LangResearchLab_NC at CMCL2021 Shared Task: Predicting Gaze Behaviour Using Linguistic Features and Tree Regressors

Raksha Agarwal, Niladri Chatterjee
Indian Institute of Technology Delhi
Hauz Khas, Delhi-110016, India
raksha.agarwal@maths.iitd.ac.in
niladri@maths.iitd.ac.in

Abstract

Analysis of gaze data behaviour has gained momentum in recent years for different NLP applications. The present paper aims at modelling gaze data behaviour of tokens in the context of a sentence. We have experimented with various Machine Learning Regression Algorithms on a feature space comprising the linguistic features of the target tokens for prediction of five Eye-Tracking features. CatBoost Regressor performed the best and achieved fourth position in terms of MAE based accuracy measurement for the ZuCo Dataset.

1 Introduction

Eye-Tracking data or Gaze data compiles millisecond-accurate records about where humans look while reading. This yields valuable insights into the psycho-linguistic and cognitive aspects of various tasks requiring human intelligence. Eye-Tracking data has been successfully employed for various downstream NLP tasks, such as part of speech tagging (Barrett et al., 2016), named entity recognition (Hollenstein et al., 2018), sentiment analysis (Mishra et al., 2018), text simplification (Klerke et al., 2016), and sequence classification (Barrett and Hollenstein, 2020) among others. Development of systems for automatic prediction of gaze behaviour has become an important topic of research in recent years. For example, Klerke et al. (2016) and Mishra et al. (2017) used bi-LSTM and CNN, respectively for learning different gaze features. In the present work, Eye-Tracking features for words/tokens of given sentences are learned using Tree Regressors trained on a feature space comprising the linguistic properties of the target tokens. The proposed feature engineering scheme aims at encoding shallow lexical features, possible familiarity with the readers, interactions of a target token with other words in its context, and statistical language model features.

2 Task Setup

The shared task is designed to predict five Eye-Tracking features namely, number of fixations (nF), first fixation duration (FFD), total reading time (TR), go-past time (GP) and, fixation proportion (fxP). ZuCo Eye-Tracking dataset is used for the present task (Hollenstein et al., 2021, 2020, 2018). The dataset contains three subsets corresponding to Train, Trial and Test which contains 700, 100, and 191 sentences, respectively. Their respective token counts are 13765, 1971, and 3554. Each input token is uniquely represented by a tuple $< sid, wid >$, where sid is the sentence_id and wid is the word_id. Mean Absolute Error (MAE) is used for evaluation.

3 Feature Engineering

For the above-mentioned task, linguistic features for a given input token are extracted in order to encode the lexical, syntactic, and contextual properties of the input token. Additionally, familiarity of the input token and its collocation with surrounding words is also modelled as explained below.

3.1 Shallow Lexical Features

It is intuitive that the lexical properties of a given input token have an effect on the amount of time spent on reading the word. Features, such as Number of letters (Nlets), vowels (Nvow), syllables (Nsyl), phonemes (Nphon), morphemes (Nmorph), and percentage of upper case characters (PerUp) in the input token are used to model shallow lexical characterstics of the target token. A feature (IsNamed) is used to indicate whether the input token is a Named Entity. The language of etymological[1] origin, e.g., Latin, French of the target token is also considered as a feature, named EtyOrig.

In addition, several Boolean features have been used for characterization of the input token. The

[1] https://pypi.org/project/ety/

Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, pages 79–84

input tokens, which are the last words of the respective sentences, are suffixed by the string <EOS>. These are identified by a Boolean feature (IsLast). The <EOS> string is removed for further feature extraction. Two Boolean features (IsNumber, Hyphen) are used to indicate whether the input token is numeric, and whether the target token contains multiple words connected using hyphens, respectively. To indicate that the input token is a possesive word, a Boolean feature is used (IsPossessive). The identification has been done with the help of POS tag of SpaCy library and presence of apostrophe. A Boolean feature (StartPunct) is used to identify inputs starting with a puntuation character, these punctuations are removed for further feature extraction. Furthermore, we have considered two sentence level features namely, the total number of tokens in the sentence (LenSent), and the relative position (Relpos) of the input token in the sentence.

3.2 Modelling Familiarity

In the present work, the familiarity of a token is modelled using various frequency based features as described below.

A Boolean feature (IsStopword) is used to indicate whether the token is a stopword or not. It has been observed that the gaze time for stopwords, such as *a*, *an*, *of*, is much less in comparison with uncommon words, such as *grandiloquent* $< 457, 20 >$, and *contrivance* $< 715, 4 >$. This feature has been extracted using NLTK's list of English stopwords.

Corpus based features are used to indicate the common usage of input tokens. A Boolean feature (InGoogle) indicates whether the input token belongs to the list of the 10,000 most common English words, as determined by n-gram frequency analysis of the Google's Trillion Word Corpus[2]. Similarly, to indicate the presence of input tokens in the list of 1000 words included in Ogden's Basic English[3], a Boolean feature (InOgden) is used.

Frequency based features are also used to model the familiarity of input tokens. The following features are used: Frequency of input token in Ogden's Basic English (OgdenFreq), Exquisite Corpus (ECFreq) and, SUBTLEX (SUBTFreq). Exquisite Corpus[4] compiles texts from seven different domains. SUBTLEX contains frequency of 51 million words calculated on a corpus of Movie Subtitles. Contextual Diversity (ConDiversity) reported in SUBTLEX is also used as a feature. Contextual Diversity is computed as the percentage of movies in which the word appears. Furthermore, frequency of the input tokens given in the L count of (Thorndike and Lorge, 1944), and London-Lund Corpus of English Conversation by (Brown, 1984) are also used as features (TLFreq, BrownFreq).

The probability of the input token calculated using a bigram and trigram character language models are also considered as feature (CharProb2, CharProb3). The probability is lower for words where letters have unusual ordering. For example, consider the tokens *crazy* $< 350, 3 >$ and *czar* $< 525, 28 >$, $CharProb2(crazy) > CharProb2(czar)$ because the letter bigram *cr* (cry, crazy, create, cream, secret) is more common than bigram *cz* (czar, eczema) amongst English words. The letter bigram and trigram probabilities are calculated using letter counts from Google's Trillion Word Corpus[5]. Suppose a word W consist of N letters $W = l_1 \dots l_N$ then, the corresponding feature value is calculated as:

$$CharProb2(W) = \frac{1}{N-1} \sum_{i=1}^{N-1} log_{10} P(l_i l_{i+1})$$

$$CharProb3(W) = \frac{1}{N-2} \sum_{i=1}^{N-2} log_{10} P(l_i l_{i+1} l_{i+2})$$

3.3 Modelling Context

There is a significant variation in the amount of time spent on comprehending the semantics of a word in different sentences. Variation in fixation time for the token *early* in different sentences is presented in Table 1. To model this variation, it is important to include features with respect to the context of the input word. Both simple Universal POS tag (UniTag) and detailed Penn POS tag (PennTag) of the input token extracted using SpaCy are considered as features. The POS of a target word depends on the context in which it appears as shown in Table 2.

Number of synsets (Nsyn), hyponyms (Nhypo) and hypernyms (Nhyper) extracted from NLTK WordNet are also used as features. These features are calculated considering the synsets having the same POS tag as the input token. The Dependency tree of a sentence helps to understand the relationship between different words of a given sentence.

[2]https://github.com/first20hours/google-10000-english
[3]http://ogden.basic-english.org
[4]https://pypi.org/project/wordfreq/
[5]http://norvig.com/ngrams/count_2l.txt, http://norvig.com/ngrams/count_3l.txt

id	nF	FF	GP	TR	fxP
$< 252, 3 >$	21.91	4.44	7.55	7.08	100
$< 618, 5 >$	15.33	4.18	4.18	5.28	83.3
$< 533, 15 >$	9.19	3.03	3.22	3.22	61.1

Table 1: Variation in fixation time for the token ***early***

sid	Sentence	Uni/Penn
366	After the show was cancelled, he played a handyman on the series The Facts of Life.	NOUN/ NN
460	A classy item by a legend who may have nothing left to prove but still has the chops and drive to show how its done.	VERB/ VB

Table 2: POS feature for the token ***show***

In this respect, the dependency tag of the input token with its syntactical head (DepTag) and, POS tag of the head (HeadPOS) are considered as features. Additionally, two features are extracted from the dependency tree, namely, depth of the input token in the tree (TokDepth), and the number of children of the input token (NChild).

3.4 Language Model Features

Statistical n-gram language models help to model collocation of words in sentences, and to determine the probability of a sequence of words. In the present work, we use a trigram language model trained on the Gigaword corpus[6] to extract two features (FragScore3, FragScore5) which measure the language model score of a word sequence containing the input token and the context words in the sentence in a window of 3 and 5, respectively.

Suppose the input sentence is denoted by $S = w_1 w_2 \dots w_N$ and w_n is the target token where $n \in 1, 2, \dots N$. Let P_3 denote the trigram language model probability then,

$$FragScore3(w_n) = log_{10} P_3(w_j \dots w_n \dots w_k)$$

$$FragScore5(w_n) = log_{10} P_3(w_r \dots w_n \dots w_t)$$

where $j = max(1, n-3)$, $k = min(N, n+3)$, $r = max(1, n-5)$ and $t = min(N, n+5)$.

We use an n-gram language model to calculate the conditional probability of a word given the preceding n-1 words. In particular, two features corresponding to the average conditional probabilities (AvgCondP3,AvgCondP2) have been extracted using the aforementioned trigram language model and a bigram model trained on Google's Trillion Word Corpus[7]. For words near the sentence boundary, the average is adjusted accordingly. If P_2 denotes the bigram language model probability then,

$$AvgCondP3(w_n) = \frac{1}{3} \sum_{k=n}^{n+2} P_3(w_k \mid w_{k-1}, w_{k-2})$$

$$AvgCondP2(w_n) = \frac{1}{2} \sum_{k=n}^{n+1} P_2(w_k \mid w_{k-1})$$

Sentences with higher perplexity have uncommon word sequences which may require more time to comprehend. Perplexity of the sentence calculated using tri-gram language model is also considered as a feature (Perplexity).

$$Perplexity(S) = \sqrt[N]{1/P_3(w_1 w_2 \dots w_N)}$$

4 Description of Algorithms

Experiments were conducted using the following machine learning regression algorithms:

- Partial Least Square Regression (PLS): This method aims at fitting a linear regression model by projecting the dependent and independent variables into a new space.
- Neural Network (NN): NN based regression method aims at predicting the value of the dependent variable as a function of input variables via a collection of interconnected nodes.
- Decision Tree (DT): The regression model is built in the form of a tree structure by breaking the dataset into smaller subsets.
- Random Forest (RF): RF regressor fits a multitude of decision trees on various sub-samples of the dataset, and uses averaging to improve accuracy and control over-fitting.
- XGBoost (XG) : Here, weakly learned decision trees are turned into strong learners by training upon residuals instead of aggregation (Chen and Guestrin, 2016).
- Light Gradient Boosting Machine (LG) : This method uses a histogram-based boosting algorthim which uses a specialised Gradient-based one-sided sampling of data points of large gradients (Ke et al., 2017).

[6] lm_giga_64k_nvp_3gram.zip

[7] http://norvig.com/ngrams/

- CatBoost (CB): This method takes advantage of the categorical features which are otherwise converted to numerical features in traditional gradient boosting algorithms. CB uses oblivious trees as base predictors which uses same splitting criterior accross the entire level of the tree, and hence are less prone to overfitting (Prokhorenkova et al., 2018).

Since five target Eye-Tracking metrics had to be predicted, Multioutput (MO) and Regressor Chain (RC) algorithms were deployed using sklearn.

5 Experimental Details

The input tokens containing only punctuations were removed. The Eye-Tracking feature for token '*&*' is assigned a fixed value [8]. For all other punctuation tokens, the assigned Eye-Tracking feature value is 0. SpaCy[9] is used for POS tagging, lemmatization, dependency parsing and NER. Stopword feature, Corpus features and Frequency features as described in Section 3.2 were extracted after lower casing and lemmatizing the input token. For RC the order is tuned between the 120 possibilities and the *max_depth* denoted as d, is tuned between 1 to 15. For NN the number of intermediate dense layers is tuned between 1 to 4, the layer dimension is tuned between {10, 25, 50, 100, 150, 200, 250, 300, 500} and dropouts is tuned randomly between 0 to 1. ReLU activation function is used in the intermediate dense layers, batch size is set to 32, learning rate is set to 0.005, and MAE is minimized using Adam optimizer (Kingma and Ba, 2015).

6 Results

The individual MAE for the five predicted features along with overall MAE for various regression techniques are reported in Table 3. For NN, two dense layers with dimension 100 and 200, respectively and corresponding dropouts 0.13 and 0.02, respectively were used. In the present work, CB outperforms other regression algorithms. This can be attributed to the permutation-driven ordered boosting technique of CB and effective use of categorical features. It can be observed that CB+MO performed the best on the Test Dataset. CB+RC with order (0,4,1,2,3) improved the performance for the Trial Dataset however, it did not have the same effect for the Test Data. The MAE of the proposed system is within 0.14 of the top performer.

[8] mean of Eye-Tracking values of '*&*' in the training set
[9] https://spacy.io/

7 Analysis

System predictions are presented in Table 4. The model had the highest MAE for the token $< 824, 16 >$ which contained alphanumeric characters because the features failed to capture its properties. For the token $< 900, 9 >$, the gold labels are 0, but the system predicts positive values. The true gaze features nF, GP, and TR for multi-hyphenated and repeated token, viz. $< 874, 20 >$ is found to be higher than the predicted values. However, the prediction of the system for the tokens $< 951, 5 >$ and $< 976, 26 >$ are close to the true values. The MAE for the token '*with*' in sentence 828 is very low while in sentence 933, it is very high. This is because there is large variation in the true Eye-Tracking values while the variation is low in the predicted values.

To analyze the importance of each feature, the corresponding feature is eliminated and the CB+MO model is trained on the reduced feature space. It was observed that elimination of each individual feature increased the error and thus, each feature plays an important role in the overall performance of the system. The MAE on the Trial Set corresponding to individual features are reported in Table 5. The feature *Relpos*, which indicates the relative position of token in the sentence, emerged as the most important feature.

8 Conclusion and Future Work

Automatic prediction of Gaze features without human intervention is important for scalability of these features for tasks involving large datasets. The Shared Task aims at prediction of five Eye-Tracking features for each token of a given sentence. In the present work, a set of linguistic features focused on representing the shallow lexical characteristics of the token, rarity of the token, and interaction and collocation of the target token with its context are extracted. CB+MO regressor trained on the above feature space secured fourth rank on the Shared Task. Error analysis indicates that there is high variation of Eye-Tracking features for the same words in different contexts. However, the proposed system does not capture this variation. In future we would like to incorporate more features in order to represent the context of the target token more effectively.

		Trial						Test					
Technique	**d**	**nF**	**FF**	**GP**	**TR**	**fxP**	**MAE**	**nF**	**FF**	**GP**	**TR**	**fxP**	**MAE**
CB+RC (0,4,1,2,3)	6	3.92	0.64	2.25	1.49	10.7	**3.79**	4.04	0.68	2.27	1.56	11.3	3.98
CB+MO	6	3.92	0.63	2.27	1.51	10.7	3.81	4.04	0.67	2.25	1.57	11.2	**3.95**
RF+MO	11	4.03	0.64	2.41	1.56	10.9	3.90	4.21	0.69	2.37	1.64	11.4	4.06
LG+MO		4.04	0.64	2.43	1.56	10.8	3.90	4.10	0.67	2.35	1.59	11.2	3.99
XG+ MO		4.05	0.65	2.40	1.57	10.9	3.92	4.21	0.69	2.40	1.63	11.5	4.09
DT+MO	7	4.32	0.68	2.58	1.70	11.6	4.18	4.51	0.73	2.53	1.78	12.3	4.37
NN		4.65	0.75	2.55	1.77	12.9	4.52	4.90	0.78	2.65	1.89	13.9	4.82
PLS+MO		4.79	0.73	3.10	1.85	13.2	4.74	4.95	0.78	3.21	1.93	13.8	4.93

Table 3: Mean Absolute Error values

		Predicted					Gold					
id	**word**	**nF**	**FF**	**GP**	**TR**	**fxP**	**nF**	**FF**	**GP**	**TR**	**fxP**	**MAE**
$< 824, 16 >$	111Senator	24.9	4.1	8.7	8.2	88.3	97.7	5.8	33.4	41.1	100	28.8
$< 900, 9 >$	counts.<EOS>	13.6	3.7	16.5	5.3	71.3	0.0	0.0	0.0	0.0	0.0	22.1
$< 874, 20 >$	great-great-great-great-great	37.8	4.7	17.2	14.8	96.1	86.1	3.8	30.8	31.1	89.7	17.1
$< 951, 5 >$	side-splittingly	42.5	4.9	12.1	15.3	99.8	42.5	4.3	14.3	14.8	100	0.69
$< 976, 26 >$	Rice's	17.8	4.0	6.4	6.5	83.4	17.2	4.4	6.4	6.4	83.3	0.23
$< 828, 9 >$	with	10.6	2.4	3.2	3.2	56.7	10.3	2.1	3.2	2.7	58.3	0.55
$< 933, 5 >$	with	11.0	2.5	3.4	3.5	58.3	14.9	3.2	7.0	5.1	75.0	5.31

Table 4: System predictions

Feature Group	**Feature Space**	**MAE**	**Feature Group**	**Feature Space**	**MAE**
Shallow Lexical	w/o Nlets	3.8474	Familiarity	w/o IsStopword	3.8158
	w/o Nvow	3.8169		w/o InGoogle	3.8190
	w/o Nsyl	3.8264		w/o InOgden	3.8177
	w/o Nphon	3.8231		w/o OgdenFreq	3.8256
	w/o Nmorph	3.8255		w/o ECFreq	3.8173
	w/o PerUp	3.8206		w/o SUBTFreq	3.8161
	w/o IsNamed	3.8180		w/o ConDiversity	3.8154
	w/o EtyOrig	3.8214		w/o TLFreq	3.8118
	w/o IsLast	3.8243		w/o BrownFreq	3.8160
	w/o IsNumber	3.8209		w/o CharProb2	3.8203
	w/o Hyphen	3.8261		w/o CharProb3	3.8236
	w/o IsPossesive	3.8176	Context	w/o UniTag	3.8112
	w/o StartPunct	3.8183		w/o PennTag	3.8186
	w/o LenSent	3.8388		w/o NSyn	3.8194
	w/o RelPos	**3.8725**		w/o Nhypo	3.8201
Language Model	w/o FragScore3	3.8166		w/o Nhyper	3.8233
	w/o FragScore5	3.8313		w/o DepTag	3.8206
	w/o AvgCondP2	3.8263		w/o HeadPOS	3.8192
	w/o AvgCondP3	3.8190		w/o TokDepth	3.8247
	w/o Perplexity	3.8169		w/o NChild	3.8178

Table 5: MAE scores for individual feature elimination

Acknowledgements

Raksha Agarwal acknowledges Council of Scientific and Industrial Research (CSIR), India for supporting the research under Grant no: SPM-06/086(0267)/2018-EMR-I. The authors thank Kushagri Tandon and Shivani Choudhary for helpful discussions.

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.

Maria Barrett and Nora Hollenstein. 2020. Sequence labelling and sequence classification with gaze: Novel uses of eye-tracking data for natural language processing. *Language and Linguistics Compass*, 14(11):1–16.

Gordon DA Brown. 1984. A frequency count of 190,000 words in the london-lund corpus of english conversation. *Behavior Research Methods, Instruments, & Computers*, 16(6):502–532.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA. Association for Computing Machinery.

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 Modeling 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(1):1–13.

Nora Hollenstein, Marius Troendle, Ce Zhang, and Nicolas Langer. 2020. ZuCo 2.0: A dataset of physiological recordings during natural reading and annotation. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 138–146, Marseille, France. European Language Resources Association.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. 2016. Improving sentence compression by learning to predict gaze. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1528–1533, San Diego, California. Association for Computational Linguistics.

Abhijit Mishra, Kuntal Dey, and Pushpak Bhattacharyya. 2017. Learning cognitive features from gaze data for sentiment and sarcasm classification using convolutional neural network. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 377–387, Vancouver, Canada. Association for Computational Linguistics.

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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. Catboost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Edward Lee Thorndike and Irving Lorge. 1944. The teacher's word book of 30,000 words.