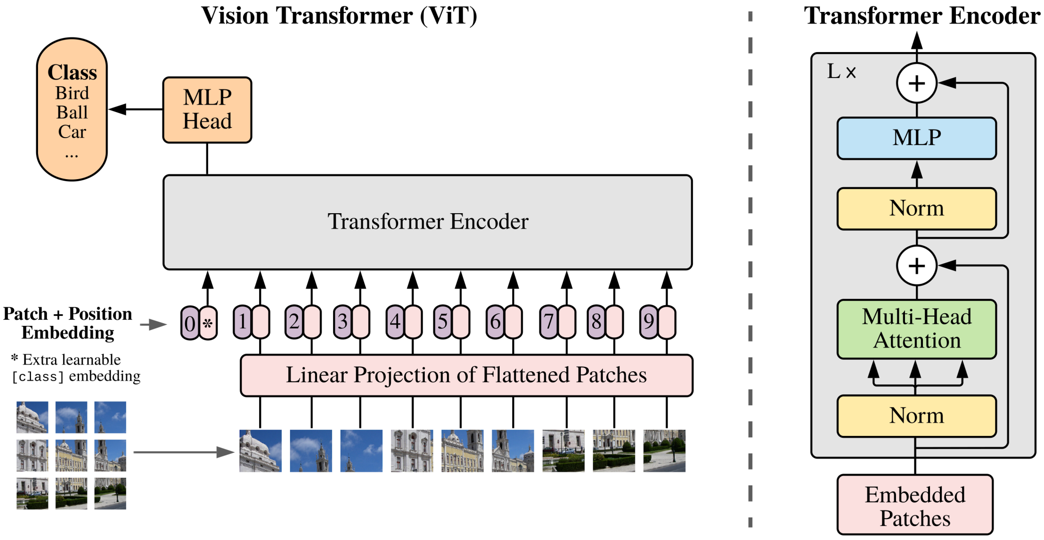


Figure 2.7: ViT architecture

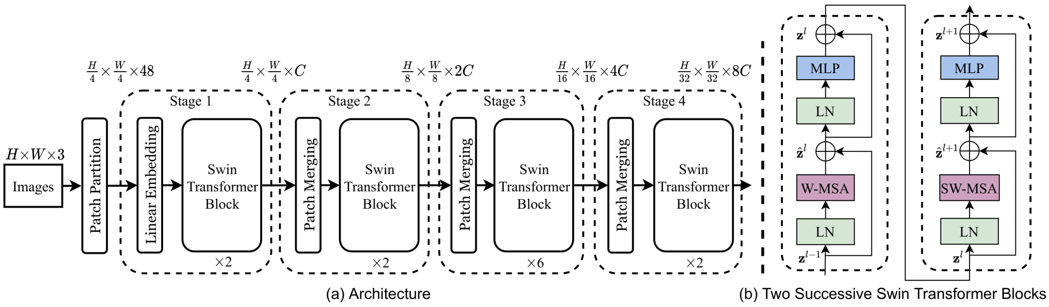


Figure 2.8: Swin Transformer architecture

Chapter 3

Attention Mechanism

3.1 Vanilla Attention Mechanism

The attention mechanism is used in deep learning to enable a neural network to selectively focus on certain parts of the input when processing it. Rather than treating the input equally, the idea of the attention technique is to allow the model to give different weights to different parts of the input, effectively highlighting the most relevant features or regions in the input. This is achieved by computing attention scores for various components of the input, depending on the specific task of the model.

The importance of attention mechanisms can be observed in their ability to handle long sequences, capture dependencies, and incorporate context information effectively. Traditional approaches such as RNN (Hopfield 1982), LSTM (Hochreiter & Schmidhuber 1997), and BRNN (Schuster & Paliwal 1997) often struggle with long-range dependencies or global context reasoning as they treat all parts of the input equally or rely on fixed-length windows. Attention mechanisms address these limitations by attending to specific elements in the input based on their importance, enabling better modelling of relationships and context.

3.2 Key Parameters

The key parameters that attention mechanism uses are three types of vectors, including keys (K), values (V), and queries (Q). Typically, the keys and values are generated by the encoder and are representations of the input data. On the other hand, queries are generated by the decoder and represent the current state of the model. Generally, the computation of the attention mechanism can be divided into two steps. The first one is measuring the similarity or compatibility between the query and the corresponding key of each element in the input. That is done with the score function, and after obtaining the scores, a normalisation is performed to ensure that the attention weights form a valid probability distribution. Commonly used for norm normalisation is softmax. All scores have to be between 0 and 1 and sum up to 1. The second step is the generation of context vectors. In this step, the normalised attention weights and values are computed by such a function that returns a single vector given the set of values and their corresponding weights, typically a weighted sum of V.

3.3 Types of Attention Mechanisms

When applying the attention mechanism, the choice of the right score function to compute the correlation between keys, values, and queries becomes crucial, as it heavily depends on the specific task and type of model. Effectively capturing the relationships between these components is paramount for the success of the model. There are various scoring functions that enable this computation, each with its own benefits.

In this section, attention mechanisms are categorized into two distinct groups based on their characteristics. The first category revolves around the score function, which determines the relationship between keys, values, and queries within the mechanism. The second category, on the other hand, focuses on the input representation used by the attention mechanism. By dividing the types of attention mechanisms into these two categories, we can gain a deeper understanding of their different approaches and functionalities.

3.3.1 Types Based on the Score Function

3.3.1.1 Additive Attention

Additive attention, introduced by Bahdanau et al. (Bahdanau et al. 2014), is the first type of attention mechanism. It uses a feedforward neural network to model the alignment between queries and keys. By applying a shared feedforward layer and tanh or another non-linear activation function, such as ReLU, it captures complex relationships and determines attention scores.

$$Additive\,attention(Q, K, V) = softmax(\tanh(Q + K + b)V) \tag{3.1}$$

Where W_Q and W_K represent the learnable parameters associated with the feedforward neural network that is used to compute the attention scores.

3.3.1.2 Multiplicative (dot-product) Attention

Multiplicative attention (Britz, Goldie, Luong & Le 2017) directly computes the attention scores by taking the dot product between the queries and keys. Although this approach is computationally less expensive, it is slightly outperformed by additive attention.

$$Multiplicative\,attention(Q, K, V) = softmax(Q^T K)V \tag{3.2}$$

Where Q, K and V represent the queries, keys and values packed together into matrices.

3.3.1.3 Scaled Multiplicative (dot-product) Attention

Addressing the multiplicative attention issue of large dot product values that can lead to unstable gradients during training, Vaswani et al. (Vaswani et al. 2017) introduce a new type of attention called the scaled dot-product for their transformer architecture. They scale the dot product by the square root of the dimension of the query or key vectors.

$$Scaled\,multiplicative\,attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3.3}$$

3.3.1.4 General Attention

General attention (Luong, Pham & Manning 2015) is similar to multiplicative attention but with new parameter W which represents a learnable weight matrix.

$$General\,attention(Q, K, V) = softmax(Q^T W K)V \tag{3.4}$$

3.3.1.5 Similarity Attention

Similarity attention (Graves, Wayne & Danihelka 2014) compares the cosine similarity between the K and Q.

$$Similarity\,attention(Q, K, V) = softmax(Cosine\,similarity(K, Q))V \tag{3.5}$$

3.3.2 Types Based on the Input Representation

3.3.2.1 Self-Attention

Self-Attention (Vaswani et al. 2017) uses the keys, values, and queries from the same input sequence. It allows the model to attend to different positions within the same sequence when processing each element. The key advantage of this technique is its ability to capture long-range dependencies within a sequence, as each position can attend to any other position. This makes it particularly effective in tasks such as machine translation, where the model needs to consider dependencies between words that are far apart in the sentence.

3.3.2.2 Co-Attention

Co-attention (Lu, Yang, Batra & Parikh 2016) uses the keys, values, and queries from more than one input sequence. Unlike self-attention, co-attention aims to model the interactions and dependencies between multiple sequences. The key advantage of this technique is its ability to understand and align information from different sources.

3.3.2.3 Hierarchical Attention

Hierarchical attention (Yang, Yang, Dyer, He, Smola & Hovy 2016) captures the hierarchical structure of input data. It combines lower-level and higher-level attention to focus on the important information at each level of the hierarchy while considering the interactions between the levels. It enables the model to effectively capture the context and dependencies within and between the hierarchical components, leading to improved performance in tasks that involve hierarchical or nested data structures.

3.4 Softness of Attention

The softness of attention refers to the principle which the attention mechanism assigns weights to different parts of the input sequence. There are two types of softness:

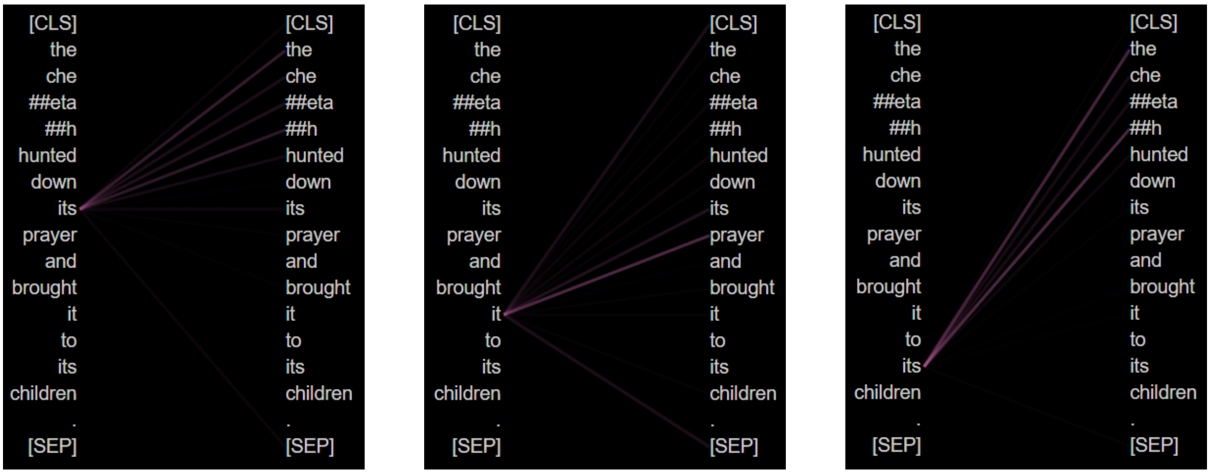


Figure 3.1: Example of isolated attentions from different words for attention layer 6 head 7.

- **soft** (Bahdanau et al. 2014) is when the normalised attention weights are between 0 and 1. This allows the attention mechanism to focus on different parts of the input sequence to different degrees.
- **hard** (Xu, Ba, Kiros, Cho, Courville, Salakhudinov, Zemel & Bengio 2015) is when the normalised attention weights are 0 or 1. This means that the attention mechanism can only focus on a single part of the input sequence at a time.

Both types of softness exhibit their own advantages and disadvantages, and their suitability depends on the specific task. Soft attention offers the advantages of flexibility and adaptability. It allows the model to smoothly distribute attention across input elements, capturing their relevance and contributions. However, it incurs additional computational complexity due to the continuous nature of attention weights.

In contrast, hard attention, which assigns attention to a single element, provides a deterministic focus. This can be advantageous when explicit and precise decisions are required, allowing the model to selectively attend to the most important input element. Hard attention reduces computational complexity as attention is allocated to a single element at each step. However, the discrete nature of hard attention limits the model's ability to capture information from multiple elements simultaneously.

3.5 Visualization of the Attention Mechanism

The visualization of attention mechanisms holds a crucial role in understanding whether a model is effectively focusing on the relevant elements. It serves as a powerful tool to gain insights into the model's decision-making process and ensure that its attention is directed towards the right aspects of the input. The visualization in Figure 3.1 illustrates the model's comprehension of the sentence *"The cheetah hunted down its prayer and brought it to its children."* The visualisation showcases the attention weights assigned by the model to different words in the sentence, highlighting the important words that the model focuses on for its understanding and interpretation. By analysing the attention distribution, we gain insights into the model's prioritisation of certain words and their contributions to the overall comprehension of the sentence.

Chapter 4

Attention Mechanism in Computer Vision

4.1 Categories of Attention Mechanism

The large variety of data in the field of computer vision, including images and videos, brings its own distinct characteristics and challenges. Furthermore, the diverse range of tasks in this domain adds complexity and variability. To address this diversity-specific challenge, different categories of attention mechanisms have been developed in the field. These categories provide specialised techniques for capturing and leveraging specific dependencies and patterns within the data. By employing these distinct categories of attention mechanisms, computer vision models are empowered to effectively focus on relevant information, extract meaningful features, and achieve enhanced performance across a wide range of tasks and data types.

4.1.1 Channel Attention

Channel attention focuses on capturing dependencies and relationships between channels or feature maps within a CNN. It aims to enhance the representation and discriminative power of feature maps by using techniques such as global pooling, convolutional operations, or self-attention mechanisms to assign weights to different channels. By learning to emphasise informative channels while suppressing less relevant ones, channel attention mechanisms enable models to effectively capture and utilise channel-wise information during image processing tasks. This technique has been shown to improve the performance of computer vision models in various tasks such as image classification, object detection, and semantic segmentation.

An example of channel-wise attention is the squeeze-and-excitation (SE) block introduced by Hu et al. (Hu et al. 2017), which is commonly used in different variations of computer vision tasks Figure 4.1. The SE block aims to capture channel dependencies by adaptively recalibrating the importance of each channel within a feature map. It archives that into two phases: squeeze and excitation. The squeeze phase consists of performing global average pooling to obtain channel-wise statistics, and the excitation phase consists of using a small, fully connected network to generate channel-wise attention weights. By applying the SE block, models can effectively emphasise informative channels while suppressing less relevant ones, leading to improved feature representation and discriminative power. The authors demonstrate that the SC block enhances and is suitable for various CNN baselines with various training settings and tasks while being cost-efficient.

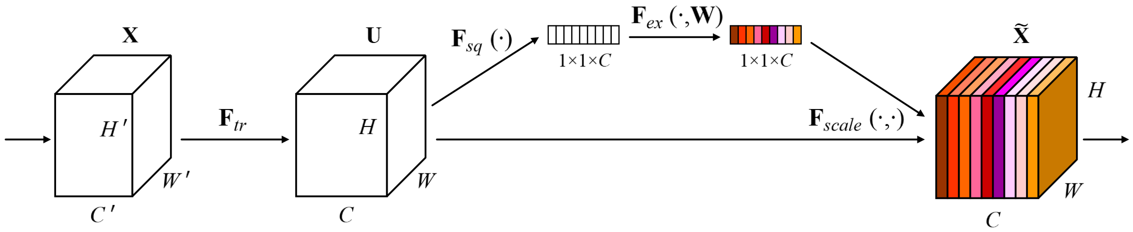


Figure 4.1: Squeeze-and-Excitation Block

4.1.2 Spatial attention

Spatial attention operates at the level of spatial dimensions within a feature map. It aims to highlight important spatial regions while downplaying irrelevant or background regions. Spatial attention mechanisms typically utilise techniques such as convolutional operations or self-attention mechanisms to assign importance weights to different spatial locations. By learning to attend to informative regions and suppress noise or background regions, spatial attention enables models to effectively focus on relevant image regions, leading to improved performance in tasks such as object localization, image segmentation, and visual reasoning. Spatial attention has been widely used in various computer vision models and has shown significant improvements in capturing spatial context and improving the accuracy of spatially related tasks.

Oktay et al. (Oktay et al. 2018) introduced a spatial attention mechanism called the Attention Gate (AG) to address the issue of excessive and wasteful use of computational resources and model parameters in specific regions of interest Figure 4.2. The AG approach aims to selectively allocate computational resources to relevant regions while suppressing unnecessary computations in less important areas. This is achieved by a convolutional or pooling operation followed by a non-linear activation function. This operation helps capture the spatial dependencies and patterns within the input features. The resulting attention weights are then applied to the input features, allowing the model to focus on the most relevant regions while suppressing less informative areas. By utilising the AG mechanism, models can focus their attention on specific spatial regions, improving computational efficiency and reducing redundant computations. This spatial attention mechanism has proven to be effective in various computer vision tasks, allowing models to allocate resources efficiently and achieve better performance with reduced computational costs.

DERT (Carion et al. 2020), ViT (Dosovitskiy et al. 2020), and Swin Transformer (Liu et al. 2021), which are discussed later in this paper, are categorised as spatial attention.

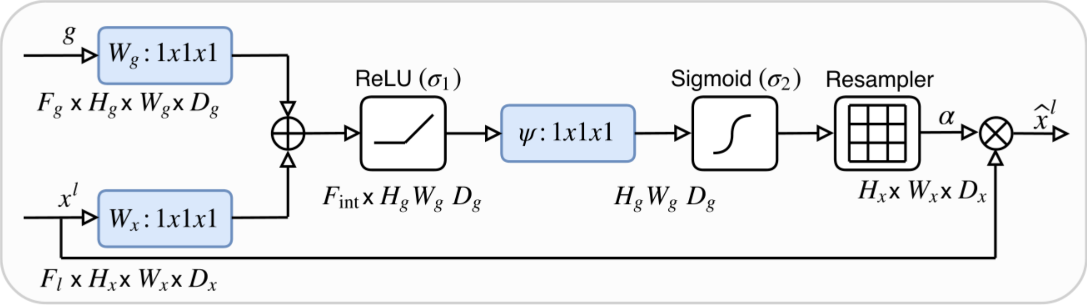


Figure 4.2: Attention Gate

4.1.3 Temporal attention

Temporal attention is an important component in tasks involving sequential or time-dependent data, such as video analysis or action recognition. It allows the model to effectively capture temporal dependencies and patterns by assigning varying weights to different time steps or frames in the sequence. By attending to relevant temporal segments or frames, the model can better capture long-range dependencies and temporal dynamics, leading to improved performance in tasks that rely on temporal understanding. Temporal attention mechanisms play a crucial role in modelling the temporal aspect of sequential data, enabling the model to extract meaningful information and make accurate predictions.

An example of a temporal attention mechanism is global-local temporal representation (GLTR) Figure 4.3, introduced by Li et al. (Li, Wang, Tian, Gao & Zhang 2019) GLTR incorporates two key components for capturing multi-scale temporal cues in video sequences: The first component is the dilated temporal pyramid (DTP), which focuses on local temporal context learning. The DTP employs a dilated convolutional architecture that captures information at different temporal scales by using varying dilation rates. This allows the model to capture both short-term and long-term dependencies within the video sequence. The second component of GLTR is the temporal self-attention (TSA) module, which is responsible for capturing global temporal interaction. This module leverages self-attention mechanisms to model the relationships between different temporal segments across the entire video sequence. By attending to relevant temporal segments and capturing their interactions, the model can understand the overall temporal dynamics and long-range dependencies present in the video. By combining the local temporal context learning from the DTP and the global temporal interaction captured by the temporal self-attention module, GLTR achieves a comprehensive representation of the multi-scale temporal cues in the video sequence. This enables the model to effectively understand and analyse the temporal dynamics of the video, leading to improved performance in tasks that require temporal understanding and analysis.

4.1.4 Branch attention

Branch attention refers to attention mechanisms that aim to capture dependencies and interactions between different branches or paths in multi-branch architectures. The purpose of branch attention is to allow the model to selectively attend to and exchange information between different branches based on their relative importance or relevance to the task. This enables the model to dynamically allocate resources and focus on the most informative branches while suppressing less relevant ones. This is achieved by involving gate mechanisms that control the flow

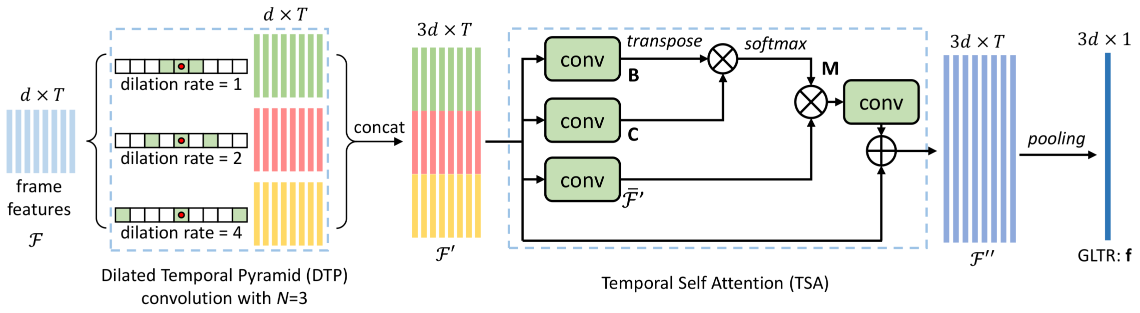


Figure 4.3: GLTR Architecture

of information between branches based on learned weights or activation values. By incorporating branch attention, models can benefit from the collaborative processing and information exchange between different branches, enhancing their overall understanding and representation of the input data in tasks such as object detection, semantic segmentation, or image classification.

In Selective Kernel Networks (SKNet) (Li, Wang, Hu & Yang 2019), the branch attention mechanism is applied in the selective kernel (SK) convolution Figure 4.4, which was introduced by Li et al. The SK convolution consists of three main operations: split, fuse, and select. During the split operation, the input feature map is processed by applying transformations with different kernel sizes. This allows the network to capture information at various receptive field sizes, enabling it to handle objects of different scales. Next, the information from all branches is fused together via element-wise summation in the fuse operation. This fusion step aggregates the feature maps obtained from the different branches, combining their representations. The gate vector is then computed using the fused feature maps. The gate vector acts as a control mechanism for information flow between the multiple branches. It assigns weights or importance to each branch, allowing the network to selectively emphasise or suppress the contributions of different branches based on their relative importance. Finally, the output feature map is obtained by aggregating the feature maps from all branches, guided by the gate vector. This aggregation process ensures that the network effectively combines the information captured at different scales, resulting in a more comprehensive and informative representation.

The branch attention mechanism in the SK convolution of SKNet allows the network to adaptively adjust the receptive field sizes and selectively utilise the information from different branches. This enables SKNet to capture both local and global contextual information effectively, enhancing its ability to handle objects of varying scales and complexities.

4.2 Benefits of Attention Mechanisms in CV

The concept of directing models to focus on specific parts of the input has made attention mechanisms a widely utilised technique in the field of computer vision, offering numerous remarkable benefits.

4.2.1 Enhanced Performance

Attention mechanisms allow models to focus on relevant regions or features in an image, leading to improved performance in tasks such as image classification, object detection, and segmen-

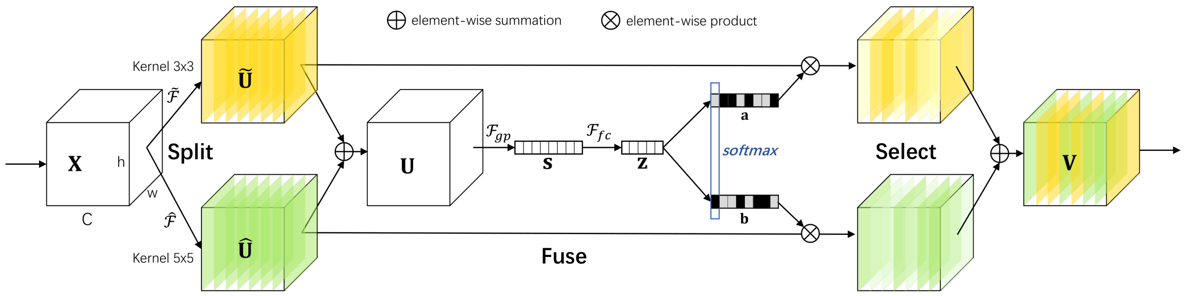


Figure 4.4: Selective Kernel Convolution.

tation. By attending to important information and suppressing irrelevant regions, attention mechanisms help extract more discriminative features and improve the model's ability to understand and interpret visual data.

4.2.2 Better Interpretability

The visualisation of the attention mechanism provides insights into the decision-making process of the model by highlighting the regions that the model attends to. This helps in understanding which parts of the image contribute most to the model's predictions, making the model more interpretable and providing valuable insights for analysis and improvement.

4.2.3 Robustness with Noisy Data

Attention mechanisms provide models with the ability to effectively handle noise and other difficult conditions commonly encountered in images. By selectively attending to salient regions and suppressing noisy areas, attention mechanisms enhance the model's robustness and enable more accurate predictions, even in challenging visual scenarios.

4.2.4 Attention Mechanism Adaptability

Attention mechanisms exhibit a high degree of adaptability and can be seamlessly integrated into various network architectures across different types of data and tasks in computer vision. This adaptability empowers models to effectively capture and utilise pertinent information across a wide range of datasets and tasks, enhancing their performance and flexibility.

4.3 Challenges of Attention Mechanisms in CV

Even though attention mechanisms in the field of computer vision offer numerous benefits, they also encounter specific challenges. One of the primary challenges is the computational complexity associated with attention mechanisms. The need for complex computations to capture and utilise attention can pose difficulties in processing large-scale or high-resolution image and video data. The computational demands may limit real-time performance or strain the resources of computational devices. Additionally, attention mechanisms can present challenges related to memory consumption. Particularly, self-attention mechanisms that require storing pairwise attention scores for all input can result in memory-constrained devices or an inefficient use of the memory.

Chapter 5

Attention Mechanism in Natural Language Processing

5.1 Early Attention in NLP

In the field of natural language processing, attention mechanisms play a really important role. The first use of attention mechanisms for machine translation tasks was introduced by Bahdanau et al. (Bahdanau et al. 2014) Their work, known as *RRNsearch*, utilised a bidirectional recurrent neural network for both the encoder and decoder. The introduction of attention in this architecture proved to be highly inspiring for researchers to adopt attention mechanisms in various neural network models.

Since then, attention mechanisms have gained widespread popularity and have become a fundamental building block in many NLP tasks. They have been successfully applied to tasks such as machine translation, sentiment analysis, question answering, and text summarization. However, challenges such as computational complexity and the ability to effectively handle long sequences of data remained unsolved.

5.2 Transformer - A Groundbreaking NLP Architecture

The Transformer architecture Figure 2.2, introduced by Vaswani et al. (Vaswani et al. 2017), has revolutionized the field of NLP. It represents a shift away from traditional recurrent models like RNNs and LSTMs, offering several advantages in terms of performance and efficiency.

The Transformer architecture is a purely attention-based approach, and it is specifically designed for tasks such as machine translation, language modeling, and text generation. It relies heavily on the attention mechanism to capture dependencies between input and output sequences. Unlike traditional sequence models that process inputs sequentially, the Transformer allows for parallel processing, resulting in faster training and inference times.

5.2.1 Attention Mechanism in the Transformer

The Transformer uses a multi-head self-attention mechanism Figure 2.3 with a scaled dot-product for the score function in both the encoder and decoder, while the decoder uses a masked multi-head self-attention mechanism.

The reason why the Transoformer uses self-attention is because of its lower total computational complexity, the number of computations that can be parallelized, and the fact that it is the best way to capture the long-range dependencies within sequential data. The self-attention mechanism calculates attention weights for each word by comparing it to all the other words in the sequence. These attention weights determine the importance of each word to the other words in the sequence.

The idea behind multi-head attention is to get more accurate attention weights for each word. If it is performed with a single attention, it leads to an averaging effect that limits the resolution of the learned representations. For example, for every word, the attention vector may weight its relation with itself much higher, and that is not going to be useful for the model. The solution is to use multi-head attention to determine more attention vectors per word and then take a weighted average to compute the final attention vector. That way, the relation with each other word will be higher.

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O \tag{5.1}$$

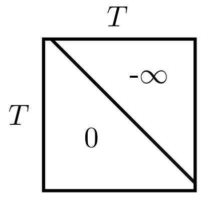
Where W^O is a weight matrix that is used to linearly project the learned representations back to the original embedding dimensionality.

$$head_i = \mathit{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{5.2}$$

Where W_i^Q, W_i^K and W_i^V are learned projection matrices, i indexes over the heads and $W_i^Q \in \mathbb{R}^{D \times d_k}$, $W_i^K \in \mathbb{R}^{D \times d_k}$ and $W_i^V \in \mathbb{R}^{D \times d_v}$. The value vector dimensionality is set to be equal to the embedding dimensionality divided by the number of heads $d_k = d_v = \frac{D}{h}$ and $W^O \in \mathbb{R}^{D \times D}$.

To maintain the temporal dependency of the output sentence in the decoder, a crucial component is the masked multi-head self-attention mechanism. During deployment, the output sentence is generated sequentially, one word at a time. It is essential that the prediction at each position only depend on the previously sampled words. This requirement is achieved by incorporating a mask during self-attention in the decoder, which enforces the restriction while enabling parallel computation during training. In each masked attention layer, when updating the representation at a specific position, it is essential to completely ignore any information from subsequent sequence positions.

$$mask = \tag{5.3}$$



$$\mathit{Attention}(Q, K, V) = softmax(mask + \frac{QK^T}{\sqrt{d_k}})V$$

Where the mask is matrix with negative infinities above the main diagonal and zeros everywhere else. The negative infinities values will become zeros after the softmax function.

5.2.2 Other Key Techniques in the Transformer

- **Position-wise Feed-Forward Network block** is a simple network of two fully connected layers with ReLU activation between them to each word representation. It enables the model to capture complex, non-linear relationships within the input sequence.
- **Residual connections and Normalization** are used to add a copy of the word representations before a multi-head attention block to the word representations after the multi-head attention block. It is used to solve the vanishing gradient problem and improve the flow of gradients during training. Then is used softmax to normalize the vector.
- **Positional Encoding block** is needed because of the lack of any inherent sequential information, such as the recurrence in recurrent neural networks. Positional encodings are used in the input embeddings at the bottoms of the encoder and decoder stacks.

$$\begin{aligned} PE_{(pos,2i)} &= \sin(pos/10000^{2i/d_{model}}) \\ PE_{(pos,2i+1)} &= \cos(pos/10000^{2i/d_{model}}) \end{aligned} \tag{5.4}$$

  Where *pos* is the position and i is the dimension.

5.3 Advantages of the Transformer over the Traditional NLP Models

The efficient architecture of the Transformer has revolutionized the field of NLP, offering significantly lower training costs over traditional models such as RNNs (Hopfield 1982), LSTM (Hochreiter & Schmidhuber 1997), and BRNNs (Schuster & Paliwal 1997). By addressing the limitations of these models, the Transformer has emerged as a game-changing solution in the NLP domain.

The Transformer is designed to capture long-range dependencies in sequences more effectively compared to traditional models. The architecture can attend to any position within the input sequence, allowing it to model relationships between distant words more accurately. This is particularly beneficial for tasks that require understanding contextual information across the entire sequence, such as machine translation or document classification.

Another key advantage of the Transformer is its ability to process the input sequence in parallel. As a result, computations across different positions can be performed independently, enabling faster training and inference times, especially when dealing with longer sequences. Vaswani et al. showcased that by demonstrating that their model achieved better accuracy compared to the previous state-of-the-art model while requiring less than a quarter of the training time.

Traditional models like RNNs (Hopfield 1982) suffer from sequential processing, leading to higher computational complexity, especially for long sequences. In contrast, the Transformer's self-attention mechanism allows for parallel computation across positions, reducing the overall computational complexity. This makes the Transformer more efficient and scalable for handling large-scale NLP tasks.

Furthermore, because of the self-attention mechanism, the Transformer is able to capture global context effectively. By attending to all positions in the input sequence, the Transformer can incorporate information from both nearby and distant words, capturing a more comprehensive understanding of the input. This is particularly advantageous for tasks that require a holistic view of the input, such as language modelling or sentiment analysis.

5.4 Applications of the Transformer in NLP

The Transformer architecture has many applications in the NLP field, such as machine translation, language generation, question answering, sentiment analysis, text classification, and many more. But two of the revolutionary developments that the Transformer led to are the powerful pre-trained language models Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. 2018) and Generative Pre-trained Transformer (GPT) (Radford et al. 2018), which have achieved state-of-the-art results in various NLP tasks through transfer learning.

The BERT architecture uses only the Transformer encoder. It is trained on two unsupervised tasks simultaneously: masked language modelling (MLM) and next sentence prediction (NSP). The objective is to minimise the combined loss function of these two strategies. In the MLM strategy, approximately 15% of the words in each sequence are replaced with a [MASK] token before being fed into the model. Then the model attempts to predict the original value of the masked words based on the contextual information provided by the other, non-masked, words in the sequence. The NSP strategy, on the other hand, involves taking pairs of sentences and determining whether they are related or not, treating it as a binary classification problem. By incorporating both strategies, BERT gains an understanding of the context across different sentences. After pre-training, the BERT model can be fine-tuned for specific NLP tasks.

In contrast with BERT, GPT uses only the Transformer decoder and focuses solely on the language modelling task. During training, GPT receives an input sequence of tokens and is trained to predict the next token in the sequence based on the preceding context. The GPT model is trained using unsupervised learning on a large corpus of text data. By learning from the vast amount of data, GPT develops an understanding of language patterns, semantics, and context. This pre-training phase enables the model to capture meaningful representations of words and sentences. Following pre-training, GPT can be fine-tuned for specific downstream tasks. This involves adding a task-specific output layer to the pre-trained model and training it using supervised learning. By fine-tuning, GPT can be adapted to various NLP tasks such as text classification, text generation, or question answering.

5.5 Challenges of Attention Mechanism in NLP

Although attention mechanisms have revolutionised NLP by enabling models to focus on relevant information, they also present several challenges that need to be addressed. These challenges arise due to the scalability and memory requirements of attention, as well as the need for interpretability and generalisation.

5.5.1 Scalability

The scalability is primary challenge of the attention mechanisms in NLP. Scalability is a primary challenge for the attention mechanisms in NLP. Traditional attention mechanisms typically require computing attention weights for each element in the input sequence, resulting in quadratic complexity with respect to the sequence length. This can make it computationally expensive to apply attention to long sequences, limiting the applicability of attention-based models to tasks involving lengthy inputs.

5.5.2 High Memory Consumption

Another primary challenge is high memory consumption, which can be caused by storing the attention weights for all input elements, especially when dealing with large-scale models or long sequences. This can pose challenges in terms of memory availability and efficiency, particularly on resource-constrained devices or when working with limited computational resources.

5.5.3 Interpretation of the Attention Weights

While attention mechanisms provide valuable insights into the model's decision-making process, interpreting attention weights can be challenging. Understanding how the model assigns importance to different input elements requires careful analysis and visualisation techniques. Interpreting attention patterns becomes more complex as models grow larger and more complex, making it harder to extract meaningful information from the attention weights.

5.5.4 Generalization

Attention mechanisms may face challenges in generalising well to diverse input data. Depending on the model architecture and attention formulation, attention weights can be sensitive to small changes in input data, leading to potential issues of overfitting or limited generalisation performance. Finding effective attention strategies that generalise well across different datasets and domains remains an unsolved problem.

Chapter 6

Computer Vision Tasks with Transformer

The attention-based Transformer architecture, initially developed for natural language processing tasks, has gained significant attention and success. However, many researchers have seen the Transformer architecture as more like a general-purpose differentiable computer that is expressive, efficient to train, and optimizable than architectures for NLP tasks. As a result, the Transformer architecture is highly adaptable and can be effectively used in various domains, including computer vision. The adaptations in applying the Transformer to computer vision tasks mainly involve transforming the input data to a suitable format for the Transformer and making adjustments to the layer norms.

6.1 Hybrid Architectures for CV Tasks

One of the first adaptations of Transformer for the object detection task is the Detection Transformer (DETR) (Carion et al. 2020), introduced by Carion et al. The DETR is hybrid architecture which means to process image data with the Transformer, the DETR model incorporates a convolutional neural network (CNN) backbone. The role of the CNN backbone is to extract features from the input image and transform it into a lower-resolution activation map. This lower-resolution representation is combined with a positional encoding and then passed as an input to the subsequent Transformer encoder, allowing the model to capture spatial dependencies. Then the output of the encoder is combined with the decoder's first attention block output and passed to other blocks of the decoder. The decoder's first attention block output is feature-encoded N random vectors representing the N class-label bounding boxes. The output embeddings of the decoder are passed to a feed-forward network that predicts the class and bounding box of the detection Figure 6.1.

The DETR model can be easily adopted for panoptic segmentation tasks. That can be achieved by adding only a mask head on top of the decoder outputs. The use of it is to predict a binary mask for each of the predicted boxes.

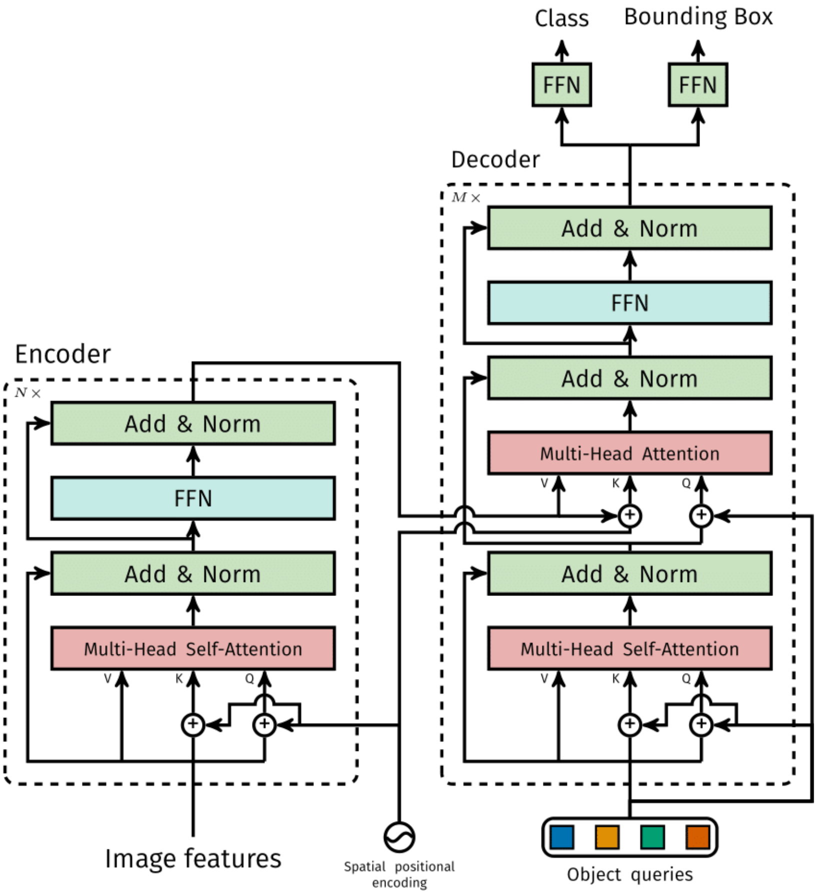


Figure 6.1: DETR Transformer architecture

6.2 Pure Transformer for CV Tasks

After the introduction of the DETR architecture, numerous hybrid architectures combining Transformers with other components emerged for various computer vision tasks. However, Dosovitskiy et al. (Dosovitskiy et al. 2020) proposed a novel architecture for image recognition tasks that deviates from conventional hybrid approaches. Their architecture, known as the Vision Transformer (ViT) Figure 2.7, relies solely on the Transformer model for image processing, eliminating the need for additional components such as CNNs. This departure from traditional hybrid architectures highlights the versatility and adaptability of the Transformer architecture for handling computer vision tasks effectively.

To maintain consistency with the original Transformer architecture, the writers divide the input image into a sequence of patches. These patches are extracted from the image and then flattened to create a sequence of tokens. Similar to the [CLS] token used in BERT (Devlin et al. 2018), an additional [class] token is embedded into the token representations. This [class] token serves as a learnable representation for the entire image and captures global information. By incorporating this patch-based tokenization scheme, the ViT model ensures that the image information is represented in a format compatible with the Transformer architecture, allowing for efficient processing and leveraging the power of self-attention mechanisms.

The ViT architecture has demonstrated impressive performance in image classification tasks. Even though the CNN-based model ResNet152x2 may outperform ViT when trained on smaller pre-training datasets, ViT demonstrates promising results when trained with larger datasets,achieving nearly 5% better accuracy. This highlights one of the advantages of the Transformer architecture: its superior optimizability. With increased data and computational resources, ViT has the potential to capture complex patterns and dependencies in the image data, leading to enhanced performance. This flexibility and scalability of the Transformer architecture make it a promising choice for computer vision tasks, particularly in scenarios where abundant training data is available.

A recent advancement in the field of computer vision with Transformer, called Swin Transformer, introduced by Liu et al. (Liu et al. 2021), addresses the limitations of the ViT in processing high-resolution images. They introduce a hierarchical design Figure 2.8 that divides the input image into smaller, non-overlapping patches. Unlike the ViT, the Swin Transformer incorporates a hierarchical transformer architecture that enables it to capture both local and global dependencies within the image.

The key innovation of the Swin Transformer lies in its use of shifting operations, which allow for efficient communication across patches and enable the model to handle large image resolutions effectively. By leveraging shifted windows at different scales, the Swin Transformer achieves a balance between capturing fine-grained details and capturing global context information.

The Swin Transformer has shown impressive performance on various computer vision tasks, including image classification, object detection, and semantic segmentation. It has demonstrated competitive results with state-of-the-art models while being more efficient in terms of memory consumption and computation.

The Swin Transformer exemplifies the strong interest among researchers in harnessing the Transformer architecture for computer vision applications. Its innovative design and impressive performance highlight the potential of utilizing the Transformer paradigm in this domain. By pushing the boundaries of what is possible in computer vision tasks, the Swin Transformer showcases the growing enthusiasm for exploring new models that leverage the power of the Transformer architecture.

6.3 Strengths of the Transformer-based models in CV

The adaptability of the Transformer architecture in the computer vision domain is indeed one of its significant strengths. The ability to apply the Transformer architecture to diverse tasks, such as image classification, object detection, and semantic segmentation, highlights its versatility and effectiveness in handling different visual recognition problems. This adaptability allows researchers and practitioners to leverage the power of Transformers across various computer vision applications, promoting flexibility and scalability in model design.

Another key strength of the Transformer architecture is its optimisation. The ViT (Vision Transformer) paper demonstrated that with large-scale datasets, the Transformer model can achieve remarkable performance, surpassing other models. The Transformer's ability to effectively utilise large amounts of data during pre-training contributes to its superior optimisation capabilities. As datasets continue to grow in size and more powerful hardware becomes available, the Transformer architecture becomes increasingly advantageous in exploiting the abundance of data and extracting meaningful representations.

6.4 Challenges of the Transformer-based models in CV

While Transformer-based models in computer vision demonstrate several strengths and remarkable results, as discussed in the previous sections, they also encounter certain challenges. These challenges need to be addressed for the widespread adoption and further improvement of these models in the field of computer vision.

6.4.1 High Computational Cost

One significant challenge is the increased computational cost compared to traditional CNN-based models. The reason for that is that these models involve complex attention mechanisms and parallel processing across a large number of tokens or patches, leading to increased computational requirements. As the size of the model and the input resolution increase, the computational cost grows exponentially. This can limit the scalability and real-time performance of Transformer-based models, especially for applications that require processing large volumes of high-resolution images. To train Transformers effectively, powerful hardware resources, such as high-end GPUs or TPUs, are often required. The need for substantial computational resources can limit the accessibility of Transformer-based models for researchers and practitioners with limited access to such hardware.

6.4.2 Large Datasets for Pre-training

While Transformers are known for their optimizability compared to CNN models, they do require significantly large datasets for effective pre-training. This is because CNNs possess inherent features like translation invariance, weight sharing, and partial scale invariance, which enable them to achieve impressive performance even with limited data. However, Transformers need to process extensive datasets to capture image-specific concepts and learn intricate relationships. Therefore, the need for larger datasets arises when training Transformers to excel in computer vision tasks.

6.4.3 Substantial Memory Requirement

Another significant challenge is the substantial memory requirement that the Transformer-based models need for computer vision tasks. Especially when processing high-resolution images, the self-attention mechanism requires storing pairwise attention scores for all input tokens or patches, leading to significant memory consumption. This can become a limiting factor, especially when dealing with limited computational resources or deploying models on resource-constrained devices.

Chapter 7

Choosing the Right Attention Mechanism

In the context of attention mechanisms, the selection of the right mechanism depends on various factors, including the field of artificial intelligence, the specific task, the architectural design, the nature of the data, and more. To facilitate a comprehensive analysis and discussion, this section is divided into two sections, allowing for a more focused exploration of their suitability for specific applications.

7.1 In the Field of Computer Vision

Channel attention mechanisms are commonly employed in tasks such as object recognition and image classification. These mechanisms enable the model to understand which features or channels within the input it should pay attention to, enhancing its ability to capture relevant information for accurate classification or recognition.

Spatial attention mechanisms are particularly well-suited for tasks like object localization and image segmentation. They enable the model to focus selectively on specific regions of the input while disregarding irrelevant background information. By concentrating its attention on the relevant regions, the model can achieve more accurate and precise localization or segmentation results.

Temporal attention mechanisms find extensive application in tasks such as video analysis or action recognition where the input data is sequential in nature, such as video. These mechanisms enable the model to attend to specific frames within the video, capturing temporal dependencies and dynamics crucial for accurate analysis or recognition of actions.

Branch attention mechanisms come into play when the architectural design involves multiple branches or pathways. These mechanisms allow the model to determine which branch or pathway to pay attention to based on their relative importance or relevance to the task. By selectively allocating attention to the most informative branches, models can leverage the collaborative processing and information exchange between different branches, improving their overall understanding and representation of the input data.

7.2 In the Field of NLP

There is a wide variety of attention mechanisms available in traditional NLP architectures, such as additive attention, multiplicative attention, and more. These mechanisms can enhance the traditional NLP model's accuracy by capturing relationships between words, but they often come with increased time complexity due to the sequential processing of data.

The introduction of the attention-based Transformer architecture has provided a solution to this problem. This architecture is highly suitable for many NLP tasks and offers parallel processing capabilities, leading to improved time and accuracy performance. Transformer-based models, such as BERT (Devlin et al. 2018) and GPT (Radford et al. 2018), have gained significant popularity in the NLP community.

BERT is particularly well-suited for tasks that require contextual understanding, such as text classification, sentiment analysis, and question answering. It can effectively capture dependencies between words and provide accurate predictions based on contextual information.

On the other hand, GPT shines in tasks that involve language generation, such as text generation, machine translation, and dialogue generation. Its generative nature allows it to generate coherent and contextually appropriate responses by predicting the next word based on the preceding context.

Chapter 8

Attention mechanism: Benefits and Limitations

Despite the diverse applications and approaches of attention mechanisms in both computer vision and natural language processing, their benefits and limitations can be grouped together due to the shared underlying concept and similar techniques used to achieve them. The fundamental idea behind attention mechanisms remains consistent across domains, aiming to focus on relevant information while suppressing irrelevant information. However, while attention mechanisms bring significant advantages to both CV and NLP tasks, they also come with certain limitations that need to be considered. This section will explore the benefits and limitations of attention mechanisms.

8.1 Benefits

8.1.1 Improved Performance of the Model

Attention mechanisms have greatly improved the modelling of long-range dependencies in various domains, such as natural language processing and computer vision. By enabling models to capture relationships between distant elements in an input sequence or image, attention mechanisms enhance the ability to model long-range dependencies effectively. This is particularly valuable in tasks where understanding the context or relationships between elements separated by significant distances is crucial for accurate predictions or analysis. Attention mechanisms enable the model to allocate its focus selectively, attending to relevant information and capturing the dependencies that exist across the entire input. As a result, models equipped with attention mechanisms can better understand and model complex patterns, leading to improved performance in tasks such as machine translation, image captioning, and sentiment analysis.

8.1.2 Contextual Information Capture

Attention mechanisms play a crucial role in capturing contextual information within the input. By selectively focusing on relevant parts of the input, attention mechanisms enable models to capture and utilise contextual information effectively. This is achieved by assigning weights or importance scores to different elements of the input based on their relevance to the task at hand. By attending to the most informative and contextually relevant parts, models can incorporate and leverage contextual information to make more accurate predictions or decisions. This ability to capture contextual information is particularly valuable in tasks where understanding the surrounding context is essential for proper interpretation and analysis. Attention mechanisms empower models to dynamically adapt their focus and incorporate relevant context, leading to improved performance in tasks such as named entity recognition, sentiment analysis, and machine comprehension.

8.1.3 Flexibility in Handling Variable-Length Inputs

Another notable benefit of attention mechanisms is their flexibility in handling variable-length sequences or images. Unlike traditional approaches that require fixed-size context windows, attention mechanisms can adapt to inputs of different lengths without losing important information. By incorporating attention mechanisms, models can dynamically allocate their focus to different parts of the input sequence or image, regardless of the sequence's or image's length. This allows the models to capture relevant information from the entire input, even when the length varies. Whether it is processing sentences of different lengths in natural language processing tasks or analysing images with varying dimensions in computer vision tasks, attention mechanisms provide the flexibility to handle such inputs seamlessly.

8.1.4 Enhanced Interpretability

In natural language processing tasks, attention mechanisms offer valuable insights into the model's decision-making process. They reveal the specific words or phrases that the model focuses on when generating translations or performing sentiment analysis. Similarly, in computer vision tasks, attention mechanisms provide visual cues by highlighting the key regions of an image that the model deems most important for tasks like object recognition or image captioning. By visualising the attention of the model, researchers can gain a deeper understanding of how the model perceives and processes the input data. This enhanced interpretability is a crucial benefit, as it allows for better transparency and verification of whether the model is attending to the correct features or regions in the input.

8.2 Limitations

8.2.1 Computational and Memory Requirements

Attention mechanisms can introduce computational and memory challenges, particularly when dealing with large-scale or high-dimensional inputs. Computing attention weights for every element in the input can lead to increased computational overhead and memory usage. The complexity of attention calculations grows with the input size, making it crucial to consider the efficiency and scalability of attention mechanisms in resource-constrained environments.

8.2.2 Lack of Interpretability of Attention Weights

While attention mechanisms offer insights into the model's focus areas, interpreting the exact meaning or significance of attention weights can be complex. Understanding the precise relationship between attention weights and the underlying input elements may pose challenges, limiting the interpretability of the model's decision-making process.

8.2.3 Scalability to large inputs

Attention mechanisms may encounter difficulties when applied to extremely long sequences or high-resolution images. As the input size increases, the computational and memory requirements for computing attention weights for each element also grow significantly. This scalability challenge makes it harder to efficiently apply attention mechanisms to very large inputs.

Chapter 9

Further Researches And Conclusion

9.1 Further Researches

Ongoing research in the field of attention mechanisms is actively addressing the limitations discussed in the previous chapter. Researchers are exploring various approaches to mitigate the computational and memory requirements associated with attention mechanisms, making them more scalable for large-scale or high-dimensional inputs. Additionally, efforts are being made to improve the interpretability of attention weights through visualisation techniques, attention mapping, and model explanation methods. By advancing these areas of research, the aim is to enhance the efficiency, interpretability, and overall performance of attention mechanisms in various applications across natural language processing, computer vision, and other domains.

9.2 Conclusion

In conclusion, attention mechanisms have emerged as powerful tools in the fields of natural language processing and computer vision. They enable models to capture long-range dependencies, effectively capture contextual information, handle variable-length inputs, and enhance interpretability. However, attention mechanisms also come with challenges such as computational and memory requirements, a lack of interpretability of attention weights, and scalability to large inputs. Despite these limitations, further research is being conducted to address these issues and explore new possibilities for attention mechanisms in both NLP and CV. As attention mechanisms continue to evolve, they hold great potential for advancing the capabilities of models in various tasks and applications.

Chapter 10

LSEP

It is the responsibility of the researcher to take into account legal, social, ethical and professional issues. While conducting the research, it has been important to ensure that there are no unethical or legal principles that this incorporates. Due to the nature of this paper and approaches that have been taken a big part of the concerns have been mitigated or significantly reduced. The following subsections will go over the main topics that had to be considered.

10.1 Legal Issues

10.1.1 Privacy and Data Protection

As part of this project, there has not been any data collection from people. Furthermore, no public personal data has been used to aid the research. Therefore, there is no need for specific privacy concerns or data protection requirements.

10.1.2 Intellectual Property Rights

In this research project, all papers that are discussed or mentioned are referenced and are publicly available. This paper collates and analyses the different approaches to attention mechanisms and provides an informed recommendation on which are best suited in different scenarios. This research project does not develop novel algorithms or techniques; therefore, concerns with intellectual property must be considered.

10.1.3 Confidentiality of Data

This project utilises open-source, publicly available data that does not contain any sensitive or personally identifiable information for training models or visualising results. That ensures compliance with legal requirements regarding data confidentiality.

10.2 Social Issues

10.2.1 Social Implications

The aim of this project is to provide a comprehensive and critical review of attention mechanisms in deep learning, covering their mathematical foundations, practical applications, and limitations. By gaining a deeper understanding of attention mechanisms, researchers and practitioners can harness their potential to enhance the performance of deep learning models. The implications of this research are significant for various industries and sectors. For instance, in healthcare, attention mechanisms can be leveraged to improve medical image analysis and diagnosis, leading to more accurate and timely treatments. In the finance industry, attention mechanisms can aid in fraud detection and risk assessment, bolstering the security and stability of financial systems. By identifying the strengths and weaknesses of attention mechanisms, this paper paves the way for future research directions that will not only improve their effectiveness but also foster innovation in the field of deep learning.

10.3 Ethical Issues

10.3.1 Public Interest. Do Not Harm

In the context of this project, the primary objective is to provide a comprehensive analysis of attention mechanisms and their applications. Throughout the research process, utmost care has been taken to ensure that the project does not pose any risk of harm to individuals or violate their privacy rights.

10.4 Professional Issues

10.4.1 Professional Conduct and Collaboration

Professional conduct and collaboration are not applicable for this project since it does not require the involvement or collaboration of external individuals or organisations.

Bibliography

Bahdanau, D., Cho, K. & Bengio, Y. (2014), 'Neural machine translation by jointly learning to align and translate', *arXiv preprint arXiv:1409.0473* .

Britz, D., Goldie, A., Luong, M.-T. & Le, Q. (2017), 'Massive exploration of neural machine translation architectures', *arXiv preprint arXiv:1703.03906* .

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. & Zagoruyko, S. (2020), End-to-end object detection with transformers, *in* 'Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16', Springer, pp. 213–229.

Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2018), 'Bert: Pre-training of deep bidirectional transformers for language understanding', *arXiv preprint arXiv:1810.04805* .

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. et al. (2020), 'An image is worth 16x16 words: Transformers for image recognition at scale', *arXiv preprint arXiv:2010.11929* .

Graves, A., Wayne, G. & Danihelka, I. (2014), 'Neural turing machines', *arXiv preprint arXiv:1410.5401* .

Hochreiter, S. & Schmidhuber, J. (1997), 'Long short-term memory', *Neural Computation* **9**(8), 1735–1780.

Hopfield, J. J. (1982), 'Neural networks and physical systems with emergent collective computational abilities.', *Proceedings of the National Academy of Sciences* **79**(8), 2554–2558.
URL: `https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554`

Hu, J., Shen, L. & Sun, G. (2017), 'Squeeze-and-excitation networks', *CoRR* **abs/1709.01507**.
URL: `http://arxiv.org/abs/1709.01507`

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2017), 'Imagenet classification with deep convolutional neural networks', *Communications of the ACM* **60**(6), 84–90.

Li, J., Wang, J., Tian, Q., Gao, W. & Zhang, S. (2019), 'Global-local temporal representations for video person re-identification', *CoRR* **abs/1908.10049**.
URL: `http://arxiv.org/abs/1908.10049`

Li, X., Wang, W., Hu, X. & Yang, J. (2019), 'Selective kernel networks', *CoRR* **abs/1903.06586**.
URL: `http://arxiv.org/abs/1903.06586`

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S. & Guo, B. (2021), Swin transformer: Hierarchical vision transformer using shifted windows, *in* 'Proceedings of the IEEE/CVF international conference on computer vision', pp. 10012–10022.

Lu, J., Yang, J., Batra, D. & Parikh, D. (2016), 'Hierarchical question-image co-attention for visual question answering', *Advances in neural information processing systems* **29**.

Luong, M.-T., Pham, H. & Manning, C. D. (2015), 'Effective approaches to attention-based neural machine translation', *arXiv preprint arXiv:1508.04025* .

Niu, Z., Zhong, G. & Yu, H. (2021), 'A review on the attention mechanism of deep learning', *Neurocomputing* **452**, 48–62.

Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B. et al. (2018), 'Attention u-net: Learning where to look for the pancreas', *arXiv preprint arXiv:1804.03999* .

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I. et al. (2018), 'Improving language understanding by generative pre-training'.

Schuster, M. & Paliwal, K. (1997), 'Bidirectional recurrent neural networks', *IEEE Transactions on Signal Processing* **45**(11), 2673–2681.

Sutskever, I., Vinyals, O. & Le, Q. V. (2014), 'Sequence to sequence learning with neural networks'.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. (2017), 'Attention is all you need', *Advances in neural information processing systems* **30**.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R. & Bengio, Y. (2015), Show, attend and tell: Neural image caption generation with visual attention, *in* 'International conference on machine learning', PMLR, pp. 2048–2057.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A. & Hovy, E. (2016), Hierarchical attention networks for document classification, *in* 'Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies', pp. 1480–1489.