Opleiding Data Science & AI

The capabilities of LLMs in assisting in a Language Production task

Rajeev Nathie

Supervisors:
Xiaochen Zheng & Zhaochun Ren

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

15/01/2026

Abstract

In this thesis, methods are explored to use model embeddings to assist in predicting response times for a picture naming task. We are interested in producing meaningful model representations. If this succeeds, we are interested whether these model representations can be improved by using a model which is more aligned to human cognition. Three models will be assessed for this task. These are a multilingual CLIP VLM model, a multilingual transformer model, and a multilingual ELMo model. The latter being a model aligning to human cognition to evaluate if these embeddings will be more meaningful than that of the other models. We use the cosine distance, surprisal and spearman correlation to evaluate the models and compare them with each other using linear mixed models and the Bayesian Information Criterion (BIC) score.

Contents

1 Introduction 1
1.1 Thesis overview . . . 2

2 Previous studies and task 4
2.1 The task . . . 4
2.1.1 The Experiment . . . 4
2.1.2 Data . . . 5
2.2 On cognitive modeling of language processing . . . 5
2.3 Prior studies using LLMs to predict RTs . . . 6

3 Methodology 8
3.1 CLIP . . . 8
3.1.1 CLIP Architecture . . . 8
3.1.2 Application . . . 10
3.1.3 Validation encoders . . . 11
3.1.4 Validation pairs . . . 14
3.2 Cognitively aligned model . . . 16
3.2.1 LSTM . . . 16
3.2.2 ELMO . . . 17
3.2.3 Validation . . . 17
3.2.4 ELMo setup . . . 18
3.3 Prediction . . . 18
3.3.1 Statistical testing . . . 18

4 Results 20
4.1 Model embeddings . . . 20
4.2 Surprisals . . . 21
4.3 Spearman correlation . . . 22
4.4 Scatter plots . . . 23
4.5 Model comparison . . . 25

5 Discussion 26

References 31

1 Introduction

Human behavior is more complex than it may seem, as simple actions like talking can be broken down into many smaller tasks, such as formulating sentences, using the right grammar, producing the right vocal tones and choosing the right words. Understanding these smaller tasks can help provide a better understanding and explanation of our behavior. This is what the field of cognitive psychology aims to achieve. This can be studied by collecting behavioral data from experiments on language tasks.

Language tasks can be subdivided into two main categories: Firstly, language comprehension, which is about understanding the language. Mainly studied using reading and listening tasks as it deals with lexical access, parsing and semantic interpretation. Tasks that are studied in this category are reading or listening tasks.

Language production tasks regard to the output of language. Conceptualization, formulation and syntactic planning are main cognitive processes involved with production tasks. An important task to study language production is picture naming. This can be evaluated by performing an experiment in which participants will have to name pictures out loud. In this thesis, data from such an experiment will be studied. This experiment has been performed in Dutch, with a total of 39 participants.

For language tasks humans predict words that follow each other [PG07]. When the predicted words fall within the context of the current event, this is known as facilitation comprehension. Interference comprehension is the opposite, which triggers an error or delayed response.

An efficient way to understand and explain processes like that of human behavior, is by using (large) language models. Many concepts within the field of Artificial Intelligence (AI) are based on human biology and psychology. Our mental working memory system which lets us temporarily hold and manipulate information has inspired multiple architectures in AI, think of (self) attention, scaling and the Long Short Term Memory (LSTM). The latter being a type of Recurrent Neural Network (RNN) which sequentially processes information through different types of gates. Predictive models in the field of AI, such as LSTMs, can be trained to try and perform a human task and accurately predict trends or sequential data. Large language models (LLMs) have proven to be an efficient method for assisting in natural language processing tasks [NKQ+25].

In this thesis we will be using multilingual models, as the experiment was performed in Dutch, to assess and compute the data of the experiment to provide us meaningful embeddings. We will use the CLIP Visual Language Model, as this model is parallel to the recognition task in humans. A setup using the multilingual Roberta transformer is also used as this is the state-of-the-art method to handle and interpret text. And finally an ELMo model is used, because this model aligns to human cognition since information is processed sequentially. Therefore, only the embeddings from the forward layers are used for the latter model.

Since this thesis takes a leap into psychology, some terms unfamiliar for those who have no background in this field will be explained so that it can be understood why certain choices were made for the methodology in this thesis.

Representations describe how a concept is conceptualized in a system. For example, a mental representation of a car could be a simple outline of wheels with a rectangular shape above the wheels. While models can represent the car by means of embeddings. In this thesis models will represent 'their idea' of sentences and labels using embeddings, these embeddings will be used to predict response times.

As mentioned before, a closer look will be take at a picture naming task in this thesis. In this task participants were presented with different sentences followed by a work that would either fit the context or not. When a participant is presented with stimuli that negatively affects the context of the prior stimuli, this is known as interference. When the opposite is the case and the presented stimuli positively affects the prior stimuli, we speak of facilitation. Both interference and facilitation are studied using either response times (RTs) or error rate. Interference in cognitive psychology can have negative effects on a participants response time, while facilitation can improve it as was studied by J. Neely et al. [Nee77].

Since language tasks involve predictions, the context of the task is important, as the agent will continue to predict the succeeding stimulus. When a stimulus adheres to the previously established context, the stimulus-context pair is congruent. If the opposite is the case, the pair is incongruent. A stimulus can only be congruent or incongruent when the prior context is constraining, meaning that only stimuli from certain categories would fit the context presented. It could also be the case that the established context is not constraining. In these cases, the stimulus-context pair is neutral. E. Wilcox and J. Gauthier explored the use of deep transformer models and LSTMs on a language comprehension task [WGH$^+$20]. Suprisals were used, measuring how uncertain the model is about its prediction, as a means of representation to predict human reading time. They found that all transformer models outperformed the RNN models and that there is a linear relation between word-level surprisal and human reading time. L. Salicchi and A. Lenci [SL21] compared different word embedding models to predict human reading patterns, they used the best cosine distance and surprisal metrics from each model to outperform baseline methods, showing how these metrics can be a good measure to optimize for predictions.

1.1 Thesis overview

In this thesis, different approaches using different models will be used to provide embeddings to help predict human RTs.

The main research question is:
Can we use model-based representations to predict human response times in a language production task?

That is, given the data from a language production task, is it possible for a model to provide meaningful embeddings, in which a clear distinction can be found regarding congruency. Following this question we are also interested in:

SQ1: Can we improve these embeddings by using models more aligned with human cognition?

To answer the secondary question, we will compare the models with each other. In our case we will be comparing an LSTM setup vs a transformer and a CLIP vs a non-CLIP setup. We do think that the human aligned model will improve the embeddings, since it is more aligned and thus processes the information sequentially in a similar way to humans.

In a recent study by Zheng et al. (in prep) an EEG experiment has been conducted in which participants were asked to read words on a screen that were followed by a picture. The participants were required to only name the picture as fast and accurate as possible. These pictures could fall within the context of the sentence (congruent) or not (incongruent), or be neutral for sentences which do not constrain the picture to be in a certain context, leading to a diversity in response times (RT) by the participants. The data from this experiment will be used in this thesis. The full experiment will be discussed in Chapter 2.1

In this thesis I will first provide a background on research that has been conducted using LLMs on language tasks and explain terminology that is used to understand the psychological concepts better in chapter 2. In chapter 3 I will dive in the methodology, explaining how I will tackle the presented problem by taking a closer look at the models and the metrics that I will be using to help predict response times. In chapter 4 I will present my results and compare these results to those of the participants of the language production task mentioned before. In Chapter 5 I will discuss my work, its limitations, and what further work can be done.

2 Previous studies and task

In this chapter, the task that will be studied will be explained in detail and relevant studies will be brought up to understand what the current state of language production tasks and predicting response times is. In 2.2 studies using LLMs are inspected in which ways to evaluate their data are brought up, this will clarify choices for our evaluation methods. Finally in 2.3 prior studies will be addressed to provide context on the current state of the work done on language tasks using LLMs.

2.1 The task

The language production task that will be taken a closer look at in this thesis is a contextualized picture naming task. Data were collected from 39 participants in the Sylvius lab in Leiden. Participants were required to be native Dutch speakers, since the experiment mandates them to read dutch words and name pictures in dutch out loud. Furthermore, participants were between the ages of 18 and 26, right-handed and had normal or corrected-to-normal vision. Participants' EEG were recorded using a BioSemi 64-channel system, following 10-20 electrode placement system so that neural activity could be measured while performing the task.

2.1.1 The Experiment

During the experiment, participants would first be shown a fixation mark ('+') for 500 ms. Following the fixation mark, dutch words would be shown one by one for a duration of 500 ms on a screen. Between each word an empty slide would be shown. A total of 4-6 words were shown on the screen, after which the pre-picture interval ('...') would be shown for 800 ms. Following the interval, the picture was shown until the participant named the picture or ran out of time. The particpants were instructed to name the pictures as fast as possible.

In total there are 60 target pictures used, each occurred once in one of the three conditions. So in total there are 180 trials per subject.

After certain sentence-picture pairs, a question regarding the context was asked, a probe, to ensure that the participant would pay attention to the context of the sentence rather than ignoring the words and only naming the picture as fast as possible. An example of the probe is present in the top right of figure 1.

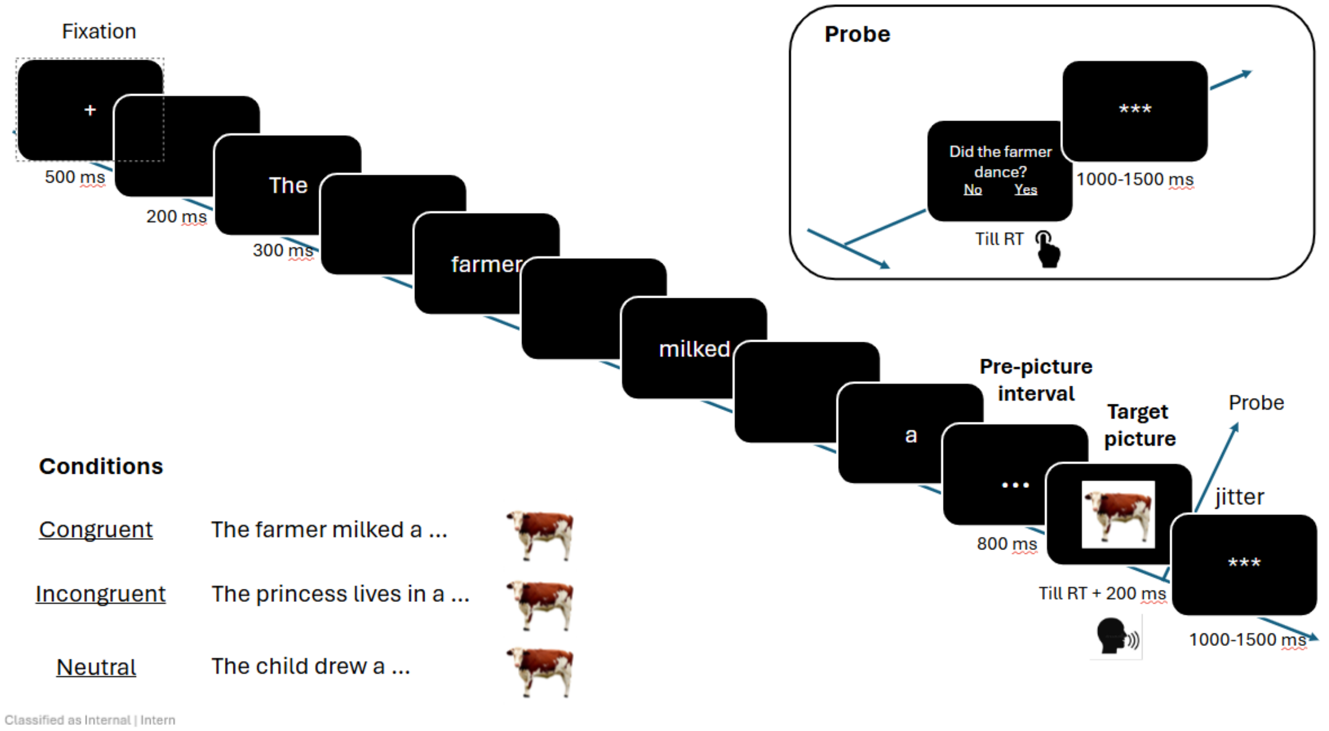


Figure 1: An illustration of the picture-naming task. The black slides on the diagonal show the timeline of an example sentence. On the bottom left three different examples for each congruency category are displayed. In the top right, an example probe is displayed. Note that the actual stimuli in the experiment were in Dutch.

The sentence-picture pairs that were shown to participants can be clustered in three different categories based on congruency. Figure 1 showcases examples of these categories. The context for the congruent and incongruent pairs is identical and leads to a clear prediction, while for the neutral class this is not the case.

2.1.2 Data

In this thesis, I will use the data collected in the contextualized picture naming task. The analysis will focus on the naming RTs. The crucial variables include the presented sentences, pictures, response times, congruency class and the final expected word.

2.2 On cognitive modeling of language processing

The brain implicitly makes predictions to assist in language processing. M. Heilbron, K. Armeni et al. [HAS+22] concluded that neural responses to speech are ”modulated by continuous linguistic predictions”. The brain is predicting upcoming language and whenever a violation is perceived, an error response was observed in the EEG and MEG data of participants listening to audiobooks. Providing proof that prediction by the brain is continuous and probabilistic, they used a GPT-2 model to calculate the surprisal:

$$S(x_t) = -log_2 P(x_t|context) \tag{1}$$

The surprisal indicates how unexpected the next word is given the previous context. There is a negative correlation between the suprisal and the logarithmic probability, since for a high probability that the next word is for example 'car' the surprisal is low.

Stefan L. et al. [FOGV15] found that the amplitude of the N400 wave was strongly correlated to the surprisal not the entropy and that this relationship was linear. Besides, two models were used to correlate brain data with. These were an RNN and N-gram model. The results showed that the RNN fitted the brain data better than the N-gram model.

L. Wang et al. [WKJ18] used MEG in combination with Representational similarity analysis (RSA) to show that the brain activates signals for a expected word before the word pops up on screen. RSA is a powerful framework which compares relationships with each other and visualizes this by means of a Representational Dissimilarity Matrix (RDM). The study showed that neural patterns were more similar when different contexts predicted the same words, than when they predicted different words. They also found that the brain predicts specific concepts instead of categories.

These studies all used LLMs and provided metrics that can be used to validate and visualize model representations, such as by using the surprisal or RDMs.

2.3 Prior studies using LLMs to predict RTs

Studies has been conducted on language comprehension tasks using LLMs to predict RTs of participants. E. Wilcox, J. Gauthier et al. [WGH+20] studied the relation between surprisal and human reading times using N-gram models and transformer models. They found a linear relation between the surprisal of a model and human reading times. The main founding of the paper was that to predict human reading times this is mainly based on the models capacity to predict future words.

T. Kuribayasi et al. [KOBI22] found that by limiting the context window of a model, instead of letting the model perceive the entire history improved the model's ability to predict human reading times, supporting memory based theories. This was mostly helpful in the cases were memory retrieval is costly for humans.

D. Merkx, S. Frank et al. [MF21] studied the architecture of human language processing. Their hypotheses were that humans process language sequentially, aligning with how RNNs process, and also retrieve past information, aligning with transformers. They trained both transformers and RNNs on the same data to predict surprisal estimates. The results showed that transformer models outperform the RNN models in both predicting human reading times and predicting the N400 wave. Transformer models were able to capture interference effects better than the RNN models, since the RNN models only forget context over time, while transformer models can forget due to a change in context.

These studies all used LLMs to predict response times. Their results indicated that these are valid models to use to capture patterns from the data, with the Transformer models outperforming the RNN models. Additionally, the surprisal showed to be a meaningful metric to interpret the relation between human response and transformer models.

3 Methodology

In order to answer the research questions, different models will be set up, validated, and acquired embeddings from. First, these models will be examined to establish an understanding about them, taking a closer look at how they work. Secondly, the models need to be validated, to check if meaningful embeddings can be provided rather than random outputs.

First, a setup using the Contrastive Learning Image Pre-training model (CLIP) will be used to acquire model representations. We chose to use this model because it is parallel to the behavioral task of recognition. CLIP has proven to be a powerful tool for aligning images and text, both of which are used for the production task. Another setup using a transformer will also be studied. We are also interested in finding if the task effect is in place, meaning that the behavior of the model is driven by the constraints or semantics of the task, this should apply to all models. I will explain the CLIP architecture and the hyperparameter choices which were made, in the following sections.

3.1 CLIP

3.1.1 CLIP Architecture

A. Radford et al., 2021 [RKH+21] introduced CLIP. CLIP consists of two encoders, an image encoder and a text encoder. For this thesis, paired multilingual encoders will be used, since these are able to handle Dutch data. These encoders are trained on the WebImageText (WIT) dataset, created by the authors themselves, which contains over 400 million text-image pairs in English. Later, datasets such as LAION-5B [SBV+22] were created by the community. This data set contains over 5 billion pairs and is multilingual.

There are two types of image encoders that can be chosen to assist in tasks. The ResNet encoders and the Vision Transformer (ViT) encoders. The ResNet encoders processes images using convolutions, which allows it to handle translation invariance and capture textures and edges in images highly detailed. ResNet encoders are slower to train and additionally have a worse global understanding, since relationships between distant objects are harder to understand. This is due to the convolutions looking at small local neighborhoods.

The ViT encoders on the other hand, processes the image in patches which it will flatten and feed into a Transformer. The Transformer uses Self-Attention [VSP+17] to form relations between each patch. The Self-Attention mechanism reads everything simultaneously and calculates relationships between each element using three small neural networks known as the Query(Q), Key(K) and Value(V). For each element the Q will be compared against all K elements calculating a score:

$$Score = Q \cdot K^T \tag{2}$$

The V of each matching element will be taken and weighted by the score to create a better representation of the current element. By using a Class Token (CLS), data from all other patches are collected as the image data passes through the network. ViT encoders are much more computationally efficient than ResNet encoders, allowing massive models to be trained. ViT encoders are also able to capture global context easily. By increasing the image resolution, the number of patches increases quadratic, making a model slower while ResNet encoders handle image size more linearly.

Different text encoders can be used. The original paper [RKH+21] used a transformer encoder, while a encoder-only [JYX+21] or encoder-decoder architecture [CBW+23] can also be used. In section 3.1.2 the choice of text encoder for this research and its architecture will be clarified.

Both the image and text encoder will work in parallel and first encode and apply L2-normalization to the input projecting to a vector space of the same size, for example 512 which is used in this thesis. This is one of the smaller dimension spaces and we have chosen to use this since the sentences and labels, which are the one word image descriptions, that are used are short (4-6 words) and describe simple objects (think of a cow, or book or newspaper). Figure 2 shows this process. In this thesis, the CLIP model is only used for inference, however during training of the model, contrastive learning is applied, minimizing the distance between the pairs (maximizing similarity) on the diagonal and maximizing the distance for the other pairs.

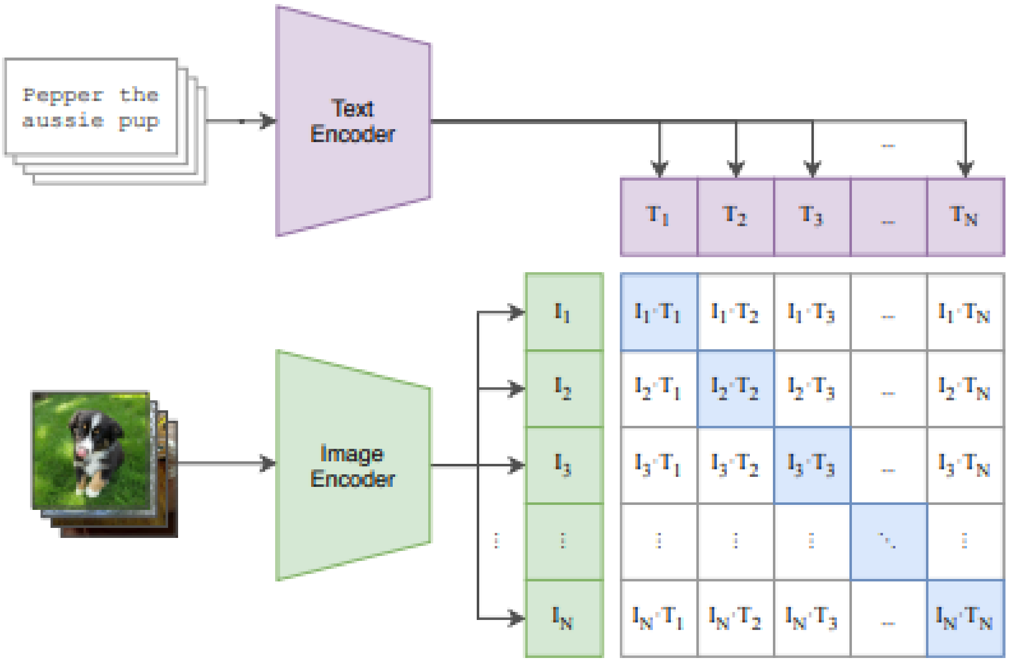

Pepper the aussie pup
Text Encoder
Image Encoder
T_1 T_2 T_3 ... T_N
I_1 I_2 I_3 ⋮ I_N
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
$I_2 \cdot T_1$ $I_2 \cdot T_2$ $I_2 \cdot T_3$ ... $I_2 \cdot T_N$
$I_3 \cdot T_1$ $I_3 \cdot T_2$ $I_3 \cdot T_3$ ... $I_3 \cdot T_N$
⋮ ⋮ ⋮ ⋱ ⋮
$I_N \cdot T_1$ $I_N \cdot T_2$ $I_N \cdot T_3$ ... $I_N \cdot T_N$

Figure 2: Visualization [RKH+21] of the CLIP joint image and text encoder setup. The diagonal displays the matching pairs for which the distance should be minimized using contrastive loss. The distance for the remaining pairs should be maximized

In this thesis contrastive learning is not used, since we do not train the model, however the cosine similarity and the loss functions do play an important part as these metrics are also used in our evaluation. A brief description of the contrastive learning concept follows.

given N image-text pairs with image features: $I_1, I_2, ..., I_N$ and text features: $T_1, T_2, ..., T_N$ CLIP aims to maximize similarity between I_i and T_i and minimize similarity between I_i and T_j where $j \neq i$. The similarity between each pair is calculated using the cosine distance:

$$S = I \cdot T^T \cdot e^\tau \tag{3}$$

in equation 3, I represents the image embedding matrix, T the text embedding matrix and τ the learned temperature value to sharpen or flatten the softmax distribution in the loss function.

Two cross-entropy losses are calculated: Image-to-Text loss $\mathcal{L}_{I\to T}$ and Text-to-Image loss $\mathcal{L}_{T\to I}$:

$$\mathcal{L}_{I\to T} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(S_{i,i})}{\sum_{j=1}^{N}\exp(S_{i,j})} \tag{4}$$

$$\mathcal{L}_{T\to I} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(S_{j,j})}{\sum_{j=1}^{N}\exp(S_{i,j})} \tag{5}$$

With total loss:

$$\mathcal{L} = \frac{1}{2}(\mathcal{L}_{I\to T} + \mathcal{L}_{T\to I}) \tag{6}$$

This loss function 6 ensures that matching pairs will have a high cosine similarity and non-matching pairs will have a low cosine similarity.

3.1.2 Application

For this thesis no model will be trained. Instead, a model will be selected to use for inference, therefore; the choices for the image and text encoders are the most important. Since the language task was carried out using dutch image-sentence pairs, the text and image encoders should be able to support the dutch language. Since the sentences are at most 6 words long and the images contain simple concepts or objects, not many features have to be extracted from them. A simple model would suffice. Therefore, only multilingual encoders with output in a 512-dimensional space were mainly considered for this task.

The following image encoders were considered: The image encoder of the multilingual ViT base model[RG19]. The original ViT-32 encoder [DBK+21], lastly an attempt was made with the ViT-H image encoder [DBK+21], which outputs embeddings in a 1024 dimensional space. By applying a linear layer, the projection dimension can be reduced from 1024 to 512 to align with the text encoder.

Regarding the text encoders, the following were considered: The text encoder of the multilingual ViT base model[RG19], Roberta [CERS22] and the laion text encoder[CBW+23][SBV+22].

The ViT multilingual model was trained using multilingual knowledge distillation [RG20]. The image encoder remained unchanged, using the original Clip-ViT-B-32 image encoder. The ViT-32 model was trained on the WIT dataset [RKH+21]. And the ViT-H model was trained on the LAION-2B dataset [SBV+22].

The latter multilingual ViT CLIP model has its own pair of image and text encoder and we will therefore validate this pairing. Furthermore the base ViT-32 image and text encoder pair is validated as well, since this model is also trained on a multilingual dataset and is not large, being able to handle shorter data as is applicable for our data. The combination of the ViT-H image encoder and Laion text encoder is examined as well, as these were trained together and since they output to a higher dimension, possibly this model would be able to distinguish different semantic groups better. Lastly, we chose to pair the multilingual ViT text encoder with the original ViT base image encoder since these individual encoders showed the best results in validation as discussed in the next section. To check if the individual encoders would deliver meaningful results, as validation step, Representational Dissimilarity Matrixes (RDMs) were used to analyze how different the embeddings from each encoder were in comparison with each other.

An RDM is a square matrix in which each row and also each column represents a different stimulus, in our case a different label or image. Each cell contains the dissimilarity value of the corresponding row-column pairing. The value is represented using a color, in our case the more yellow the cell, the closer the vectors are in the dimensional space, the more similar they are.

3.1.3 Validation encoders

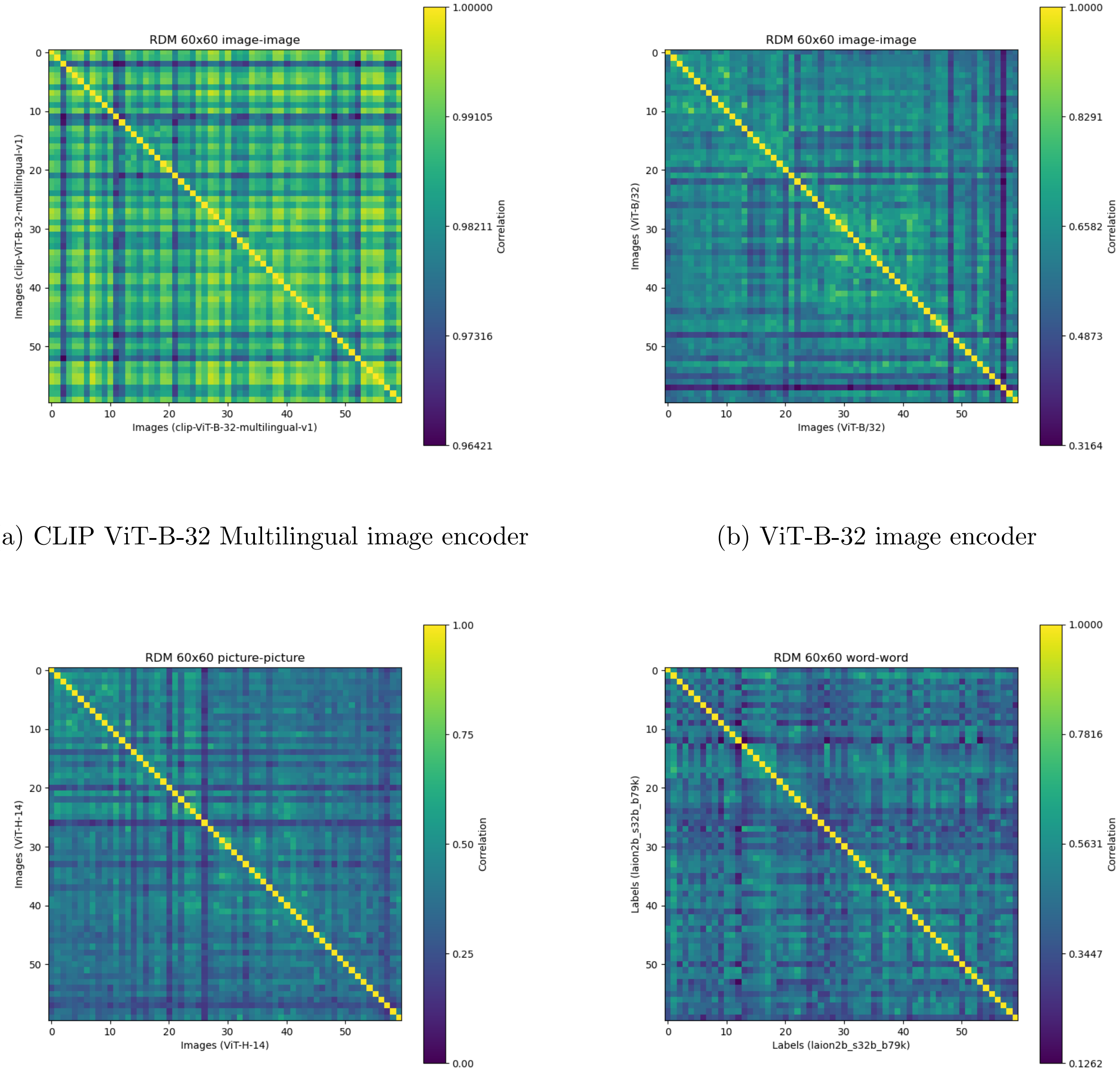
First, for each image and text encoder, each output embedding was compared to all other output embeddings provided by the same encoder, to understand if relations between these text-text pairs or image-image pairs can be extracted and correlation values would not be too high or too low. We will do this by feeding the encoders all 60 labels to check if the diagonal will contain the most similar values. Besides, both image and text inputs were semantically categorized as shown in table 1, such that it was easier to check for semantic relations between groups. This should be represented in the RDM by each member of a category having higher similarity with the other members of the same category. If the models were to capture context and semantics this would be visible through squares indicating a higher correlation along the diagonal. The diagonal itself should consist of only the value 1, since the embedding of each input is paired with itself. RDMs are used to visualize the correlation of the outputs by the encoders.

Semantic category	**Range**
(parts of) Animals	000-010
Nature	011-020
Food	021-025
Tools	026-044
Literature	045-047
Objects	048-059

Table 1: Semantic categorization of label elements and image elements input to the encoders for the validation process

Secondly, for our chosen image-text encoder pairings, we will inspect the chosen pairs by checking their image-sentence pair RDM and specifically looking at whether the highest value is on the diagonal (matching image-label pair) and if semantic categorization can be seen along the diagonal based on the ordering we did.

The images used for this validation process were the images used in the production task. The labels are a single word describing the depiction. For each encoder, 60 inputs were embedded in the order displayed in table 1. The following RDMs were produced for this process:



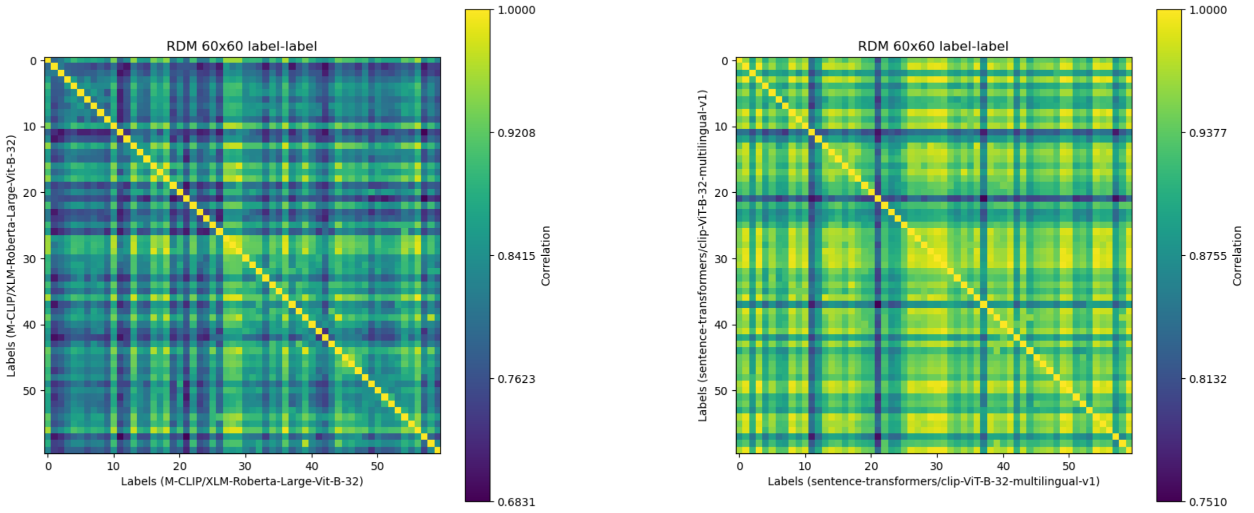
(a) CLIP ViT-B-32 Multilingual image encoder

(b) ViT-B-32 image encoder

(c) ViT-H-14 image encoder

(d) LAION-2B text encoder

Figure 3: Comparison of Model RDMs (Part 1 of 2) showing 3 image encoders and 1 text encoder



(e) M-CLIP XLM-R Large text encoder

(f) multilingual text encoder

Figure 3: Comparison of Model RDMs (Part 2 of 2) showing the remaining 2 text encoders.

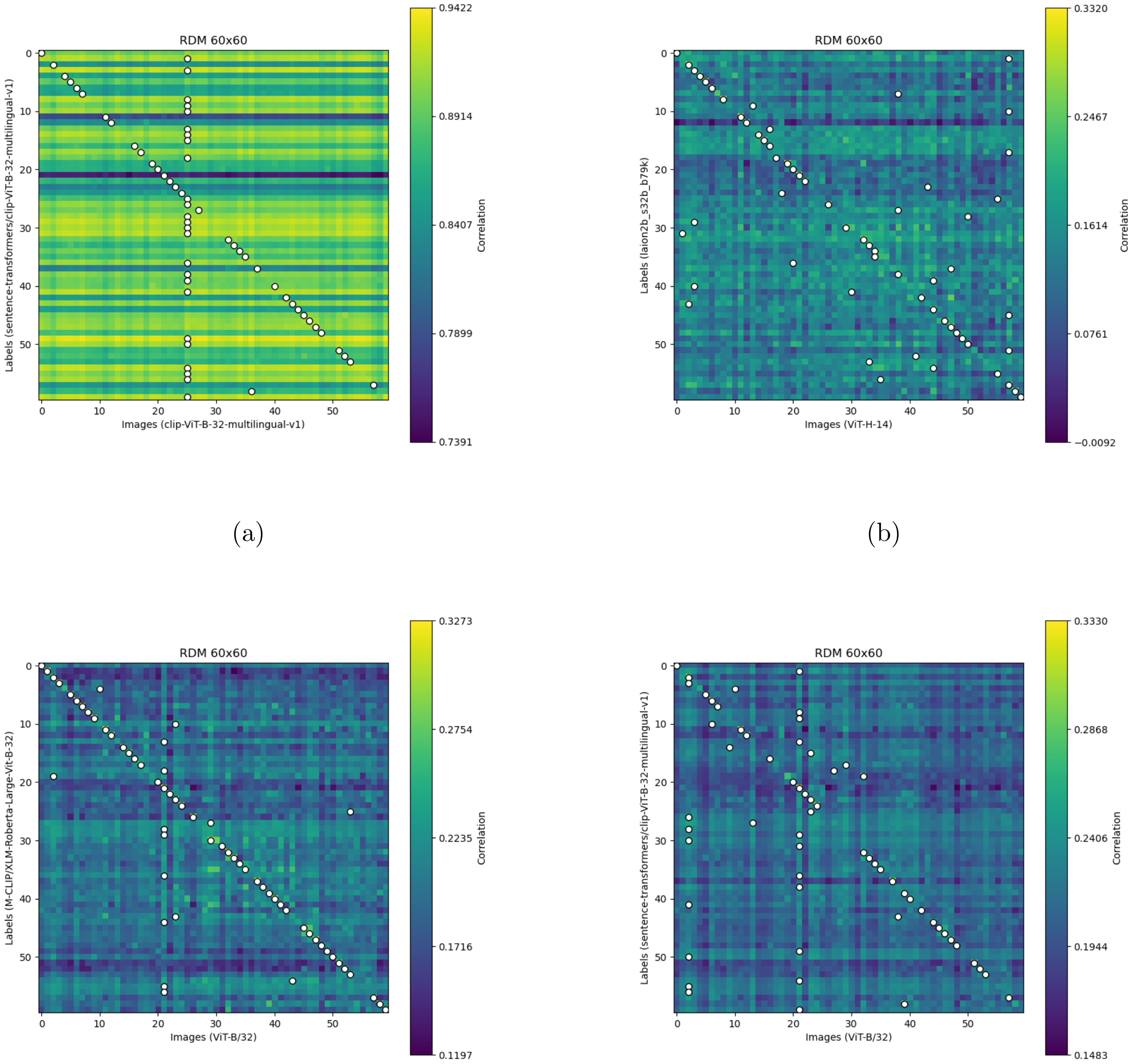
Comparing the RDMs of each image encoder with one another, for figure 3b the correlations look best, since correlations between semantic groups can be observed along the diagonal as brighter squares, while low correlation for other word pairing is perceived. For the encoder in figure 3a, there are strong correlations between pairs that semantically have no strong correlations as can be seen by the amount of squares not along the diagonal. The image encoder in figure 3c depicts some semantic correlation along the diagonal, however the image embeddings are not as clearly distinguishable as in figure 3b.

The results for the text encoders show that for figure 3d some semantic grouping around the diagonal can be found, however no strong correlations. Figure 3e shows strong correlations, however also for pairs not relevant. Figure 3f shows very strong correlations all around, making it less viable.

Based on these results, the pairings to be tested will be the following: the multilingual image and text encoder since these have been trained together. The original ViT-32 image encoder has the best results, and therefore will be paired with the Roberta transformer and the multilingual text encoder. Finally, the ViT-H encoder and laion text encoder are paired since these output both to a higher dimension and might have better distinctions between semantic groupings, these too have been trained together as well. The validation of the encoders was necessary to see if different encoders from different pairs could work together, since it could be the case that only one out of the two encoders would interpret the data meaningfully.

3.1.4 Validation pairs

Setting up different image-sentence encoder pairs using the same images and labels as used previously to check the similarity by embeddings for each encoder. This validation is needed, since eventhough individual text or image encoders can perform well, their embeddings might have been overfit to the training data. By performing this validation, we get to see whether new data pairs can also show meaningful results. The following results were observed, note that for each row in the RDMs the highest value is highlighted with a white dot:



(a)

(b)

(c)

(d)

Figure 4: Text-image encoder RDMs

Firstly, an indication of a good pairing and meaningful results would be that most of the white dots will be visible on the diagonal, as the model correctly predicts which label matches to which image the most. The results shown in figure 4a show a clean diagonal except for one image that is correlated with a number of labels. After removing this image from the data, these labels would correlate to two other images. Repeating this process these labels would then continue to correlate with other images, this indicates that this setup would wrongly predict many image-text pairs and would be unreliable. Figure 4b still shows too many mismatches all over the place while in figure 4c a clear diagonal can be seen and have provided the best results by any pair. Figure 4d have similar results but more mismatches than in figure 4c. Therefore the image-text encoder pair from figure 4c will be used to provide the embeddings for the image-sentence pairs.

3.2 Cognitively aligned model

Since we are also interested whether a model more aligned to human cognition can improve these embeddings, such a model will be explored as well. Since humans process information sequentially, as do RNNs, we will use this model type. First, a closer look at the Long Short-Term Memory (LSTM) network will be taken, since this model has much in common with how humans process information, mainly its sequential approach is of interest here. We will specifically look at ELMO which can be used to explore this problem.

3.2.1 LSTM

Examining the architecture of the LSTM, clear inspiration and foundations from human cognition can be identified. The LSTM processes information based on the prior seen context, which resembles working memory in humans. Figure 5 illustrates the architecture of an LSTM.

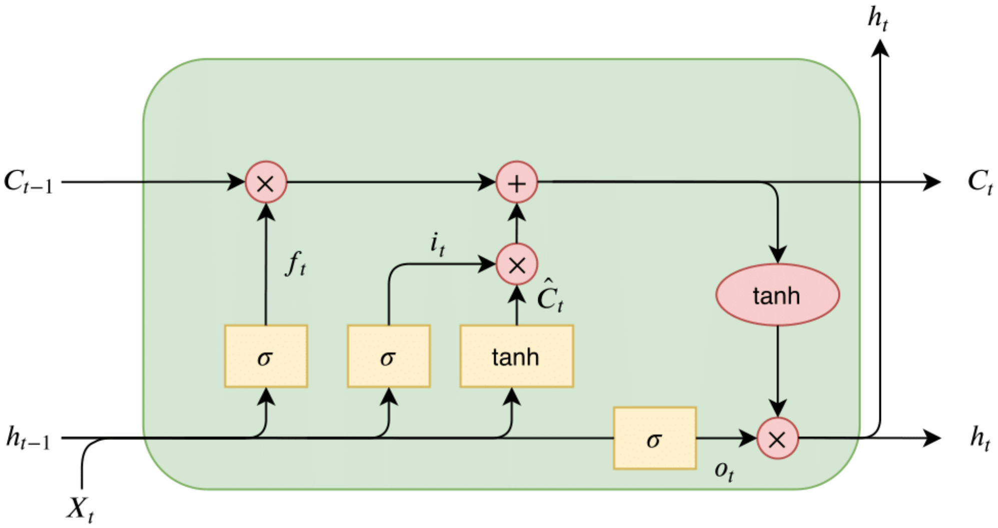


Figure 5: The architecture of a single cell part of a LSTM network [Ing21]

In figure 5 depicted by C_{t-1} and C_t is the cell-state at timestep t, which is a data pathway through the entire sequence, mimicking working memory. The forget gate is depicted as f_t. The σ describes the sigmoid activation with outputs between 0 and 1. The forget gate is in control of how much information contained by the cell state passes through to the long-term memory onto the next state, based on the current input x_t and hidden state h_{t-1}, and thus is in control of how much information is remembered. This aligns with inhibition in human cognition.

The input gate i_t determines how much of the new information $\tilde{C}_t$ is stored in the long term memory or cell-state C_t and mimics selective attention in humans. $\tilde{C}_t$ is encapsulated by a tanh function with outputs between -1 and 1 as the new information could contradict the already stored information. The output gate o_t decides what output should be revealed to the next layer based on the current state.

3.2.2 ELMO

For this thesis, the ELMoForManyLangs [CLW+18] was chosen as model to align with human cognition. This model is a bidirectional language model. A clear representation of its architecture is shown in figure 6. After feeding the model text as input, word vectors will first be computed by the character-CNN. These vectors will then be fed to the bi-directional LSTM. Since we are interested in the unidirectional processing feature of an LSTM, the text will only be fed into the forward layers, which processes the sentence in-order. The backward layers will be discarded, such that these embeddings will not be used. The first forward layer will capture syntax, aligning to the cognitive task of parsing and structure building. The second layer captures semantics.

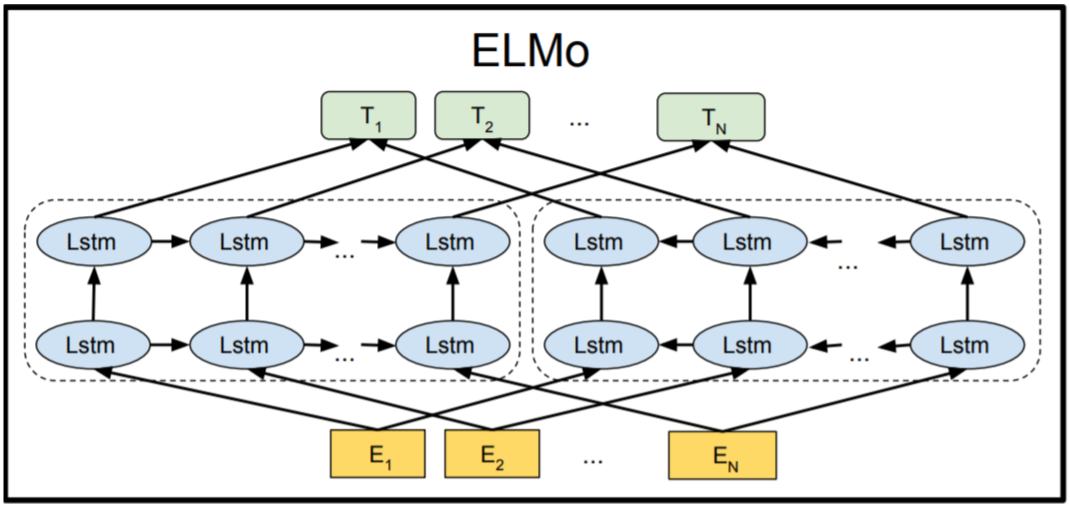


Figure 6: Architecture of ELMo[PNI+18] [DCLT19]. ELMo uses trained left-to-right and right-to-left LSTMs combining the vectors at the end. This thesis will only use both forward LSTMs and discard the backward LSTM layers embeddings for the output.

3.2.3 Validation

To ensure that only the forward layers are used, this will have to be validated. In our CLIP model we validated that the usage of the encoders would be meaningful by first checking for embedding dissimilarities using RDMs and semantically grouped labels after which we assembled the image and text encoder pairs. The validation process for ELMo looks different that that of the CLIP model, as the main concern is that the embeddings of words w remain the same even in two sentences x, y where every word up until n is equal. This means that for sentence x with word order $x_1, x_2, x_3, ...x_n, x_{n+1}, ..x_{n+i}$ and sentence y with word order $y_1, y_2, y_3, ...y_n, y_{n+1}, ..y_{n+i}$ where the words up until index n have constraint $x_i = y_i$, the embeddings up until that point should be the same for both sentences. An example would be that the embedding for the sentence "Dit is een voorbeeldzin" and "Dit is een voorbeeldzin met meer tekst" would have the same embeddings for the part of "Dit is een voorbeeldzin". Examples of sentences used to validate this model is shown in table 2. No history should be tracked and therefore will not change the value of the embeddings. The ELMo model has an implemented function which allows the embeddings after each layer to be observed, and since the forward and backward layers embed the input separately, the output of the second forward layer can easily be extracted. The model was tested using five sentences. For each of these sentences the embeddings showed to be equal.

Sentence 1	**Sentence 2**
This is an example sentence	This is an example sentence with extra text,
The cat sits on the mat	The cat sits on the mat and stares outside.
The weather is nice	The weather is nice and sunny.

Table 2: Example sentences used to validate the ELMo embeddings. The full sentence in colum 'Sentence 1' is used to compare against the part in 'Sentence 2' which is similar. The sentences used for validation are in Dutch. Note that no final dot is present in the sentences in the first column, as this will change an embedding of a sentence.

3.2.4 ELMo setup

The ELMo model will be used in a setup which does not include the CLIP encoders. In this setup, the ELMo model will embed both sentences and labels. Afterwards, the cosine distance between these embeddings will be calculated; this step mirrors that of the CLIP setup. The ELMo model can substitute a text encoder in the CLIP setup, however since this pair has not been trained together, the cosine similarity and surprisal outputs will likely be meaningless.

3.3 Prediction

The models mentioned in the previous sections will run the sentence-pairs and output a cosine distance, this will be used to input to our statistical model which will in turn predict the RTs correlating to those distances. The model setups are (1) the CLIP setup with a transformer text encoder and image encoder outputting text and image embeddings, (2) the transformer text encoder which will output embeddings for both the sentence and label pairs and finally, (3) the ELMo model which will output sentence and label embeddings. After the corresponding cosine distances have been calculated, these will be used as input in a linear mixed model, the linear mixed effects regression (lmer) model.

3.3.1 Statistical testing

Lmer models are an established standard, as these models tend to grasp the structure of the data better than standard regression. Besides, an lmer model prevents the 'Simpson's Paradox' from occuring, in which a model can be tricked by a group average. The lmer model will show the relation between fixed effects and account for variance by the random effects, like the condition or participant.

Three lmer models will be setup where each model uses the cosine distance of the corresponding LLM setup. Each of these lmer model will use this cosine distance as fixed effect. The random effects that will be added to the equation are the participant and the label of the image that was presented. The general equation is presented below, in equation 7. The cosine distance is the only fixed effect since it is assumed that if the cosine distance changes, so will the reaction time. The random effects present the variation in the model that we want to control for. There is a variation in each participant that can not be controlled for. For example, the participant could have had a rough night before participating in the experiment, which could have resulted in slower RTs. The final word could also cause variation since some words might be easier for a participant to name than others. As an example: “Cow” might be easier to name than “Miter”.

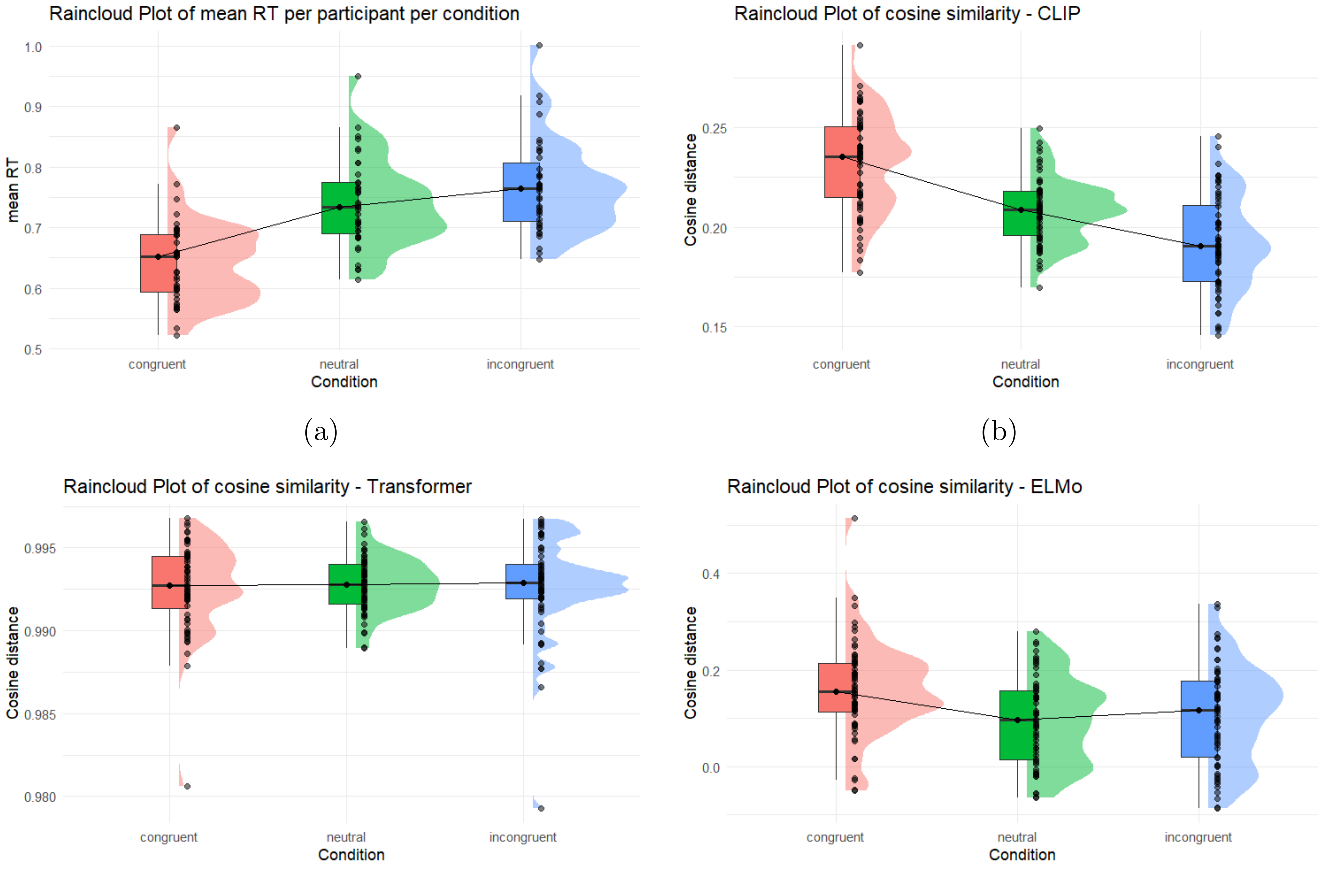
$$log(namingRT) \sim demean(cosine\ distance) + (1|subject) + (1|target\ picture) \tag{7}$$

4 Results

After having established how the experiment has been set up, the findings of our experiments are highlighted in this section. First we will take a closer look at the cosine distances that are between the sentence and target picture for each model. The cosine distance is a good indicator of how semantically close the embeddings are from each other. Secondly we will take a look at the surprisals that can be calculated for the transformer and ELMo embeddings as this is a probabilistic method which computes preferability for a target label. We will also consider the spearman correlation as this is a ranked based statistical method measuring the strength of a relation. Finally, we will dive into the linear mixed model results as these will show us correlation and whether this is by random chance or significant.

4.1 Model embeddings

As mentioned before, the models each deliver embeddings with which cosine distances can be computed. By plotting the cosine distance of each sentence-image or sentence-label pair per condition, a clear indication can be established whether the model is able to distinguish these classes correctly, meaning that the relation of the difference in cosine distance between classes is similar to the differences in response times per condition by the participants of the experiment. In the experiment, on average the highest response times are recorded in the incongruent class, followed by the neutral class and finally the lowest response times in the congruent class. Expected is to see the inverse relation for the cosine similarity, as a higher cosine similarity indicates that the words are close to each other in the vector space and thus semantically are closer to each other requiring a shorter response time.



(a)

(b)

(c)

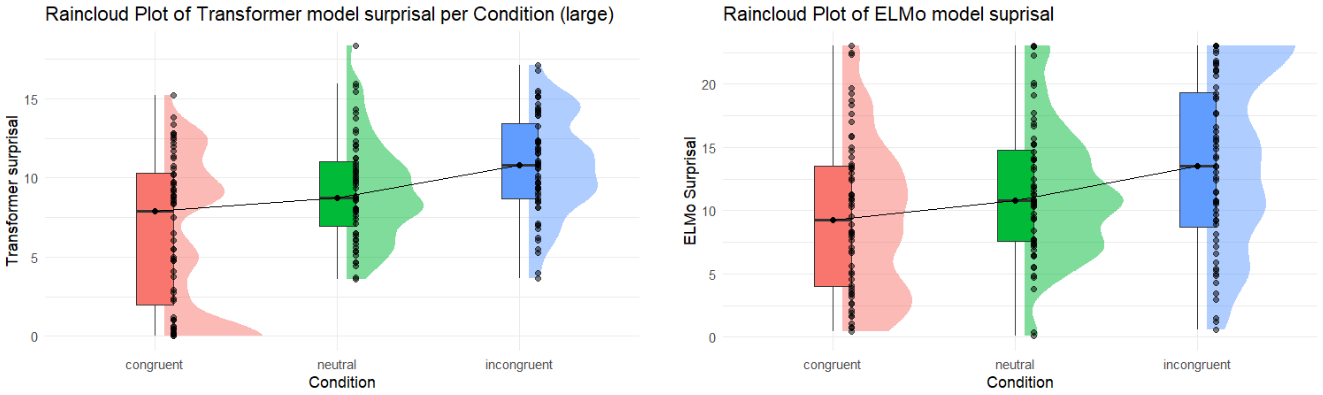
(d)

Figure 7: Raincloud for each sentence-image or sentence-label pair graphed per condition (congruent, incongruent or neutral), scattered items, box-plot and distribution which are visible in this plot

From figure 7 we can observe that the relation for the CLIP model in 8b is inverse to that of the participants in the experiment as shown in 8a. The transformer model, Roberta, was unable to distinguish the difference in 7c, this is due to anisotropy, where the vectors output by the model have no meaning in comparison to each other and are pointing in the same direction. This is likely due to the model not being trained using the cosine distance. The cosine distances embedded by the ELMo model in 7d show that neutral embeddings are perceived as least similar, whilst this should have been for the incongruent case.

4.2 Surprisals

To take a leap back to our background study, the surprisal as described in 1, has been a viable method to assist in predicting mean response times. Using the surprisal we can calculate the unexpectedness of finding the provided target. Since CLIP is using images as targets, calculating a surprisal would be very costly as many images would have to be embedded, since a full 'vocabulary' has to be used to calculate the surprisal. The surprisal of the transformer model can be calculated by appending all sentences with a mask token and using that token as a way to substitute all different words from the vocabulary used during training. The ELMo surprisal uses a constrained vocabulary. Since most libraries use transfer learning, the 'head' of the model is gone, meaning that the true full vocabulary can not be accessed. Therefore the vocabulary used in this thesis is used. A fair comparison between the models is therefore also not possible.



(a)

(b)

Figure 8: The surprisal raincloud plot per condition. (a) shows the surprisal of the Roberta transformer on a full vocabulary (b) shows the suprisal of the ELMo on our constrained vocabulary.

From the surprisals we can conclude that the upward trend is visible along the conditions indicating that the models can distinguish the different conditions correctly as higher surprisal scores are expected for incongruent pairs over other pairs.

4.3 Spearman correlation

For each of the models, a spearman correlation value was computed vs the average human RT. The spearman correlation (ρ) tests for a monotonic relation, by assigning each data point a rank after which each of these are compared to each other. The spearman correlation cares about the trend more than the specific values. When running the spearman correlation test for our models we get the values as shown in table 3.

Model	**Spearman correlation (ρ)**	**p-value**
CLIP	-0.5222054	2.2e-16
Transformer	0.07187465	0.3373
ELMo	-0.3373376	4.17e-06

Table 3: Spearman Correlations and corresponding p-values for each model vs average human response time

From this table it can be deduced that each model has a negative relation with the human response times except for the transformer model. Since the p-values for each correlation are lower than 0.05, for each case the relation is statistically significant again with an exception for the transformer model as the p-value is very high. Only for the CLIP model the absolute value is bigger than 0.5, meaning that the relation between the CLIP cosine similarity and human response times is strong.

The absolute value of the spearman correlation of the ELMo model is bigger than 0.3 meaning that there is a moderate relation.

4.4 Scatter plots

Using lmer models, each model output will be fit to the data of human response times, to understand if a clear relation between the two outputs can be established. For each lmer model we set up the equation as shown in equation 7. In each case $demean(cos\_sim)$ represents the Z-score standardization for which the cosine similarity scores for each model is first centered and then scaled using the standard deviation. This step is needed, since it makes the output easier to interpret, as we are able to understand how much of a jump the data makes every standard deviation unit from the mean. Following this translation, the following regression models were created, as shown in figures 9, 10 and 11.

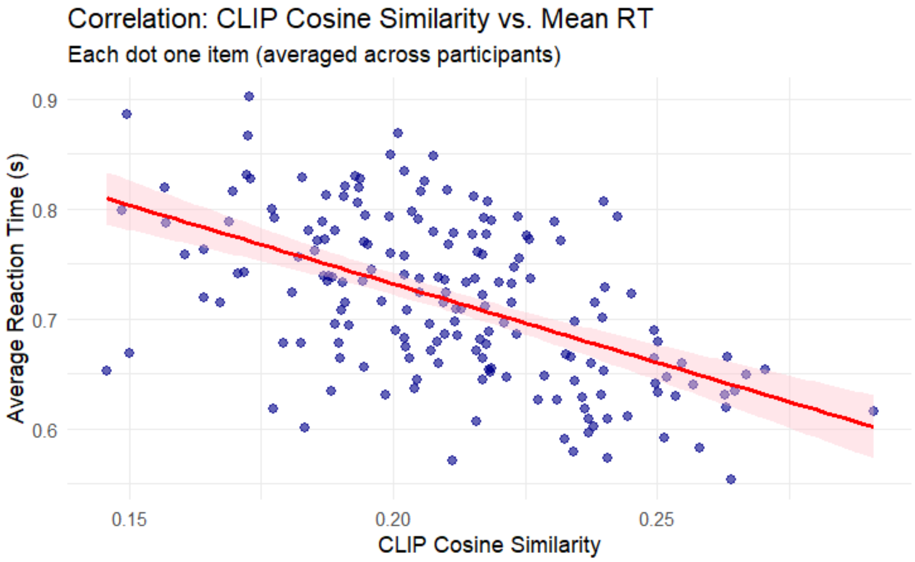


Figure 9: Scatter plot of CLIP cosine similarity vs Average RT

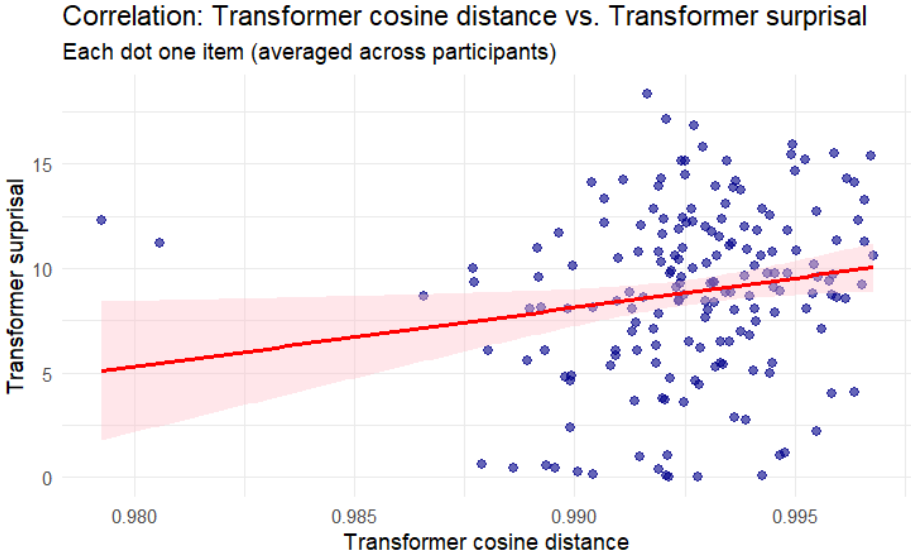


Figure 10: Scatter plot of Transformer cosine similarity vs Average RT

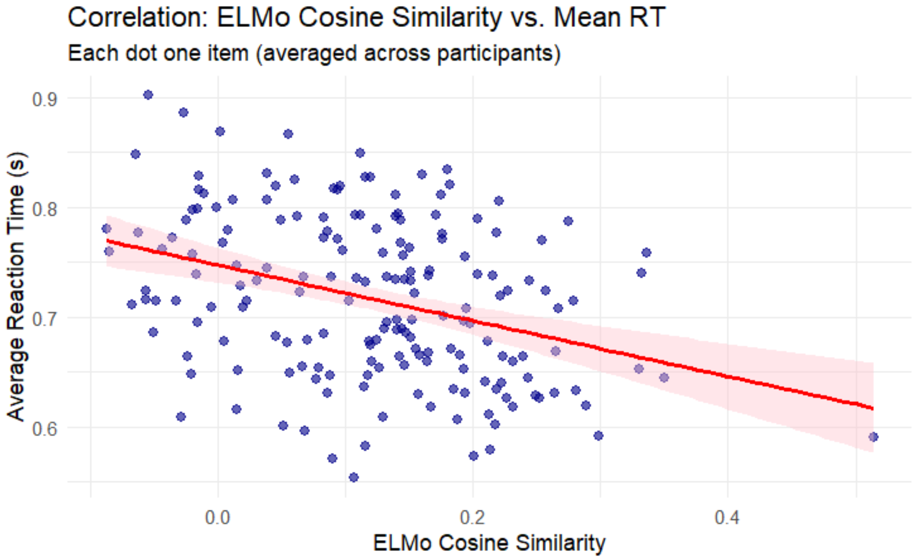


Figure 11: Scatter plot of ELMo cosine similarity vs Average RT

From these regression models, no clear correlation can be found as the points are dispersed. The CLIP model in figure 9 still showed the best correlation and negative trend and the points show a meaningful trend with some variance along the trend line. The Transformer model showed to provide useless results as the points are scrambled all over the place and the trend line looks to be a random guess. The ELMo model correlation shows a negative trend as well, however with a wider variance than that of the CLIP model.

4.5 Model comparison

Model comparison was performed using Bayesion Information Criterion (BIC). BIC is a relative metric, distinguishing different models. The model BIC value is determined using the goodness of fit and penalty for complexity measures. A lower value is preferred. Table 4 shows the BIC values for each model based. Note that the models are based on equation 7.

Model	**BIC**
CLIP	1920.939
Transformer	2198.951
ELMo	2084.349

Table 4: BIC values for the different models based on the lmer equation

From the table clearly can be deduced that the CLIP model is most preferred.

The linear mixed effect models show us the direction and trend of each model based on its random and fixed effects. We focus on the fixed effects, cosine similarity for the CLIP model and ELMo model, surprisal for the Transformer, as these provided better model indications. The lmers gave us the following outputs:

Model	**Intercept**	**Slope**	**t-value**
CLIP	-3.627e-01	-6.655e-02	-17.05
Transformer	-0.362859	0.073391	13.41
ELMo	-0.36261	-0.05079	-11.04

Table 5: lmer model fixed effects. CLIP model based on cosine similarity. Transformer and ELMo model based on surprisal.

From the table we can conclude that for each model the intercept is negative. For the cosine similarity this is expected. For the surprisal this is not expected, as the relation between the conditions and congruency classes have a similar trend to that of the mean RT of the participants. For the transformer model a positive trend was expected as the surprisal was used in the lmer. Therefore, the model embeddings are inaccurate. For each model the t-value is either bigger than 2 or smaller than negative 2 indicating these results are not obtained by random chance.

5 Discussion

In this study, we explored whether LMs can help predict response time behavior depending on different pairs of congruence. The pair would consist of a sentence and a label or image. This pair can either be congruent, incongruent or neutral. Three different model setups have been evaluated to address this problem, a CLIP model, transformer model and ELMo (LSTM based) model. The last setup is an approach for which the model architecture aligns to human cognition, by sequentially processing data, to get an insight whether such a setup would improve model embeddings. The first setup is a CLIP model using the Roberta text encoder and the ViT-B-32 image encoder. The second setup used in this research is a transformer setup using the Roberta text encoder to encode the sentence and corresponding label. The final setup uses the first two forward layers of the ELMo model to encode the sentence and paired label.

To evaluate our results we compared the cosine similarity between each model using the BIC metric. This metric showed that the CLIP model was most preferred over the other models. Besides, our lmers showed that the CLIP model has a strong inverse trend in comparison to the mean RT of each participant. The transformer model was unable to distinguish the classes and had no meaningful results. Our human aligned model, the ELMo model, did find a slight negative relation between its surprisal and average RT of each participant. The raincloud plots depicting the distribution of cosine distance per condition showed that the CLIP model performed best in distinguishing each congruency class. The ELMo model performed worst at this task, indicating that the similarity between incongruent sentence-label pairs is higher than that of neutral pairs. This does answer the question whether the human-aligned model embeddings can improve our predictions, which it does not. When performing regression, our results showed that no real correlation can be found for this or any of the other models. Even though a negative trend could be found, the data points are dispersed showing unreliability. The spearman correlation did indicate that the CLIP model and the human RTs have a strong relation not based on random chance. Although for the other two models the correlations were weak and moderate. We can conclude that model embeddings can be used to predict response times, however our LSTM model aligning to human cognition did not improve these embeddings.

There is room for a lot of improvement in this thesis. Firstly, a better VLM can be chosen for this task. CLIP was one of the first VLMs that was created, however nowadays there exist better models already, such as the gemini model. Secondly, a better transformer can be used for comparison. In this thesis the same transformer was used in the clip setup and as separate model. The results for the transformer model showed to be lacking, however these models are known to have a good semantic understanding because of the self-attention mechanism. Thirdly, beforehand it could have been known already that an LSTM will not outperform a transformer or CLIP model, however since this was of interest to the study, this approach had to be evaluated. For the CLIP model, hyperparameters such as the output dimensionality can be optimized as well. In this thesis, a 512 output dimensionality has been a set parameter, this can be scaled up or down as well in order to improve our results even more.

The field of AI is ever changing, and better models do show up from time to time. This thesis tried to answer the question if model-based representations can be used to human response times in a language production task. Although the results in this thesis did not show a strong correlation between the model representations and the average human RT, not every option is explored. CLIP has shown to be a good choice for a vision language model (VLM). However there are better VLMs such as Gemini [TG23] or InternVL[CWW^{+}23]. Besides, in this thesis the models are not trained, rather existing models are used to infer on new data. Another way to improve the model representations would be to train the model first on this data, however this would be an expensive task possibly taking up a lot of time and resources.

Although the results did not indicate a good relation between the model representation and average human response time, the standard of the CLIP model used does provide a good baseline from which improvements can be made.

Appendix

The code that was used in this thesis can be found here: https://github.com/rajeevnathie/thesis

References

[CBW+23] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829, 2023.

[CERS22] Fredrik Carlsson, Philipp Eisen, Faton Rekathati, and Magnus Sahlgren. Cross-lingual and multilingual clip. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6848–6854. European Language Resources Association, 2022.

[CLW+18] Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 55–64, Brussels, Belgium, October 2018. Association for Computational Linguistics.

[CWW+23] Zhe Chen, Jiannan Wu, Pipu Wang, Wanzhen Wang, Guo Guo, Shizhe Gong, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023.

[DBK+21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[FOGV15] Stefan L. Frank, Leun J. Otten, Giulia Galli, and Gabriella Vigliocco. The erp response to the amount of information conveyed by words in sentences. *Brain and Language*, 140:1–11, 2015.

[HAS+22] Micha Heilbron, Kristijan Armeni, Jan-Mathijs Schoffelen, Peter Hagoort, and Floris P. de Lange. A hierarchy of linguistic predictions during natural language comprehension. *Proceedings of the National Academy of Sciences*, 119(32):e2201968119, 2022.

[Ing21] Thorir Mar Ingolfsson. Insights into lstm architecture. https://thorirmar.com/post/insight_into_lstm/, November 2021. Accessed: 2025-11-29.

[JYX$^+$21] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021.

[KOBI22] Tatsuki Kuribayashi, Yohei Oseki, Ana Brassard, and Kentaro Inui. Context limitations make neural language models more human-like. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 10421–10436. Association for Computational Linguistics, 2022.

[MF21] Danny Merkx and Stefan L. Frank. Human sentence processing: Recurrence or attention? In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics (CMCL)*, pages 12–22. Association for Computational Linguistics, 2021.

[Nee77] James H. Neely. Semantic priming and retrieval from lexical memory: Roles of inhibitionless spreading activation and limited-capacity attention. *Journal of Experimental Psychology: General*, 106(3):226–254, 1977.

[NKQ$^+$25] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *ACM Trans. Intell. Syst. Technol.*, 16(5), August 2025.

[PG07] Martin J. Pickering and Simon Garrod. Do people use language production to make predictions during comprehension? *Trends in Cognitive Sciences*, 11(4):105–110, 2007.

[PNI$^+$18] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[RG19] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

[RG20] Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. *CoRR*, abs/2004.09813, 2020.

[RKH$^+$21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021.

[SBV$^+$22] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[SL21] Lavinia Salicchi and Alessandro Lenci. PIHKers at CMCL 2021 shared task: Cosine similarity and surprisal to predict human reading patterns. In Emmanuele Chersoni, Nora Hollenstein, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus, editors, *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 102–107, Online, June 2021. Association for Computational Linguistics.

[TG23] Gemini Team and Google. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[VSP$^+$17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[WGH$^+$20] Ethan Gotlieb Wilcox, Jon Gauthier, Jennifer Hu, Peng Qian, and Roger Levy. On the predictive power of neural language models for human real-time comprehension behavior, 2020.

[WKJ18] Lin Wang, Gina Kuperberg, and Ole Jensen. Specific lexico-semantic predictions are associated with unique spatial and temporal patterns of neural activity. *eLife*, 7:e39061, dec 2018.