UNIVERSITY OF TURKU

An Eye for AI: A Multimodal Bottleneck Transformer Approach for Predicting Individual Eye Movements

Towards Foundation Models for Human Factors & Neuroscience

Human Neuroscience

Master's thesis

Turku Brain and Mind Center

Author(s):

Tenzing Christopher Dolmans

16.06.2023

Turku, Finland

Master's thesis

Subject: Human Neuroscience
Author(s): Tenzing Christopher Dolmans
Title: An Eye for AI: A Multimodal Bottleneck Transformer Approach for Predicting Individual Eye Movements
Supervisor(s): Prof. Jukka Leppänen, Assoc. Prof. Prof. Antti Airola
Number of pages: 63 pages
Date: 16.06.2023

Human perception has been a subject of study for centuries. Various eye tracking methods in many study designs have shed light on individual differences in perception and visual navigation. However, accurately identifying individuals based on gaze behaviour remains a challenge. Artificial intelligence (AI) based methods have led to large successes in domains such as vision and language; they are also making their introduction in human factors & neuroscience (HFN). Leveraging AI for HFN requires quantities of data several orders of magnitude larger than the field is used to organising; there exists a clear discrepancy in the standardisation of data publication. In this work, we work towards foundation models (FM) for HFN by highlighting important data insights from AI. A multimodal bottleneck transformer is proposed, a model architecture that can effectively and efficiently represent and work with the varying modalities encountered in HFN. Results indicate that classification of individuals and prediction of gaze is possible, given more training data.

Key words: eye tracking, deep learning, standardisation, transformers, multimodal AI

Table of contents

1 Introduction

Human perception has been a subject of study for centuries. Due to the development of various research methodologies, among them eye tracking, visual perception can readily be studied. We navigate the world by observing, making sense of our observations, and then making decisions. The order and importance attributed to observations varies from person to person and the extent to which individual observational patterns differ has been widely studied. The effect and underlying causes of said differences is poorly understood and remains under continuous study. Identification of differences in eye-movements takes on many forms, for example, facial scanning recognition strategies, in which distinctions like “eye-looker” and “nose-looker” are discernible (Peterson & Eckstein, 2013). Moreover, idiosyncrasies in observed oculomotor patterns can be used in various contexts, such as security and privacy applications (Katsini et al., 2020). Another notable field of application is the widespread use of eye-tracking based medical screening, which utilises the fact that eye movements rely on the integrity of a distributed network of brain areas, and so may provide useful information in an initial assessment of, for example, Parkinson’s disease, or autism spectrum disorder (Itti, 2015). While purely eye tracking based medical diagnoses are unlikely to be prevalent in the near future, studying features of individual oculomotor characteristics may aid in the identification of markers that are diagnostic of a neurological condition. Notably, visual orientation is not solely done with information from the ocular modality, it integrates multiple senses. For example, when balancing, the visual system works closely together with the proprioceptive and the vestibular system to integrate various hints about posture and body positioning (Redfern et al., 2001); studying gaze behaviour is a multimodal endeavour.

On a high level, analysis of gaze behaviour, with a focus on saliency, may provide a basis of successful differentiability of individuals. Saliency is a property ascribed to entities that are perceived to be significant or important in some way, such as objects, people, and emotions (J. Xu et al., 2014). When exploring our visual surroundings, we tend to orient our eyes to regions that are in high salience and evidence suggests that a saliency map is constructed by the brain to keep track of salient entities in space (Itti & Koch, 2000). Distinguishing patterns in the selection of such entities, or even the construction of such maps may provide a fruitful basis for the classification of gaze-related idiosyncrasies. De Haas et al. (2019) found stable individual differences in the allocation of attention to different categories, demonstrating the

predictability of salience. However, these differences were not predicted by other known idiosyncratic properties in gaze, such as the tendency for visual exploration. Nonetheless, saliency dimensions found by de Haas et al. (2019) are excellent predictors of gaze on a group level.

Human vision is also subject to study on a lower level. The human visual field is relatively large, spanning over 200 degrees horizontally and 135 degrees vertically. However, it is highly foveated, meaning that only a fraction of the entire field, less than two degrees, falls in the central part of the retina and is thus subject to highly detailed vision (Zigmond et al., 1999). Given this, we explore our environments and construct saliency maps using a series of rapid, ballistic movements, saccades, that direct our fovea (Zigmond et al., 1999). Research on the oculomotor systems that give rise to these saccades can aid in the identification of individual differences. These differences lie in things such as temporal characteristics of saccades and pursuits (Bargary et al., 2017a) and saccadic (de)acceleration (Rigas & Komogortsev, 2016). In addition to more traditional approaches, where features are selected and crafted based on the literature's understanding of eye-movements, more recently developed approaches seek to tackle the problem with artificial intelligence (AI). AI-based methods automatically extract patterns from the data and learn a meaningful representation for a given task.

This work is structured as follows: The goals and hypotheses are explained, after which the achieved contributions are summarised. Then, the theoretical background is built up by highlighting three key research works. The first work is one on foundation models (FM) by Bommasani et al. (2021); second, the transformer architecture by Vaswani et al. (2017); third and final, the multimodal bottleneck transformer architecture by Nagrani et al. (2021). Combined, these works serve to construct a basis for the present study. After that, important literature for the study of human gaze is discussed, covering primarily AI-based methods. Then, through the combination of the key works and discussed literature, a novel approach to the multimodal study of human gaze will be introduced and developed. The architecture of the approach is such that it can be expanded to the study of all human factors and human neuroscience (HFN). Concrete methods based on the concepts and insights developed in the framework are described, after which they are situated in the current meta of HFN research. Finally, a discussion will take place surrounding the limitations of the current implementation and recommendations for the future are presented.

1.1 Goals

The overarching goal of the present study is to work towards the standardisation of a system in which human-factor & neuroscience (HFN) data are collected and processed. As it stands, many of the studies that combine neuroscience and AI are conducted on metaphorical islands. Each island has its own language, making it very difficult to bundle the knowledge of said islands. One has to go on a search for such islands, learn to speak the language, translate the findings (through which some of the knowledge will be misrepresented), and bundle all gained insights and data. Only then can a study be done on a larger collection of concepts. Recently, many fields have started moving towards open science. Thousands of datasets are published each year, in a myriad of fields. Given ample time and expertise, new, AI-suitable datasets can be created. However, this process is prohibitively arduous and expensive. There is simply a lack of consensus in the field when it comes to standardisation. Dolmans et al. (2021) talk about modularity and generality (MG) as a set of tools by which methods are to be evaluated. Similar to the concept of polymorphism, MG seeks to mould research pipelines into ones that can be grasped and reused with minimal effort; these MG requirements should also apply to datasets. By following a set of guidelines, a common format can be agreed upon and enforced by a field. What are valuable guidelines for standardisation of behavioural and human-factor related data for AI usage?

The overarching goal of working towards standardisation is applied to the classification of individuals based on eye tracking data collected during image viewing, as a proof of concept. Xu et al. (2014) constructed a dataset of 700 images for which semantic-level information is available, named Object and Semantic Images and Eye-tracking (OSIE). They show that semantic attributes significantly explain the saliency of viewed images. Properties that are particularly impactful for saliency are the presence of a face, both with and without an emotion, text, and watchability (objects that are designed to be looked at, like paintings and signs). De Haas et al. (2019) further show that (i) the saliency weights of these semantic attributes vary substantially between individuals and (ii) the variation between individuals is consistent across different images within a viewing session and across viewing sessions on different days.

The present study seeks to outline a framework for accurately classifying individuals in eye tracking studies, as well as to predict an individual's gaze pattern for both seen and unseen stimuli. As previously discussed, the study of vision is a multimodal study by nature; the

framework must be modality extensible to HFN. A neural network that is able to distinguish individual viewers based on their gaze behaviour will be devised. Furthermore, due to recent advancements in sequence-to-sequence modelling, the same network can predict raw gaze, given an understanding of individual viewing patterns and a stimulus. Can such a network conceivably be designed and trained? The OSIE paradigm provides an excellent example case since it lends itself to multimodal study by combining images, their respective semantic labels, and eye tracking data. On the basis of the above, the following goals are identified:

- G1: Create an example case that meaningfully and concretely contributes to the standardisation of data organisation in eye tracking, making use of multimodal AI.
    - G1.a: Determine whether it is possible to classify individuals using a multimodal bottleneck transformer in eye tracking tasks.
    - G1.b: Determine whether it is possible to predict a specified individual's gaze behaviour using a multimodal bottleneck transformer.
- G2: Provide concrete guidelines for the contribution of data and pipelines to human factors & neuroscience foundation models.

1.2 Hypotheses

We hypothesise that the design and implementation of a modular and generalisable (MG) multimodal bottleneck transformer (MBT) meaningfully contributes to the standardisation of HFN research; imposing the MG criteria forces the expansion of scope such that the devised methodology can be applied to arbitrary multimodal configurations. We further hypothesise that the large-scale data ingestion requirements and the necessary workflow of data contributes to the communication about and publication of data in HFN.

1.3 Contributions

This present work contributes several key components to the field of eye tracking studies and human factors and neuroscience (HFN). First and foremost, a flexible implementation of a multimodal bottleneck transformer (MBT) for usage in HFN is developed. The MBT was developed and implemented with modularity and generalisability in mind (Dolmans et al., 2021); to be modular, new modalities should be easy to add and their data should fit within the data processing pipeline with minimal structural implications. Generalisability refers to

that the addition of new modalities should contribute to the overall performance of the system. The code is openly published on GitHub (Eye4AI, 2023), see Figure 1 for a high-level overview of the model architecture.

Secondly, several key bottlenecks for the development of next-generation AI-based platforms for research are identified and suggestions are made for overcoming said bottlenecks. Concretely, consensus on the generation and publication of data in common format for AI purposes needs to be reached. Several suggestions are made to overcome these issues in a multimodal setting through vector-based organisation of data. Further, embedding methods are introduced for various modalities in similar fashion to language (Pennington et al., 2014; Peters et al., 2018). Recommendations for the generalisation to arbitrary modalities are made.

Third, we expand on the scope in which multimodal AI is approached by moving away from what is conventionally considered multimodal: language-image understanding and generation (J. Li et al., 2022; OpenAI, 2023). We expand to the prospect of including an extensible number and variety of modalities in a single unified model inductively through the inclusion of eye tracking, vision, and semantic labels.

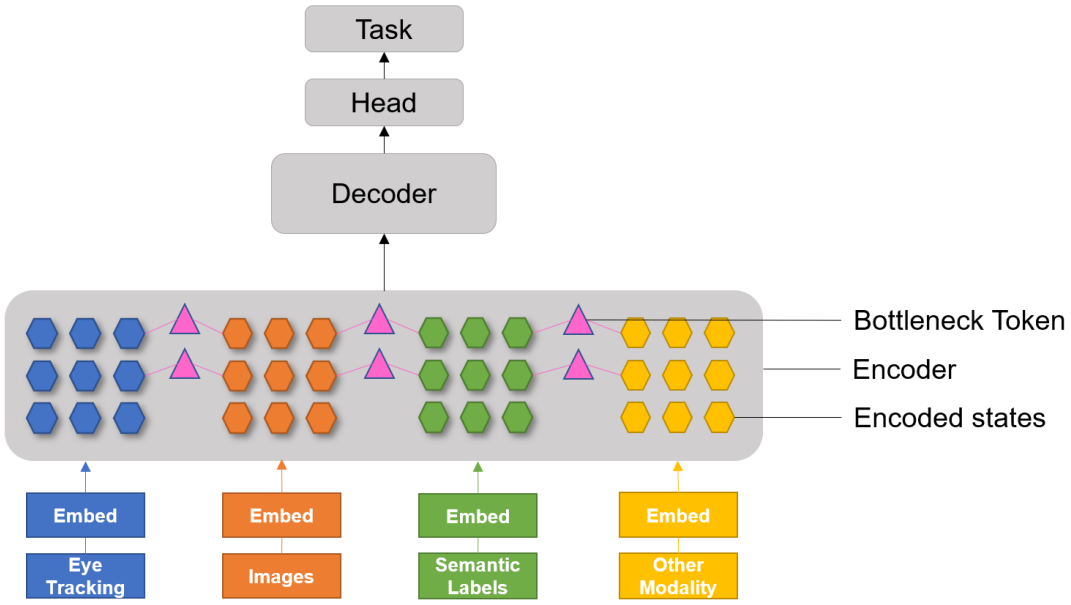

Figure 1 High-level overview of the multimodal bottleneck transformer (MBT) implementation. From bottom to top, modalities are embedded and fed to an encoder. The encoder builds a representation of the data using multiple layers. Encoded states = representations of input data after encoder layers; encoder = MBT encoder, adapted from (Nagrani et al., 2021); bottleneck token = tokens that are used to communicate information between the encoded states of different modalities; decoder = transformer decoder (Vaswani et al., 2017); head = head network that converts decoded states to the correct dimensionality; task = downstream objective of the model.

2 Key Works

2.1 Foundation models

As previously indicated, one of the key works in the context of this work is the introduction of Foundation Models by Bommasani et al. (2021). A foundation model (FM) is a model that is trained on a vast collection of broad data in a specific domain or modality. From said FM, a wide range of applications can be developed through down-stream adaptation, or so-called finetuning. The strength of FMs follow from their emergent capabilities and homogenisation. Emergence refers to the "spontaneous" behaviour that is implicitly induced, rather than explicitly crafted, whereas homogenisation refers to the convergence of machine learning methods for a wide range of applications. Hence, the availability of an FM in a domain allows for the creation of a variety of downstream implementations through careful finetuning. Bommasani et al. (2021) attribute the possibility of FMs to three factors: (i) the continuing improvements of hardware (specifically GPUs and TPUs); (ii) the availability of much more data; (iii) the widespread implementation of the Transformer architecture (Vaswani et al in 2017), which will be extensively discussed in later sections. The quality of the emergent capabilities inherently depends on the quality of the FM.

As per (i, ii), modern AI has been able to make such large strides due to the exponential increase in data availability and computational power. Consumer-grade computational power continues to make significant strides, all the while still reaping the benefits of Moore's law and other cost-reducing innovations (Shalf, 2020). Regarding increased data availability, almost 82% of all websites use images in PNG format, closely followed by JPEG at almost 78%; SVGs also hold a sizable 54% stake (W3Techs 2023). Images are often also conveniently accompanied by a written description or title, providing ample training data of text-image pairs for model training. Furthermore, the open availability of over 750GB of text data allows for the training of increase large, large language models (LLM) (Thompson, 2022). There exists such a widespread consensus on the format and organisation of text and image data that truly massive compilations of data can be crafted. Anyone is able to take a stab at the problem of building FMs, provided they can afford the significant costs associated with the expertise and hardware. For a sense of scale, generative pre-trained transformers (GPT) such as GPT-4, OpenAI's latest FM, cost more than $100 million to train, according to CEO Sam Altman (WIRED, 2023). However, training costs are not all. According to SemiAnalysis' chief analyst Dylan Patel, daily operation costs for ChatGPT (a finetuned

version of the GPT class FM) could be around $700,000 per day (Patel & Mok, 2023). Midjourney, a model created by an independent research lab with the same name can synthesise images from textual data (Midjourney, 2023). Given the availability of vast collections of images and art alike, Midjourney can be considered an FM in the visual domain.

If one were to attempt to reproduce the creation of FMs such as GPT-4 and Midjourney for HFN by compilation data such as EEG, fMRI, eye-movement data, etc., one would fail. The problem is twofold. First, not nearly all data are publicly available, restricting the size of the theoretically creatable dataset. Second, the formats and organisation in which these data are presented are incredibly diverse; cracking the experimental paradigms, research objectives, and quirky programming habits that "encode" the data is prohibitively expensive. Luckily, there has been a recent push for standardisation for the collection and reporting methods in eye tracking studies by Holmqvist et al. (2022). Their paper presents various factors that affect the data quality in eye tracking studies and further outlines standard practices that can be used to improve reproducibility. However, even this most up-to-date guideline for eye tracking research does not address the need to provide a structure in which eye tracking data can be stored and shared such that the data are easily reusable from an AI perspective. The commonly used FAIR (Findable, Accessible, Interoperable, and Re-usable) principles provide a framework for concrete actions that aid research collaboration (The FAIR Data Principles – FORCE11, 2023). Section R1.3 states that (meta)data should meet domain-relevant community standards. However, such standards do not exist for many of the aforementioned modalities.

The fields of HFN and AI are too different, leading to discrepancies in understanding from either side. There exists a gap between what is understood in the field of HFN and what is needed for the large-scale sharing of data. Data principles that might suffice in the reproducibility of individual research works fall short when it comes to the compilation of datasets on the scale of the abovementioned available text and image sets. Conversely, the field of AI is inadequately equipped to interpret and make the most of HFN data, even if it were available. There exists a clear discrepancy in the currently available model architectures for the purpose of HFN. It is up to those who have knowledge of both fields to start connecting the disciplines. It is also pertinent to improve communication between experts in the two fields, rather than rely solely on individuals with simultaneous expertise in both. In

this work, we will work towards a framework in which a FM for HFN can be constructed, from data creation to deployment.

In order to be able to concretely work towards FMs in HFN, several key components are missing. Bommasani et al. (2021) provide five clear stages in the development of FMs. In this section, we will explore what these five stages entail in the field of HFN. For the sake of brevity, we limit the scope of the following considerations to eye-tracking and neuroimaging data. Later, in the more practical parts of this work, we focus on eye-tracking only.

2.1.1 Data Creation

The creation of data is the first step in this process. Data are typically generated for a specific purpose. In HFN, the purpose generally pertains to the study of, for example, human psychology, decision making, physiology, and perhaps more importantly, their dynamical interaction. The recent push for unification of data creation by Holmqvist et al. (2022) bodes well for the field of eye-tracking. Other fields, such as (f)MRI and radiology already enjoy a more unified data organization through the NIFTI, MINC, and DICOM formats (Sriramakrishnan et al., 2019). Of course, these formats are not without problems, see Larobina & Murino (2014) for a discussion. Nonetheless, through several software packages, such as MRIcro, researchers are able to navigate most of the prevalent data formats (MRIcro: Tool/Resource Info).

2.1.2 Data Curation

Unfortunately, no perfect distribution of data has ever or will ever exist. Some level of curation is always required. Depending on the considerations that underlie the creation of an FM, legal and ethical concerns must be addressed, preferably at the data creation stage. Trade-offs between data quality and quantity are to be carefully considered during this stage. Currently, it is hard to find amply large datasets that consists of intrinsically well-aligned sample/label pair in HFN. For example, the largest publicly available eye tracking related dataset is TEyeD, which contains 20 million real-world eye images for the purpose of improving gaze estimation in VR and AR (Fuhl et al., 2021); one of the larger MRI (and other modalities) dataset contains some 1350 participants (Snoek et al., 2021). We will later discuss just how much data is required for FMs. Creating such large datasets requires considerable effort and collaboration between labs, universities, and countries. As put wisely by a

colleague: “Fields that can generate data in common format are going to move faster than fields that can’t.”

2.1.3 Training

The architecture of the model and training pipeline are determined prior to the commencement of the training. Data ingestion methods that retain domain information must be designed, which requires expertise. Besides the conservation of signal, the ingestion methods must focus on efficiency. On large scale projects like the creation of FMs, an efficiently organised training sequence can save several orders of magnitude; the difference between 1023 and 1025 FLOPs (Hoffmann et al., 2022). This affects not only the end-to-end time, but also limits the environmental impact of training large models. Gopher, a 280B parameter model, was trained on 4096 TPUv3 chips, which emitted an estimated 380 tons of CO2, rather than the estimated emissions of 552 tons of CO2 as a result of training the 175B parameter GPT-3 (Rae et al., 2021). It is evident that efficiency considerations should be made carefully, they have significant impacts on emissions and energy usage.

The Chinchilla model was specifically trained and evaluated with the performance to cost ratio in mind (Hoffmann et al., 2022). It was found that current LLMs are significantly undertrained, and performance could be improved by increasing the training data. Conversely, the same performance can be achieved with much smaller models. In general, Hoffman et al (2022) recommend 20 times the number of parameters worth of tokens. However, in the case of FMs for HFN, it becomes unclear what “one token” pertains to, since datatypes vary significantly. This issue is comprehensively discussed in the Embedding section.

2.1.4 Adaption

After training a domain specific FM, it must be finetuned to the context in which it will be deployed (Hu et al., 2021). During training, the range of possible contexts in which the model will operate should be taken into consideration to ensure that it is able to generalise over the scope of indented adaptations. This phase also defines the rules that guide behaviour, objectives, as well as the restrictions of the network. Undesirable qualities and behaviours must be identified and resolved. In LLMs, these undesirable qualities are often referred to as bias and toxicity (Chowdhery et al., 2022). Extensive testing is done to assess the degree to which an LLM tends to display such behaviours. Similarly, the bias and toxicity of any FM should be assessed in ways that fulfil requirements of the respective domain.

In HFN, bias is the largest concern, as these models are used for classification, intervention, and medical decisions and advice. A recent work developed a unified and generalist Biomedical GPT, which effectively makes use of the methodological breakthroughs achieved in language modelling, such as self-supervised learning (Zhang et al., 2023). The authors find that BiomedGPT is competitive in tasks spanning vision, language, and various multimodal domains in the biomedical sector, such as disease predition. Notably, Zhang et al. (2023) further draw attention to the sensitivity to training data balance, indicating that careful planning and extensive testing is required to find a balance in data diversity. Concerningly, BiomedGPT sometimes fails to understand instructions and outputs irrelevant data types to the task at hand; such mode collapses should be identified and resolved prior to deployment.

2.1.5 Deployment

During deployment, any interested party might ideally be able to publish and work with their adapted application for an FM. The impacts of every deployment should carefully be assessed prior to release to the public. Every domain has specific requirements. In the case of HFN, there is an increased need for privacy-oriented implementation. One of the largest strengths of FMs are the emergent capabilities and behaviours. However, this might also be one of the largest pitfalls; current legislation is not oriented towards this rapidly developing regime and will fall short in the protection of human rights (Ienca & Andorno, 2017). At a low level, it is currently often not possible to perform inference without giving up privacy, simply because the infrastructure does not support it. The concept of MPCFormer outlines a design which tackles encrypted inference, allowing for inference without compromising privacy or speed (D. Li et al., 2022). More fundamentally, the dawning age of neuroinformation threatens several, currently relatively undefined, human rights. Lenca & Andorno (2017) identify an initial four human rights that are threatened by current developments: cognitive liberty, mental privacy, psychological continuity, and mental integrity. While these concerns may seem relatively out of scope for our current state, the increasingly rapid development of FMs urges us to leave no stone unturned in our considerations.

2.1.6 Proposed: Evaluation

We propose a sixth stage to the development of foundation models: evaluation. We are entering an era where FMs are displaying human-level capabilities, increasingly in multimodal settings (OpenAI, 2023; Zhang et al., 2023). With the increasing capabilities

comes increasing evaluation difficulties. There exist many benchmarks that can be used to evaluate models, with many more in development. Examples are planning (Valmeekam et al., 2022), vision-language tasks (Zhou et al., 2022), and using API's (M. Li et al., 2023). However, many fields and modalities are still lacking clear standards. Furthermore, as will be discussed further in later sections, performance metrics are not always to be trusted due to various issues such as unresolved session bias. An eye-tracking example of session bias: if train and test data are collected on the same day, there exists inherent uncertainty about what the data are classified on. Rather than classifying solely on eye-movement features, the network can classify on noise, such as the angle at which a participant is viewing the screen, the brightness of environmental light, or even the presence of electrical noise in the signal. Hence, we deem it essential that that evaluation and specifically the open communication about said evaluation is emphasised during the creation of new FMs.

2.2 Transformers

In this section, the transformer architecture as introduced by Vaswani et al. (2017) is discussed. Transformers are the second key concept relevant for the present work. This forms the basic building block for later sections of theoretical background, as well as the practical implementation of the present work. Crucially, the success of transformers can be attributed to three properties: self-attention, universality, and scalability. Self-attention extensively discussed in the following section. The universality is demonstrated through the application of transformers in the modelling of many different modalities, such as language (OpenAI, 2023), vision (Dosovitskiy et al., 2020), eye tracking (Rolff et al., 2022), biomedical imaging (Zhang et al., 2023), and many more. It is considered to be a "universal" architecture that can be adapted to any arbitrary organisation of information. Lastly, transformers are highly scalable when distributed across modern GPUs with many cores, allowing for bigger models and thus performance improvements (T. Lin et al., 2022). However, scalability remains an open challenge for transformers (Peng et al., 2023). Transformer's time and space costs scale quadratically with sequence length. I.e., time: $O(T^2 d)$; space: $O(T^2 + Td)$ where T is the number of tokens in the sequence and d is the hidden dimension of the network.

Transformers come in various flavours, depending on how its components are combined. Specifically, the encoder and decoder components. The encoder is used to map an input sequence to a sequence of representations of said sequence. The decoder is used to generate the output sequence based on both its own representations, as well as the representations

produced by the encoder. Transformers thus come in the following varieties: encoder only, decoder only, and encoder-decoder (J. Yang et al., 2023). Encoder only and encoder-decoder models are generally used in masked learning, where items in a sequence are masked and predicted. Decoder only models are generally autoregressive, meaning that they produce future items based on past items (Weber & Gühmann, 2021).

2.2.1 Attention

One of the key characteristics of transformer networks is the implementation of self-attention, more specifically scaled dot-product attention as introduced by Vaswani et al. (2017). Their attention maps an input to an output by computing a similarity measure. Attention is computed using several vectors, namely the query, key, and value vectors, by computing a weighted sum of their values. The query vector represents the data or positions in a sequence for which we want to compute attention. The key vector represents the tokens or positions that are being attended to. It contains information that can be compared to the query vector to measure the relevance and comparability between the query and the keys. Finally, the value vector contains the actual information that is used to compute the weighted sum during the attention mechanism. It is associated with each token in the input sequence. Each of these vectors is obtained by multiplying the input sequence with a learned weight matrix. It is important to note that the input sequence is not raw data, rather a latent representation, or embedding, of said data. The mechanisms underlying embedding are discussed in the embedding section.

The dimensionalities of the query, key, and value vectors are d_q, d_k and d_v, respectively. The weight assigned to each value vector is computed by a scaling factor $\sqrt{d_k}$. The dot product of the query with all keys is calculated simultaneously in matrix form. A SoftMax function is applied to obtain the weights on the values. Equation 1 describes the computation of attention as described above. The benefit of dot-product attention is its efficiently scaling and space-efficient matrix-multiplication based implementation. This implementation is identical to dot-product attention, except for the scaling factor of $\frac{1}{\sqrt{d_k}}$.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

Vaswani et al. (2017) found that linearly projecting the queries, keys, and values h times allows the model to jointly attend different (latent)subspaces of the input, which is beneficial

for generalisation. This leads to the concept of multi-head attention (MHA). By scaling the dimensionality of each head down, the total computational cost is comparable to single-headed attention in a scenario where full dimensionality is retained. Equation 2 describes the implementation of MHA.

$$MultiHead(Q,K,V) = Concat\big(head_1, \dots, head_h\big)W^O \quad (2)$$
$$where\ head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

Where the projections are learnable parameter matrices W_i^Q, W_i^K, W_i^V, and W^O is a learnable matrix which' weights balance the contribution of the various heads. Figure 2 visualises the construction of scaled dot-product attention and MHA. The weight matrices are what define the effectivity of the model's learning and they are subject to optimisation through backpropagation. Backpropagation was introduced by Finnish mathematician Seppo Linnainmaa in their master's thesis, albeit without reference to neural networks (Linnainmaa, 1970). To make use of backpropagation, a metric is defined by which the model's performance is evaluated by introducing a loss function. The loss function defines how well the model performs on a single input-output pair. If the loss is non-zero, backpropagation computes the gradients of the loss with respect to the weights in an efficient manner. From this gradient, a step can be taken in the direction that reduces the loss, this process is called stochastic gradient descent and was introduced by Robbins & Monro in 1951. Modern adaptations of these core concepts have accelerated AI considerably.

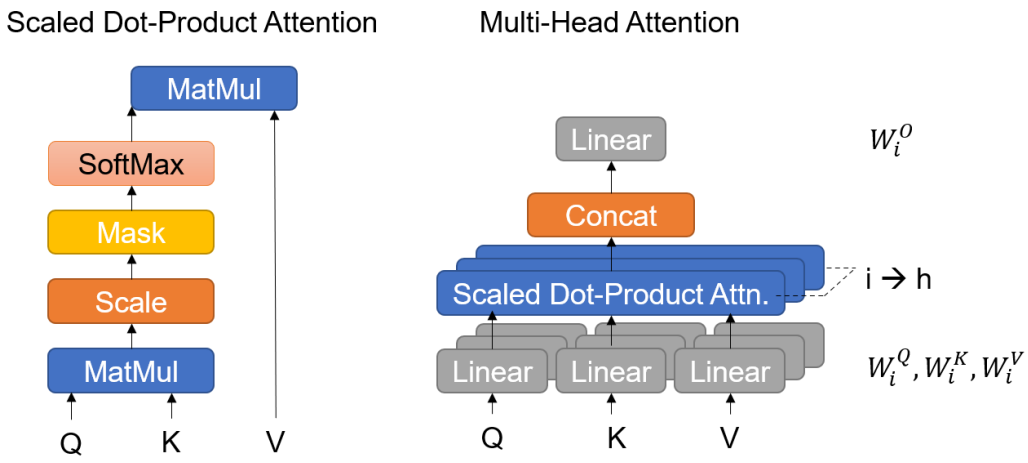

Figure 2: Adapted from (Vaswani et al., 2017). Scaled dot-product attention and the multi-head implementation of it. Q, K, and V represent query, key, and value, respectively. W_i refers to a weight matrix where the letter in superscript refers to Q, K, V, or output (O) and i refers to the current head of the multi-head block.

2.2.2 Transformer Blocks

In order to effectively use MHA, it is used in transformer blocks. These blocks are constructed by stacking and connecting multiple components. The blocks are similar in both the encoder and decoder, the differences will be discussed in the paragraph following the current one. For the construction of transformer blocks, first, MHA is computed over $Q,\ K,\ V$ and stacked with layer normalisation (LN). The LN also ingests a residual connection from $Q,\ K,\ V$ (He et al., 2015). In the original model architecture, the LN follows the MHA. Networks that use this order of operation are known post-LN transformers (Vaswani et al., 2017). After many attempts at improving the transformer architecture, it was found that one of the very few meaningful changes that can be made to the architecture, is shifting from a post-LN to a pre-LN architecture. This small change leads to more stable initialisation when using a warm-up stage and is thus favourable for some tasks (Xiong et al., 2020). However, in this work, we apply a post-LN architecture, because we do not deploy the model in such a way that this improvement would bear fruits. After the MHA and LN two components, the hidden dimension of the network is expanded by a certain factor and fed to a densely connected feed-forward layer. Conventionally, the expansion factor is kept at four. After the forward expansion, another LN is done on the outputs of the linear layer together with another residual connection from first LN. Then, and finally, dropout is introduced as a regularisation (Srivastava et al., 2014). This completes the basic implementation of a transformer block. The blocks are stacked, leading to "layers". The definition of layers in this context differs from the definition of layers in the context of multilayer perceptrons (MLPs), where it refers to the number of densely connected linear feed-forward layers (Rosenblatt, 1958).

The transformer block described above is the "vanilla" block as encountered in the encoder section of a network. In the decoder, the block is similar, barring a few adjustments. The decoder transformer block consists of the same stacking of LN an MHA but is repeated an extra time, rather than being immediately followed by a feed-forward layer. Depending on the implementation of the transformer, the decoder block performs its second iteration of LN and MHA over queries produced by the first MHA and the outputs of the final encoder block. Similar to the encoder, the decoder also utilises residual connections (He et al., 2015). Furthermore, the self-attention in the decoder can be masked such that the predictions for item i in the sequence cannot depend on $i + 1$. Attending items that follow the current item would

lead to a form of cheating, since the network would be able to copy what follows the current item as its prediction. See Figure 3 for visual intuition on the interaction between encoder and decoder blocks.

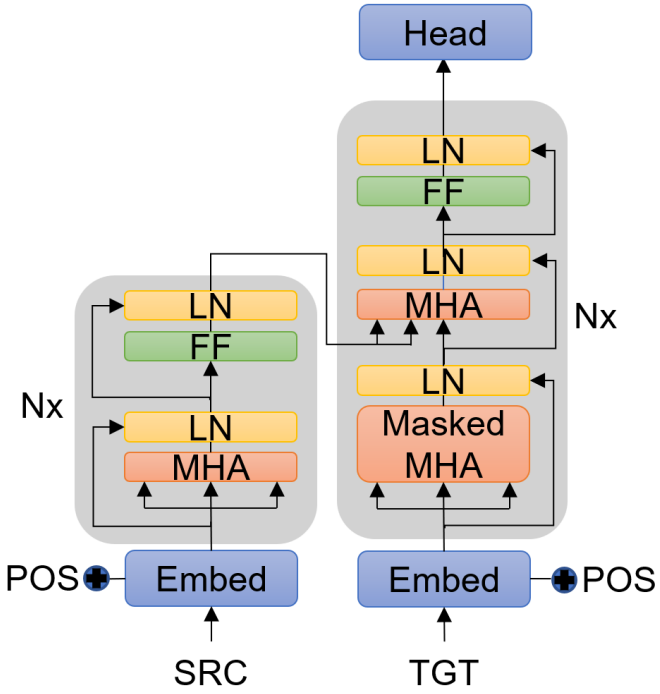

Figure 3: Encoder and Decoder blocks in the transformer architecture, adapted from (Vaswani et al., 2017). Embed = embedder class for input data; POS = positional embedding; MHA = multi-head attention, described in Figure 1; LN = layer normalization, FF = feed forward; Head = head network; Nx = number of layers for the Encoder and Decoder.

2.3 Multimodal Bottleneck Transformers

The third and final key work is the work on attention bottlenecks for multimodal fusion (Nagrani et al., 2021). Transformers can be extended to a multimodal context by computing attention over multiple sequences of length T, one for each modality. These sequences are essentially 'fused' into one, allowing for the attendance of tokens from multiple modalities in a single stroke. As highlighted in a survey of SotA transformer architectures by Xu et al. (2022), there are several paths that lead to multimodal network capabilities, with various options for early, intermediate, or late fusion. Options span from early concatenation of data, to using multi-stream architectures that fuse somewhere, to architectures that keep modalities separate for several transformer blocks and later concatenate to one stream (J. Lin et al., 2020; P. Xu et al., 2022). Intermediate fusion is favourable due to the improved performance that is

observed compared to early and late fusion (Nagrani et al., 2021); such fusion requires self-attendance across modalities.

However, as previously indicated, the time and space complexity scales quadratically with T (Peng et al., 2023). Hence, computing attention over multiple modalities imposes large restrictions on T. Naively extending T is computationally very expensive. Several options for improving performance through multimodal fusion are available. First, by concatenating sequences of modalities such that

$$T = \sum_{i=1}^{M} T_i$$

where M is the number of modalities and T_i is the sequences length of the i^{th} modality and accepting the increased computational overhead. The second option is to truncate T_i hyperbolically in order to occupy equal time and space:

$$T = \sum_{i=1}^{M} \frac{T_i}{M}$$

Context length is directly related to performance, increasing M thus leads to increasingly diminished utility from the same computational footprint (Takahashi & Tanaka-Ishii, 2018). The third option for tackling efficient communication about cross-modal interactions is using a cross-modal attention bottleneck (Nagrani et al., 2021). The bottleneck consists of a relatively small, and crucially constant, number of learnable tokens that are shared across encoders for all modalities. Nagrani et al (2021) introduce these tokens and demonstrate how limiting the time and space for the exchange of information between modalities is beneficial to the performance of the network. To make effective use of these bottleneck tokens, attention is computed over the concatenation of each modality's tokens and the bottleneck tokens. The bottleneck is updated separately and simultaneously with information from all modalities. Because all communication between modalities is now highly restricted, the model learns to populate the bottleneck with only highly relevant information.

There exists contrasting evidence on where the bottlenecked attention sharing should be done. You et al. (2022) indicate that freely allowing cross-model attention yields the best results. Conversely, Nagrani et al. (2021) claim that allowing free attendance is not as effective as restricting it to several of the later layers in a network. The ideas of multi-stream cross-attention and MBT allow for the design of efficient, cross-modal self-attending networks in

any context. However, this does require that the input space can be somehow tokenised and embedded representatively, which requires domain knowledge.

2.3.1 Bottlenecking in the Brain

The brain has implemented communication between regions in a bottlenecked fashion. White matter tracts facilitate communication between regions by, among other things, providing functional connectivity (Bijsterbosch et al., 2018). The authors further find that spatial ‘connectivity-fingerprints’ are more predictive of cross-subject variance than temporal equivalents. This suggests that the organisation of connectivity gives rise to substantial variability from brain to brain. The axons that make up white matter tracts provide ‘long-distance’ communication between regions. Identifying faulty communication between brain regions, or faulty bottlenecks, has led to the improved classification of mild cognitive impairment (MCI) associated with Alzheimer's disease (AD) (Wee et al., 2012). Of course, the brain does not solely consist of such white matter tracts, suggesting that not all matter in the brain is structured around communication between regions. A parallel can be drawn between the functional localisation in the brain and the processing of multimodal data. The visual cortex then becomes, for example, a vision transformer (Dosovitskiy et al., 2020). The white matter tracts connecting the visual cortex to the motor cortex then become the bottleneck: communication is expensive and should thus be reserved for highly relevant information. Expand this functional localisation and functional connectivity to the full faculty of our senses and a complex system is born: the brain.

3 Further Literature

Previous work has been done on the use of AI-based methods for the classification of individuals in eye tracking studies. In this section, a variety of works are discussed to identify (un)promising methodologies, gain perspective on the effectiveness of these methods, and identify shortcomings. Then, various developments in deep learning are summarised and issues are highlighted. Lastly, the conversion from data to transformer-ingestible data is discussed. This process is referred to as embedding.

3.1 Deep Learning for Eye Tracking

Researchers in the field have been tirelessly working to accurately determine what individual differences in eye movements may be and what physical or cognitive processes underlie them. Experimental designs can be categorised into having a set of properties which help quickly understand a paradigm. For one, a distinction is made between explicit and implicit tasks. In explicit tasks, individuals perform a set of precise and fixed patterns of eye-movements within a pre-determined space. Key low-level characteristics, such as fixations, smooth pursuit, and saccadic information, are then used to identify an individual (Katsini et al., 2020). In implicit tasks, individuals freely view a stimulus and inference can be done based on, for example, gaze-behaviour (Biedert et al., 2012) and scan-patterns (Buswell, 1935). A further distinction can be made between a controlled versus uncontrolled task, and a dynamic versus static task. In a controlled task, the visual task consumes the full attention of the individual. Conversely, an uncontrolled task would be, for example, traversing an environment or browsing the web freely. A dynamic task would involve moving targets, whereas a static task is non-variable. Bargary et al. (2017) performed a controlled, static, explicit study with over 1000 healthy young adults. They were able to identify individuals approximately 60% of the time in a subsample of retested participants (N=104) based on their position in a multidimensional space created by 18 crafted saccade features. They also found that over 90% of the classified users were in the top 10 closest predictions. They further indicate that the rich variation in eye-movements between individuals is poorly understood and that many differentiating factors are likely to be associated with this variation. Differences can come from implicit origins, such as central decision-making processes (Carpenter et al., 2009), but also more explicit origins, such as muscle tissue and differences in motor control (Komogortsev et al., 2012).

Existing methods of user classification with eye-movement data using deep learning are scarce. Li & Chen (2018) found that there are distinct individual differences in both low- and high-level abstractions of eye movement. They believe that deep features learnt by neural networks, or latent representations, are valuable for saliency mapping. While some deep features are valuable for the distinction between individuals, they also found that some features are too similar across individuals. This suggests that there exists a learnable set of features that are valuable for understanding human gaze behaviour in general. However, this set contains both usable and unusable features for the classification of individuals. This provides a balancing challenge in the training of neural networks: general understanding of human gaze behaviour, or eye tracking, versus individual distinguishability.

Jäger et al. (2020) also attempted to tackle the issue of biometric identification using learned latent representations of eye movements. They demonstrated that the addition of hand-crafted features, such as fixational or saccadic features lead to inferior performance compared to using latent features alone. This begs the question how effective and predictive these metrics are in conventional (fixation and saccade based) eye tracking studies. Based on a series of explanatory experiments, Jäger et al. (2020) indicate that the information contained in manual heuristic categorisation, i.e., classifying the types of fixations, contributes very little to the performance of their deep neural network. Rather than high-level features, low-level features, such as involuntary micro-movements provide valuable insight into to individuality. This suggests that raw data, including low-level information, such as angular velocity of eye movements, contains valuable information for identification purposes. Feature extraction/crafting provides noise reduction, but at the cost of potentially useable information, leading to lower performance. Lastly, they demonstrate that deep learning can lead to a significant speed increase compared to traditional methods. Most traditional works in biometric identification make use of sequences of data that are of the considerable length of around a minute. However, Jäger et al. (2020), show 91.4% accuracy in a 75-class problem after only one second of data, climbing up to 99.77% when using 10 seconds of data. The authors do not comment on time saving from the perspective of the researcher.

This deep learning-based approach shows promising results. Not only is classification much quicker, reliant on less data and feature crafting, it also far exceeds the traditionally achieved accuracies. However, the approach suffers from serious drawbacks, the most pressing of which is the opaque decision making of neural networks. While Jäger et al. (2020) perform some explanatory tests, the used dataset by Makowski et al. (2019) demonstrates a commonly

occurring key vulnerability: The excellent classification results are achieved when classification is done based on a ‘leave one out’ paradigm, meaning that classification is done on a single unseen segment of data. This problem of session bias is somewhat circumvented by collecting training and testing data on separate days, or at least in separate sessions. Jäger et al. (2020) show that removing the session bias by classifying on a different session reduces the accuracy by nearly 20% in a binocular setting and nearly half in a monocular setting. Increasing the number of sessions to three allows for the accuracy to increase, though not match the performance that includes session bias. This clearly demonstrates the uncertainty about the basis of classification and begs the question whether deep learning-based methods are really outperforming classical approaches based on feature crafting.

A recent work by Rolff et al. (2022) introduces a pipeline that utilises transformer-based deep learning for the prediction of gaze angle in virtual environments. The impact on performance of a total of five different pre-processing methods is evaluated. As input data for the network, they use the following four measures: horizontal and vertical gaze position (in degrees), head velocity, and several of the final rendered frames of their virtual environment. These input data are combined with early fusion, meaning that they are fused prior to injection into the transformer. Alternatives to early fusion are intermediate and late fusion, which are described by Dolmans et al. (2020). Rolff et al. (2022) propose five pre-processing methods, after each of which the fused data is encoded by a small MLP network and fed to the transformer. Results indicate that the devised pre-processing methods either contribute very little or even reduce performance compared to directly feeding the fused data to the transformer. Their implementation that does not make use of pre-processing outperforms the state of the art when evaluated on the mean angular error of gaze.

The discussed work of Rolff et al. (2022) confirms the findings of Li & Chen (2018) and Jäger et al. (2020): raw, time-series, gaze data is favourable. Furthermore, a transformer architecture that ingests and predicts raw gaze is achievable. However, to the best of our knowledge, no work that predicts raw gaze currently exist. Lastly, the opaque nature of latent spaces does not lend itself to human-interpretable decision making, an ever-present issue in deep learning (Salahuddin et al., 2022).

3.2 Sequence Modelling

Yang & Molano-Mazón (2021) discuss the future of recurrent neural networks (RNNs) in cognitive neuroscience, because of their ability to work with time series data. They argue that

an approach ought to be found which strikes a balance between functionality and biological realism. Concretely, they propose modelling distinguishable biological functions through connecting several smaller RNNs to tackle issues such a varying temporal dimensions, or varying modalities across tasks. Furthermore, this multi-network approach allegedly helps prevent the issues of vanishing and exploding gradients. When moving to a set of tasks that differs from the trained tasks, there is a distributional shift in the data, as well as the evaluation of the network. This shift will likely lead to vanishing or exploding gradients (Ma et al., 2022). The reason for this is that during training, backpropagation updates the weights of the network on each iteration. The size of the update is proportional to the weight that is to be updated. When the partial derivative of the error function with respect to the current weight is vanishingly small, the weight cannot be meaningfully updated (Basodi et al., 2020). On the other hand, exploding gradients occur when the partial derivatives of the error function with respect to the weights become very large. This can cause the weights to update with excessively large values, leading to instability and convergence issues during training. The occurrence of this is thus when a variety of tasks is done interspersedly, or when moving from unimodal to multimodal tasks. RNNs can be resilient to gradient issues (Ribeiro et al., 2020) and they can be functionally separated (Yang & Molano-Mazón, 2021).

However, the gap between modalities and cross-modal communication is not bridged by the framework that Yang & Molano-Mazón (2021) outline. Furthermore, RNNs demonstrate scalability issues. The strength (and weakness) of RNNs lies in the recurrence relations. The strength is that temporal dependencies can be captured through recurrent operations (Chung et al., 2014). The weakness is that the formulation of such temporal dependencies is reliant on sequential operations. In other words, operations must be performed in order and are therefore not parallelisable over time. A recent work by Peng et al. (2023) aim to combine the effective temporal recurrence capturing of RNNs with the parallelisability of transformer networks by introducing the receptance weighted key value (RWKV) architecture. The architecture introduces a linearly scaling attention mechanism, which efficiently operates on modern hardware, leading to the first non-transformer architecture that was scaled to tens of billions of parameters. More research on the effectiveness of this architecture is required.

3.3 Embedding

The attention mechanism of transformers allows for the excellent performance. However, this attendance comes at a heavy computational cost, sometimes even for individual modalities,

such as vision, which relies on a raster of pixels. At high resolutions, sequence lengths can quickly scale out of hand if no truncation is done. Therefore, embedding, that is, the conversion from raw data to usable tokens, is an essential step in processing various types of data for use in machine learning models. It involves breaking down data into smaller, representative units or tokens, which can then be processed efficiently by the model. In the context of different data types, embedding methods vary, adapting to the unique characteristics of each data modality. Embeddings should capture the underlying structure and semantics of the data while preserving sufficient information for downstream tasks. This requires balancing between the granularity of the tokenisation and the computational complexity associated with processing high-dimensional embeddings. Furthermore, finding the optimal tokenisation method and embedding size for a given task can be a complex and data-dependent process. Despite these challenges, converting data from raw to latent embeddings using learned networks has proven to be effective in various applications, such as language and vision (Dosovitskiy et al., 2020; Ryoo et al., 2021; Sennrich et al., 2015).

Embedding data in various modalities is non-trivial. In language modelling, tokenisation typically involves dividing text into words, subwords, or even characters, traditionally based on the Byte Pair Encoding (BPE) algorithm as introduced by Gage (1994). More recently, sophisticated algorithms that leverage statistical information to identify frequently occurring subword units have been employed for the learning of efficient token representations of language (Sennrich et al., 2015). The latter approach is essential for handling out-of-vocabulary, or previously unseen, words and improving the model's ability to generalise across languages and text domains. In vision, embedding becomes problematic because of the high dimensionality of the data. One proposed solution relies on reshaping an image into a sequence of 2D patches and tokenising said patches (Dosovitskiy et al., 2020). Alternatively, there exist hybrid approaches in which an initial network is used to embed the tokens. For example, a CNN can be used to embed the tokens into a latent space, this embedding can then be attended by a transformer head (Song et al., 2022). In multimodal settings, each modality can be embedded by a separate learnt embedder class.

Tokenising time-series data requires a different approach, as the data consists of sequential observations measured at regular intervals. In video, a common method is treating individual frames like images and embedding them into a fixed-size vector as a whole (Dosovitskiy et al., 2020). Alternatively, sliding window segmentation can be done, where a fixed-size window is moved across the time series, and each window's data is treated as a separate token

(Bagnall et al., 2017). This results in spatiotemporal features from a sequence of frames, represented in tokens. The latter approach results in a more compact and expressive representation for videos (Feichtenhofer et al., 2019; Ryoo et al., 2021).

Embeddings can be learnt at scale for a specific domain and can be fine-tuned or adapted to specific tasks or domains, improving their versatility and applicability across a wide range of problems. Several pre-trained language embedders are available, such as ELMo (Peters et al., 2017) as used in the BERT model (Devlin et al., 2018). The benefit of using pre-trained representation is the reduced need of labour-intensive engineering for task-specific architectures. Once the effectivity of an embedder has been verified, the weights can be frozen such that it becomes deterministic. This improves reproducibility and consistency, giving one more control over other parameters that determine network performance. A drawback of learnt embedders is that they are subjected to the same calculated loss as the rest of the network. This means that the evaluation and thus the backpropagation that adjust the embedder's weights is highly dependent on the task used to train the embedder. If the task does not line up well with the intended use of the embedder, this may lead to issues.

Finetuning pre-trained embedders can offer a solution to the issue of suboptimally aligned embedders. An embedder can be prepended to a neural network, with a small adapter layer (Pfeiffer et al., 2020). Instead of having to adjust all weights of the entire embedder network, only the small number of parameters of the adapter are adjusted. This is referred to as parameter efficient fine-tuning (PEFT) and can be done with low-rank adapters (LoRA), which makes use of matrix decomposition to reduce workloads (Hu et al., 2021). Open-source communities, such as HuggingFace (Hugging Face, 2023) leverage the dynamic environment of modern AI by taking on the challenge of large pre-trained models to specific tasks with limited hardware. This allows for increased flexibility in the usage of large pretrained models without the need for high-cost hardware. Thus, pretrained embedders with small adapters are identified as a promising development for multimodal research.

3.4 Summary

In conclusion, idiosyncrasies in eye-tracking have been extensively studied and have been shown to correlate with both low and high-level features. However, the challenge lies in distinguishing personalised saliency from the overall saliency in the performance of neural networks. Transformer based approaches are promising because they show good scalability, domain agnosticism, and effective cross-modal learning through self-attention. Tokenisation

and learning embeddings are essential components in the processing pipeline for transformers applied to diverse data types. By understanding the unique challenges and opportunities associated with each data modality, researchers can develop more effective and efficient transformer-based models, leading to improved performance across a wide range of tasks and domains. Furthermore, modern platforms allow for easy and efficient sharing of (pre-trained) embedding techniques. However, to the best of our knowledge, no such embedder classes currently exist for HFN data. If the abovementioned challenges are properly overcome, the way is paved for the utilisation of foundation models in HFN. This would allow for applications such as screening of MCI associated with AD (Wee et al., 2012), sped-up development of medicine (Acosta et al., 2022), and increased access to high-quality personalised healthcare (Zhang et al., 2023). The possibilities seem endless.

4 On the Current State of Data

To work towards a unified framework of collecting, processing, and publishing eye movement data, it is important to understand the various forms of it. There are many methodologies that are used to measure eye movements, two of which will be highlighted here: Electro-OculoGraphy (EOG), and video-based combined pupil and corneal reflection (PC-R) eye tracking (Duchowski, 2007; Holmqvist et al., 2022). EOG is a more traditional approach in which electrodes that are placed around the eye measure the skin's electrical changes as a result of eye movements. This method is still used today, especially in combination with electroencephalography with the purpose of improving signal quality by removing eye movement artifacts (Schlögl et al., 2007). The currently dominant eye tracking method is PC-R. Many modern eye tracking devices, such as those developed by Tobii, work by beaming infrared onto the face of participants and using one or more video cameras capture the reflection on the cornea (Gibaldi et al., 2017). The trackers are calibrated, after which they can output a variety of pre-processed data types. Generally, eye trackers produce "raw" data in several possible formats and a sampling frequency. For example, gaze position, or X and Y coordinates on a 2D plane from one or both eyes. Alternatively, gaze direction, or vectors that describe the line of sight for one or both eyes, can be output (Holmqvist et al., 2022). However, these raw data are rarely published, instead, the undergo processing into high-level analysable chunks such as fixations and saccades. Fixation data is obtained by processing raw eye tracking data into segments that are semantically meaningful and during which the eye is relatively stationary. Saccades are the movements between fixates. Pre-processing the data into this type of data is undesirable for AI methods, as described by (A. Li & Chen, 2018; Makowski et al., 2020; Rolff et al., 2022).

Transformer networks are incredibly data hungry, requiring upwards of 109 examples to reach state of the art performance, depending on the domain (Chowdhery et al., 2022). There exists no dataset for eye tracking of this size. As previously discussed, the largest publicly available eye tracking related dataset is TEyeD, which contains 20 million real-world eye images for gaze estimation, falling several orders of magnitude short (Fuhl et al., 2021). Several other mid-sized datasets are available, such as GazeCapture (Krafka et al., 2016), GazeBase (GB) (Griffith et al., 2021), and Gaze in Wild (GW) (Kothari et al., 2020). GazeCapture uses crowdsourcing to gather data from a total of 1474 subjects that each perform a dot-tracking task (Krafka et al., 2016). GB collects data with the same participants up to nine times,

performing 7 different tasks each time (Griffith et al., 2021). These tasks vary from lower-level tasks e.g., tracking dots and horizontal saccades, to more ecologically valid tasks e.g., video viewing and reading. GW contains eye and head movements, as well as video footage of the tasks: indoor navigation, ball catching, object search, and tea making.

Unfortunately, even though the above-mentioned datasets are of considerable size, each of them datasets report different eye tracking measures: gaze direction, cyclopean eye distribution vectors in a custom coordinate system, and yaw and pitch relative to head position. The structure of the data organisation varies greatly, some datasets are organised by task, whereas others are organised by participant. Furthermore, the data itself is published in different formats. For example, GB is organised in .CSV files per round, per session, per task, per participant (Griffith et al., 2021), whereas GW provides .mat files per participant that contain all data across tasks and provide the relevant labels in structs. This variation in data reporting habits inhibits the field's ability to combine large datasets by requiring significant consideration and work to 'stitch' all of them together. In order to perform this task, one must first carefully decipher the original format. Then, provided metrics in the original format should be converted such that they align with metrics from other datasets. Then, the organisation of the files should make sense from the perspective of using them with the same pieces of code. The difficulty of the latter lies in the varying tasks and goals of the datasets, each requiring different information in addition to the eye tracking. Clearly, a solution to the organisation of eye tracking data is non-trivial.

A crucial requirement for a comprehensive framework is the establishment of standardized metrics and units. Normalizing the data over the input dimension is a simple technique that can help mitigate the problem of one dataset overpowering the rest. By mapping the data to the same range, it becomes easier to compare and combine datasets. However, it is essential to recognize that variability in units and dimensionality still exists. Therefore, efforts should be made to establish a consistent set of metrics and units that can be universally applied across different datasets. This standardization would enable researchers to interpret and compare eye tracking measures accurately. Additionally, assuming some universal agreement, existing datasets can be adapted to the newly established agreement. This would enable use of previously collected data, given that efforts are made to convert the data.

It appears most reasonable to ground the reported data to the stimuli, rather than the orientation of the eyes since eye tracking only gains value when it is understood in the context

of the stimuli being observed. The stimuli, which represent the visual environment, play a significant role in interpreting eye movement patterns. Therefore, a comprehensive framework should include ample information about the stimuli presented during the eye tracking experiments. This information could encompass various aspects such as visual properties, task instructions, and contextual and timing cues. For instance, in dynamic scenes or videos, precise temporal alignment between the eye tracking data and the corresponding frames or events is vital for analysing gaze behaviour accurately. All recorded data should incorporate mechanisms for synchronizing and annotating eye tracking data with the stimuli in a precise, consistent, and automated manner. This would facilitate more robust analysis and enable researchers to investigate the relationship between visual stimuli and eye movements more effectively.

Eye tracking experiments can encompass various domains, such as reading, visual search, scene perception, and biometric identification (Broda & de Haas, 2022; Jäger et al., 2020; Mézière, 2022). Each task may require different additional information beyond eye tracking data, such as behavioural responses, task performance metrics, or cognitive measures. The richer the data, the richer the potential understanding. A multimodal framework can accommodate diverse data types and task-specific information, allowing for comprehensive analyses across multiple domains. By combining stimuli-related data with eye tracking data, researchers, as well as trained AI, can gain a more comprehensive understanding of visual attention and gaze behaviour. This understanding can then be distilled into a generalised understanding of human gaze, which may then be applied to novel stimuli and environments. Fixation datasets are often organised by participant, by stimulus, by fixation. Several options for the conversion to tensors are available and the selected organisation depends on the hold-out paradigm that is used for the training and testing of the neural network.

4.1 Selected Data

Several datasets that can be used were identified. Generally, the available datasets fall in either of two categories: raw, and fixation data. Table 1 provides an overview of the authors, task, number of participants, and the estimated number of usable samples for each dataset, as well as the type of data. In the methods section, a generalised approach is discussed for the use of one the two dataset types. GazeBase (Griffith et al., 2021), referred to as GB, is used for the validation of the pipeline for the raw data variant. In principle, the process described to

integrate GB into the MBT pipeline applies to datasets that consist of raw eye tracking data. Furthermore, the process for the integration of fixation datasets is also described.

GB (Griffith et al., 2021) provides raw monocular data of 332 participants performing seven different tasks, sampled at 1000Hz. Participants participated in up to nine sessions, spanning 37 months. Collecting data in multiple different sessions with the same participants is valuable for the identification and elimination of session bias. The tasks in GB vary from lower level e.g., tracking dots and horizontal saccades, to more ecologically valid tasks e.g., video viewing and reading. Data in the OSIE task as described in Table 1 contain fixation data, images used as stimuli, as well as semantic labels of said images.

Table 1: Overview of currently available datasets and their estimated number of usable samples. OSIE = Object and Semantic Images and Eye-tracking and contains 700 images and their respective semantic labels. Various = seven tasks spanning lower level e.g., tracking dots and horizontal saccades, to more ecologically valid tasks e.g., video viewing and reading. Comments indicate the stage of training where the data can be used.

Authors	**Task**	**N**	**Samples**	**Comments**
de Haas et al. (2019)	OSIE	117	~245,700	Fixation, Finetune
Broda & de Haas (2022)	OSIE (and videos)	44	~92,400	Fixation, Finetune
Linka & de Haas (2020)	OSIEshort	103	~278,100	Fixation, Finetune & Test
GazeBase, Griffith et al. (2021)	Various	322	~250,000	Raw, Pretrain
Total		**755**	**~796,000+**	

4.2 Data Cards

Google Research published a guideline and format for “Data Cards” (Pushkarna et al., 2022). As stated by the authors, the purpose of data cards is to provide a clear and thorough understanding of the dataset’s creation, curation, purposes, considerations, and maintenance. Whereas a full-fledged data card may be overkill for many small datasets, the creation of such data cards will facilitate authors in their considerations. Data cards serve as a valuable resource for both data creators and data consumers. For data creators, the process of developing data cards encourages a thorough examination of various aspects related to the dataset. This includes considerations such as data collection methodologies, pre-processing techniques, quality assurance measures, and ethical considerations. By documenting these details in a structured manner, data creators can ensure transparency, accountability, and reproducibility in their research endeavours. Data consumers, on the other hand, benefit from the availability of data cards as they provide essential information about the dataset's

characteristics and limitations. Understanding the dataset's creation and curation processes enables researchers to assess its suitability for their specific research questions or applications.

Additionally, data cards highlight any potential biases, anomalies, or data quality issues that may impact the validity and reliability of the results derived from the dataset. Furthermore, the existence of data cards promotes efficient data reuse and repurposing. Researchers often spend a significant amount of time and effort understanding the intricacies of a new dataset before they can effectively utilize it. Data cards can somewhat alleviate this burden by offering a structured overview of the dataset, saving researchers valuable time and resources. By providing a standardized format for presenting essential information, data cards facilitate seamless integration and comparison of different datasets, leading to more comprehensive analyses and meaningful insights. Moreover, data cards contribute to the broader goals of data sharing and collaboration. The availability of data cards enables researchers from diverse backgrounds to explore and build upon existing datasets, leading to new discoveries and advancements in various fields of study. Furthermore, it allows for the bundling of several datasets for the purpose of training AI.

In conclusion, a comprehensive framework for eye tracking data sharing must address not only the issues of inconsistent measures, data organization, and formats but also several other requirements. These include standardizing metrics and units, incorporating stimuli-related information, establishing accurate correlations between stimuli and eye tracking data, and accommodating the diverse range of tasks and stimuli encountered in the natural world. Lastly, the framework should have multimodality at its core. By meeting these requirements, the field can overcome the challenges associated with combining large eye tracking datasets and foster more effective collaboration and knowledge advancement in the future, as well as retroactively. Data cards may provide both a framework for standardisation, as well as retrofitting existing data to a more modern standard.

5 Methods

In this section, the methods are discussed. First, the available and used data will be discussed. Then, the designed network will be discussed in detail. In addition to the MBT, an MLP is implemented to serve as a baseline for comparisons. Then, the training, finetuning and testing are discussed. It is recommended to keep the GitHub repository on which all code is published handy while reading the MBT Implementation section, as several references will be made to it [https://github.com/tcdolmans/Eye4AI] (Eye4AI, 2023). As stated in the Goals of this work, this section contains recommendations for the implementation of an MBT in HFN. However, no meaningful results are to be expected given that there simply is not enough data.

The proposed MBT implementation can operate in two modes: classification and prediction. In classification mode, the network will seek to classify which participant a sample of eye-tracking data belongs to. The network is provided with all available modalities, such as eye tracking as well as stimuli, and is tasked the prediction of the participant label. In other words, given some unseen stimulus, can the eye tracking data be connected to a specific participant? The second mode is prediction, in which the network is provided with the participant label, as well as the stimulus, but crucially not the eye tracking data; it is tasked with the prediction of eye tracking data. In other words, given a known participant and an unseen stimulus, can the gaze behaviour be predicted? The predicted gaze's format depends on the input format and can be reconstructed to visually interpretable data.

All experiments were done on a Lenovo Legion 5 Pro (Intel) with the following hardware: 12th Gen Intel® Core ™ i7-12700H, 16.0 GB RAM, with an NVIDIA GeForce RTX 3070 Laptop GPU.

5.1 Datasets and Pre-Processing

For present purposes, the data of GB have been organised in tensor files that are organised by round (a total of nine rounds), session (two sessions per round), and task (seven tasks per round), resulting in a total of 126 tensor files. Each of the files contains data from all participants for that specific task. The data has been sanitised by checking for validity of observations at a threshold of 85%. If the proportion of observations to NaNs is below the threshold, the data is rejected. If the proportion of observations is above the threshold, NaNs are replaced by values that are linearly interpolated between valid samples. After sanitation, the data are downsampled by a factor of 10 by finding the mode of 10 observations and

iterating over the total length of the sample. This is done by the `replace_nans` and `downsample` functions in `pipeline/convert_sets.py`, respectively (Eye4AI, 2023). Other raw datasets can be treated in similar ways by organising the data sample-wise. This structure adheres to the data principles relevant for AI. Practically, a tensor-based storage is used since this method allows for flexible use of the data by allowing for easy access to and grouping of samples. Access meaning the quick and easy loading of tensors to memory for training, testing, or manipulation. Grouping meaning the quick and easy reorganisation of the tensors for e.g., participant or task-wise holdout methods.

In order to make use of the second type, fixation data, there are several options. Frist, an embedding method can be applied to the fixation data directly. Second, they can be converted to a "raw" signal by reconstructing the intra-fixation gaze behaviour, so as to simulate saccades. During fixations, slight noise can be added to the datapoints, to simulate realistic gaze characteristics. The reconstructed data is then embedded. The former option is favourable for the sake of simplicity. Furthermore, the reconstructed information might not contain valuable information for the classification and prediction of eye tracking data since the data is not based on reality but rather simple assumptions about reality. Nonetheless, these 're-rawing' methods may be necessary to prevent exploding and vanishing gradients. Regardless, this means that only high-level information about gaze behaviour can be used for classification and prediction from fixation datasets; lower-level information, such as saccadic (de)acceleration and micro vibrations, are simply not present.

As an example case for the inclusion of fixation-based data, the OSIE paradigm is selected. This paradigm uses images and stimuli and provides semantic labels for every image. Image names range from 1001 to 1700, totalling 700 images and they are stored in .jpg format. Pillow is used to load the images, then they are converted to a PyTorch Tensor of size `3,600,800` and stacked in a tensor of size `700,3,600,800` (Pillow, n.d.; PyTorch, n.d.). This tensor is kept in memory and sliced by selecting the relevant image based on the sample. To integrate OSIE's visual stimuli, two approaches can be used. First, a pretrained network, such as ResNet (He et al., 2015), can be used to create embeddings by using the penultimate layer's output. The benefit to this approach is that ResNet has a validated performance on visual tasks. Alternatively, a custom image embedder can be trained for the OSIE set. The benefit to this approach is that the learned embedder is more specialised for OSIE. A generic visual classifier may not be optimal, since it is not specifically trained for the OSIE task, but for image classification in general.

The semantic labels of the OSIE paradigm are organised in twelve dimensions, being the presence of faces, emotions, text, operable objects, etc. For each semantic dimension, a pixel-wise mask is constructed that indicates the presence of the respective dimension. Multiple dimensions can be flagged present in any given pixel e.g., a face in addition to an emotion. The `pipeline/Osie.m` file combines each of the twelve semantic dimensions into a single matrix of size `12,600,800`. In similar fashion to the images, these semantic matrices are stacked into a single tensor of size `700,12,600,800` which is sliced by indexing for stimuli.

5.1.1 Dataloaders

After conversion to the desired organisation of tensor files, they can be made available to the MBT for training and testing. In the current implementation, PyTorch's Dataset and DataLoader are used. (PyTorch, 2023). First, a Dataset is constructed, this creates a map that connects keys to data samples. For raw data, such as GB, we then further split the tensor files into samples of the desired length that is passed as an argument. For fixation data, individual fixations or collections of fixations can serve as samples; function arguments can be passed to determine the number of fixations, which is subject to hyperparameter optimisation. When the dataset is constructed, it is converted to a DataLoader, which combines the dataset with a sampler and provides an iterable with which the Dataset can be sampled per item or batch of items. A split is made between training and testing data, which is available to `train_dl` and `test_dl`, respectively. Please refer to `models/dataloaders.py` for details.

5.2 MBT Implementation

In this section, the implementation of a multimodal bottleneck transformer (MBT) is described. Learning multimodal representations gives a more complete understanding of the environment and its predictors and thus leads to better classification; the MBT leverages information from one modality to assist in the comprehension of another (P. Xu et al., 2022). The MBT is designed to work with the following modalities: eye tracking (referred to as `ET`), images (`img`), and semantic labels of said images (`sem`). However, extensibility is kept in mind and the implementation allows for easy addition of modalities.

5.2.1 Encoder

Building on the earlier-described transformer blocks, we will now describe the `encoder` architecture in our implementation of an MBT. First, a positional encoding is added to the

constructed embedding of the input data for each modality. This ensures that the attention mechanism is able to use information about the location and order of things in the input sequences. Transformer networks have no way of determining the absolute or relative position of tokens in a sequence inherently, therefore, it is essential to inject such information into the embedded data. Vaswani et al. (2017) indicate that the position embeddings can either be learned or fixed, with very little effect on their performance. For the sake of simplicity, we follow their implementation, which summates an index-based sine and cosine value for even and odd indices, respectively. Second, for each modality, an encoder is initialised. The operations of the transformer blocks are performed L times, where L is the number of layers.

On the basis of Nagrani et al. (2021), we introduce bottleneck tokens to the `encoder`, allowing for limited communication between the transformer blocks of each modality. When the `forward` method of the `encoder` has reached the fusion layer (L_f), the bottleneck tokens are concatenated to the output of the previous layer. L_f then performs operations as normal, after which the bottleneck tokens are sliced from the outputs and `bottleneck` is updated and appended to the inputs for L_{f+1}. As such, the inputs for layer L_{f+1} are the outputs of L_f + *bottleneck*. For an intuitive understanding of this process, please refer to both Figure 4 as well as the `models/mbt_encoder.py` file in the GitHub repository (Eye4AI, 2023).

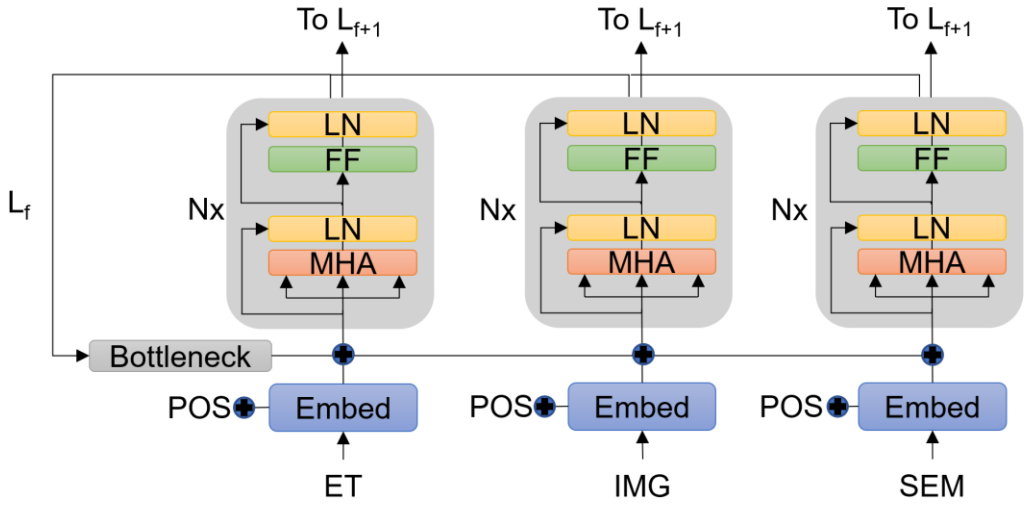

Figure 4: Multimodal Bottleneck Transformer Encoder. ET = Eye tracking, IMG = image, SEM = semantic labels, these serve as input to; Embed = embedder classes; POS = positional embedding; Bottleneck = learnable bottleneck tokens; MHA = multi-head attention, described in Figure 1; LN = layer normalization, FF = feed forward; Nx = number of layers; Lf = fusion layer. The bottleneck tokens are concatenated to embedded data and fed through encoder blocks if N = Lf. Then, the bottleneck is sliced from the outputs and updated, to be concatenated for layers N > Lf.

5.2.2 Decoder

The current implementation of the MBT uses PyTorch's standard implementation of a `decoder` (*TransformerDecoder — PyTorch 2.0 Documentation*, n.d.). It receives the outputs of the `encoder`, as well as a position-encoded target sequence, which is subject to prediction. As previously described, in order to make sure that the `decoder` does not attend future time points in the sequence as a basis for prediction, a mask is applied which renders them unusable until they are permitted to be used. The `decoder` contains the number of layers L as the `encoder`. In future versions of the MBT, experiments can be carried out with the goal of discovering whether a custom multimodal decoder with bottleneck contributes to the task. The prospect of this is further explored in the Discussion section.

5.2.3 Embedders

As discussed in the Embedding section, in order to properly use input data in transformer networks, input data must be embedded into a different representation. The embedding methods for each data type will be discussed, the code for the embedder classes can be found in `models/embedders.py`.

In the GB task, `ET` is split into of samples by the `GazeBaseDataset` dataloader that have size `4,300`, where the columns are the timestamp, x-coordinate, y-coordinate, and a pupil size measure. The length of the sample is set to three seconds of data, corresponding to 300 observations after downsampling by a factor of 10. The splitting into sections of three seconds is done to conform to the stimulus presentation length in the OSIE task; Pre-training the network on the same length of inputs will aid the consistency of the pipeline. In the OSIE task, `ET` samples are of size `4,300`, where the columns are the timestamp, x-coordinate, y-coordinate, and a pupil size measure.

All `ET` samples are fed to the `ETPatchEmbed` class which sequentially performs several operations that are selected from commonly used eye tracking processing techniques (Makowski et al., 2021). In order of operation: A 1D convolution, a ReLU activation function, 1D adaptive average pooling, and finally, a layer normalisation. The output dimensions of the class depend on the selected hyperparameters for embedding dimension and kernel size, which are subject to optimisation. The optimisation of all hyperparameters is discussed in the Hyperparameters section. Similar to `ET`, `img` is processed with a 2D convolution, a ReLU activation function, followed by 2D adaptive average pooling and a

layer normalisation in the `ImagePatchEmbed` class. Samples of `img` are of size `3,600,800`. 2D convolutional layers are commonly used for image processing and the thereafter following steps are kept the same as `ET` for the sake of consistency. As previously discussed, a pretrained ResNet model can also be implemented for the embedding of images (He et al., 2015). The processing of `sem` is identical to that of `img`, barring the input dimensionality of the data: `12,600,800`. It takes place in the `SemanticPatchEmbed` class.

5.2.4 Modes

The network operates in two different modes to achieve the various goals of this research. These modes are `classification` and `et_prediction`. In `classification` mode is provided with `ET`, `img`, and `sem`, and is tasked the prediction of the participant label `p_num`. In this mode, the loss is calculated through cross entropy (CrossEntropyLoss — PyTorch 2.0 Documentation, 2023). The second mode is `et_reconstruction`, in which the network is provided with `p_num`, `img`, and `sem`, and is tasked with the prediction of `ET`. The predicted `ET` can be reconstructed to visually interpretable data. In this mode, the loss is calculated using the mean squared error between the prediction and the target (MSELoss — PyTorch 2.0 Documentation, 2023).

5.2.5 Head Networks

The MBT can select between two head networks that are used for the two modes. In `classification`, the head consists of a single linear layer that essentially converts the decoder logits to the number of classes as indicated in `config`. The outputs of this layer are directly fed into the loss function. In `et_prediction`, the head consists of a single linear layer that converts the decoder logits to a matrix the size of the missing snipper of `ET`, which is also governed by `config`.

5.2.6 Hyperparameters

Several hyperparameters can be searched in order to optimise model performance, which will be discussed in this section. They are summarised in Table 2. Optuna is used to automatically search over possible combinations of hyperparameters (Optuna, 2023). An objective function is defined `n_trials` times in each of which the model is initiated based on a configuration dictionary. `config` contains all parameters that are set for the network, from embedder settings, to current hyperparameters. `objective` further trains and tests the model based on

config. If a trial is unpromising according to Optuna, it is pruned. If the results of the trial are the best so far, the checkpoint of the model is saved, and the best parameters are returned.

Table 2: Overview of hyperparameters that are searched. For each hyperparameter, the symbol, description, and searched over range are provided.

Symbol	Parameter Description	Value range
l	Number of layers (encoder and decoder)	{2, 8}
h	Number of heads (MHA)	{8, 16}
FE	Forward expansion factor in transformer blocks	{2, 8}
d	Dropout rate	{0.1, 0.5}
α	Learning rate	{1e-5, 1e-3}
$batch$	Batch size	{64, 256}
B	Number of bottleneck tokens	{4, 16}
L_f	Fusion layer	{2, 8}
N_e	Number of epochs	{50, 200}
$L2$	L2 Weight decay	{1e-5, 1e-3}
d	Embedding dimension	{40*h, 80*h}

5.2.7 Trainer Flow

This section describes the order of operations of the models/trainer.py file. First, the mode and task are set, from which a DataLoader will be constructed. Optuna is called and the toolbox suggests hyperparameters to initialise the model with, which are stored in config. To initialise the model, config is passed as an argument. Training is started with the hyperparameters that are selected by Optuna. As previously discussed, the passed data depends on the mode. In classification, the p_num is withheld from the input data. Then the loss is calculated using CrossEntropyLoss over the difference between the model output and the true p_num. Conversely, if the model is in et_prediction, it is provided with only a snippet of ET with the objective of predicting the remaining part of the original ET sample. The length of the snippet is adjustable and can also be considered a hyperparameter. The loss is then determined by calculating the MSELoss over the predicted ET versus the original ET.

The MBT flexibly ingests a dictionary of data from various modalities: x. Currently supported modalities are described in the embedders section; adding additional modalities is relatively straightforward with very few network adjustments. The present modalities in x are embedded by the respective embedders, after which the time dimension is adjusted to ensure compatibility. The time dimension is governed by config and should therefore be consistent, the adjustment is thus done as a sanity check. From the embedded sequences, a source and

target sequences are constructed, `src` and `tgt`, respectively. `src` is the original sequence of tokens retrieved from the embedders. `tgt` is the original sequence of interest, in our case `ET`, offset by one token and preceded by a start token to indicate the start of the sequence. After this, `src` and `bottleneck` are passed to the `encoder`, which returns a dictionary with logits for each of the modalities. The logits for `ET` are extracted and passed to the `decoder` along with `tgt` and a mask that prevents the network from attending future tokens. The output logits from the decoder are passed to the correct head network for the mode. During training, data is provided by `train_dl`, the loss is evaluated, and the weights are updated with backpropagation. When the training is completed, the weights are frozen and `test_dl` provides samples for testing.

5.3 MLP Baseline

As a baseline for the performance, a densely connected MLP is constructed and tested for unimodal embedded inputs of `ET`. The implementation consists of four linear layers that are stacked with ReLU activation functions. After the final linear layer, a dropout layer randomly drops a portion of the connections. The portion of dropout is defined as a hyperparameter. For the precise implementation, please refer to `models/MLP.py` in the GitHub repository (Eye4AI, 2023). The MLP baseline is only tested in `classification` mode.

5.4 Evaluation

During testing, GazeBase will be evaluated (Griffith et al., 2021). This set is beneficial since data for participants is collected on nine different occasions, which allows for the elimination of session bias. Furthermore, it provides raw gaze. Data from the first seven sessions are used as training data, the final two sessions are used for testing. Participants that made it to the last round are present in all rounds, leading to improved session bias elimination (Jäger et al., 2020). The performance of the model will be evaluated by the following metrics:

1. Classification accuracy in `classification` mode: match the correct individual to the label. This is tested on a top-k basis, referring to the top k likely predictions. When k=5, if the correct participant is in the top 5 most likely predictions, the sample is considered correct.

   a. GB contains data of 322 participants, therefore, random chance for k=5 gives an accuracy rating of 0.0156, or 1.56%. Always guessing the same participant (contained in the testing set) gives an accuracy rating of 0.322, or 3.22%.

2. Prediction accuracy in `et_reconstruction` mode: MeanSquareError is calculated over the predicted gaze (*MSELoss — PyTorch 2.0 Documentation*, n.d.). Furthermore, by treating the prediction of the model as though it were ET, the 2D timeseries can be visualised and further evaluated visually.

6 Results

In this section the preliminary results are discussed. We emphasize that the expected number of samples required for optimal performance of a network of this size is several orders of magnitude larger than the number of samples that are available. Therefore, this section serves more as an indication of how results can be reported, rather than as a section containing meaningful results.

6.1 MLP Performance

The MLP performed as follows on the classification task: Loss: 4.498 with a Top-5 accuracy of 0.2377, or 23.77%. This was achieved with the following parameters: {dropout: 0.5, learning rate: 1.45e-4, batch size: 256, number of epochs 4, L2: 9.68e-4, embedding dimension: 144}

6.2 MBT Performance

The MBT performed as follows on the classification task: Loss: 4.764 with a Top-5 accuracy of 0.1426, or 14.26%. This was achieved with the following parameters: {number of layers: 4, heads: 8, forward expansion: 4, dropout: 0.5, learning rate: 7.58e-4, batch size: 128, number of bottlenecks: 16, fusion layer: 3, L2: 1.88e-4, embedding dimension: 128}.

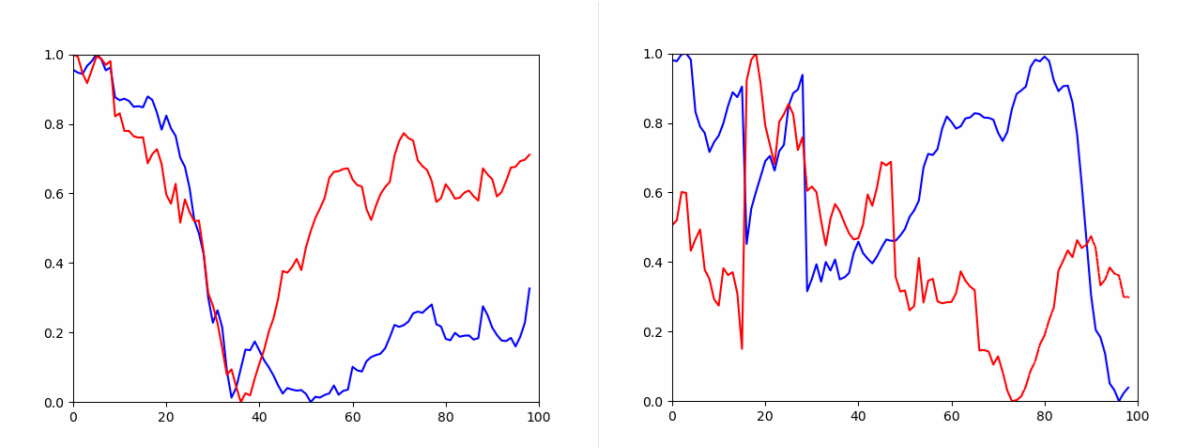

Figure 5: Visualisation of randomly selected sample of data from the GB task (Griffith et al., 2021). Left: True sample of gaze. Right: corresponding prediction of gaze Blue visualises the x-coordinate over time. Red visualises the y-coordinate over time. The y-axis is normalised so that all values fit beteween zero and one. The x-axis corresponds to the sample number, 1 per millisecond.

In prediction mode, the loss remains meaninglessly high since the evaluation is done over a matrix of size (batch size, snippet length, number of columns). However, the predictions can be visualized. Figure 5 visualises a single randomly selected sample of length 100 with the corresponding prediction of the network. Figure 6 visualises the errors between various measures.

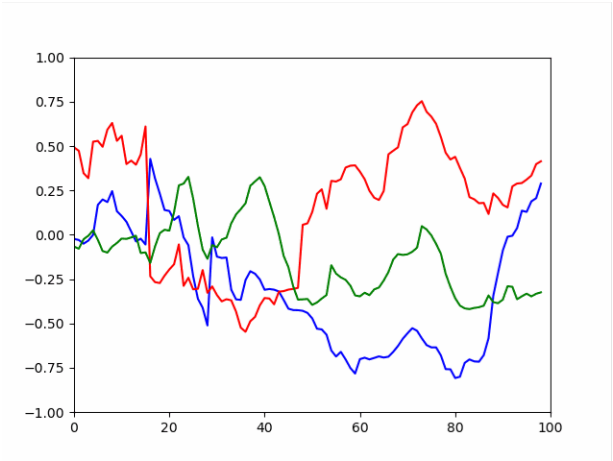

Figure 6: Differences between true and predicted gaze. Blue visualises the error in x-coordinate over time. Red visualises the error in y-coordinate over time. Green visualises the error in pupil diameter over time. The y-axis is normalised so that all values fit beteween zero and one. The x-axis corresponds to the sample number, 1 per millisecond.

7 Discussion

We set out to create an example case that meaningfully and concretely contributes to the standardisation of data organisation in eye tracking, making use of multimodal AI. Through the implementation of the MBT, we believe that important insights were gathered. First, it seems possible that accurate classification of individuals and prediction of their gaze can be achieved, given more data. Performance of the MLP baseline and MBT implementation reached 23.77% and 14.26%, respectively. This is significantly above random guessing: 1.56%; and always guessing the same participant (contained in the testing set):3.22%. We provide a clear pipeline for the data preparation in our methods. Furthermore, recommendations are made for the format of storage. Lastly, the provided infrastructure allows for easy addition of multiple modalities. However, not without limitations and drawbacks, which are reflected on in the following section.

7.1 Limitations

The main limitation of the present study is the lack of concrete results due to data and scaling issues. The availability of data plays a crucial role in assessing the performance of any network effectively. In our case, there is not enough data available to meaningfully evaluate the proposed network's performance. Insufficient data can lead to unreliable or inconclusive results, making it challenging to draw robust conclusions about the effectiveness of the proposed approach. Furthermore, several large issues that are currently out of scope may present themselves when enough data becomes available. However, the MBT demonstrates GM properties as proposed by (Dolmans et al., 2021). Nonetheless, it remains unclear whether AI for eye tracking is viable enough to replace handcrafted features and simple classifiers. The lack of clear standards and benchmarks make this consideration difficult. Once more data becomes available, evaluation of the models can and should be further expanded in order to achieve better results. Better evaluation methods also aid communication about results and shortcomings. Due to the lack of a realistic baseline performance, it becomes challenging to validate the effectiveness of the proposed method against existing approaches or benchmarks. Without a baseline for comparison, it is difficult to assess whether the introduced modifications or techniques truly contribute to improved performance. Future research should work towards establishing good benchmarks and comprehensive evaluation of models.

Another shortcoming to consider is the vulnerability of the pretraining process, particularly in the context of multimodal tasks. Multimodal issues, such as mode collapse and vanishing/exploding gradients, have been highlighted in previous research (Ma et al., 2022). As it currently stands, there is no clear solution that guarantees the retention of valuable learnt parameters when the MBT is pretrained on a different set of modalities than it will be deployed in. Failure to devise a solution for this shortcoming would prevent the flexible use of the MBT since it would only be usable in the exact configuration that it was trained in. As highlighted by Zhang et al. (2023), training multimodal networks requires careful balancing of input data to achieve the desired performance. The employed embedders are currently learned and are therefore entirely dependent on the task. While this flexibility allows the embedders to adapt to the task requirements, it also introduces a limitation. GB consists of seven different tasks, the effects of these tasks on the distribution of the eye tracking data is unknown (Griffith et al., 2021). In the future, it might be beneficial to explore the possibility of pretraining the embedders in a verified manner and then freezing them. By pretraining and freezing the embedders, they become deterministic and less sensitive to the specific environment in which they were trained. This should lead to more task-robust systems. This limitation highlights potential directions for future research in enhancing the stability and generalization capabilities of the embedders in multimodal tasks.

The available hardware was also a limiting factor in the present work. Large models require large GPUs and ample RAM to work with large datasets. The used hardware limited the size of the hyperparameter search, leading to suboptimal configurations of networks. This is directly reflected in the performance of the MBT. The MLP could be scaled up to the required size, whereas the MBT could not. This could be a possible explanation for the performance discrepancy. A few hyperparameters that strongly influence network sizes and performance are the number of layers, the number of heads in MHA, and the hidden dimension of the network (`embed_dim`) in the implementation. These parameters determine the depth and breadth of the attention mechanism and are therefore key players in performance.

Based on the visualisations as a result of the prediction mode, working with fixation rather than raw data should be reconsidered for multiple reasons. First, the loss of fixation prediction is evaluated over smaller matrices, likely leading to improved effectiveness of gradient descent since a clearer direction can be identified. Furthermore, the field has primarily produced fixation data, allowing for the incorporation of such data allows for good backwards compatibility. The field is used to working with fixation data and this data is more

interpretable than raw data in some contexts. Lastly, the visualisation of fixation data might make more intuitive sense than visualisation methods for raw gaze. This may not always be the case, in for example, video or dynamic environment-based eye tracking studies.

The selection of a transformer-based model also imposes some restrictions. As previously discussed, the quadratically scaling complexity of transformers quickly renders them unusable for many types of hardware (Peng et al., 2023). Furthermore, even with ample hardware to train and run the models, their use should be considered carefully as this produces considerable energy consumption (Rae et al., 2021). A recent work, proposed by Assran et al. (2023) highlights a new approach for learning highly semantic image representations in a self-supervised manner. The Image-based Join-Embedding Predictive Architecture is the first implementation of the overarching Hierarchical-JEPA efforts by Yann LeCun (Lecun, 2022). I-JEPA for embedding, combined with transformers for vision tasks is highly scalable and efficient Assran et al. (2023).

Autoregressivity, the property of predicting one token from the previous one, severely limits the space in which prediction can be done (Weber & Gühmann, 2021). Transformers cannot plan and look ahead because of this property. A single poor selection of a token forces transformers to keep generating with the token. When asking LLMs to reflect on their response, they will often be able to identify the incorrectness, they simply don’t have to ability to address the issue (Yao et al., 2023). A recent work aims to solve this problem by allowing LLMs to generate several tokens and select paths that seem promising, improving performance significantly (Yao et al., 2023). On the technical side, a concrete logical next step in this research would be to move from image to video content understanding. Such understanding would lead to a plethora of novel applications, ranging from novel research paradigms in which moving stimuli can be used. From there, ecologically valid research can be done by making use of head-mounted eye trackers. Making sense of such unstructured and dynamic environments is essential for the creation of AI-based agents that can navigate and interact with the world independently.

Overall, these limitations highlight areas for improvement in several categories. They also highlight shortcomings of the field as a whole. Addressing these shortcomings can enhance the reliability, scalability, and generalizability of multimodal models, ultimately leading to more robust and effective solutions for various real-world applications.

7.2 Implications

There are various areas in which the present study might have an impact. First, human-cantered developments. With knowledge of individual gaze saliency and preferences, we may be able to develop educational software/user interfaces that adapts to the user by adjusting the pace of information, reducing distractors, or providing a varying degree of guidance. Similarly, productivity improvement may be done with adapting apps by better balancing workload over time and reducing individually specific sets of distractions. Furthermore, it may lead to insights into understanding gaze patterns in ASD, which may then be used in the coaching of individuals. Expanding the model capabilities by including multiple modalities further improves the expected impact. Certain things are not visible in the patterns of small datasets. Similarly, through a lot of data can see things that are not apparent on small subsets of the data. Imagine a doctor that has seen 1000 patients versus a doctor that has seen 100 million patients. Which doctor is more likely to provide healthcare that is specifically tailored? Which doctor is more likely to understand odd edge cases?

Perhaps, the likeliness of an individual to engage in interaction with salient entities in an environment can be estimated based on their visual interaction with the environment. Visual interaction may include the length of time an individual spends looking at a particular object or person, the frequency of looking at that object or person, or the nature of the interaction (such as whether the individual makes eye contact or gestures towards the object or person). By analysing an individual's visual interaction with their environment, researchers or analysts may be able to gain insights into that person's behaviour, interests, and priorities. These insights can then be used to predict behaviour and interaction.

This research also lends itself to applications which require more careful consideration and come with a host of ethical issues, such as advertising and biometric identification. Given a personalised profile of saliency and preferred gaze patterns, advertising, when combined with generative AI, can become extremely personalised and targeted. Conversely, websites may gather usage statistics the gain insight into the organisation and legibility of the contents with the intention to improve the design. This may then lead to increased democratisation of information, since it will be more accessible. Such applications should truly be considered to be double edged swords and should therefore remain under constant scrutiny. Furthermore, it is essential that international cooperation legislates the use of AI. The European Parliament recently adopted a new AI Act, providing support for innovation, but also restriction AI from

being used for specific purposes (European Parliament, 2023). These purposes include, for example, real-time biometric identification systems in publicly accessible spaces, predictive policing systems (based on profiles, location, or past criminal behaviour), emotion recognition systems, and untargeted scraping of facial images to create facial recognition databases.

Then there are philosophical implications: Are we all predictably different? Do we fall in specific categories? What degree of difference in our perception of the world does this lead to? Do we all "pass out our lives in private perceptual worlds" as stated by Mollon et al. (2017)? How do these differences affect our choices, social behaviours (e.g., errors of omission), and preference in life? Can we make meaningful changes in one's preferences by nudging or adapting saliency at a subliminal level? These questions are becoming increasingly accessible with increasingly powerful systems that are based on large scale datasets.

8 Conclusion

The main goal of the present work is the re-directing of gaze towards the new way of doing HFN research with AI. This requires significant efforts on the data organization and publication side; through the given overview and proposal in this work, we have contributed meaningfully towards this goal. We highlight several important works and place them into the literature relevant for eye tracking. A multimodal bottleneck transformer was developed, and results indicated that classification of individuals and prediction of gaze is possible, given more training data. We demonstrate the feasibility of storing and processing eye tracking data for AI purposes and highlight a path for large-scale modern AI-based research in human factors and neuroscience.

9 Acknowledgements

I would like to express my gratitude to those who have supported me throughout the writing of this thesis. In particular, I would like thank Jukka Leppänen and Antti Airola for their excellent supervision, good discussions, and valuable feedback. I would further like to thank Myrthe Tilleman for her patience in discussion with me, as well as her unwavering support and assistance.

ChatGPT (3.5 and 4 backend) and Bing Chat were used to assist with the writing of both text and code. Our interaction appears to be mutualistic.

10 Data and Code Availability

All data that were used in this work were downloaded from the GazeBase Data Repository (GazeBase Data Repository, n.d.). All code is available for viewing and reuse on GitHub (Dolmans, 2023).

References

Acosta, J. N., Falcone, G. J., Rajpurkar, P., & Topol, E. J. (2022). Multimodal biomedical AI. *Nature Medicine 2022 28:9*, *28*(9), 1773–1784. https://doi.org/10.1038/s41591-022-01981-2

Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., & Ballas, N. (2023). *Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture*. https://arxiv.org/abs/2301.08243v3

Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, *31*(3), 606–660. https://doi.org/10.1007/S10618-016-0483-9/FIGURES/17

Bargary, G., Bosten, J. M., Goodbourn, P. T., Lawrance-Owen, A. J., Hogg, R. E., & Mollon, J. D. (2017a). Individual differences in human eye movements: An oculomotor signature? *Vision Research*, *141*, 157–169. https://doi.org/10.1016/J.VISRES.2017.03.001

Bargary, G., Bosten, J. M., Goodbourn, P. T., Lawrance-Owen, A. J., Hogg, R. E., & Mollon, J. D. (2017b). Individual differences in human eye movements: An oculomotor signature? *Vision Research*, *141*, 157–169. https://doi.org/10.1016/j.visres.2017.03.001

Basodi, S., Ji, C., Zhang, H., & Pan, Y. (2020). Gradient amplification: An efficient way to train deep neural networks. *Big Data Mining and Analytics*, *3*(3), 196–207. https://doi.org/10.26599/BDMA.2020.9020004

Biedert, R., Frank, M., Martinovic, I., & Song, D. (2012). Stimuli for gaze based intrusion detection. *Lecture Notes in Electrical Engineering*, *164 LNEE*(VOL. 1), 757–763. https://doi.org/10.1007/978-94-007-4516-2_80/COVER

Bijsterbosch, J. D., Woolrich, M. W., Glasser, M. F., Robinson, E. C., Beckmann, C. F., Van Essen, D. C., Harrison, S. J., & Smith, S. M. (2018). The relationship between spatial configuration and functional connectivity of brain regions. *ELife*, *7*. https://doi.org/10.7554/ELIFE.32992

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., …

Liang, P. (2021). *On the Opportunities and Risks of Foundation Models*. https://doi.org/10.48550/arxiv.2108.07258

Broda, M. D., & de Haas, B. (2022). Individual fixation tendencies in person viewing generalize from images to videos. *I-Perception*, *13*(6). https://doi.org/10.1177/20416695221128844

Buswell, G. T. (1935). How people look at pictures: a study of the psychology and perception in art. In *How people look at pictures: a study of the psychology and perception in art.* Univ. Chicago Press.

Carpenter, R. H. S., Reddi, B. A. J., & Anderson, A. J. (2009). A simple two-stage model predicts response time distributions. *The Journal of Physiology*, *587*(16), 4051–4062. https://doi.org/https://doi.org/10.1113/jphysiol.2009.173955

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., … Fiedel, N. (2022). *PaLM: Scaling Language Modeling with Pathways*. https://doi.org/10.48550/arxiv.2204.02311

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. https://arxiv.org/abs/1412.3555v1

CrossEntropyLoss — PyTorch 2.0 documentation. (n.d.). Retrieved June 6, 2023, from https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html?highlight=cross+entropy#torch.nn.CrossEntropyLoss

De Haas, B., Iakovidis, A. L., Schwarzkopf, D. S., & Gegenfurtner, K. R. (2019). Individual differences in visual salience vary along semantic dimensions. *Proceedings of the National Academy of Sciences of the United States of America*, *116*(24), 11687–11692. https://doi.org/10.1073/pnas.1820553116

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, *1*, 4171–4186. https://arxiv.org/abs/1810.04805v2

Dolmans, T. C. (2023). *tcdolmans/Eye4AI*. https://github.com/tcdolmans/Eye4AI

Dolmans, T. C., Poel, M., van 't Klooster, J.-W. J. R., & Veldkamp, B. P. (2021). Perceived Mental Workload Classification Using Intermediate Fusion

Multimodal Deep Learning. *Frontiers in Human Neuroscience*, *14*. https://www.frontiersin.org/articles/10.3389/fnhum.2020.609096

Dolmans, T. C., Poel, M., van't Klooster, J.-W. J., & Veldkamp, B. P. (2020). Data synchronisation and processing in multimodal research. *Measuring Behavior 2020-21 Volume*, *1*, 26–32.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. https://doi.org/10.48550/arxiv.2010.11929

Duchowski, A., & Duchowski, A. (2007). Eye tracking techniques. *Eye Tracking Methodology: Theory and Practice*, 51–59.

Feichtenhofer, C., Fan, H., Malik, J., & He, K. (2019). *SlowFast Networks for Video Recognition* (pp. 6202–6211). https://github.com/

Fuhl, W., Kasneci, G., & Kasneci, E. (2021). Teyed: Over 20 million real-world eye images with pupil, eyelid, and iris 2D and 3D segmentations, 2D and 3D landmarks, 3D eyeball, gaze vector, and eye movement types. *Proceedings - 2021 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2021*, 367–375. https://doi.org/10.1109/ISMAR52148.2021.00053

Gage, P. (1994). *A New Algorithm for Data Compression*. https://doi.org/10.5555/177910.177914

GazeBase Data Repository. (n.d.). Retrieved June 16, 2023, from https://figshare.com/articles/dataset/GazeBase_Data_Repository/12912257

Gibaldi, A., Vanegas, M., Bex, P. J., & Maiello, G. (2017). Evaluation of the Tobii EyeX Eye tracking controller and Matlab toolkit for research. *Behavior Research Methods*, *49*(3), 923–946. https://doi.org/10.3758/S13428-016-0762-9/TABLES/3

GPT-3.5 + ChatGPT: An illustrated overview – Dr Alan D. Thompson – Life Architect. (n.d.). Retrieved January 16, 2023, from https://lifearchitect.ai/chatgpt/

Griffith, H., Lohr, D., Abdulin, E., & Komogortsev, O. (2021). GazeBase, a large-scale, multi-stimulus, longitudinal eye movement dataset. *Scientific Data 2021 8:1*, *8*(1), 1–9. https://doi.org/10.1038/s41597-021-00959-y

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition, *2016-December*, 770–778. https://doi.org/10.1109/CVPR.2016.90

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. de Las, Hendricks, L. A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., Driessche, G. van den, Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., … Sifre, L. (2022). *Training Compute-Optimal Large Language Models*. https://doi.org/10.48550/arxiv.2203.15556

Holmqvist, K., Örbom, S. L., Hooge, I. T. C., Niehorster, D. C., Alexander, R. G., Andersson, R., Benjamins, J. S., Blignaut, P., Brouwer, A. M., Chuang, L. L., Dalrymple, K. A., Drieghe, D., Dunn, M. J., Ettinger, U., Fiedler, S., Foulsham, T., van der Geest, J. N., Hansen, D. W., Hutton, S. B., … Hessels, R. S. (2022). Eye tracking: empirical foundations for a minimal reporting guideline. *Behavior Research Methods 2022*, *32*, 1–53. https://doi.org/10.3758/S13428-021-01762-8

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2021). *LoRA: Low-Rank Adaptation of Large Language Models*. https://arxiv.org/abs/2106.09685v2

Hugging Face – The AI community building the future. (2023, May 17). https://huggingface.co/

Ienca, M., & Andorno, R. (2017). Towards new human rights in the age of neuroscience and neurotechnology. *Life Sciences, Society and Policy*, *13*(1), 1–27. https://doi.org/10.1186/S40504-017-0050-1/METRICS

Itti, L. (2015). New Eye-Tracking Techniques May Revolutionize Mental Health Screening. *Neuron*, *88*(3), 442–444. https://doi.org/10.1016/J.NEURON.2015.10.033

Itti, L., & Koch, C. (2000). A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, *40*(10–12), 1489–1506. https://doi.org/10.1016/S0042-6989(99)00163-7

Jäger, L. A., Makowski, S., Prasse, P., Liehr, S., Seidler, M., & Scheffer, T. (2020). Deep Eyedentification: Biometric Identification Using Micro-movements of the Eye. In U. Brefeld, E. Fromont, A. Hotho, A. Knobbe, M. Maathuis, & C. Robardet (Eds.), *Machine Learning and Knowledge Discovery in Databases* (pp. 299–314). Springer International Publishing.

Katsini, C., Abdrabou, Y., Raptis, G., Khamis, M., & Alt, F. (2020). *The Role of Eye Gaze in Security and Privacy Applications: Survey and Future HCI Research Directions*. https://doi.org/10.1145/3313831.3376840

Komogortsev, O. v., Karpov, A., Price, L. R., & Aragon, C. (2012). Komogortsev et al 2021 Biometric Authentication via oculomotor plants characterestics. *IEEE Xplore*. https://doi.org/10.1109/ICB.2012.6199786

Kothari, R., Yang, Z., Kanan, C., Bailey, R., Pelz, J. B., & Diaz, G. J. (2020). Gaze-in-wild: A dataset for studying eye and head coordination in everyday activities. *Scientific Reports*, *10*(1). https://doi.org/10.1038/S41598-020-59251-5

Krafka, K., Khosla, A., Kellnhofer, P., Kannan, H., Bhandarkar, S., Matusik, W., & Torralba, A. (2016). *Eye Tracking for Everyone* (pp. 2176–2184). http://gazecapture.csail.mit.edu.

Larobina, M., & Murino, L. (2014). Medical image file formats. *Journal of Digital Imaging*, *27*(2), 200–206. https://doi.org/10.1007/S10278-013-9657-9/TABLES/1

Lecun, Y. (2022). *A Path Towards Autonomous Machine Intelligence*.

Li, A., & Chen, Z. (2018). Personalized Visual Saliency: Individuality Affects Image Perception. *IEEE Access*, *6*, 16099–16109. https://doi.org/10.1109/ACCESS.2018.2800294

Li, D., Shao, R., Wang, H., Guo, H., Xing, E. P., & Zhang, H. (2022). *MPCFormer: fast, performant and private Transformer inference with MPC*. https://doi.org/10.48550/arxiv.2211.01452

Li, J., Li, D., Xiong, C., & Hoi, S. (2022). *BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation*. https://arxiv.org/abs/2201.12086v2

Li, M., Song, F., Yu, B., Yu, H., Li, Z., Huang, F., & Li, Y. (2023). *API-Bank: A Benchmark for Tool-Augmented LLMs*. https://arxiv.org/abs/2304.08244v1

Lin, J., Yang, A., Zhang, Y., Liu, J., Zhou, J., & Yang, H. (2020). *InterBERT: Vision-and-Language Interaction for Multi-modal Pretraining*. https://doi.org/10.48550/arxiv.2003.13198

Lin, T., Wang, Y., Liu, X., & Qiu, X. (2022). A survey of transformers. *AI Open*, *3*, 111–132. https://doi.org/10.1016/J.AIOPEN.2022.10.001

Linka, M., & de Haas, B. (2020). OSIEshort: A small stimulus set can reliably estimate individual differences in semantic salience. *Journal of Vision*, *20*(9), 1–9. https://doi.org/10.1167/JOV.20.9.13

Linnainmaa, S. (1970). Algoritmin kumulatiivinen pyöristysvirhe yksittäisten pyöristysvirheiden Taylor-kehitelmänä [he representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors]. *University of Helsinki, Master's Theses*.

Ma, M., Ren, J., Zhao, L., Testuggine, D., & Peng, X. (2022). *Are Multimodal Transformers Robust to Missing Modality?* (pp. 18177–18186).

Makowski, S., Jäger, L. A., Abdelwahab, A., Landwehr, N., & Scheffer, T. (2019). A discriminative model for identifying readers and assessing text comprehension from eye movements. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11051 LNAI*, 209–225. https://doi.org/10.1007/978-3-030-10925-7_13/FIGURES/3

Makowski, S., Jager, L. A., Prasse, P., & Scheffer, T. (2020). Biometric identification and presentation-attack detection using micro- And macro-movements of the eyes. *IJCB 2020 - IEEE/IAPR International Joint Conference on Biometrics*. https://doi.org/10.1109/IJCB48548.2020.9304900

Makowski, S., Prasse, P., Reich, D. R., Krakowczyk, D., Jager, L. A., & Scheffer, T. (2021). DeepEyedentificationLive: Oculomotoric Biometric Identification and Presentation-Attack Detection Using Deep Neural Networks. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, *3*(4), 506–518. https://doi.org/10.1109/TBIOM.2021.3116875

MEPs ready to negotiate first-ever rules for safe and transparent AI | News | European Parliament. (n.d.). Retrieved June 16, 2023, from https://www.europarl.europa.eu/news/en/press-room/20230609IPR96212/meps-ready-to-negotiate-first-ever-rules-for-safe-and-transparent-ai?utm_source=marktechpost-newsletter.beehiiv.com&utm_medium=newsletter&utm_campaign=ai-news-llm-blender-for-superior-results-eu-s-groundbreaking-ai-regulations-lu-nerf-s-accurate-pose-estimation-chatdb-s-symbolic-memory-tulu-s-advancements-and-webglm-s-web-enhanced-qa-system

Mézière, D. C. (2022). *Using Eye Movements to Develop an Ecologically-valid AI Measure of Reading Comprehension*. https://doi.org/10.33612/DISS.216916144

Midjourney. (n.d.). Retrieved January 16, 2023, from https://www.midjourney.com/home/

Mollon, J. D., Bosten, J. M., Peterzell, D. H., & Webster, M. A. (2017). Individual differences in visual science: What can be learned and what is good experimental practice? *Vision Research*, *141*, 4–15. https://doi.org/10.1016/J.VISRES.2017.11.001

MSELoss — PyTorch 2.0 documentation. (n.d.). Retrieved June 6, 2023, from https://pytorch.org/docs/stable/generated/torch.nn.MSELoss.html

Nagrani, A., Yang, S., Arnab, A., Jansen, A., Schmid, C., Sun, C., & Research, G. (2021). Attention Bottlenecks for Multimodal Fusion. *Advances in Neural Information Processing Systems*, *34*, 14200–14213.

NITRC: MRIcro: Tool/Resource Info. (n.d.). Retrieved January 27, 2023, from https://www.nitrc.org/projects/mricro

OpenAI. (2023). *GPT-4 Technical Report*. https://arxiv.org/abs/2303.08774v3

OpenAI's CEO Says the Age of Giant AI Models Is Already Over | WIRED. (n.d.). Retrieved June 13, 2023, from https://www.wired.com/story/openai-ceo-sam-altman-the-age-of-giant-ai-models-is-already-over/

Optuna - A hyperparameter optimization framework. (n.d.). Retrieved June 6, 2023, from https://optuna.org/

Patel, D., & Mok, A. (n.d.). *How Much Does ChatGPT Cost to Run? $700K/day, Per Analyst*. Retrieved June 13, 2023, from https://www.businessinsider.com/how-much-chatgpt-costs-openai-to-run-estimate-report-2023-4?r=US&IR=T

Peng, B., Alcaide, E., Anthony, Q., Albalak, A., Arcadinho, S., Cao, H., Cheng, X., Chung, M., Grella, M., GV, K. K., He, X., Hou, H., Kazienko, P., Kocon, J., Kong, J., Koptyra, B., Lau, H., Mantri, K. S. I., Mom, F., … Zhu, R.-J. (2023). *RWKV: Reinventing RNNs for the Transformer Era*. https://arxiv.org/abs/2305.13048v1

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 1532–1543. https://doi.org/10.3115/V1/D14-1162

Peters, M. E., Ammar, W., Bhagavatula, C., & Power, R. (2017). Semi-supervised sequence tagging with bidirectional language models. *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, *1*, 1756–1765. https://doi.org/10.18653/v1/P17-1161

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, *1*, 2227–2237. https://doi.org/10.18653/v1/n18-1202

Peterson, M. F., & Eckstein, M. P. (2013). Individual Differences in Eye Movements During Face Identification Reflect Observer-Specific Optimal Points of Fixation. *Psychological Science*, *24*(7), 1216–1225. https://doi.org/10.1177/0956797612471684

Pfeiffer, J., Rücklé, A., Poth, C., Kamath, A., Vulić, I., Ruder, S., Cho, K., & Gurevych, I. (2020). *AdapterHub: A Framework for Adapting Transformers*. 46–54. https://doi.org/10.18653/v1/2020.emnlp-demos.7

Pillow (PIL Fork) 9.5.0 documentation. (n.d.). Retrieved May 24, 2023, from https://pillow.readthedocs.io/en/stable/

Pushkarna, M., Zaldivar, A., & Kjartansson, O. (2022). Data Cards: Purposeful and Transparent Dataset Documentation for Responsible AI. *ACM International Conference Proceeding Series*, 1776–1826. https://doi.org/10.1145/3531146.3533231

PyTorch. (n.d.). Retrieved May 24, 2023, from https://pytorch.org/

Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, F., Aslanides, J., Henderson, S., Ring, R., Young, S., Rutherford, E., Hennigan, T., Menick, J., Cassirer, A., Powell, R., Driessche, G. van den, Hendricks, L. A., Rauh, M., Huang, P.-S., … Irving, G. (2021). *Scaling Language Models: Methods, Analysis & Insights from Training Gopher*. https://doi.org/10.48550/arxiv.2112.11446

Redfern, M. S., Yardley, L., & Bronstein, A. M. (2001). Visual influences on balance. *Journal of Anxiety Disorders*, *15*(1–2), 81–94. https://doi.org/10.1016/S0887-6185(00)00043-8

Ribeiro, A. H., Tiels, K., Aguirre, L. A., & Schön, T. (2020). *Beyond exploding and vanishing gradients: analysing RNN training using attractors and smoothness* (pp. 2370–2380). PMLR. https://proceedings.mlr.press/v108/ribeiro20a.html

Rigas, I., & Komogortsev, O. (2016). Biometric Recognition via Eye Movements: Saccadic Vigor and Acceleration Cues. *Appl. Percept*, *13*(6). https://doi.org/10.1145/2842614

Robbins, H., & Monro, S. (1951). A Stochastic Approximation Method. *Https://Doi.Org/10.1214/Aoms/1177729586*, *22*(3), 400–407. https://doi.org/10.1214/AOMS/1177729586

Rolff, T., Harms, H. M., Steinicke, F., & Frintrop, S. (2022). GazeTransformer: Gaze Forecasting for Virtual Reality Using Transformer Networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *13485 LNCS*, 577–593. https://doi.org/10.1007/978-3-031-16788-1_35/TABLES/3

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408. https://doi.org/10.1037/H0042519

Ryoo, M. S., Piergiovanni, A. J., Arnab, A., Dehghani, M., & Angelova, A. (2021). TokenLearner: Adaptive Space-Time Tokenization for Videos. *Advances in Neural Information Processing Systems*, *34*, 12786–12797. https://github.com/google-research/

Salahuddin, Z., Woodruff, H. C., Chatterjee, A., & Lambin, P. (2022). Transparency of deep neural networks for medical image analysis: A review of interpretability methods. *Computers in Biology and Medicine*, *140*, 105111. https://doi.org/10.1016/J.COMPBIOMED.2021.105111

Schlögl, A., Keinrath, C., Zimmermann, D., Scherer, R., Leeb, R., & Pfurtscheller, G. (2007). A fully automated correction method of EOG artifacts in EEG recordings. *Clinical Neurophysiology*, *118*(1), 98–104. https://doi.org/10.1016/J.CLINPH.2006.09.003

Sennrich, R., Haddow, B., & Birch, A. (2015). Neural Machine Translation of Rare Words with Subword Units. *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers*, *3*, 1715–1725. https://doi.org/10.18653/v1/p16-1162

Shalf, J. (2020). The future of computing beyond Moores Law. *Philosophical Transactions of the Royal Society A*, *378*(2166). https://doi.org/10.1098/RSTA.2019.0061

Snoek, L., van der Miesen, M. M., Beemsterboer, T., van der Leij, A., Eigenhuis, A., & Steven Scholte, H. (2021). The Amsterdam Open MRI Collection, a set of multimodal MRI datasets for individual difference analyses. *Scientific Data 2021 8:1*, *8*(1), 1–23. https://doi.org/10.1038/s41597-021-00870-6

Song, Y., Zheng, Q., Liu, B., & Gao, X. (2022). EEG Conformer: Convolutional Transformer for EEG Decoding and Visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. https://doi.org/10.1109/TNSRE.2022.3230250

Sriramakrishnan, P., Kalaiselvi, T., Padmapriya, S. T., Shanthi, N., Ramkumar, S., & Kalaichelvi, N. (2019). An medical image file formats and digital image conversion. *Int. J. Eng. Adv. Technol*, *9*(1S3), 74–78.

Srivastava, N., Hinton, G., Krizhevsky, A., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, *15*, 1929–1958.

Takahashi, S., & Tanaka-Ishii, K. (2018). Cross Entropy of Neural Language Models at Infinity—A New Bound of the Entropy Rate. *Entropy*, *20*(11). https://doi.org/10.3390/E20110839

The FAIR Data Principles – FORCE11. (n.d.). Retrieved June 13, 2023, from https://force11.org/info/the-fair-data-principles/

TransformerDecoder — PyTorch 2.0 documentation. (n.d.). Retrieved May 31, 2023, from https://pytorch.org/docs/stable/generated/torch.nn.TransformerDecoder.html

Usage Statistics of Image File Formats for Websites, January 2023. (n.d.). Retrieved January 16, 2023, from https://w3techs.com/technologies/overview/image_format

Valmeekam, K., Olmo, A., Sreedharan, S., & Kambhampati, S. (2022). *Large Language Models Still Can't Plan (A Benchmark for LLMs on Planning and Reasoning about Change)*. https://arxiv.org/abs/2206.10498v3

Vaswani, A., Brain, G., 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*.

Weber, D., & Gühmann, C. (2021). Non-Autoregressive vs Autoregressive Neural Networks for System Identification. *IFAC-PapersOnLine*, *54*(20), 692–698. https://doi.org/10.1016/j.ifacol.2021.11.252

Wee, C. Y., Yap, P. T., Zhang, D., Denny, K., Browndyke, J. N., Potter, G. G., Welsh-Bohmer, K. A., Wang, L., & Shen, D. (2012). Identification of MCI individuals using structural and functional connectivity networks. *NeuroImage*, *59*(3), 2045–2056. https://doi.org/10.1016/J.NEUROIMAGE.2011.10.015

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., & Liu, T.-Y. (2020). *On Layer Normalization in the Transformer Architecture* (pp. 10524–10533). PMLR. https://proceedings.mlr.press/v119/xiong20b.html

Xu, J., Jiang, M., Wang, S., Kankanhalli, M. S., & Zhao, Q. (2014). Predicting human gaze beyond pixels. *Journal of Vision*, *14*(1). https://doi.org/10.1167/14.1.28

Xu, P., Zhu, X., & Clifton, D. A. (2022). *Multimodal Learning with Transformers: A Survey*. https://doi.org/10.48550/arxiv.2206.06488

Yang, G. R., & Molano-Mazón, M. (2021). Towards the next generation of recurrent network models for cognitive neuroscience. *Current Opinion in Neurobiology*, *70*, 182–192. https://doi.org/10.1016/J.CONB.2021.10.015

Yang, J., Jin, H., Tang, R., Han, X., Feng, Q., Jiang, H., Yin, B., & Hu, X. (2023). *Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond*. https://arxiv.org/abs/2304.13712v2

Yao, S., Yu, D., Deepmind, G., Zhao, J., Griffiths, T. L., Cao, Y., & Narasimhan, K. (2023). *Tree of Thoughts: Deliberate Problem Solving with Large Language Models*. https://arxiv.org/abs/2305.10601v1

You, H., Zhou, L., Xiao, B., Codella, N., Cheng, Y., Xu, R., Chang, S.-F., & Yuan, L. (2022). Learning Visual Representation from Modality-Shared Contrastive Language-Image Pre-training. *LNCS*, *13687*, 69–87. https://doi.org/10.1007/978-3-031-19812-0_5/FIGURES/4

Zhang, K., Yu, J., Yan, Z., Liu, Y., Adhikarla, E., Fu, S., Chen, X., Chen, C., Zhou, Y., Li, X., He, L., Davison, B. D., Li, Q., Chen, Y., Liu, H., & Sun, L. (2023). *BiomedGPT: A Unified and Generalist Biomedical Generative Pre-trained Transformer for Vision, Language, and Multimodal Tasks*. https://arxiv.org/abs/2305.17100v1

Zhou, W., Zeng, Y., Diao, S., & Zhang, X. (2022). *VLUE: A Multi-Task Benchmark for Evaluating Vision-Language Models*. https://arxiv.org/abs/2205.15237v1

Zigmond, M. J., Bloom, F. E., Landis, S. C., Roberts, J. L., & Squire, L. R. (1999). *Fundamental neuroscience* (Vol. 207). Academic press San Diego.