CogAlign: Learning to Align Textual Neural Representations to Cognitive Language Processing Signals

Yuqi Ren and Deyi Xiong *
College of Intelligence and Computing, Tianjin University, Tianjin, China
{ryq20, dyxiong}@tju.edu.cn

Abstract

Most previous studies integrate cognitive language processing signals (e.g., eye-tracking or EEG data) into neural models of natural language processing (NLP) just by directly concatenating word embeddings with cognitive features, ignoring the gap between the two modalities (i.e., textual vs. cognitive) and noise in cognitive features. In this paper, we propose a CogAlign approach to these issues, which learns to align textual neural representations to cognitive features. In CogAlign, we use a shared encoder equipped with a modality discriminator to alternatively encode textual and cognitive inputs to capture their differences and commonalities. Additionally, a text-aware attention mechanism is proposed to detect task-related information and to avoid using noise in cognitive features. Experimental results on three NLP tasks, namely named entity recognition, sentiment analysis and relation extraction, show that CogAlign achieves significant improvements with multiple cognitive features over state-of-the-art models on public datasets. Moreover, our model is able to transfer cognitive information to other datasets that do not have any cognitive processing signals. The source code for CogAlign is available at https://github.com/tjunlp-lab/CogAlign.git.

1 Introduction

Cognitive neuroscience, from a perspective of language processing, studies the biological and cognitive processes and aspects that underlie the mental language processing procedures in human brains while natural language processing (NLP) teaches machines to read, analyze, translate and generate human language sequences (Muttenthaler et al., 2020). The commonality of language processing shared by these two areas forms the base of cognitively-inspired NLP, which uses cognitive language processing signals generated by human brains to enhance or probe neural models in solving a variety of NLP tasks, such as sentiment analysis (Mishra et al., 2017; Barrett et al., 2018), named entity recognition (NER) (Hollenstein and Zhang, 2019), dependency parsing (Strzyz et al., 2019), relation extraction (Hollenstein et al., 2019a), etc.

In spite of the success of cognitively-inspired NLP in some tasks, there are some issues in the use of cognitive features in NLP. First, for the integration of cognitive processing signals into neural models of NLP tasks, most previous studies have just directly concatenated word embeddings with cognitive features from eye-tracking or EEG, ignoring the huge differences between these two types of representations. Word embeddings are usually learned as static or contextualized representations of words in large-scale spoken or written texts generated by humans. In contrast, cognitive language processing signals are collected by specific medical equipments, which record the activity of human brains during the cognitive process of language processing. These cognitive processing signals are usually assumed to represent psycholinguistic information (Mathias et al., 2020) or cognitive load (Antonenko et al., 2010). Intuitively, information in these two types of features (i.e., word embeddings and cognitive features) is not directly comparable to each other. As a result, directly concatenating them could be not optimal for neural models to solve NLP tasks.

The second issue with the incorporation of cognitive processing signals into neural models of NLP is that not all information in cognitive processing signals is useful for NLP. The recorded signals contain information covering a wide variety of cognitive processes, particularly for EEG (Williams et al., 2019; Eugster et al., 2014). For different tasks, we may need to detect elements in the recorded signals,

*Corresponding author

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 3758–3769
August 1–6, 2021.

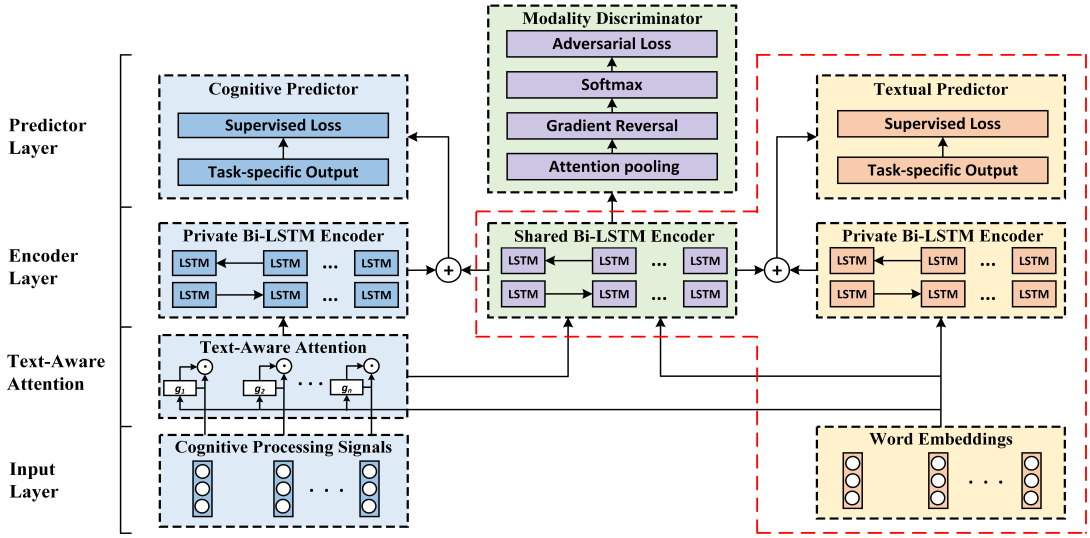


Figure 1: Neural Architecture of the proposed CogAlign. For inference, only the components in the red dashed box are used.

which are closely related to specific NLP tasks, and neglect features that are noisy to the tasks.

In order to address the two issues, we propose **CogAlign**, a multi-task neural network that learns to align neural representations of texts to cognitive processing signals, for several NLP tasks. As shown in Figure 1, instead of simply concatenating cognitive features with word embeddings, we use two private encoders to separately encode cognitive processing signals and word embeddings. The two encoders will learn task-specific representations for cognitive and textual inputs in two disentangled spaces. To align the representations of neural network with cognitive processing signals, we further introduce an additional encoder that is shared by both data sources. We alternatively feed cognitive and textual inputs into the shared encoder and force it to minimize an adversarial loss of the discriminator stacked over the shared encoder. The discriminator is task-agnostic so that it can focus on learning both differences and deep commonalities between neural representations of cognitive and textual features in the shared encoder. We want the shared encoder to be able to transfer knowledge of cognitive language processing signals to other datasets even if cognitive processing signals are not available for those datasets. Therefore, CogAlign does not require cognitive processing signals as inputs during inference.

Partially inspired by the attentive pooling network (Santos et al., 2016), we propose a text-aware attention mechanism to further align textual inputs and cognitive processing signals at the word level. The attention network learns a compatibility matrix of textual inputs to cognitive processing signals. The learned text-aware representations of cognitive processing signals also help the model to detect task-related information and to avoid using other noisy information contained in cognitive processing signals.

In a nutshell, our contributions are listed as follows:

- We present CogAlign that learns to align neural representations of natural language to cognitive processing signals at both word and sentence level. Our analyses show that it can learn task-related specific cognitive processing signals.
- We propose a text-aware attention mechanism that extracts useful cognitive information via a compatibility matrix.
- With the adversarially trained shared encoder, CogAlign is capable of transferring cognitive knowledge into other datasets for the same task, where no recorded cognitive processing signals are available.
- We conduct experiments on incorporating eye-tracking and EEG signals into 3 different NLP tasks: NER, sentiment analysis and relation extraction, which show CogAlign achieves new state-of-the-art results and significant improvements over strong baselines.

2 Related Work

Eye-tracking for NLP. Eye-tracking data have proved to be associated with language comprehension activity in human brains by numerous research in neuroscience (Rayner, 1998; Henderson and Ferreira, 1993). In cognitively motivated NLP, several studies have investigated the impact of eye-tracking data on NLP tasks. In early works, these signals have been used in machine learning approaches to NLP tasks, such as part-of-speech tagging (Barrett et al., 2016), multiword expression extraction (Rohanian et al., 2017), syntactic category prediction (Barrett and Søgaard, 2015). In neural models, eye-tracking data are combined with word embeddings to improve various NLP tasks, such as sentiment analysis (Mishra et al., 2017) and NER (Hollenstein and Zhang, 2019). Eye-tracking data have also been used to enhance or constrain neural attention in (Barrett et al., 2018; Sood et al., 2020b,a; Takmaz et al., 2020).

EEG for NLP. Electroencephalography (EEG) measures potentials fluctuations caused by the activity of neurons in cerebral cortex. The exploration of EEG data in NLP tasks is relatively limited. Chen et al. (2012) improve the performance of automatic speech recognition (ASR) by using EEG signals to classify the speaker's mental state. Hollenstein et al. (2019a) incorporate EEG signals into NLP tasks, including NER, relation extraction and sentiment analysis. Additionally, Muttenthaler et al. (2020) leverage EEG features to regularize attention on relation extraction.

Adversarial Learning. The concept of adversarial training originates from the Generative Adversarial Nets (GAN) (Goodfellow et al., 2014) in computer vision. Since then, it has been also applied in NLP (Denton et al., 2015; Ganin et al., 2016). Recently, a great variety of studies attempt to introduce adversarial training into multi-task learning in NLP tasks, such as Chinese NER (Cao et al., 2018), crowdsourcing learning (Yang et al., 2018), cross-lingual transfer learning (Chen et al., 2018; Kim et al., 2017), just name a few. Different from these studies, we use adversarial learning to deeply align cognitive modality to textual modality at the sentence level.

3 CogAlign

CogAlign is a general framework for incorporating cognitive processing signals into various NLP tasks. The target task can be specified at the predictor layer with corresponding task-specific neural network. CogAlign focuses on aligning cognitive processing signals to textual features at the word and encoder level. The text-aware attention aims at learning task-related useful cognitive information (thus filtering out noises) while the shared encoder and discriminator collectively learns to align representations of cognitive processing signals to those of textual inputs in a unified semantic space. The matched neural representations can be transferred to another datasets of the target task even though cognitive processing signals is not present. The neural architecture of CogAlign is visualized in Figure 1. We will elaborate the components of model in the following subsections.

3.1 Input Layer

The inputs to our model include textual word embeddings and cognitive processing signals.

Word Embeddings. For a given word x_i from the dataset of a target NLP task (e.g., NER), we obtain the vector representation h_i^{word} by looking up a pre-trained embedding matrix. The obtained word embeddings are fixed during training. For NER, previous studies have shown that character-level features can improve the performance of sequence labeling (Lin et al., 2018). We therefore apply a character-level CNN framework (Chiu and Nichols, 2016; Ma and Hovy, 2016) to capture the character-level embedding. The word representation of word x_i in NER task is the concatenation of word embedding and character-level embedding.

Cognitive Processing Signals. For cognitive inputs, we can obtain word-level eye-tracking and EEG via data preprocessing (see details in Section 5.1). Thus, for each word x_i, we employ two cognitive processing signals h_i^{eye} and h_i^{eeg}. The cognitive input h_i^{cog} can be either a single type of signal or a concatenation of different cognitive processing signals.

3.2 Text-Aware Attention

As not all information contained in cognitive processing signals is useful for the target NLP task, we propose a text-aware attention mechanism to assign text sensitive weights to cognitive processing signals. The main process of attention mechanism consists of learning a compatibility matrix between word embeddings $H^{word} \in R^{d_w \times N}$ and cognitive representations $H^{cog} \in R^{d_c \times N}$ from the input

layer and preforming cognitive-wise max-pooling operation over the matrix. The compatibility matrix $G \in R^{d_w \times d_c}$ can be computed as follows:

$$G = \tanh(H^{word} U H^{cog^T}) \quad (1)$$

where d_w and d_c are the dimension of word embeddings and cognitive representations, respectively, N is the length of the input, and $U \in R^{N \times N}$ is a trainable parameter matrix.

We then obtain a vector $g^{cog} \in R^{d_c}$, which is computed as the importance score for each element in the cognitive processing signals with regard to the word embeddings, by row-wise max-pooling over G. Finally, we compute attention weights and the text-aware representation of cognitive processing signals $H^{cog'}$ as follows:

$$\alpha^{cog} = \text{softmax}(g^{cog}) \quad (2)$$

$$H^{cog'} = \alpha^{cog} H^{cog} \quad (3)$$

3.3 Encoder Layer

We adopt Bi-LSTMs to encode both cognitive and textual inputs following previous works (Hollenstein and Zhang, 2019; Hollenstein et al., 2019a). In this work, we employ two private Bi-LSTMs and one shared Bi-LSTM as shown in Figure 1, where private Bi-LSTMs are used to encode cognitive and textual inputs respectively and the shared Bi-LSTM is used for learning shared semantics of both types of inputs. We concatenate the outputs of private Bi-LSTMs and shared Bi-LSTM as input to the task-specific predictors of subsequent NLP tasks. The hidden states of the shared Bi-LSTM are also fed into the discriminator.

3.4 Modality Discriminator

We alternatively feed cognitive and textual inputs into the shared Bi-LSTM encoder. Our goal is that the shared encoder is able to map the representations of the two different sources of inputs into the same semantic space so as to learn the deep commonalities of two modalities (cognitive and textual). For this, we use a self-supervised discriminator to provide supervision for training the shared encoder.

Particularly, the discriminator is acted as a classifier to categorize the alternatively fed inputs into either the textual or cognitive input. For the hidden state of modality k, we use a self-attention mechanism to first reduce the dimension of the output of the shared Bi-LSTM $H_k^s \in R^{d_h \times N}$:

$$\alpha = \text{softmax}(v^T \tanh(W_s H_k^s + b_s)) \quad (4)$$

$$h_k^s = \sum_{i=1}^{N} \alpha_i H_{k_i}^s \quad (5)$$

where $W_s \in R^{d_h \times d_h}$, $b_s \in R^{d_h}$, $v \in R^{d_h}$ are trainable parameters in the model, h_k^s is the output of self-attention mechanism. Then we predict the category of the input by softmax function:

$$D(h_k^s) = \text{softmax}(W_d h_k^s + b_d) \quad (6)$$

where $D(h_k^s)$ is the probability that the shared encoder is encoding an input with modality k.

3.5 Predictor Layer

Given a sample X, the final cognitively augmented representation after the encoder layer can be formulated as $H^{'} = [H^p; H^s] \in R^{2d_h \times N}$. H^p and H^s are the result of private Bi-LSTM and shared Bi-LSTM, respectively.

For sequence labeling tasks like NER, we employ the conditional random field (CRF) (Lafferty et al., 2001) as the predictor as Bi-LSTM-CRF is widely used in many sequence labeling tasks (Ma and Hovy, 2016; Luo et al., 2018) due to the excellent performance and also in cognitively inspired NLP (Hollenstein and Zhang, 2019; Hollenstein et al., 2019a). Firstly, we project the feature representation $H^{'}$ onto another space of which dimension is equal to the number of NER tags as follows:

$$o_i = W_n h_i^{'} + b_n \quad (7)$$

We then compute the score of a predicted tag sequence y for the given sample X:

$$score(X, y) = \sum_{i=1}^{N} (o_{i,y_i} + T_{y_{i-1}, y_i}) \quad (8)$$

where T is a transition score matrix which defines the transition probability of two successive labels.

Sentiment analysis and relation extraction can be regarded as multi-class classification tasks, with 3 and 11 classes, respectively. For these two tasks, we use a self attention mechanism to reduce the dimension of $H^{'}$ and obtain the probability of a predicted class via the softmax function.

4 Training and Inference

4.1 Adversarial Learning

In order to learn the deep interaction between cognitive and textual modalities in the same semantic space, we want the shared Bi-LSTM encoder to output representations that can fool the discriminator. Therefore we adopt the adversarial learning strategy. Particularly, the shared encoder acts as the generator that tries to align the textual and cognitive modalities as close as possible so as to mislead the discriminator. The shared encoder and discriminator works in an adversarial way.

Additionally, to further increase the difficulty for the discriminator to distinguish modalities, we add a gradient reversal layer (GRL) (Ganin and Lempitsky, 2015) in between the encoder layer and predictor layer. The gradient reversal layer does nothing in the forward pass but reverses the gradients and passes them to the preceding layer during the backward pass. That is, gradients with respect to the adversarial loss $\frac{\partial L_{Adv}}{\partial \theta}$ are replaced with $-\frac{\partial L_{Adv}}{\partial \theta}$ after going through GRL.

4.2 Training Objective

CogAlign is established on a multi-task learning framework, where the final training objective is composed of the adversarial loss L_{Adv} and the loss of the target task L_{Task}. For NER, we exploit the negative log-likelihood objective as the loss function. Given T training examples $(X^i; \overline{y}^i)$[1], L_{Task} is defined as follows:

$$L_{Task} = -\sum_{i=1}^{T} \log p(\overline{y}^i | X^i) \quad (9)$$

where $\overline{y}$ denotes the ground-truth tag sequence. The probability of $\overline{y}$ is computed by the softmax function:

$$p(\overline{y}|X) = \frac{e^{score(X,\overline{y})}}{\sum_{\widetilde{y} \in Y} e^{score(X,\widetilde{y})}} \quad (10)$$

For sentiment analysis and relation extraction tasks, the task objective is similar to that of NER. The only difference is that the label of the task is changed from a tag sequence to a single class.

The adversarial loss L_{Adv} is defined as:

$$L_{Adv} = \min_{\theta_s}(\max_{\theta_d} \sum_{k=1}^{K} \sum_{i=1}^{T_k} \log D(S(X_k^i))) \quad (11)$$

where θ_s and θ_d denote the parameters of the shared Bi-LSTM encoders S and modality discriminator D, respectively, X_k^i is the representation of sentence i in a modality k. The joint loss of CogAlign is therefore defined as:

$$L = L_{Task} + L_{Adv} \quad (12)$$

4.3 Inference

After training, the shared encoder learns a unified semantic space for representations of both cognitive and textual modality. We believe that the shared space embeds knowledge from cognitive processing signals. For inference, we therefore only use the textual part and the shared encoder (components in the red dashed box in Figure 1). The private encoder outputs textual-modality-only representations while the shared encoder generates cognitive-augmented representations. The two representations are concatenated to feed into the predictor layer of the target task. This indicates that we do not need cognitive processing signals for the inference of the target task. It also means that we can pretrain CogAlign with cognitive processing signals and then transfer it to other datasets where cognitive processing signals are not available for the same target task.

5 Experiments

We conducted experiments on three NLP tasks, namely NER, sentiment analysis and relation extraction with two types of cognitive processing signals (eye-tracking and EEG) to validate the effectiveness of the proposed CogAlign.

5.1 Dataset and Cognitive Processing Signals

We chose a dataset[2] with multiple cognitive processing signals: Zurich Cognitive Language Processing Corpus (ZuCo) (Hollenstein et al., 2018). This corpus contains simultaneous eye-tracking and EEG signals collected when 12 native English speakers are reading 1,100 English sentences. Word-level signals can be divided by the duration of each word.

The dataset includes two reading paradigms: normal reading and task-specific reading where subjects exercise some specific task. In this work, we only used the data of normal reading, since this paradigm accords with human natural reading. The materials for normal reading paradigm

[1] X can be either textual or cognitive input as we alternatively feed word embeddings and cognitive processing signals into CogAlign.

[2] The data is available here: https://osf.io/q3zws/

EARLY	first fixation duration (FFD)	the duration of word w that is first fixated
	first pass duration (FPD)	the sum of the fixations before eyes leave the word w
LATE	number of fixations (NFIX)	the number of times word w that is fixated
	fixation probability (FP)	the probability that word w is fixated
	mean fixation duration (MFD)	the average fixation durations for word w
	total fixation duration (TFD)	the total duration of word w that is fixated
	n re-fixations (NR)	the number of times word w that is fixated after the first fixation
	re-read probability (RRP)	the probability of word w that is fixated more than once
CONTEXT	total regression-from duration (TRD)	the total duration of regressions from word w
	w-2 fixation probability (w-2 FP)	the fixation probability of the word w-2
	w-1 fixation probability (w-1 FP)	the fixation probability of the word w-1
	w+1 fixation probability (w+1 FP)	the fixation probability of the word w+1
	w+2 fixation probability (w+2 FP)	the fixation probability of the word w+2
	w-2 fixation duration (w-2 FD)	the fixation duration of the word w-2
	w-1 fixation duration (w-1 FD)	the fixation duration of the word w-1
	w+1 fixation duration (w+1 FD)	the fixation duration of the word w+1
	w+2 fixation duration (w+2 FD)	the fixation duration of the word w+2

Table 1: Eye-tracking features used in the NER task.

consist of two datasets: 400 movie reviews from Stanford Sentiment Treebank (Socher et al., 2013) with manually annotated sentiment labels, including 123 neutral, 137 negative and 140 positive sentences; 300 paragraphs about famous people from Wikipedia relation extraction corpus (Culotta et al., 2006) labeled with 11 relationship types, such as award, education.

We also tested our model on NER task. For NER, the selected 700 sentences in the above two tasks are annotated with three types of entities: PERSON, ORGANIZATION, and LOCATION. All annotated datasets[3] are publicly available. The cognitive processing signals and textual features used for each task in this work are the same as (Hollenstein et al., 2019a).

Eye-tracking Features. Eye-tracking signals record human gaze behavior while reading. The eye-tracking data of ZuCo are collected by an infrared video-based eye tracker EyeLink 1000 Plus with a sampling rate of 500 Hz. For NER, we used 17 eye-tracking features that cover all stages of gaze behaviors and the effect of context. According to the reading process, these features are divided into three groups: **EARLY**, the gaze behavior when a word is fixated for the first time; **LATE**, the gaze behavior over a word that is fixated many times; **CONTEXT**, the eye-tracking features over neighboring words of the current word. The 17 eye-tracking features used in the NER task are shown in the Table 1. In the other two tasks, we employed 5 gaze behaviors, including the first fixation duration (FFD), the number of fixations (NFIX), the total fixation duration (TFD), the first pass duration (FPD), the gaze duration (GD) that is the duration of the first time eyes move to the current word until eyes leave the word.

EEG Features. EEG signals record the brain's electrical activity in the cerebral cortex by placing electrodes on the scalp of the subject. In the datasets we used, EEG signals are recorded by a 128-channel EEG Geodesic Hydrocel system (Electrical Geodesics, Eugene, Oregon) at a sampling rate of 500 Hz with a bandpass of 0.1 to 100 Hz. The original EEG signals recorded are of 128 dimensions. Among them, 23 EEG signals are removed during preprocessing since they are not related to the cognitive processing (Hollenstein et al., 2018). After preprocessing, we obtained 105 EEG signals. The left EEG signals are divided into 8 frequency bands by the frequency of brain's electrical signals: $theta1$ (t1, 4-6 Hz), $theta2$ (t2, 6.5-8 Hz), $alpha1$ (a1, 8.5-10 Hz), $alpha2$ (a2, 10.5-13 Hz), $beta1$ (b1, 13.5-18 Hz), $beta2$ (b2, 18.5-30 Hz), $gamma1$ (g1, 30.5-40 Hz) and $gamma2$ (g2, 40-49.5 Hz). The frequency bands reflects the different functions of brain cognitive processing. For NER, we used 8 EEG features that are obtained by averaging the 105 EEG signals at each frequency band. For the other two tasks, EEG features were obtained by averaging the 105 signals over all frequency bands. All used EEG features are obtained by averaging over all subjects and normalization.

5.2 Settings

We evaluated three NLP tasks in terms of precision, recall and F1 in our experiments. Word embeddings of all NLP tasks were initialized with the publicly available pretrained GloVe (Pennington

[3]https://github.com/DS3Lab/zuco-nlp/

Signals	Model	NER			Sentiment Analysis			Relation Extraction		
		P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
	Base*	89.34	78.60	83.48	59.47	59.42	58.27	79.52	75.67	75.25
eye	(Hollenstein et al., 2019a)	86.2	**84.3**	85.1	65.1	61.9	62.0	61.4	61.7	61.5
	Base	90.56	81.05	85.43*	64.26	61.96	61.19*	82.01	78.23	77.95*
	Base+TA	90.75	81.77	85.93*	64.63	62.71	61.41*	**83.26**	76.47	78.04*
	CogAlign	90.76	82.52	86.41*	62.86	64.10	62.30*	78.33	82.06	78.56*
EEG	(Hollenstein et al., 2019a)	86.7	81.5	83.9	**68.3**	64.8	65.1	60.5	60.2	60.3
	Base	89.82	80.55	84.76*	64.09	60.29	59.79*	82.79	77.16	77.61*
	Base+TA	89.54	82.22	85.62*	62.20	62.19	60.91*	80.83	78.46	77.81*
	CogAlign	89.87	83.08	86.21*	63.11	65.38	62.81*	77.94	**82.60**	78.66*
eye +EEG	(Hollenstein et al., 2019a)	85.1	83.2	84.0	66.3	59.3	60.8	59.8	60.0	59.8
	Base	89.70	81.11	85.11*	62.86	61.49	60.84*	79.00	76.52	77.72*
	Base+TA	90.75	82.94	86.31*	65.22	63.88	63.23*	82.24	77.53	78.12*
	CogAlign	**91.28**	83.02	**86.79***	65.11	**65.94**	**65.40***	78.66	82.07	**78.93***

Table 2: Results of CogAlign and other methods on the three NLP tasks augmented with eye-tracking features (eye), EEG features (EEG), and both (eye+EEG). 'Base*' denotes that the model does not use any cognitive processing signals. 'Base' is a neural model that consist of a textual private encoder and textual predictor, and combines cognitive processing signals with word embeddings via direct concatenation, similar to previous works. 'Base+TA' is a neural model where direct concatenation in the base model is replaced by the text-aware attention mechanism. Significance is indicated with the asterisks: * = $p<0.01$.

et al., 2014) vectors of 300 dimensions. For NER, we used 30-dimensional randomly initialized character embeddings. We set the dimension of hidden states of LSTM to 50 for both the private Bi-LSTM and shared Bi-LSTM. We performed 10-fold cross validation for NER and sentiment analysis and 5-fold cross validation for relation extraction.

5.3 Baselines

We compared our model with previous state-of-the-art methods on ZuCo dataset. The method by Hollenstein et al. (2019a) incorporates cognitive processing signals into their model via direct concatenation mentioned before.

5.4 Results

Results of CogAlign on the three NLP tasks are shown in Table 2. From the table, we observe that:

- By just simply concatenating word embeddings with cognitive processing signals, the Base model is better than the model without using any cognitive processing signals, indicating that cognitive processing signals (either eye-tracking or EEG signals) can improve all three NLP tasks. Notably, the improvements gained by eye-tracking features are larger than those obtained by EEG signals while the combination of both does not improve over only using one of them. We conjecture that this may be due to the low signal-to-noise ratio of EEG signals, which further decreases when two signals are combined together.

- Compared with the Base model, the Base+TA achieves better results on all NLP tasks. The text-aware attention gains an absolute improvement of 0.88, 2.04, 0.17 F1 on NER, sentiment analysis, and relation extraction, respectively. With Base+TA, the best results for most tasks are obtained by the combination of eye-tracking and EEG signals. This suggests that the proposed text-aware attention may have alleviated the noise problem of cognitive processing signals.

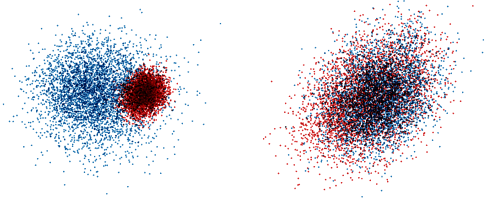
- The proposed CogAlign achieves the highest F1 over all three tasks, with improvements of 0.48, 2.17 and 0.87 F1 over Base+TA on NER, sentiment analysis and relation extraction, respectively, which demonstrates the effectiveness of our proposed model. In addition, CogAlign with both cognitive processing signals obtains new state-of-the-art performance in all NLP tasks. This suggests that CogAlign is able to effectively augment neural models with cognitive processing signals.

5.5 Ablation Study

To take a deep look into the improvements contributed by each part of our model, we perform ablation study on all three NLP tasks with two cognitive processing signals. The ablation test includes: (1) **w/o text-aware attention**, removing text-aware attention mechanism; (2) **w/o cognitive loss**, discarding the loss of the cognitive predictor whose inputs are cognitive processing signals; (3) **w/o modality discriminator**, removing the discriminator to train parameters with the task loss. Table 3 reports the ablation study results.

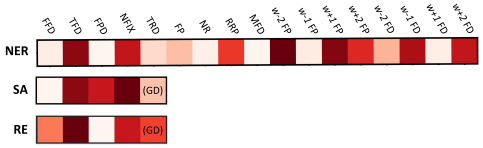
Model	NER			Sentiment Analysis			Relation Extraction		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
CogAlign (eye+EEG)	91.28	83.02	86.79*	65.11	65.94	65.40*	78.66	82.07	78.93*
- text-aware attention	90.51	82.45	86.19*	64.75	65.30	63.90*	77.67	83.14	78.68*
- cognitive loss	90.20	81.11	85.45*	64.48	65.42	63.77*	77.79	81.24	77.75*
- modality discriminator	89.63	83.66	86.09*	64.11	66.24	63.28*	78.61	80.71	78.46*

Table 3: Ablation study on the three NLP tasks. Significance is indicated with the asterisks: * = p<0.01.

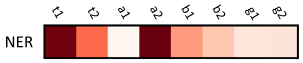


(a) without adv (b) with adv

Figure 2: The visualization of hidden states from the shared Bi-LSTM layer. 'adv' denotes the adversarial learning. Red dots are the hidden representations of cognitive processing signals while blue dots hidden representations of textual inputs. Both are at the word level via t-SNE (Van der Maaten and Hinton, 2008).



(a) eye-tracking



(b) EEG

Figure 3: The visualization of attention weights over cognitive processing signals by the text-aware attention in the three NLP tasks. Darker colors represent higher attention weights.

The absence of the text-aware attention, cognitive loss and modality discriminator results in a significant drop in performance. This demonstrates that these components all contribute to the effective incorporation of cognitive processing signals into neural models of the three target tasks. CogAlign outperforms both (2) **w/o cognitive loss** and (3) **w/o modality discriminator** by a great margin, indicating that the cognitive features can significantly enhance neural models.

Furthermore, we visualize the distribution of hidden states learned by the shared Bi-LSTM to give a more intuitive demonstration of the effect of adversarial learning. In Figure 2, clearly, the modality discriminator with adversarial learning forces the shared Bi-LSTM encoder to align textual inputs to cognitive processing signals in the same space.

6 Analysis

6.1 Text-aware Attention Analysis

In addition to denoising the cognitive processing signals, the text-aware attention mechanism also obtains the task-specific features. To have a clear view of the role that the text-aware attention mechanism plays in CogAlign, we randomly choose samples and visualize the average attention weights over each signal in Figure 3.

For eye-tracking, signals reflecting the late syntactic processing, such as 'NFIX' (number of fixation), 'TFD' (total fixation duration), play an important role in the three tasks. These results are consistent with findings in cognitive neuroscience. In cognitive neuroscience, researchers have shown that readers tend to gaze at nouns repeatedly (Furtner et al., 2009) (related to the eye-tracking signal NFIX, the number of fixations) and there is a dependency relationship between regression features and sentence syntactic structures (Lopopolo et al., 2019). In other NLP tasks that infused eye-tracking features, the late gaze features have also proved to be more important than early gaze features, such as multiword expression extraction (Rohanian et al., 2017). Moreover, from the additional eye-tracking used in NER, we can find that the cognitive features from the neighboring words are helpful to identify entity, such as 'w-2 FP' (w-2 fixation probability), 'w+1 FP' (w+1 fixation probability).

Since a single EEG signal has no practical meaning, we only visualize the attention weights over EEG signals used in the NER task. Obviously, attentions to 't1' ($theta1$) and 'a2' ($alpha2$) are stronger than other signals, suggesting that low frequency electric activities in the brain are obvious when we recognize an entity.

Model	Wikigold			SST		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
baseline	80.70	70.67	75.19	56.67	57.58	56.40
baseline (two encoders)	80.16	73.39	75.73	56.76	58.05	56.89
CogAlign (eye)	80.39	72.59	76.17	58.05	59.69	57.27
CogAlign (EEG)	80.54	71.91	75.93	57.25	58.34	57.10
CogAlign (eye+EEG)	81.71	74.17	77.76	58.60	58.33	58.32

Table 4: Results of CogAlign in transfer learning to other datasets without cognitive processing signals. 'baseline' is a model trained and tested with one encoder for textual inputs. 'baseline (+ZuCo text)' is the baseline trained with both Zuco textual data and target dataset (i.e., Wikigold or SST). 'baseline (two encoders)' is the same as CogAlign (the inference version), where cognitive processing signals are replaced by textual inputs.

6.2 Transfer Learning Analysis

The cognitively-inspired NLP is limited by the collection of cognitive processing signals. Thus, we further investigate whether our model can transfer cognitive features to other datasets without cognitive processing signals for the same task. We enable transfer learning in CogAlign with a method similar to the alternating training approach (Luong et al., 2016) that optimizes each task for a fixed number of mini-batches before shifting to the next task. In our case, we alternately feed instances from the ZuCo dataset and those from other datasets built for the same target task but without cognitive processing signals into CogAlign. Since CogAlign is a multi-task learning framework, model parameters can be updated either by data with cognitive processing signals or by data without such signals, where task-specific loss is used in both situations. Please notice that only textual inputs are fed into trained CogAlign for inference.

To evaluate the capacity of CogAlign in transferring cognitive features, we select benchmark datasets for NER and sentiment analysis: Wikigold (Balasuriya et al., 2009) and Stanford Sentiment Treebank (Socher et al., 2013). Since no other datasets use the same set of relation types as that in ZuCo dataset, we do not test the relation extraction task for transfer learning. To ensure that the same textual data are used for comparison, we add a new baseline model (baseline (+Zuco text)) that is trained on the combination of textual data in ZuCo and benchmark dataset. Additionally, as CogAlign uses two encoders for inference (i.e., the textual encoder and shared encoder), for a fair comparison, we setup another baseline (baseline (two encoders)) that also uses two encoders fed with the same textual inputs. The experimental setup is the same as mentioned before.

Results are shown in the Table 4. We can observe that CogAlign consistently outperforms the two baselines. It indicates that CogAlign is able to effectively transfer cognitive knowledge (either eye-tracking or EEG) from ZuCo to other datasets. Results show that the best performance is achieved by transferring both eye-tracking and EEG signals at the same time.

7 Conclusions

In this paper, we have presented CogAlign, a framework that can effectively fuse cognitive processing signals into neural models of various NLP tasks by learning to align the textual and cognitive modality at both word and sentence level. Experiments demonstrate that CogAlign achieves new state-of-the-art results on three NLP tasks on the Zuco dataset. Analyses suggest that the text-aware attention in CogAlign can learn task-related cognitive processing signals by attention weights while the modality discriminator with adversarial learning forces CogAlign to learn cognitive and textual representations in the unified space. Further experiments exhibit that CogAlign is able to transfer cognitive information from Zuco to other datasets without cognitive processing signals.

Acknowledgments

The present research was partially supported by the National Key Research and Development Program of China (Grant No. 2019QY1802) and Natural Science Foundation of Tianjin (Grant No. 19JCZDJC31400). We would like to thank the anonymous reviewers for their insightful comments.

References

Pavlo Antonenko, Fred Paas, Roland Grabner, and Tamara Van Gog. 2010. Using electroencephalography to measure cognitive load. *Educational Psychology Review*, 22(4):425–438.

Dominic Balasuriya, Nicky Ringland, Joel Nothman, Tara Murphy, and James R. Curran. 2009. Named entity recognition in wikipedia. In *Proceedings of the 1st 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources@IJCNLP 2009, Suntec, Singapore, August 7, 2009*, pages 10–18. Association for Computational Linguistics.

Maria Barrett, Joachim Bingel, Nora Hollenstein, Marek Rei, and Anders Søgaard. 2018. Sequence classification with human attention. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 302–312.

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.

Maria Barrett and Anders Søgaard. 2015. Reading behavior predicts syntactic categories. In *Proceedings of the 19th Conference on Computational Natural Language Learning, CoNLL 2015, Beijing, China, July 30-31, 2015*, pages 345–349. ACL.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. Adversarial transfer learning for chinese named entity recognition with self-attention mechanism. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 182–192.

Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2018. Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics*, 6:557–570.

Yun-Nung Chen, Kai-min Chang, and Jack Mostow. 2012. Towards using EEG to improve ASR accuracy. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 3-8, 2012, Montréal, Canada*, pages 382–385. The Association for Computational Linguistics.

Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, 4:357–370.

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.

Emily L Denton, Soumith Chintala, Rob Fergus, et al. 2015. Deep generative image models using a Laplacian Pyramid of Adversarial Networks. *Advances in neural information processing systems*, 28:1486–1494.

Manuel J. A. Eugster, Tuukka Ruotsalo, Michiel M. A. Spapé, Ilkka Kosunen, Oswald Barral, Niklas Ravaja, Giulio Jacucci, and Samuel Kaski. 2014. Predicting term-relevance from brain signals. In *The 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '14, Gold Coast , QLD, Australia - July 06 - 11, 2014*, pages 425–434. ACM.

Marco R. Furtner, John F. Rauthmann, and Pierre Sachse. 2009. Nomen est omen: Investigating the dominance of nouns in word comprehension with eye movement analyses. *Advances in Cognitive Psychology*, 5.

Yaroslav Ganin and Victor S. Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1180–1189. JMLR.org.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27:2672–2680.

John M Henderson and Fernanda Ferreira. 1993. Eye movement control during reading: Fixation measures reflect foveal but not parafoveal processing difficulty. *Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale*, 47(2):201.

Nora Hollenstein, Maria Barrett, Marius Troendle, Francesco Bigiolli, Nicolas Langer, and Ce Zhang. 2019a. Advancing NLP with cognitive language processing signals. *arXiv preprint arXiv:1904.02682*.

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 and Ce Zhang. 2019. Entity recognition at first sight: Improving NER with eye movement information. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1–10. Association for Computational Linguistics.

Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. Cross-lingual transfer learning for pos tagging without cross-lingual resources. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2832–2838.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, pages 282–289. Morgan Kaufmann.

Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. 2018. A multi-lingual multi-task architecture for low-resource sequence labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 799–809. Association for Computational Linguistics.

Alessandro Lopopolo, Stefan L. Frank, Antal Van Den Bosch, and Roel Willems. 2019. Dependency parsing with your eyes: Dependency structure predicts eye regressions during reading. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*.

Ling Luo, Zhihao Yang, Pei Yang, Yin Zhang, Lei Wang, Hongfei Lin, and Jian Wang. 2018. An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition. *Bioinform.*, 34(8):1381–1388.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

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

Sandeep Mathias, Diptesh Kanojia, Abhijit Mishra, and Pushpak Bhattacharya. 2020. A survey on using gaze behaviour for natural language processing. In *Twenty-Ninth International Joint Conference on Artificial Intelligence and Seventeenth Pacific Rim International Conference on Artificial Intelligence IJCAI-PRICAI-20*.

Abhijit Mishra, Diptesh Kanojia, Seema Nagar, Kuntal Dey, and Pushpak Bhattacharyya. 2017. Leveraging cognitive features for sentiment analysis. *arXiv preprint arXiv:1701.05581*.

Lukas Muttenthaler, Nora Hollenstein, and Maria Barrett. 2020. Human brain activity for machine attention. *CoRR*, abs/2006.05113.

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 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Keith Rayner. 1998. Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124(3):372.

Omid Rohanian, Shiva Taslimipoor, Victoria Yaneva, and Le An Ha. 2017. Using gaze data to predict multiword expressions. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, Varna, Bulgaria, September 2 - 8, 2017*, pages 601–609. INCOMA Ltd.

Cicero dos Santos, Ming Tan, Bing Xiang, and Bowen Zhou. 2016. Attentive pooling networks. *arXiv preprint arXiv:1602.03609*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y 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.

Ekta Sood, Simon Tannert, Diego Frassinelli, Andreas Bulling, and Ngoc Thang Vu. 2020a. Interpreting attention models with human visual attention in machine reading comprehension. *arXiv preprint arXiv:2010.06396*.

Ekta Sood, Simon Tannert, Philipp Müller, and Andreas Bulling. 2020b. Improving natural language processing tasks with human gaze-guided neural attention. *arXiv preprint arXiv:2010.07891*.

Michalina Strzyz, David Vilares, and Carlos Gómez-Rodríguez. 2019. Towards making a dependency parser see. *arXiv preprint arXiv:1909.01053*.

Ece Takmaz, Sandro Pezzelle, Lisa Beinborn, and Raquel Fernández. 2020. Generating image descriptions via sequential cross-modal alignment guided by human gaze. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4664–4677. Association for Computational Linguistics.

Chad C. Williams, Mitchel Kappen, Cameron D. Hassall, Bruce Wright, and Olave E. Krigolson. 2019. Thinking theta and alpha: Mechanisms of intuitive and analytical reasoning. *NeuroImage*, 189:574–580.

YaoSheng Yang, Meishan Zhang, Wenliang Chen, Wei Zhang, Haofen Wang, and Min Zhang. 2018. Adversarial learning for chinese NER from crowd annotations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1627–1635. AAAI Press.