CogNLP-Sheffield at CMCL 2021 Shared Task: Blending Cognitively Inspired Features with Transformer-based Language Models for Predicting Eye Tracking Patterns

Peter Vickers*, **Rosa Wainwright***,
Harish Tayyar Madabushi and **Aline Villavicencio**

Department of Computer Science
University of Sheffield
United Kingdom

{pgjvickers1, rhwainwright1, h.tayyarmadabushi, a.villavicencio}
@sheffield.ac.uk

Abstract

The CogNLP-Sheffield submissions to the CMCL 2021 Shared Task examine the value of a variety of cognitively and linguistically inspired features for predicting eye tracking patterns, as both standalone model inputs and as supplements to contextual word embeddings (XLNet). Surprisingly, the smaller pre-trained model (XLNet-base) outperforms the larger (XLNet-large), and despite evidence that multi-word expressions (MWEs) provide cognitive processing advantages, MWE features provide little benefit to either model.

1 Introduction and Motivation

Many researchers now agree that eye movements during reading are not random (Rayner, 1998); as a result, eye-tracking has been used to study a variety of linguistic phenomena, such as language acquisition (Blom and Unsworth, 2010) and language comprehension (Tanenhaus, 2007). Readers do not study every word in a sentence exactly once, so following patterns of fixations (pauses with the eyes focused on a word for processing) and regressions (returning to a previous word) provides a relatively non-intrusive method for capturing subconscious elements of subjects' cognitive processes.

Recently, cognitive signals like eye-tracking data have been put to use in a variety of NLP tasks, such as POS-tagging (Barrett et al., 2016), detecting multi-word expressions (Rohanian et al., 2017) and regularising attention mechanisms (Barrett et al., 2018): the majority of research utilising eye-tracking data has focused on its revealing linguistic qualities of the reading material and/or the cognitive processes involved in reading. The CMCL 2021 Shared Task of Predicting Human Reading Behaviour (Hollenstein et al., 2021) asks a slightly different question: given the reading material, is it possible to predict eye-tracking behaviour?

Our ability to quantitatively describe linguistic phenomena has greatly increased since the first feature-based models of reading behaviour (i.e. Carpenter and Just (1983)). Informed by these traditional models, our first model tests 'simple' features that are informed by up-to-date expert linguistic knowledge. In particular, we investigate information about multi-word expressions (MWEs) as eye-tracking information has been used to detect MWEs in context (Rohanian et al., 2017; Yaneva et al., 2017), and empirically MWEs appear have processing advantages over non-formulaic language (Siyanova-Chanturia et al., 2017).

Our second model is motivated by evidence that Pre-trained Language Models (PLMs) outperform feature based models in ways that do not correlate with identifiable cognitive processes (Sood et al., 2020). Since many PLMs evolved from the study of human cognitive processes (Vaswani et al., 2017) but now perform in ways that do not correlate with human cognition, we wished to investigate how merging cognitively inspired features with PLMs may impact predictive behaviour. We felt this was a particularly pertinent question given that PLMs have been shown to contain information about crucial features for predicting eye tracking patterns such as parts of speech (Chrupała and Alishahi, 2019; Tenney et al., 2019) and sentence length (Jawahar et al., 2019).

We therefore had the goals of providing a competitive Shared Task entry, and investigating the following hypotheses: A) Does linguistic/cognitive information that can be *predicted* by eye-tracking features prove useful for *predicting* eye-tracking features? B) Can adding cognitively inspired features to a model based on PLMs improve performance in predicting eye tracking features?

*Equal Contribution

Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, pages 125–133
Online Event, June 10, 2021. ©2021 Association for Computational Linguistics

2 Task Description

The CMCL 2021 Shared Task of Predicting Reading Behaviour formulates predicting gaze features from the linguistic information in their associated sentences as a regression task. The data for the task consists of 991 sentences (800 training, 191 test) and their associated token-level gaze features from the Zurich Cognitive Language Processing Corpora (Hollenstein et al., 2018, 2020). For each word, the following measures were averaged over the reading behaviour of the participants: FFD (*first fixation duration*, the length of the first fixation on the given word); TRT (*total reading time*, the sum of the lengths of all fixations on the given word); GPT (*go past time*, the time taken from the first fixation on the given word for the eyes to move to its right in the sentence); nFix (*number of fixations*, the total quantity of fixations on a word, regardless of fixation lengths) and fixProp (*fixation proportion*, the proportion of participants that fixated the word at least once). Solutions were evaluated using Mean Absolute Error (MAE). For more details about the Shared Task, see Hollenstein et al. (2021).

3 Related Work

Transformer architectures Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) is a Language Representation model constructed from stacked Neural Network attention layers and 'massively' pre-trained on large Natural Language Corpora. In contrast with traditional language models, BERT is pre-trained in two settings: a 'cloze' task where a randomly masked word is to be predicted, and next sentence prediction. BERT or derivative models have been used to achieve state-of-the-art baselines on many NLP tasks (Devlin et al., 2019; Yang et al., 2019). Analysis studies have shown that BERT learns complex, task-appropriate, multi-stage pipelines for reasoning over natural language, although there is evidence of model bias. XLNet (Yang et al., 2019) is an autoregressive formulation of BERT which trains on all possible permutations of contextual words, and removes the assumption that predicted tokens are independent of each other.

Similar studies To our knowledge, studies that attempt to *predict* cognitive signals using language models are fairly few and far between. Djokic et al. (2020) successfully used non-Transformer word embeddings to decode brain activity recorded during literal and metaphorical sentence disambiguation. Since RNNs may be considered more 'cognitively plausible' than Transformer based models, Merkx and Frank (2020) compared how well these two types of language models predict different measures of human reading behaviour, finding that the Transformer models more accurately predicted self-paced reading times and EEG signals, but the RNNs were superior for predicting eye-tracking measures.

In a slightly different task, Sood et al. (2020) compared LSTM, CNN, and XLNet attention weightings with human eye-tracking data on the MovieQA task (Tapaswi et al., 2016), finding significant evidence that LSTMs display similar patterns to humans when performing well. XLNet used a more accurate strategy for the task but was less similar to human reading.

Though these studies may indicate that Transformer models are not the most suited to eye-tracking prediction, they are still considered State of the Art in creating broad semantic representations and general linguistic competence (Devlin et al., 2019). As such, we hoped they would allow us to investigate Carpenter and Just's speculation that the dominance of word length and frequency for predicting eye-tracking behaviour may reduce "as the metrics improve for describing higher-level factors" like semantic meaning (1983, p. 290).

4 Experimental Design[1]

We pursued both feature engineering and deep learning approaches to the task; though both methods performed well independently, there was little improvement in predictive capability when combining their features (see Table 1). As such, we developed and submitted two models: Model 1 (Feature Rich) and Model 2 (XLNet). Additional details about the feature combinations used in our final models can be found in Appendices A and C.

4.1 Linguistic Features

Each word in the training vocabulary was encoded as a one-hot vector. Since function words are more likely to be fixated than open class words (Carpenter and Just, 1983), we included POS information generated by `Spacy` (Honnibal et al., 2020) (honouring the tokenisation in the training data). We included a a binary indicator for whether a word

[1]For reproducibility purposes, our program code (including details of hyperparameters) is available here: CogNLP-Sheffield-CMCL-2021

was the first or last in its sentence to incorporate the knowledge that first and last fixations on a *line* are 5-7 letter spaces from the two respective ends (Rayner, 1998). We generated raw frequencies (proportion per million words) and Zipf frequencies (Van Heuven et al., 2014).

Finally, concreteness norms (a measure of how 'abstract' a given word is) were included as features (mean, standard deviation, and the % of participants familiar enough with the word to accurately judge its concreteness; Brysbaert et al. (2014)). We specifically tested concreteness due to the unusually large coverage of the norms.

4.2 Reading Specific Features

Word length has been empirically demonstrated as a very good predictor of gaze features in many studies (i.e. Rayner and McConkie (1976); Carpenter and Just (1983). Duration of fixation is observed to increase for words that exceed the mean saccade length (7-9 letters), and probability of fixation is reduced for words shorter than half the mean saccade length (Rayner and McConkie, 1976). Therefore, as features we included both the raw word lengths, and categorical variables representing word length as a proportion of a mean saccade length.

Since readers may store information about adjacent words (Rayner, 1975, 1998; Barrett, 2018), we also experimented with supplying features from previous and future words to each target word.

4.3 Type Summary Statistics from GECO

Following Barrett et al. (2016), we used the monolingual data from the GECO corpus (Cop et al., 2017) to generate type-level summary statistics for each word. Specifically, we averaged the gaze features across the 12 participants who completed the reading task, and normalised these features to reflect the normalisation of the Shared Trask training data. We then averaged these values again at the type (word) level. For words present in the task training data but not the GECO data, we estimated the values using means for words in the GECO data of a similar frequency (according to the `wordfreq`).

4.4 Multi-word Expression Features

We generated an MWE lexicon and summary metrics using the Wikitext-103 corpus (Merity et al., 2016) and `mwetoolkit` (Ramisch, 2012). We chose Wikitext-103 since it provided a large variety of possible MWEs in a similar context to the ZuCo reading material (Hollenstein et al., 2020). We produced two indicator features for the presence of MWEs: a binary indicator, and a categorical variable summarising the syntactic pattern of the MWE, motivated by Yaneva et al.'s evidence that MWEs of different syntactic patterns display different eye-tracking characteristics (2017).

Following the method of Cordeiro et al. (2019), we joined component words of MWEs in Wikitext-103 using underscores (i.e. *climate change* became *climate_change*) and then generated Skip-gram word embeddings (Mikolov et al., 2013) for all single words and MWEs identified in Wikitext-103. Using the `feat_comp` function in `mwetoolkit` (Ramisch, 2012), these MWE embeddings were used to compute compositionality scores and weights (Cordeiro et al., 2019). [2]

MWEs identified in the training data were assigned MWE embeddings and compositionality information as features, and non-MWEs were assigned single word embeddings and zero values for compositionality.

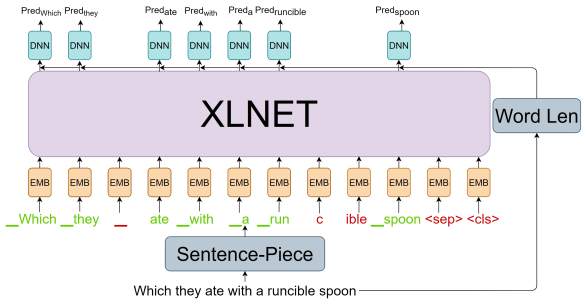

Figure 1: XLNET Feature Prediction Model

4.5 XLNet

In order to obtain Massively Pre-trained Language Model features we used XLNet. We finetuned a model that was pre-trained on BooksCorpus (Zhu et al., 2015), English Wikipedia, Giga5 (Courtney Napoles, Matthew R. Gormley, 2012), ClueWeb 2012-B (Callan et al., 2009), and Common Crawl text (Crawl, 2019). For predictions, we took the final hidden representation of the first sub-word token encoding of each word. We concatenated this feature with an integer representing the total word length in characters to encourage the model to explicitly attend to word length. We tested the effectiveness of sub-word aggregation but found this

[2] The score represents the degree to which the meaning of the MWE can be worked out from the meanings of its constituent words (i.e. 'climate change' has high compositionality, 'cloud nine' has low compositionality), and the weights estimate the semantic contribution of each word in the expression.

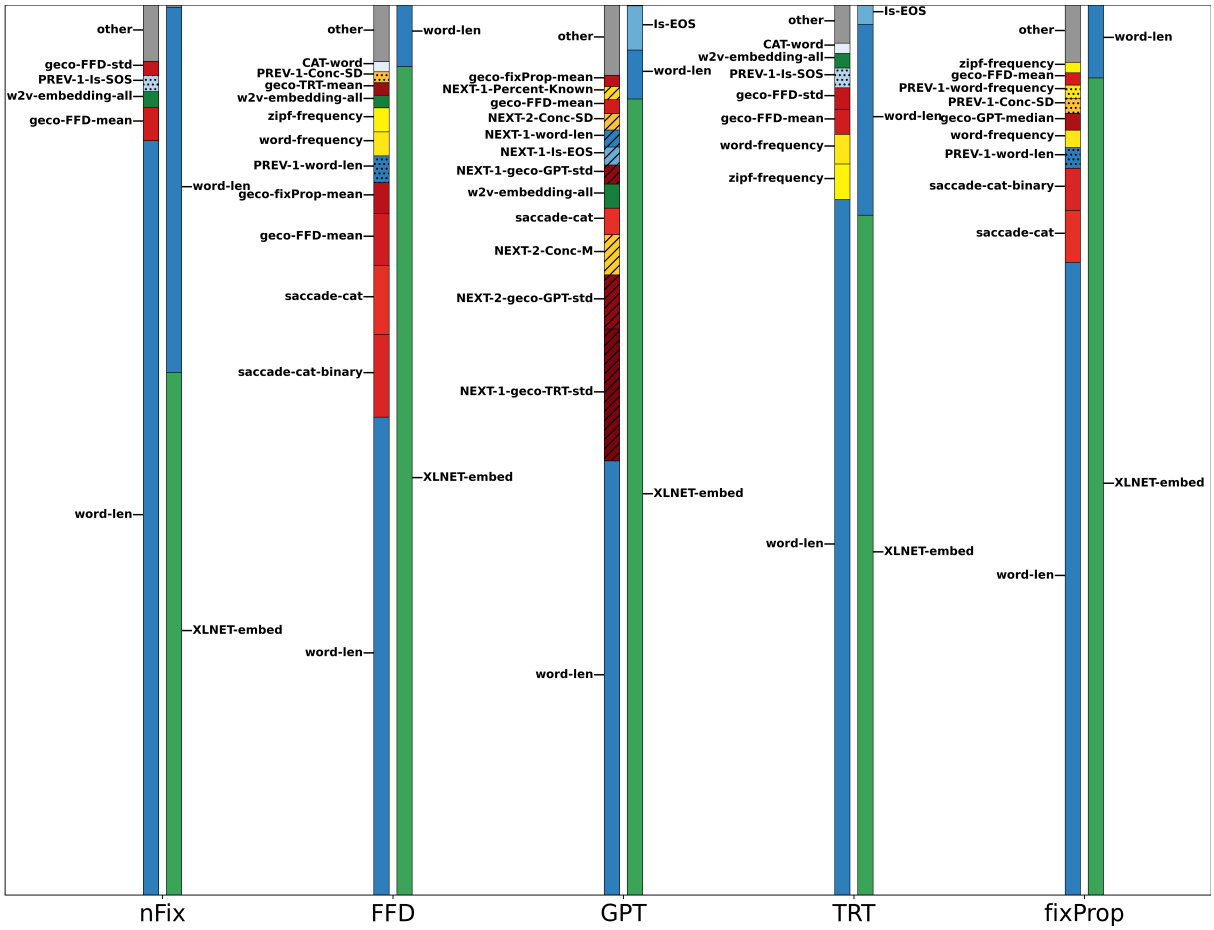

Figure 2: Feature Importance by Target for Model 1 (Left) and Model 2 (Right).

reduced the model's accuracy by an average of 0.04 MAE, which we speculate is due loss of information in the pooling operation whilst head sub-word units already contain contextual information. We then passed the concatenated sub-word and word-length features to a 3-layer dense Neural Network which was used to predict the Shared Task's five target features. This 3-layer multi-feature Network was found to be optimal through experimentation. For stability, we used the Huber loss objective, which approximates L2 loss for small values and L1 loss for large values. We trained using the AdamW optimiser and with learning rates and training duration chosen through grid search across 3-fold cross-validation, obtaining an optimal learning rate of 0.00001 and 800 epochs.

4.6 Regressors

To form predictions for the Feature Rich model we used a Random Forest Regressor implemented by `scikit-learn` (Pedregosa et al., 2011) with parameters `[max_depth = 7, n_estimators = 100, max_features = None]`. For the XLNet model, we collected the XLNet final state embeddings (identical to those fed into the DNN in Figure 1) along with the features `[word-len, CAT-pos, zipf-frequency, Is-EOS, Is-SOS]`. We then trained `scikit-learn's ElasticNetCV` for 5-fold validation with parameters `[max_iter = 10000, l1_ratio=[0.1,0.3,0.5,0.7,1], cv=5]`.

5 Results

In Table 1 we present the MAE on validation splits of the training data. This information informed our choice of model submissions alongside a preference for models using more cognitive features.

Model/Split	1	2	3	Mean
ElasticNet(XLNet + ALL Features)	3.918	3.927	3.891	3.912
Feature Rich/Model 1	**4.017**	**4.023**	**3.981**	**4.007**
BERT-base-cased	4.030	4.045	3.977	4.012
ElasticNet(BERT-base-cased)	3.986	4.024	3.969	3.993
XLNet-base-cased	3.988	3.956	3.935	3.959
XLNet-base-cased (random init)	4.608	4.722	4.695	4.675
XLNet-large-cased	3.929	4.039	3.960	3.976
ElasticNet(XLNet-base-cased)/Model 2	**3.921**	**3.924**	**3.896**	**3.914**

Table 1: Model MAE on Development Splits

We submitted two sets of predictions from Model 2 (ElasticNet(XLNet-base-cased)) and one set of predictions from Model 1 (Feature Rich). Table 2 shows the ranking of Models 1 and 2 in

Rank	Team (model)	MAE
1	LAST	3.8134
2	TALEP	3.8328
	...	
5	CogNLP@Sheffield (XLNet/Model 2)	3.9565
	...	
7	MTL782_IITD	4.0639
-	CogNLP@Sheffield (Feature Rich/Model 1)	4.0689
	...	
-	MEAN BASELINE	7.3699
13	IIIT_DWD	9.7615

Table 2: Ranking on the CMCL Shared Task Test Data.

the overall task. Our overall standing is shown to be 5th, with an MAE delta of 0.143 behind the best model. Whilst a prediction which combined Models 1 and 2 was slightly more accurate (see Table 1), we regard this improvement as within margin of error. We therefore focussed on Models 1 and 2 separately since this allowed for clearer comparisons between the two approaches.

6 Analysis and Discussion

Our results (Table 1) support both our hypotheses introduced in Section 1.

We did not anticipate that XLNet-base would outperform XLNet-large, which had more pre-training data and layers. This is possibly due to the limited amount of training data specific to the task for fine-tuning, resulting in the larger model underfitting. We are able to confirm that the knowledge XLNet learns through massive pre-training crucial to its performance in this arena - removal of this knowledge through weight randomisation increases MAE from 3.959 to 4.675. Hence we believe that both structure and pre-training of XLNet-base contribute to its success in this task.

We use normalised permutation feature importance (see Appendix B) to better understand the value of different features and present it on a per-target basis for each model in Figure 2.

The most interesting outcome of our experiments was the fact that XLNet embeddings subsume information contained across most features except word length (especially in predicting nFix). It may be that the use of word-pieces obfuscate word length information thus requiring the explicit addition of that information. While the usefulness of features such as word length is consistent with the literature, we were surprised by the relative unimportance of MWE information given that many neurocognitive studies have demonstrated differences in how they are processed (Siyanova-Chanturia et al., 2011, 2017; Cacciari and Tabossi, 1988). An additional surprise is that even though the Skip-gram embeddings provide semantic information about single words as well as MWEs, the Feature Rich models make little use of them. Many of the Feature Rich models utilize the GECO features, which may be because they provide approximate guidance about the distributions of the various gaze features that would be difficult to learn directly given the sparsity of the training data.

7 Conclusion and Future Work

This work describes our submissions to the 2021 CMCL Shared Task: we contributed a Feature Rich model inspired by cognitive and linguistic information, and model predominantly based on contextual XLNet-base embeddings. We find that only a limited subset of the cognitive features (such as word length) are helpful in the XLNet model. To our surprise, neither XLNet-large embeddings nor MWE features provide performance improvements. However, we believe this indicates a need for further research into MWE representations as opposed to suggesting that MWEs are unimportant for creating effective cognitive models.

Acknowledgements

We are grateful to Cheng Cao, Elham Khodaei, Srivishnu Ethirajulu Krishnaraj and Ronan Ramdas Revadker for their help generating and testing the feature sets. PV and RW are supported by the Centre for Doctoral Training in Speech and Language Technologies (SLT) and their Applications funded by the UK Research and Innovation grant EP/S023062/1. RW is also supported by ZOO Digital. This work is also partially supported by the EPSRC grant EP/T02450X/1.

References

Maria Barrett. 2018. *Improving natural language processing with human data: Eye tracking and other data sources reflecting cognitive text processing*. Ph.D. thesis.

Maria Barrett, Joachim Bingel, Nora Hollenstein, Marek Rei, and Anders Søgaard. 2018. Sequence classification with human attention. In *Proceedings of CoNLL 2018*, pages 302–312.

Maria Barrett, Joachim Bingel, Frank Keller, and Anders Søgaard. 2016. Weakly Supervised Part-of-speech Tagging Using Eye-tracking Data. In *Pro-*

ceedings of ACL 2016: Short Papers, pages 579–584, Stroudsburg, PA, USA. Association for Computational Linguistics.

E. Blom and S. Unsworth. 2010. *Experimental Methods in Language Acquisition Research*. Language learning and language teaching. John Benjamins Pub. Company.

Leo Breiman. 2001. Random forests. *Machine Learning*, 45(1):5–32.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior Research Methods*, 46(3):904–911.

Cristina Cacciari and Patrizia Tabossi. 1988. The comprehension of idioms. *Journal of Memory and Language*, 27:668–683.

Jamie Callan, Mark Hoy, Changkuk Yoo, and Le Zhao. 2009. Clueweb09 data set.

P. A. Carpenter and M. A. Just. 1983. What your eyes do while your mind is reading. In Keith Rayner, editor, *Eye movements in reading: Perceptual and language processes*, pages 275–307. Academic Press., New York.

Grzegorz Chrupała and Afra Alishahi. 2019. Correlating neural and symbolic representations of language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2952–2962, Florence, Italy. Association for Computational Linguistics.

Uschi Cop, Nicolas Dirix, Denis Drieghe, and Wouter Duyck. 2017. Presenting GECO: An eyetracking corpus of monolingual and bilingual sentence reading. *Behavior Research Methods*, 49(2):602–615.

Silvio Cordeiro, Aline Villavicencio, Marco Idiart, and Carlos Ramisch. 2019. Unsupervised compositionality prediction of nominal compounds. *Computational Linguistics*, 45(1):1–57.

Benjamin Van Durme Courtney Napoles, Matthew R. Gormley. 2012. Annotated English Gigaword. *Linguistic Data Consortium*.

Common Crawl. 2019. Common Crawl.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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. Association for Computational Linguistics.

Vesna G Djokic, Jean Maillard, Luana Bulat, and Ekaterina Shutova. 2020. Decoding Brain Activity Associated with Literal and Metaphoric Sentence Comprehension Using Distributional Semantic Models. *Transactions of the Association for Computational Linguistics*.

Nora Hollenstein, Emmanuele Chersoni, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus. 2021. CMCL 2021 Shared Task on Eye-Tracking Prediction. In *Proceedings of the Workshop on Cognitive Modelling and Computational Linguistics*.

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.

Nora Hollenstein, Marius Troendle, Ce Zhang, and Nicolas Langer. 2020. ZuCo 2.0: A Dataset of Physiological Recordings During Natural Reading and Annotation. Technical report.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. 2016. Distribution-Free Predictive Inference For Regression. *Journal of the American Statistical Association*, 113(523):1094–1111.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. *CoRR*.

Lucas Mentch and Giles Hooker. 2016. Quantifying Uncertainty in Random Forests via Confidence Intervals and Hypothesis Tests. *Journal of Machine Learning Research*, 17(26):1–41.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer Sentinel Mixture Models. *Proceedings of ICLR 2017*.

Danny Merkx and Stefan L. Frank. 2020. Comparing transformers and rnns on predicting human sentence processing data.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*. International Conference on Learning Representations, ICLR.

Kristin K. Nicodemus, James D. Malley, Carolin Strobl, and Andreas Ziegler. 2010. The behaviour of random forest permutation-based variable importance measures under predictor correlation. *BMC Bioinformatics*, 11(1):110.

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.

Carlos Ramisch. 2012. A generic and open framework for multiword expressions treatment: from acquisition to applications. In *Proceedings of the ACL 2012 Student Research Workshop*, September, pages 61–66. Association for Computational Linguistics.

Keith Rayner. 1975. The perceptual span and peripheral cues in reading. *Cognitive Psychology*, 7(1):65–81.

Keith Rayner. 1998. Eye Movements in Reading and Information Processing: 20 Years of Research. *Psychological Bulletin*, 124(3):372–422.

Keith Rayner and George W. McConkie. 1976. What guides a reader's eye movements? *Vision Research*, 16(8):829–837.

Radim Rehurek and Petr Sojka. 2011. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2).

Omid Rohanian, Shiva Taslimipoor, Victoria Yaneva, and Le An Ha. 2017. Using Gaze Data to Predict Multiword Expressions. In *Proceedings of RANLP 2017*, pages 601–609.

Anna Siyanova-Chanturia, Kathy Conklin, Sendy Caffarra, Edith Kaan, and Walter J.B. van Heuven. 2017. Representation and processing of multi-word expressions in the brain. *Brain and Language*, 175:111–122.

Anna Siyanova-Chanturia, Kathy Conklin, and Norbert Schmitt. 2011. Adding more fuel to the fire: an eye-tracking study of idiom processing by native and non-native speakers. *Second Language Research*, 27(2):251–272.

Ekta Sood, Simon Tannert, Diego Frassinelli, Andreas Bulling, and Ngoc Thang Vu. 2020. Interpreting Attention Models with Human Visual Attention in Machine Reading Comprehension. In *Proceedings of CoNLL 2020*, pages 12–25, Stroudsburg, PA, USA. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, Andrew Lin, Sara Jewett, and Lance Nathan. 2018. Luminosoinsight/wordfreq: v2.2.

Michael K Tanenhaus. 2007. Spoken language comprehension: Insights from eye movements. *The oxford handbook of psycholinguistics*.

Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2016. MovieQA: Understanding Stories in Movies through Question-Answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Sam Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

Walter J B Van Heuven, Pawel Mandera, Emmanuel Keuleers, and Marc Brysbaert. 2014. SUBTLEX-UK: A new and improved word frequency database for British English. *The Quarterly Journal of Experimental Psychology*, 67(6):1176–1190.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Victoria Yaneva, Shiva Taslimipoor, Omid Rohanian, and Le An Ha. 2017. Cognitive Processing of Multiword Expressions in Native and Non-native Speakers of English: Evidence from Gaze Data. Technical report.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Y Zhu, R Kiros, R Zemel, R Salakhutdinov, R Urtasun, A Torralba, and S Fidler. 2015. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.

A Features Used

We use the following features for each model. +N and +P indicate that associated data for the two next and two preceding words were included, respectively.

A.1 Model One Features

```
[CAT-pos+N+P, CAT-word+N+P,
Conc-M+N+P, Conc-SD+N+P,
Is-EOS+N+P, Is-SOS+N+P,
Percent-Known+N+P,
comp-score+N+P, comp-weights+N+P,
geco-FFD-mean+N+P,
geco-FFD-std+N+P,
geco-GPT-median+N+P,
geco-GPT-std+N+P,
geco-TRT-mean+N+P,
geco-fixProp-mean+N+P,
geco-fixProp-std+N+P,
geco-nFix-median+N+P,
geco-nFix-std+N+P,
is-mwe+N+P, is-strange+N+P,
mwe-cat+N+P, saccade-cat+N+P,
saccade-cat-binary+N+P,
w2v-embedding+N+P,
word-frequency+N+P, word-len+N+P,
zipf-frequency+N+P]
```

A.2 Model Two Features

```
[XLNET-embed, CAT-pos, Is-EOS,
Is-SOS, word-len, zipf-frequency]
```

B Permutation Feature Importance

We use permutation feature importance (Breiman, 2001) to better understand the impact of different features on each of the different models. This method measures the base error of the model against the error when one feature is randomly permuted, allowing for quantification of importance. That is for feature i:

$$FI_i = E_{base} - E_{perm_i}$$

We note that permutation methods have a tendency of attributing higher importance to correlated features (Nicodemus et al., 2010), whilst still being informative. Alternatives include per-feature retraining (Lei et al., 2016; Mentch and Hooker, 2016) which was computationally intractable within the timeframe of the CMCL task duration.

C Description of features

Feature (generated at the word-level unless specified)	Description	Data and tools used
CAT_word	One hot word encoding	
CAT_pos	Categorical encoding of Part-of-Speech tag	Honnibal et al. (2020)
Is_EOS	Binary variable indicating if word is the last in its sentence	
Is_SOS	Binary variable indicating if word is the first in its sentence	
Conc_M	Mean concreteness norm assigned to the lemmatized form of the word. Words not covered by the dataset of norms were given a'neutral' score of 3 (concreteness rated on a Likert scale from 1-5)	Brysbaert et al. (2014)
Conc_SD	Standard deviation of concreteness values assigned to lemmatized form of word. Words not covered by the dataset of norms were assigned the mean of Conc_SD for all other words	Brysbaert et al. (2014)
Percent_Known	Proportion of participants asked to estimate concreteness norms that were familiar enough with the word to judge its concreteness. Words not covered by the dataset of norms were assigned a value of 1	Brysbaert et al. (2014)
word_len	Number of characters in the word	
saccade_cat	Categorical representation of number of characters in relation to average saccade length (categories were 1-3, 4-7, 8-10 and 11+ letters)	
saccade _cat_binary	Binary categorical representation of number of characters in relation to average saccade length (categories were 1-3 letters and 4+ letters)	
word_frequency	Frequency of word per million words	Speer et al. (2018)
zipf_frequency	Frequency of word per million words on the zipf scale	Speer et al. (2018)
NEXT_n_FEAT	Attaches FEAT for the next n words to the current word (i.e. NEXT_1_Is_EOS attaches Is_EOS for the next word to the current word)	
PREV_n_FEAT	Attaches FEAT for the previous n words to the current word	
geco_FEAT_mean	Mean average of all measurements of FEAT for this word in GECO. If the word was not present in GECO, the mean of means for words with comparable frequency in natural language was used	Cop et al. (2017)
geco_FEAT_median	Median average of all measurements of FEAT for this word GECO. If the word was not present in GECO, the mean of medians for words with comparable frequency was used	Cop et al. (2017)
geco_FEAT_std	Standard deviation of all measurements of FEAT for this word in GECO. If the word was not present in GECO, mean of standard deviations for words with comparable frequency was used	Cop et al. (2017)
is_mwe	Binary indicator showing if word is part of an MWE in this context	Ramisch (2012)
mwe_cat	Categorical representation of whether the word is part of an MWE in this context, where categories are based on syntactic patterns (i.e. adjective noun compound, verb + preposition phrase)	Ramisch (2012) Loper and Bird (2002)
w2v_embedding	300 dimensional Skip-gram embedding for the word or MWE. If the word is part of an MWE in this context, the Skip-gram embedding trained for the MWE is used instead. Embeddings are trained using the Wikitext-103 corpus, where multiword expressions are reformatted to be concatenated using underscores (i.e. *multiword_expression*)	Ramisch (2012) Mikolov et al. (2013) Rehurek and Sojka (2011) Merity et al. (2016)
comp_score	Compositionality score for the MWE calculated using mwetoolkit. Words not part of MWEs are assigned a value of 0	Ramisch (2012) Cordeiro et al. (2019)
comp_weights	Weights used for each word to calculate the comp_score for the MWE (certain words may contribute more semantic meaning to an MWE than others). Words not part of MWEs are assigned a value of 0	Ramisch (2012) Cordeiro et al. (2019)
is_strange	Binary indicator of non-standard formatting or non-alphanumeric characters in the current word (generated using regular expressions)	