Comparison between eye-tracking data with word embeddings

Yuqin Zhou
Univeristy of Copenhagen
qvk729@alumni.ku.dk

Abstract

Gaze data provides valuable information about where humans are focusing their attention, which can be useful for improving the performance of computational models in a range of applications. Despite the potential benefits of using gaze data, it is not yet clear how to effectively and robustly incorporate gaze data into computational models. In this study, we investigate the semantic properties of gaze features and evaluate the effectiveness of classical machine learning models using gaze data. Our results suggest that gaze data encodes some preliminary semantic meanings of words and its the potential to improve the performance of computational models [1].

1 Introduction

Gaze data provides insight into the underlying cognitive processes involved in language and image understanding, and has been shown to improve the performance of models in a range of applications (Barrett et al., 2016; Sugano and Bulling, 2016a; Karessli et al., 2016). Such model improvements are typically achieved by combining gaze with other types of data (e.g., pre-trained word embeddings, fMRI, and EEG data) to provide additional information about human cognition.

Despite the potential benefits of using gaze features in computational models, there are still many open questions about how to effectively incorporate gaze data into these models. One challenge is that gaze data is often noisy and difficult to interpret. To the best knowledge of us, it is not yet clear whether gaze features in vector space have any semantic properties or how they might contribute to the effectiveness of a model without the help of other types of data.

[1]The code is available at https://github.com/yuqinzhou9/CS3.

In this paper, we firstly explore whether word-level gaze features have similar properties of pre-trained word embeddings in Sec. 3 by visualizing gaze data in low-dimensional space and investigating whether word relations are encoded in gaze data. Meanwhile, we present an extensive experimental evaluation to examine the effectiveness of gaze data on classical machine learning or neural network classifiers in Sec. 4.

Our main **contribution** is that we are the first to explore whether gaze features have any semantic properties in vector space. This provides a new perspective when combine gaze data with other types of data to create more comprehensive models of human cognition.

2 Related Work

There has been a growing interest in using gaze data in various domains and tasks in recent years. In natural language processing, gaze has been used to perform text simplification (Karessli et al., 2016), part-of-Speech tagging (Barrett et al., 2016), sentiments analysis (Mishra et al., 2018), named-entity recognition (Hollenstein and Zhang, 2019). Collecting eye-tracking information from a reader has also benefited multiple tasks in computer vision, including image captioning (Sugano and Bulling, 2016b), object detection (Shcherbatyi et al., 2015) and salient objects detection in images (Li et al., 2014) or video (Shanmuga Vadivel et al., 2015).

Despite gaze has been an increasingly popular cue to support various tasks, it is not yet clear how to use gaze data in a way that is both effective and robust to these types of challenges. One key challenge is that, while individual features of gaze data such as *fixation* and *saccade* have clear definitions, the meaning of text or image that is reflected by these features together is not yet well understood. We here study the semantic properties of gaze data

and its contribution to the effectiveness of the computational models.

3 Semantic analysis

We introduce our gaze embeddings collection in Sec. 3.1. Then, we visualize the embeddings in Sec. 3.2 and specify the simplest the semantic property of gaze embeddings in Sec. 3.3.

3.1 Gaze Embeddings

Dataset. We conduct experiments on the word-level eye-tracking datasets: The Zurich Cognitive Language Processing Corpus (ZuCo, Hollenstein et al. (2018, 2019)) contains gaze data for 990 sentences read by 18 participants during a normal reading session. A detailed description of the dataset is provided in Appendix A.1.

Features. Each English text is tokenized into a sequence of tokens. The five word-level eye-tracking features are **(1)** the total number of fixations (nFix), **(2)** the duration of the first fixation (FFD), **(3)** the sum of all fixation durations, including regressions (TRT), **(4)** the sum of the duration of all fixations prior to progressing to the right, including regressions to previous words (GPT), and **(5)** the proportion of participants that fixated the word (fixProp). Fig. 1 depicts the feature value distributions in ZuCo.

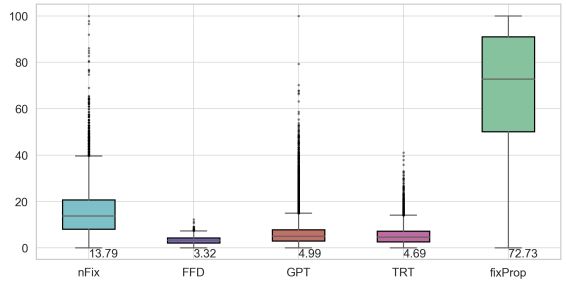


Figure 1: Boxplot showing the feature value distributions of ZuCo. Below each box is the median value of each feature. These features are averaged across all participants and scaled in the range between 0 and 100.

Gaze embeddings The gaze features of a word in different sentences vary. To obtain a fixed embedding for a word, we first discard all tokens with special characters and lowercase all remaining tokens, since this can make the results more stable. The number of remaining tokens is 16,672 and the vocabulary size is 4,400. Fig. 2 displays the top-50 frequent words in ZuCo after preprocessing. Lastly, we average all values of a same word as its embedding, for a reason we show later in Sec. 3.2.

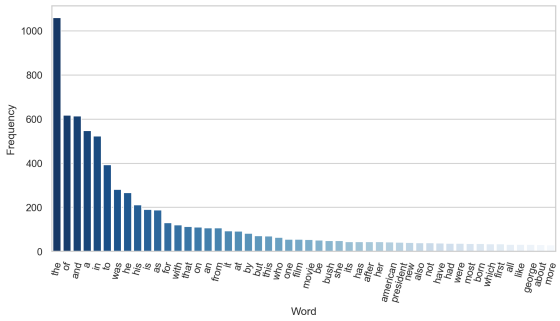


Figure 2: Top-50 frequent words in ZuCo.

3.2 Embeddings visualization

Visualizing embeddings is an important goal in helping understand, apply, and improve these models of word meaning (Jurafsky, 2000), and we here focus on a new type of embedding: gaze data.

We first use the simplest way to visualize the meaning of a word w embedded in a space, i.e. list the most similar words to w by sorting the vectors for all words in the vocabulary by their cosine with the vector for w. When we do not average all values of a same word, the similarity list does not match the results compared to using word embedding [2].

Two properties of using gaze embeddings may explain such results. First, gaze embeddings are strongly influenced by the context, and consequently an identical word has a completely different similarity lists. For example, the 5 closest words to one *secretary* in ZuCo are: *publisher*, *impressed*, *child*, *rose* and *kennedy*. The 5 closest words to another *secretary* in ZuCo are: *sing*, *questions*, *later*, *equal* and *film*. To let the gaze features of a word be more representative, one intuitive way is to combine different values of a word together. It can be done in a very computationally expensive way. For example, we can build a network model to assign weights for each gaze data of a word, and the learning objectives is the similarity list obtained from GloVe embeddings. Here, we chose the most computational-friendly function, i.e. average all values of a same word as its embedding. As shown in the following experiments, this simple pooling function is surprisingly effective to let gaze features of a word be more representative.

[2] A simple example using the GloVe embeddings is that the 7 closest words to *frog* : *frogs* , *toad* , *litoria*, *leptodactylidae*, *rana* , *lizard* , and *eleutherodactylus* (Pennington et al., 2014).

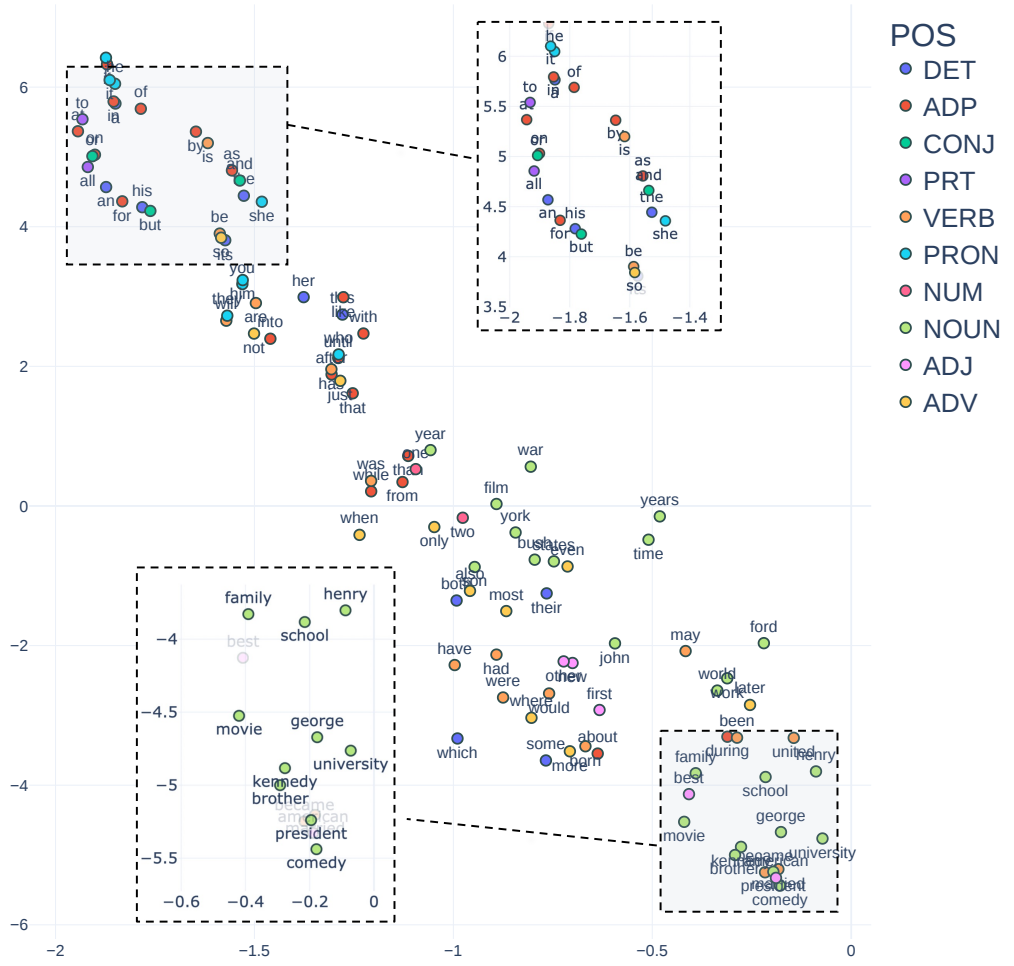
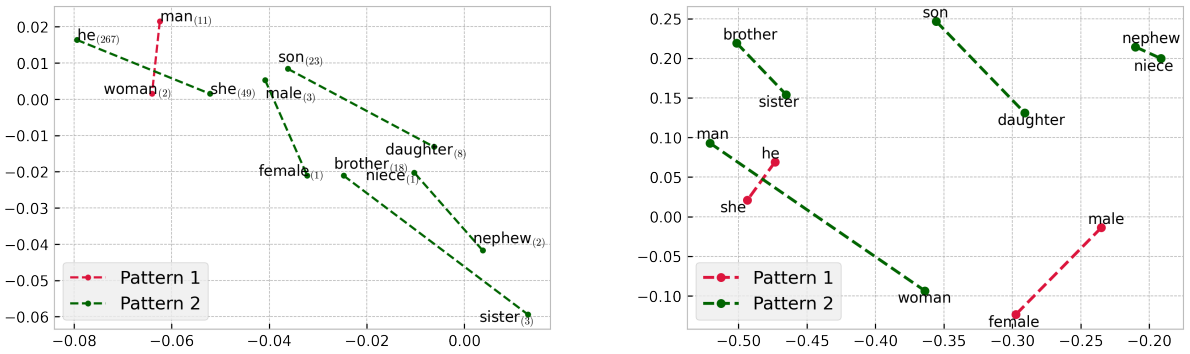


Figure 3: A two-dimensional (t-SNE) projection of embeddings for top-100 frequent words in ZuCo, showing that words requiring similar attention are nearby in space. POS here refers to the universal POS tags, and we provide a short explanation of each abbreviation in Appendix B. Since a word has multiple POS tags, we here label each word in ZuCo using the most frequent tag of a word in the Brown Corpus (Francis and Kucera, 1979).

Another property is that "similar" in gaze embedding seems not imply semantic similarity, but rather a similar degree of attention to them by the readers. For example, the 10 closest words to *was* using gaze embeddings are: *elder*, *when*, *out*, *next*, *would*, *was*, *from*, *without*, *and* and *were*. This list does not contain words with similar semantics or even some tokens of one type or (e.g., the list should include *are* and *is* if we use GloVe embeddings). In contrast, most of the tokens in the list belong to the closed class(except *elder*), which requires fewer attention compared to open classes words.

To better understand this property, we use one common visualization method, t-distributed stochastic neighbor embedding (t-SNE) (Van der Maaten and Hinton, 2008), to project the 5 dimensions of gaze data down into 2 dimensions. t-SNE is a non-linear dimensionality reduction technique that seeks to preserve the local structure of the data, so that similar data points are mapped to nearby points in the lower-dimensional space. Meanwhile, we label each word using the universal POS tags (Petrov et al., 2012). Our hypothesis is that words in the open class require more attention when read than words in the closed class. Thus, with POS tags, we can observe whether the low-dimensional projections of gaze features are distributed in a way that reflects how much attention readers pay to each word.

Fig. 3 shows that the words are distributed almost linearly, from the left-upper corner to right-



(a) Gaze features

(b) GloVe word embeddings

Figure 4: Relational properties of two types of embedding spaces after principal component analysis (F.R.S., 1901), shown by projecting embedding onto two dimensions. The subscripts to the words in Fig. 4a represent the number of times they appear in ZuCo.

lower corner. This trend is likely related to the amount of attention readers pay to a word, as we can see there are two distinct clusters of words in the figure: one belonging to the closed class and the other consisting mostly of nouns. Therefore, it can be inferred that words with more semantic meaning typically require more attention from readers, and this information can be effectively encoded by gaze features even when represented in a low-dimensional space.

After illustrating that gaze data can reflect the amount of attention given to a word, we focus on a simple semantic property of word embeddings, analogy similarity (Jurafsky, 2000), to see if gaze data can reveal any semantic relation.

3.3 Analogy Similarity

In this section, we evaluate if gaze features can capture relational meanings.

We use an early vector space model of cognition called the parallelogram model proposed by (Rumelhart and Abrahamson, 1973). Formally, given vectors $\mathbf{a}, \mathbf{b}$, and $\mathbf{a}^*$ and must find $\mathbf{b}^*$, the parallelogram method is defined as follows:

$$\hat{\mathbf{b}}^* = \arg_{\mathbf{x}} \min ||\mathbf{x}, \mathbf{a}^* - \mathbf{a} + \mathbf{b}||$$

where $\|\cdot\|$ corresponds to L^2 norm. Using this formula, we can use word embeddings like GloVe to compute an example *apple* + *tree* - *grape*, obtaining the closest answer *vine*[3]. The GolVe embedding model seems to be extracting representations of relations like MALE−FEMALE, as shown in Fig. 4b.

When it comes to gaze features, they generally outline the relation MALE−FEMALE, as words are all in the correct relative positions (except *niece* and *nephew*), as shown in Fig. 4a. The underlying concept that distinguishes *man* from *woman* or *brother* and *sister* , i.e. sex or gender, has been captured by gaze features.

However, the line between two types of words are not as parallel as that of GloVe embeddings (e.g., the line between *he* and *she*, the line between *male* and *female*). We speculate that there are two reasons for this. First, the size of ZuCo is relatively small compared to normal text datasets. Some words such as *niece* only appear once. Even if the gaze features of these infrequent words are representative enough, the parallelogram method is not as effective for these infrequent words (Linzen, 2016; Ethayarajh et al., 2019; Gladkova et al., 2016). The embedding spaces perform only well if we use frequent words, small distances, and certain relations (like relating countries with their capitals or verbs/nouns with their inflected forms) (Jurafsky, 2000).

Another reason is the information encoded by gaze features. Their goal is not to exactly capture the intricate semantic relationships between two given words by vector forms as word embeddings. They aim to reflect the activity of human minds, encoding how readers pay attention to different words. Thus, Fig. 4a shows that there is a difference when readers identify some words that contain different gender information. Such difference follow a fixed pattern in human minds but are not as fixed as the

[3]Note that this result requires to explicitly exclude the three input words and their morphological variants such as *apples*

word embedding would suggest. This observation is consistent with our intuition, as brain activity is more flexible compared to written text.

4 Effectiveness analysis

As proof-of-concept work, we evaluate the effectiveness of gaze data on a classification task using classical machine learning techniques. Our method here is to align each embedding to an individual word in a given dataset and perform a classification task on that dataset. We will describe our experiment setting in Sec. 4.1, and illustrate results in Sec. 4.2.

4.1 Experiment setting

Task, Dataset, and Metric. Our task is that, given a Twitter feed in English, determine whether its author spreads irony and stereotypes. We use the dataset provided by PAN at CLEF 2022, which was built for profiling **iro**ny and **stereo**type spreaders on Twitter (IROSTEREO)[4]. The class-balanced dataset consists of 420 XML files, and each XML file contains 200 tweets published by a twitter user. The evaluation metric is accuracy.

Text preprocessing. We use tweet tokenizer, which is a rule-based tokenizer designed to tokenize tweets. The advantage of this tokenizer is that it can avoid dividing hashtags such as `#Truth` to `#␣Truth`.

Word Alignment. After tokenizing tweets, we aline each word with embeddings, as shown in Table 1. For words that have no corresponding embeddings, we ignore these unknown words, as our goal is to see the separate effect of one type of embedding.

Embeddings	Matched	Matched Pct.
GloVe	50336	65.14%
Gaze	3911	5.10%

Table 1: Alignment between words in tweets and embeddings. Despite ZuCo is a relatively large eye-tracking dataset, the number of matched gaze embeddings is significantly smaller than that of GloVe embeddings.

Pooling functions. Since the task is to classify Twitter users, we need to use word-level embeddings to generate user embeddings. Here we have examined three computational-friendly methods as follows:

$$\mathbf{x}_{i,\ \text{sum}} = \sum_{j=1}^{n_i} x_j$$

$$\mathbf{x}_{i,\ \text{avg}} = \frac{\mathbf{x}_{i,\ \text{sum}}}{n_i}$$

$$\mathbf{x}_{i,\ \text{norm}} = \frac{\mathbf{x}_{i,\ \text{sum}}}{\|\mathbf{x}_{i,\ \text{sum}}\|}$$

where n_i represents the total number of word-level embeddings used by a Twitter user i, x_j represents the embedding of the j-th word-level embeddings, $\mathbf{x}_{i,\ k}$ represents the embedding of user i, and $\|\cdot\|$ corresponds to L^2 norm (Euclidean norm).

Baselines. We implemented three classical machine learning classifiers: Logistic Regression (LR), Support Vector Machine (SVM), and Random Forest (RF).

Hyperparameters. All models are implemented in Python 3.9 using the Scikit-Learn library (Pedregosa et al., 2011). **(1)** LR uses L_2 penalty term and the inverse of regularization strength is set to 1. **(2)** SVM uses radial basis function (rbf) kernel and squared L2 penalty. **(3)** RF builds 2000 subsamples of the dataset using bootstrap, and for each sub-sample, it randomly selects a subset of features at each split (the size of the subset is the square root of the original feature dimension).

4.2 Results and Discussion

Table3 illustrates the performance of the three classifiers using different features. If we use a single feature, such as the total number of emojis used by Twitter users, the accuracy barely exceeds 60%. With the help of gaze features, the performance of RF can be close to 70%. If we use GloVe embedding, the performance of RF can be stable above 90%. If we concatenate the gaze embedding and the glove embedding, the performance of the three models decreases slightly.

Based on these results, we can find that gaze embeddings can provide more cues than single feature in the classification task. However, they are much less informative than GloVe embeddings. This is consistent with our finding in Sec. 3.3 that gaze embedding fail to completely capture the complex semantic relationship between two words as GloVe embeddings. In addition, the concatenation of gaze features and GloVe embeddings here

[4]The information regarding PAN at CLEF 2022 is available at https://pan.webis.de/clef22/pan22-web/index.html.

Input features	LR	SVM	RF
Emoji counts	53.33%	60.48%	**61.19%**
Gaze (SUM)	67.38%	66.90%	**69.52%**
Gaze (AVG)	63.10%	49.29%	**69.04%**
Gaze (NORM)	49.76%	47.62%	**66.19%**
GloVe (SUM)	87.14%	75.95%	**90.00%**
GloVe (AVG)	86.19%	62.15%	**91.67%**
GloVe (NORM)	67.14%	64.05%	**91.19%**
Concat (SUM)	**88.81%**	65.00%	87.62%
Concat (AVG)	79.52%	48.33%	**89.76%**
Concat (NORM)	64.52%	61.67%	**90.00%**

Table 2: Performance of the three classifiers using non-embedding or embedding features. `Concat` refers to the concatenation of GloVe and Gaze embeddings.

drags down the performance of the models. This implies that although gaze features can provide additional information if incorporated into the model, we must carefully design the way this integration is attributed.

5 Conclusion

In this paper, we investigate the possibility of using gaze features as embeddings. Semantic analysis shows that the distribution of gaze data in the low-dimensional vector space has analogous similarity, reflecting the level of attention to a word. However, it fails to fully capture the semantic relationships between words as the GloVe embedding does. The effectiveness analysis shows that the use of gaze features can improve the performance of the classifiers. However, if we plan to combine gaze features with other modalities, we need to carefully design the structure of the model.

References

Maria Barrett, Joachim Bingel, Frank Keller, and Anders Søgaard. 2016. Weakly supervised part-of-speech tagging using eye-tracking data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 579–584, Berlin, Germany. Association for Computational Linguistics.

Aron Culotta, Andrew McCallum, and Jonathan Betz. 2006. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 296–303, New York City, USA. Association for Computational Linguistics.

Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. 2019. Towards understanding linear word analogies. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3253–3262, Florence, Italy. Association for Computational Linguistics.

W Nelson Francis and Henry Kucera. 1979. Brown corpus manual. *Letters to the Editor*, 5(2):7.

Karl Pearson F.R.S. 1901. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pages 8–15, San Diego, California. Association for Computational Linguistics.

Nora Hollenstein, Emmanuele Chersoni, Cassandra L Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus. 2021. Cmcl 2021 shared task on eye-tracking prediction. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 72–78.

Nora Hollenstein, Jonathan Rotsztejn, Marius Troendle, Andreas Pedroni, Ce Zhang, and Nicolas Langer. 2018. Zuco, a simultaneous eeg and eye-tracking resource for natural sentence reading. *Scientific data*, 5(1):1–13.

Nora Hollenstein, Marius Troendle, Ce Zhang, and Nicolas Langer. 2019. Zuco 2.0: A dataset of physiological recordings during natural reading and annotation. *arXiv preprint arXiv:1912.00903*.

Nora Hollenstein and Ce Zhang. 2019. Entity recognition at first sight: Improving ner with eye movement information. *arXiv preprint arXiv:1902.10068*.

Dan Jurafsky. 2000. *Speech & language processing*. Pearson Education India.

Nour Karessli, Zeynep Akata, Bernt Schiele, and Andreas Bulling. 2016. Gaze embeddings for zero-shot image classification.

Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. 2014. The secrets of salient object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 280–287.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 13–18, Berlin, Germany. Association for Computational Linguistics.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Abhijit Mishra, Srikanth Tamilselvam, Riddhiman Dasgupta, Seema Nagar, and Kuntal Dey. 2018. Cognition-cognizant sentiment analysis with multitask subjectivity summarization based on annotators' gaze behavior. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2089–2096, Istanbul, Turkey. European Languages Resources Association (ELRA).

David E Rumelhart and Adele A Abrahamson. 1973. A model for analogical reasoning. *Cognitive Psychology*, 5(1):1–28.

Karthikeyan Shanmuga Vadivel, Thuyen Ngo, Miguel Eckstein, and BS Manjunath. 2015. Eye tracking assisted extraction of attentionally important objects from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3241–3250.

Iaroslav Shcherbatyi, Andreas Bulling, and Mario Fritz. 2015. Gazedpm: Early integration of gaze information in deformable part models. *arXiv preprint arXiv:1505.05753*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Yusuke Sugano and Andreas Bulling. 2016a. Seeing with humans: Gaze-assisted neural image captioning.

Yusuke Sugano and Andreas Bulling. 2016b. Seeing with humans: Gaze-assisted neural image captioning. *arXiv preprint arXiv:1608.05203*.

A Datasets

A.1 Zurich Cognitive Language Processing Corpus (ZuCo)

The data is described in Hollenstein et al. (2021), which we will briefly summarize here. The dataset use the normal reading paradigms from ZuCo, i.e, Task 1 and Task 2 from ZuCo 1.0, and all tasks from ZuCo 2.0.

ZuCo 1.0. It is recorded from 12 healthy adults. All the participants are native English speakers originating from Canada, the USA, UK, and Australia, of which 5 females and 7 males, and are all right-handed and between 22 and 54 years old. The eye-tracking data were recorded using the EyeLink 1000 system while participants read English sentences from the Stanford Sentiment Treebank (Socher et al., 2013) and the Wikipedia dataset (Culotta et al., 2006).

ZuCo 2.0. It is recorded from 19 healthy adults and finally discarded one. All the participants are native English speakers originating from Canada, USA, UK, Australia, and South Africa, of which 10 females and 8 males, two of them are left-handed, and three of them wear glasses. The eye-tracking data were also recorded using the EyeLink 1000 system while participants read English sentences from the Wikipedia corpus (Culotta et al., 2006).

B Universal POS tags

We briefly summarize the universal POS tags, please refer to the original paper for a detailed explanation (Petrov et al., 2012).

Open class	Closed class	Other
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

Table 3: One classification of univeral POS tags. The meanings of the abbreviations are as follows: adjective (ADJ), adposition (ADP), adverb (ADV), auxiliary (AUX), coordinating conjunction (CCONJ), determiner (DET), interjection (INTJ), noun (NOUN), numeral (NUM), particle (PART), pronoun (PRON), proper noun (PROPN), punctuation (PUNCT), subordinating conjunction (SCONJ), symbol (SYM), verb (VERB) and other (X).