

Learning Sentence Representation with Guidance of Human Attention

Shaonan Wang^{1,2}, Jiajun Zhang^{1,2}, Chengqing Zong^{1,2,3}

¹ National Laboratory of Pattern Recognition, CASIA, Beijing, China

² University of Chinese Academy of Sciences, Beijing, China

³ CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai, China
 {shaonan.wang,jjzhang,cqzong}@nlpr.ia.ac.cn

Abstract

Recently, much progress has been made in learning general-purpose sentence representations that can be used across domains. However, most of the existing models typically treat each word in a sentence equally. In contrast, extensive studies have proven that human read sentences efficiently by making a sequence of fixation and saccades. This motivates us to improve sentence representations by assigning different weights to the vectors of the component words, which can be treated as an attention mechanism on single sentences. To that end, we propose two novel attention models, in which the attention weights are derived using significant predictors of human reading time, i.e., Surprisal, POS tags and CCG supertags. The extensive experiments demonstrate that the proposed methods significantly improve upon the state-of-the-art sentence representation models.

1 Introduction

To understand the meaning of a sentence is a prerequisite to solve many linguistic and non-linguistic problems: answer a question, translate the text into another language and so on. Obviously, this requires a good representation of the meaning of a sentence. Recently, neural network based sentence representation models have shown advantages in learning general-purpose sentence embeddings [Le and Mikolov, 2014; Kiros *et al.*, 2015; Wieting *et al.*, 2016]. However, these models typically treat each word in a sentence equally. This is inconsistent with the way that human read and understand sentences, namely reading some words superficially and paying more attention to others. All these factors motivate us to build sentence representation models that can selectively focus on important words, which can be treated as a task-independent attention mechanism.

The main difficulty of introducing the above attention mechanism to a single sentence is the lack of extra information to guide the computation of attention weight. In this paper, we hypothesize that the significant predictors of human reading time are such useful information. So far, extensive studies have proven that word attributes, as represented

by POS tag, length, frequency, Surprisal, etc., are all correlated with human reading time [Demberg and Keller, 2008; Barrett *et al.*, 2016]. This paper focuses on two kinds of predictors: Surprisal as a continuous variable; POS tags and Combinatory Categorical Grammar (CCG) supertags which are discretely variables. Surprisal, proposed by [Hale, 2001] and [Levy, 2008], measures the amount of information conveyed by a particular event. Generally, the higher surprisal value corresponds to the higher processing complexity and more reading time. Moreover, psycholinguistic experiments have shown that readers are more likely to fixate on words from open syntactic categories (verbs, nouns, adjectives) than on closed category items like prepositions and conjunctions [Rayner, 1998]. These findings indicate that the above factors are crucial for simulating human attention in reading.

In this paper, we propose two novel attention approaches which are called attention model with Surprisal (ATT-SUR) and attention model with POS tag or CCG supertag (ATT-POS/ATT-CCG), respectively, to improve sentence representations. One approach utilizes Surprisal directly as the attention weight. The other approach builds attention model with the help of POS tag and CCG supertag vectors which are trained together with word embeddings. Aiming at enhancing semantic representation of sentences, the proposed attention models are then combined with two state-of-the-art (unsupervised/semi-supervised) sentence representation models. Furthermore, we perform extensive quantitative and qualitative analysis to shed light on the principle of the proposed attention models and its relation with human attention mechanism in reading.

To summarize, our main contributions include:

- We present two simple but efficient attention models for sentence representations, which can also be seen as a general framework of integrating predictors of human reading time into sentence representation models.
- We have evaluated our approaches on 24 SemEval datasets on semantic textual similarity (STS) tasks, which contain a wide range of domains. The results show that our approaches can significantly improve semantic representation of sentences.
- Experimental results have indicated that the proposed attention models can selectively focus on important words and successfully predict human reading time.

2 Background

We introduce two main approaches for learning general-purpose sentence representations according to the training material used: unsupervised methods trained on raw text corpora (Section 2.1), and semi-supervised methods trained on out-of-domain annotated text corpora (Section 2.2). For each method, we *choose the state-of-the-art model as our baseline* to incorporate the proposed attention models.

2.1 Unsupervised Methods

[Mikolov *et al.*, 2013] constructed a learning criterion for obtaining word representations from unlabeled data, by predicting a word from its surrounding words. Afterwards, several approaches for learning sentence representations were proposed, extending this strategy at the sentence level by predicting a sentence from its adjacent sentences [Kiros *et al.*, 2015; Hill *et al.*, 2016a; Kenter *et al.*, 2016], or by learning extra sentence embeddings in the learning process of word embeddings [Le and Mikolov, 2014; Wang *et al.*, 2016].

Among them, the Siamese CBOW (SCBOW) model introduced by [Kenter *et al.*, 2016] is the best performing method on multiple test sets. This method utilizes successive sentences as the training corpus, e.g., sentences in the article, and trains with categorical cross-entropy method. We briefly describe the SCBOW model below:

Baseline 1: The SCBOW model represents sentences by averaging the embeddings of its constituent words. Given a word sequence with length n : $x = \langle x_1, x_2, \dots, x_n \rangle$, the sentence representation model is described as:

$$g_{sentence}(x) = \frac{1}{n} \sum_{i=1}^n W_w^{x_i}, \quad (1)$$

where W_w is the word embedding matrix. For a pair of sentences (s_i, s_j) , we define the set S^+ as the sentences that occur next to the sentence s_i , and S^- , a set of randomly chosen sentences which are not in S^+ . The probability $p_\theta(s_i, s_j)$ reflects how likely the sentence pairs are adjacent to each other in the training data and is computed as:

$$p_\theta(s_i, s_j) = \frac{\exp(\cos(s_i^\theta, s_j^\theta))}{\sum_{s_k \in \{S^+ \cup S^-\}} \exp(\cos(s_i^\theta, s_k^\theta))}, \quad (2)$$

where s_x^θ denotes the embedding of sentence s_x . The objective function is defined as:

$$L = - \sum_{s_j \in \{S^+ \cup S^-\}} p(s_i, s_j) \cdot \log(p_\theta(s_i, s_j)), \quad (3)$$

where $p(s_i, s_j)$ is the target probability the network should produce, which is $\frac{1}{|S^+|}$ if $s_j \in S^+$ and 0 if $s_j \in S^-$.

2.2 Semi-Supervised Methods

Lately, various models for learning distributed sentence representations have been proposed, ranging from simple additional composition of the word vectors to sophisticated architectures such as convolution neural networks and recurrent neural networks. However, sentence representations,

generated by most of the existing work, are tuned only for their respective task. More recently, [Wieting *et al.*, 2016] proposed the Paragram-Phrase (PP) model, which learns general-purpose sentence embeddings with supervision from the Paraphrase Database (PPDB) [Ganitkevitch *et al.*, 2013]. This simple method is extremely efficient, outperforming more complex models (e.g., LSTM model), and even competitive with systems tuned for particular tasks.

Baseline 2: The PP model constructs sentence representations with the word averaging model defined in equation (1). The training data consists of a set of phrase pairs (x_1, x_2) from the PPDB dataset and negative examples (t_1, t_2) which are the most similar phrase pairs to (x_1, x_2) generated in a mini-batch during optimization. The PP model uses a max-margin objective function to train sentence embeddings by maximizing the distance between positive examples and negative examples:

$$\min_{W_w} \frac{1}{|X|} \left(\sum_{(x_1, x_2) \in X} \max(0, 1 - W_w^{x_1} \cdot W_w^{x_2} + W_w^{x_1} \cdot W_w^{t_1}) + \max(0, 1 - W_w^{x_1} \cdot W_w^{t_2} + W_w^{x_2} \cdot W_w^{t_2}) \right) + \lambda \|W_{w_i} - W_w\|^2 \quad (4)$$

Where λ is the regularization parameter, $|X|$ is the length of training paraphrase pairs, W_w is the current word vector matrix, and W_{w_i} is the initial word vector matrix.

3 Attention-based Sentence Representation Model

This section introduces the proposed attention models (Section 3.1), and how to integrate them into sentence representation models (Section 3.2).

3.1 Attention Models

ATT-SUR: Attention Model with Surprisal.

Surprisal, also known as self-information, measures the amount of information conveyed by the target. In language processing, it is defined as:

$$s^{x_t} = -\log(P(x_t|x_1, \dots, x_{t-1})), \quad (5)$$

where the Surprisal s^{x_t} corresponds to the negative logarithm of the conditional probability of word x_t given the sentential context x_1, \dots, x_{t-1} .

Based on the assumption that words with higher surprisal value convey more information and should gain more attention, we directly use the value of Surprisal as the attention weight. The proposed ATT-SUR model is computed as:

$$attention(x_t) = \frac{\exp(s^{x_t})}{\sum_{i \in [1, \dots, n]} \exp(s^{x_i})}, \quad (6)$$

where n is the length of the word sequence.

ATT-POS (ATT-CCG): Attention Model with POS Tags (CCG Supertags).

In this work we hypothesize that POS tag and CCG supertag of words are useful factors in building the attention model for sentence representations. For instance, given a sentence a#DT man#NN with#IN a#DT hard#JJ hat#NN is#VBZ dancing#VBG, the optimal sentence

representation should give more weights to words with NN, JJ, VBZ and VBG tags, and less weights to words with DT and IN tags. To model the above observation, we assign a vector to each POS tag (CCG supertag) and compute the dot product with the corresponding word embedding vectors. The result is a scalar parameter which determines the relative power of each of the POS tag (CCG supertag), which is described as:

$$attention(x_t) = \frac{exp(W_w^{x_t} \cdot W_c^{x_t})}{\sum_{i \in [1, \dots, n]} exp(W_w^{x_t} \cdot W_c^{x_t})}, \quad (7)$$

where $W_w \in \mathbb{R}^{V_w \times d}$ is the word embedding matrix and $W_c \in \mathbb{R}^{V_c \times d}$ is the word class matrix with each line represents a POS tag (CCG tag) vector.

3.2 Incorporating the Attention Models into Sentence Representations

To incorporate attention mechanism into sentence representations learned by the SCBOW model and the PP model, we use weighted summation of word embeddings instead of the averaging model in equation (1), and the weight is calculated by equation (6) or (7). The attention-based sentence representation model is computed as:

$$g_{sentence}(x) = \frac{1}{n} \sum_{i=1}^n attention(x_i) W_w^{x_i} \quad (8)$$

4 Experiments and Results

4.1 Datasets

Following the parameter settings in [Kenter *et al.*, 2016] and [Wieting *et al.*, 2016], the SCBOW model uses the Toronto Book Corpus¹ which contains 7,087 books collected from the web. The PP model is first trained with the smaller P-PDB dataset (version XL) for 10 epochs and then trained for another 10 epochs on a much larger PPDB dataset (version XXL). To evaluate the performance of our models, we use 24 datasets from STS task, covering a wide range of domains like news, image and video descriptions, glosses, twitter, machine translation evaluation and so on.

4.2 Experimental Settings

In this paper, we use the Stanford POS tagger² and the C&C tool³ to assign POS tags and CCG supertags, respectively, for words in the training and testing datasets. In all the models, we randomly initialize the POS tag and CCG supertag vectors with 300-dimension vectors, by drawing from a normal distribution with $\mu = 0.0$ and $\sigma = 0.01$. In the ATT-SUR model, the Surprisal is calculated by a state-of-the-art large-scale neural language model released by [Jozefowicz *et al.*, 2016]. Moreover, we also train a 5 order n-gram language

model with modified Kneser-Ney smoothing, but the performance is slightly worse. Hence we only report the results of the neural language model. In the experiment, we set surprisal value x as $min(max(0, x), 10)$.

In the SCBOW model, we use two negative examples and initialize the embedding layer with the pre-trained word embeddings⁴. Then we train the model using AdaDelta for one epoch with the initial learning rate of 0.001 and the batch size of 100. For the PP model, phrases in the training dataset are not sentences or even constituents, causing worse tagging results. Hence we use the SICK data set, which consists of 10,000 English sentence pairs with human annotation, to train the attention models after training the PP model⁵. The attention models are trained by AdaGrad for ten epochs with initial learning rate of 0.05. *The code for training and evaluation will be released.*

4.3 Textual Similarity

Following previous work in evaluating the STS task, we use the Pearson’s correlation coefficient to examine relationships between the human ratings and the predicted cosine similarity scores of sentence pairs. Table 1 displays the results of the two baseline models (PP-Base, SCBOW-Base), TF-IDF baseline model⁶ and the proposed attention-based sentence representation models on all 24 textual similarity dataset⁷. In addition, we also include three more baselines: the SkipThought (ST) [Kiros *et al.*, 2015] model, that utilizes an encoder-decoder model and produces highly generic sentence representations⁸. The ParaphraseVec (PV) [Le and Mikolov, 2014] model, as an extension of the CBOW and Skip-gram models [Mikolov *et al.*, 2013], represents sentences as fixed-length vectors efficiently in a non-compositional way. The DictRep (DictR) [Hill *et al.*, 2016b] model, which learns sentence representations by mapping dictionary definitions to a pre-trained word embeddings of the words defined by those definitions, achieves the best performance on STS task within eight sentence representation models [Hill *et al.*, 2016a].

Comparing the results of the PP-Base, SCBOW-Base and three baselines models, we can see that the PP-Base and DictR models outperform other models by a large margin on most datasets. The PP model is trained with PPDB, which is carefully constructed from a big bilingual parallel corpus. The DictR model utilizes the corpora of dictionary definitions from Wordnet, Wiktionary and online dictionaries. Clearly, these structured corpora consist of rich semantic infor-

⁴The word embeddings is available at <https://github.com/mmihaltz/word2vec-GoogleNews-vectors>

⁵The trained model is available at <http://ttic.uchicago.edu/~wieting/>

⁶This model represents sentence meaning by the same method with ATT-SUR model and replaces value of word surprisal with tf-idf, in which the tf-idf is computed with all the textual similarity datasets and each sentence is viewed as a document.

⁷We have also conducted experiments on baselines like simply reserve nouns, verbs, adjectives and adverbs. The average results on STS datasets are between Base and TF-IDF model with some results better or poorer than Base or TF-IDF on specific datasets.

⁸We employ the trained model released in <https://github.com/ryankiros/skip-thoughts> for evaluation.

¹The corpus can be downloaded from <http://www.cs.toronto.edu/~mbweb/>.

² <http://nlp.stanford.edu/software/tagger.shtml>

³<http://svn.ask.it.usyd.edu.au/trac/candc/wiki/Download>

Table 1: Pearson rank correlation of model predictions with subject similarity ratings on SemEval textual similarity datasets. The bold scores in each row are the best result in the Baselines column, PP model column and the SCBOW model column, respectively. *Base* denotes the model without attention mechanism. Results of PV and DictR are reprinted from [Hill *et al.*, 2016a] where *they are reported for only Semeval 2014 datasets* in two-digit precision.

	Baselines			PP					SCBOW				
	ST	PV	DictR	Base	TF-IDF	ATT-SUR	ATT-POS	ATT-CCG	Base	TF-IDF	ATT-SUR	ATT-POS	ATT-CCG
MSRpar	0.168	-	-	0.476	0.465	0.486	0.497	0.499	0.429	0.412	0.437	0.414	0.419
MSRvid	0.437	-	-	0.774	0.792	0.802	0.846	0.842	0.620	0.611	0.672	0.702	0.734
OnWN	0.413	-	-	0.714	0.725	0.726	0.725	0.727	0.687	0.688	0.677	0.695	0.696
SMTeurop	0.413	-	-	0.481	0.521	0.505	0.493	0.493	0.537	0.542	0.533	0.538	0.552
SMTnews	0.335	-	-	0.652	0.658	0.663	0.664	0.666	0.523	0.525	0.544	0.541	0.557
2012 Average	0.353	-	-	0.619	0.632	0.636	0.645	0.645	0.559	0.556	0.573	0.578	0.592
FNWN	0.201	-	-	0.476	0.500	0.502	0.498	0.507	0.378	0.375	0.350	0.383	0.392
OnWN	0.241	-	-	0.738	0.745	0.760	0.779	0.793	0.584	0.585	0.649	0.609	0.583
headlines	0.334	-	-	0.733	0.738	0.748	0.737	0.736	0.693	0.688	0.705	0.704	0.711
2013 Average	0.259	-	-	0.649	0.661	0.67	0.671	0.679	0.552	0.549	0.568	0.565	0.562
OnWN	0.354	0.51	0.85	0.812	0.810	0.824	0.836	0.841	0.686	0.682	0.729	0.705	0.693
deft-forum	0.249	0.33	0.49	0.540	0.544	0.554	0.558	0.552	0.400	0.400	0.409	0.409	0.421
deft-news	0.369	0.42	0.65	0.739	0.738	0.746	0.741	0.755	0.724	0.719	0.715	0.726	0.733
headlines	0.332	0.46	0.57	0.707	0.713	0.721	0.723	0.722	0.652	0.653	0.657	0.666	0.668
images	0.375	0.32	0.71	0.805	0.808	0.808	0.831	0.831	0.660	0.656	0.682	0.733	0.761
tweets	0.399	0.54	0.67	0.769	0.773	0.777	0.791	0.777	0.708	0.699	0.713	0.754	0.723
2014 Average	0.346	0.43	0.67	0.729	0.731	0.738	0.746	0.746	0.638	0.634	0.651	0.665	0.666
ans-forums	0.299	-	-	0.691	0.684	0.694	0.690	0.692	0.476	0.475	0.531	0.491	0.512
ans-students	0.361	-	-	0.781	0.785	0.782	0.791	0.783	0.723	0.724	0.722	0.715	0.732
belief	0.387	-	-	0.773	0.782	0.783	0.780	0.784	0.584	0.594	0.598	0.600	0.603
headlines	0.415	-	-	0.764	0.768	0.773	0.773	0.774	0.713	0.715	0.723	0.720	0.727
images	0.423	-	-	0.837	0.838	0.841	0.851	0.853	0.740	0.734	0.740	0.773	0.787
2015 Average	0.377	-	-	0.769	0.771	0.775	0.777	0.777	0.647	0.648	0.663	0.660	0.672
answer	0.439	-	-	0.670	0.659	0.669	0.660	0.645	0.398	0.396	0.385	0.379	0.391
deadlines	0.370	-	-	0.699	0.707	0.712	0.709	0.701	0.683	0.668	0.696	0.705	0.701
plagiarism	0.559	-	-	0.802	0.815	0.819	0.814	0.825	0.708	0.699	0.731	0.736	0.736
postediting	0.690	-	-	0.828	0.830	0.831	0.824	0.830	0.708	0.697	0.713	0.700	0.706
question	0.242	-	-	0.535	0.551	0.561	0.582	0.585	0.580	0.475	0.672	0.623	0.652
2016 Average	0.46	-	-	0.707	0.712	0.718	0.717	0.718	0.615	0.587	0.639	0.629	0.637

mation which is extremely helpful for representing sentence meaning. These results suggest that further work of learning general-purpose sentence representations should consider structured corpora, which are widely available on the web.

As it is shown in Table 1, our attention-based sentence representation models outperform state-of-the-art approaches on almost all datasets. However, there are three exceptions: MSRpar-2012, answer-2016 and postediting-2016. Going back to the testing datasets, we find that sentences in MSRpar-2012 and postediting-2016 are really long, with an average length of 20.63 and 19.36. The answer-2016 dataset contains large content of daily conversation, in which function words are in majority. These characteristics may be the reason why our models do not fit in. Another observation is that the performance of the ATT-CCG model is better than the ATT-SUR model and the ATT-POS model, which indicates that the word class with more fine-grained categories (e.g., CCG supertag) is more suitable for building the attention model to improve sentence representations.

4.4 Effects of the Attention Models

To delve deeper into how our models work, we examine the attention weights calculated by the three different attention models and then investigate what have POS tag and CCG supertag vectors learned.

Do ATT-SUR, ATT-POS and ATT-CCG Differ?

The ATT-SUR model uses Surprisal directly as the word attention weight, while the ATT-POS model and ATT-CCG model train the POS tag vectors and CCG supertag vectors, respectively, and calculate the attention weight by dot product with the corresponding word embeddings. Due to different mechanism, we expect to see some difference in the learned attention values. Table 2 shows the lowest (highest) attention values of each word by averaging their attention values in all test datasets.

From Table 2, as expected, we observe quite different patterns within different attention models. For the ATT-POS_(SCBOW) and ATT-CCG_(SCBOW) models, five lowest attention words are function words like prepositions and conjunctions, while the highest attention words are content words. However, for the ATT-SUR model, the words with lowest or highest attention value do not show difference in word syntactic category. The five lowest attention words are those often co-occur with a fixed previous word. For example, *versa* often appears as *vice versa* and *laundering* is frequently used in *money laundering*. The five highest attention words are infrequent words or words occurring in the context that are not included in the training corpus. Moreover, the attention models integrated in the PP model are trained on a much smaller sentence similarity data set. Hence the results reflect

Table 2: The five words with the lowest (highest) attention values generated by the ATT-SUR, ATT-POS and ATT-CCG models.

ATT-SUR		ATT-POS(SCBOW)		ATT-CCG(SCBOW)		ATT-POS(PP)		ATT-CCG(PP)	
low	high	low	high	low	high	low	high	low	high
versa	stenographers	to	skillet	and	skillet	who	revolve	anthropology	bienging
cons	dukes	and	uterus	to	colander	consists	centum	verification	revolve
laundering	unsparing	below	colander	a	window sill	appears	texans	double-check	firstly
pong	mason	incredibly	suction	nor	robe	which	depleted	dashes	centum
ounce	ringers	really	foal	the	porch	seems	armenia	dash	depleted

corpus-specific characteristics which are dissimilar from the attention models trained in unsupervised setting.

What do POS Tag Vectors and CCG Supertag Vectors Capture?

To intuitively show the variations of attention weights by different POS tags⁹, we average the attention values of all words within the same tag and show the results in Figure 1.

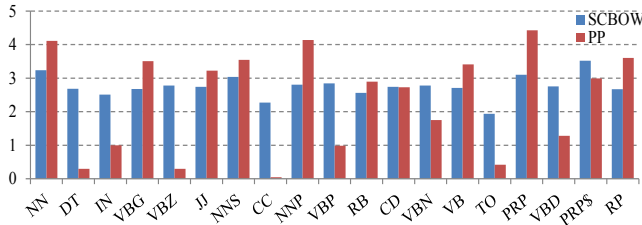


Figure 1: Percentage of attention value for the 20 most frequent POS tags, in which NN is the tag with the highest frequency in a collection of test data sets.

As it is shown in the figure, POS tag vectors learned in the PP model express more variations in attention values. Nevertheless, the similar trend is achieved for tag vectors learned in both SCBOW and PP models, i.e., nouns, verbs and pronouns get more attention, while prepositions and conjunctions gain less attention. The POS tag vectors and CCG supertag vectors are trained together with word vectors: the attention score is computed as the dot product of a tag vector with the corresponding word vector. Therefore, both L_2 norm and direction of a tag vector can affect the results of attention models.

Considering the computation methodology of tag vectors, they might be a mixture of a class of word vectors. Therefore, we calculate the most similar words with POS tag vectors using cosine similarity. The results show that most POS tags can be seen as the prototype of words in the category. For instance, the three most similar words to NN is {baby, mother, kitten}. The three most similar words to VBG is {takeoff, instrument, wrote}. However, there are a few exceptions, such as DT and TO, which return {sahel, pebbly, download}, {patent, mickey, ernst}, respectively. Regarding the L_2 norm, the POS tags with highest values are {NN, PRP\$, TO, PRP, CC} and POS tags with lowest values are {WP, POS, EX, VBZ, WRB}. Together, the results suggest that only the tags that satisfy two factors (1)

⁹Here we take POS tag for example and the same conclusion is achieved in CCG supertag.

owning higher L_2 norm, and (2) sharing similar vector with the corresponding word, such as NN, PRP\$ and PRP, achieve higher attention scores. The tags with higher L_2 norm but do no share similar vector with the corresponding word, such as TO and CC, get lower attention weights.

4.5 Relation with Human Attention

We have built attention models using significant predictors of human reading time, thus we wonder if the proposed models which are learned from large-scale corpus share similarity with human attention. Therefore, we calculate correlation between our proposed attention models and human reading times in the Dundee corpus [Kennedy *et al.*, 2003].

The Dundee corpus is widely used in researches on human reading and consists of eye-tracking data for 10 subjects reading 20 newswire articles (about 51,000 words). It contains three measures of reading time:

- First-pass time (RT_{pass}) measures the total time spent reading a word before the first fixation on any other word. It is also known as gaze duration.
- Go-past time (RT_{gopast}) measures the total time spent reading a word from the first fixation up to the first fixation on a word further to the right. This often includes the reading time on words to the left of the current word.
- Right-bounded time (RT_{rb}) measures the total time spent reading a word before the first fixation on a word further to the right.

Table 3 shows the results of the linear correlation (Pearson rank) and monotonic correlation (Spearman rank) between our models and the averaged human reading times. As it can be seen, the attention weights calculated by our models are highly correlated with human reading time, which suggests that the proposed attention models might have some cognitive plausibility. Furthermore, the attention models with more fine-grained CCG supertags are more relevant to human attention than the attention models with POS tag. Although Surprisal has long been seen as a strong predictor of human reading time, the ATT-CCG model, when incorporated into SCBOW model, reaches even higher correlation scores. These results suggest that our data-driven attention models can predict human reading time to some extent, which may lead to new insights in studies of human reading behaviors.

5 Related Work

Learning meaningful sentence representations is the first step towards the goal of language understanding, which has at-

Table 3: Pearson/Spearman rank correlation between human reading times and the attention values predicted by our models. The asterisk means significantly for p-values < 0.0001.

Models	RTfpass	RTgopast	RTrb
ATT-SUR	0.412*/0.353*	0.382*/0.288*	0.413*/0.349*
ATT-POS _(SCBOW)	0.385*/0.307*	0.370*/0.268*	0.386*/0.302*
ATT-CCG _(SCBOW)	0.430*/0.347*	0.412*/0.306*	0.431*/0.342*
ATT-POS _(PP)	0.373*/0.123*	0.353*/0.104*	0.373*/0.120*
ATT-CCG _(PP)	0.390*/0.093*	0.367*/0.076*	0.390*/0.091*

tracted a significant amount of research attention. Recently, neural network based methods have shown advantage in learning task-specific sentence representations [Tai *et al.*, 2015; Chen *et al.*, 2015] and general-purpose sentence representations [Le and Mikolov, 2014; Kiros *et al.*, 2015; Kenter *et al.*, 2016; Wieting *et al.*, 2016]. Our work addresses the problem of *learning general-purpose sentence representations* that capture textual semantics and perform robustly across tasks.

This paper incorporates the proposed attention model into the averaging model, which results a weighted additive model. It is necessary to note that the proposed attention model can also be combined with more complex sentence representation models. Though this is beyond the focus of this paper, it remains an interesting direction to explore as a future work. In compositional distributional semantics, the effectiveness of weighted addition is first emphasized by [Mitchell and Lapata, 2010]. They use weighted summation of word vectors in a text, in which weights are parameters (i.e., single value for a word) learned together with word embeddings. This idea is also used in learning word [Ling *et al.*, 2015; Wang *et al.*, 2016] and phrase [Yu and Dredze, 2015; Wang and Zong, 2017] representations. [Ling *et al.*, 2015; Wang *et al.*, 2016] uses weighted summation of context words to predict the target word in learning word representations. The weights are a set of parameters, determining the importance of each word in each relative position in the predefined context window size. Regarding phrase representations, [Yu and Dredze, 2015] employs attention weights as linear combinations of various features, whereas [Wang and Zong, 2017] utilizes L_2 norm of enhanced word representations as implicit attention weights. Different from these work, we focus on learning representations of sentences and design cognition-inspired methods to calculate attention weights.

As for sentence representations, [Lin *et al.*, 2016; Yang *et al.*, 2016] used hidden layers of the sequence model (i.e., LSTM) as input to a MLP model to get the attention weight of each word. However, this attention mechanism is designed for sequence models and need large supervised training data. In contrast, ours can be easily applied to various kinds of models and can be learned without or with a little supervision. In parallel to our work, [Arora *et al.*, 2016] proposes a similar approach to improve sentence representations by using smooth inverse frequency (SIF) to weight words in a sentence, which achieves comparable results with ours. The most crucial difference is that we employ cognitive plausible factors as auxiliary information to compute attention weights, which brings a lot of scalability, making it applicable to oth-

er tasks like recognizing textual entailments. Furthermore, we observe that the performance is improved when applying SIF to ATT-POS (ATT-CCG) model. This indicates that these models capture complementary information and thus an optimal combination of them will probably lead to better attention models, which we leave further investigation to future work.

Another branch of related work is computational models for human reading behaviours [Keller, 2016]. Recently, there are increasing research interests in improving the performance of natural language processing tasks by employing the information in an eye-tracking corpus. For instance, [Klerke *et al.*, 2016] presents a multi-task learning algorithm to improve performance of sentence compression by using an auxiliary training task that is predicting human reading times. [Barrett *et al.*, 2016] improves the state-of-the-art part of speech tagging (POS) model with various features derived from an eye-tracking corpus. The work presented in this paper, to our best knowledge, is the first study to improve sentence representations with guidance of human attention in reading. Together, these results indicate that traces or predictors of human cognitive processing, such as the eye-tracking data in reading behaviours, can be used to augment natural language processing models.

6 Conclusions

Motivated by the fact that human read sentences by selectively focusing on importance words, in this paper we have demonstrated that introducing such attention mechanism can enhance semantic representations of sentences.

To attach different weights to vectors of the component words in a sentence, we propose two novel approaches that use significant predictors of human reading time to guide the construction of attention models on single sentences. Experimental evaluations show that our approaches lead to substantial gains in accuracy on nearly all 24 SemEval datasets. Qualitative analyses have indicated that the proposed attention models can selectively focus on important words and successfully predict human reading times. These results, coupled with findings from [Klerke *et al.*, 2016] and [Barrett *et al.*, 2016], suggest that rich information contained in human cognitive processing can be used to enhance NLP models.

Acknowledgments

The research work has been funded by the Natural Science Foundation of China under Grant No. 61333018 and No. 91520204 and supported by the Strategic Priority Research Program of the CAS under Grant No. XDB02070007.

References

- [Arora *et al.*, 2016] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. *arXiv*, 2016.
- [Barrett *et al.*, 2016] Maria Barrett, Joachim Bingel, Frank Keller, and Anders Søgaard. Weakly supervised part-of-speech tagging using eye-tracking data. *The 54th Annual Meeting of the Association for Computational Linguistics*, page 579, 2016.
- [Chen *et al.*, 2015] Xinchu Chen, Xipeng Qiu, Chenxi Zhu, Shiyu Wu, and Xuanjing Huang. Sentence modeling with gated recursive neural network. *Proceedings of EMNLP*, pages 793–798, 2015.
- [Demberg and Keller, 2008] Vera Demberg and Frank Keller. Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109(2):193–210, 2008.
- [Ganitkevitch *et al.*, 2013] Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. PPDB: The paraphrase database. *HLT-NAACL*, pages 758–764, 2013.
- [Hale, 2001] John Hale. A probabilistic earley parser as a psycholinguistic model. *Proceedings of the NAACL-HLT*, pages 1–8, 2001.
- [Hill *et al.*, 2016a] Felix Hill, Kyunghyun Cho, and Anna Korhonen. Learning distributed representations of sentences from unlabelled data. *Proceedings of NAACL-HLT 2016*, pages 1367–1377, 2016.
- [Hill *et al.*, 2016b] Felix Hill, Kyunghyun Cho, Anna Korhonen, and Yoshua Bengio. Learning to understand phrases by embedding the dictionary. *Transactions of the Association for Computational Linguistics*, 4:17–30, 2016.
- [Jozefowicz *et al.*, 2016] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*, 2016.
- [Keller, 2016] Michael Hahn Frank Keller. Modeling human reading with neural attention. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 85(95):95, 2016.
- [Kennedy *et al.*, 2003] Alan Kennedy, Robin Hill, and Joël Pynte. The dundee corpus. *Proceedings of the 12th European conference on eye movement*, 2003.
- [Kenter *et al.*, 2016] Tom Kenter, Alexey Borisov, and Maarten de Rijke. Siamese cbow: Optimizing word embeddings for sentence representations. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 941–951, 2016.
- [Kiros *et al.*, 2015] Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. *Advances in neural information processing systems*, pages 3294–3302, 2015.
- [Klerke *et al.*, 2016] Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. Improving sentence compression by learning to predict gaze. *Proceedings of NAACL-HLT*, pages 1528–1533, 2016.
- [Le and Mikolov, 2014] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. *Proceedings of the 31st International Conference on Machine Learning*, pages 1188–1196, 2014.
- [Levy, 2008] Roger Levy. Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177, 2008.
- [Lin *et al.*, 2016] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A self-attentive sentence embedding. *arXiv*, 2016.
- [Ling *et al.*, 2015] Wang Ling, Lin Chu-Cheng, Yulia Tsvetkov, and Silvio Amir. Not all contexts are created equal: Better word representations with variable attention. *Proceedings of the EMNLP*, pages 1367–1372, 2015.
- [Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [Mitchell and Lapata, 2010] Jeff Mitchell and Mirella Lapata. Composition in distributional models of semantics. *Cognitive science*, 34(8):1388–1429, 2010.
- [Rayner, 1998] Keith Rayner. Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124(3):372, 1998.
- [Tai *et al.*, 2015] Kai Sheng Tai, Richard Socher, and Christopher D Manning. Improved semantic representations from tree-structured long short-term memory networks. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 1556–1566, 2015.
- [Wang and Zong, 2017] Shaonan Wang and Chengqing Zong. Comparison study on critical components in composition model for phrase representation. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 16(3):16, 2017.
- [Wang *et al.*, 2016] Yashen Wang, Huang Heyan, Feng Chong, Zhou Qiang, Gu Jiahui, and Xiong Gao. Cse: Conceptual sentence embeddings based on attention model. *The 54th Annual Meeting of the Association for Computational Linguistics*, pages 505–515, 2016.
- [Wieting *et al.*, 2016] John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. Towards universal paraphrastic sentence embeddings. *ICLR*, 2016.
- [Yang *et al.*, 2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. *Proceedings of the NAACL*, 2016.
- [Yu and Dredze, 2015] Mo Yu and Mark Dredze. Learning composition models for phrase embeddings. *Transactions of the Association for Computational Linguistics*, 3:227–242, 2015.