Hitachi at SemEval-2020 Task 3: Exploring the Representation Spaces of Transformers for the Human Perception of Word Similarity

Terufumi Morishita, Gaku Morio, Hiroaki Ozaki and Toshinori Miyoshi
Hitachi, Ltd.
Research and Development Group
Kokubunji, Tokyo, Japan.
{terufumi.morishita.wp, gaku.morio.vn, hiroaki.ozaki.yu, toshinori.miyoshi.pd}@hitachi.com

Abstract

In this paper, we present our system for SemEval-2020 task 3, Predicting the (Graded) Effect of Context in Word Similarity. Due to the unsupervised nature of the task, we concentrated on inquiring about the similarity measures induced by different layers of different pre-trained Transformer-based language models, which can be good approximations of the human sense of word similarity. Interestingly, our experiments reveal a language-independent characteristic: the middle to upper layers of Transformer-based language models can induce good approximate similarity measures. Finally, our system was ranked 1st on the Slovenian part of Subtask1 and 2nd on the Croatian part of both Subtask1 and Subtask2.

1 Introduction

In this paper, we describe our participation in SemEval-2020 task 3: Predicting the (Graded) Effect of Context in Word Similarity (Armendariz et al., 2020). The goal of the task is to understand the effect of contexts on word similarity. The task is composed of two subtasks sharing inputs: we are given a word pair (i.e., two words) and two text snippets (hereafter “contexts”) both including the word pair. For convenience of explanation, we describe Subtask2 first. In Subtask2, we predict two similarity scores between the word pair in the two given contexts. In Subtask1, we predict the difference in the above two similarity scores. More detailed descriptions of the subtasks are given in Section 3.

In both subtasks, small labeled data is available for model development. Thus, participants are required to build models in an unsupervised manner.

We formulated the tasks as the exploration of similarity measures induced in the hidden layer word representation space of pre-trained Transformer (Vaswani et al., 2017) based language models. Our expectation is that contextualized representation of Transformers could induce context dependent similarity measures that approximate the human perception.

Experimental results show that better approximations of word sense similarity can be induced in the middle to upper layers of Transformers for most languages. As a result, our system was ranked 1st on the Slovenian part of Subtask1 and 2nd on the Croatian part of both Subtask1 and Subtask2.

2 Background

Capturing the similarity between words has been considered to be one of the fundamental tasks in natural language processing research because it is strongly related to many research fields such as text search, entailment recognition, and information extraction. Recent work has been aimed at predicting the similarity of a given word pair (Camacho-Collados et al., 2017; Mikolov et al., 2013) and considers the similarity of word meanings that does not consider the effects of their contexts. Word-Sense Disambiguation (WSD) (Miller et al., 2012; Raganato et al., 2017), which is a lexicographical approach to the representation of word senses, aims at selecting an appropriate sense for a given word from word-specific sense candidates. Alongside with these two lines of the work, Armendariz et al. (2020) extended...
was almost an improvement over her room at the last hostel.

His job as a biologist didn’t leave much room for a personal life. He knew much more about human

cells than about human feelings.

Table 1: Example of data used in task. In this case, \( w_1 = “cell,” \) and \( w_2 = “room.” \) In \( c_1 \), both \( w_1 \) and \( w_2 \) refer to different kinds of rooms, so similarity score \( s_1 \) is rather high. In \( c_2 \), \( w_1 \) is used as biological term, while \( w_2 \) means abstract concept, so similarity score \( s_2 \) is low. Note that this sample is taken from official competition examples, and scores are virtual since we do not know gold ones.

similarity-based word-sense detection in SemEval-2020 task 3. This task focuses on the effect of contexts on word similarity. More concretely, the task aims at predicting the similarity of a given word pair in different contexts.

Recently proposed contextual word vectors, especially those of Transformer-based language models, are considered to be able to capture context-dependent word meanings (Ethayarajh, 2019). We utilize these vectors for the task.

3 Task Formalization

The task includes two subtasks, both of which aim at capturing the effect of context on similarities of word pairs. Each subtask has four “sub-subtasks”, each of those corresponding to each of four languages, namely, English, Finnish, Hungarian, and Slovenian.

As shown in Table 1, let \( w = (w_1, w_2) \) denote the given word pair, \( c = (c_1, c_2) \) the two different contexts, and \( s = (s_1, s_2) \) the human-annotated similarity scores of \( w \) for each context \( (c_1, c_2) \). \( s_i \) is annotated as an integer number in the range of \([0, 10]\). The higher the value is, the more similar \( w_1 \) and \( w_2 \) are.

Given \( w \) and \( c \), Subtask 1 aims at predicting \( d = s_2 - s_1 \), which expresses the change in similarity scores caused by contexts. The metric of Subtask 1 is the Pearson correlation coefficient between gold labels and predictions. Due to the translational and scale invariance of Pearson correlation, we can use the \([-1, +1]\) range instead of \([0, 10]\). Using this same input for Subtask 1, Subtask 2 aims at predicting \( s_1 \) and \( s_2 \) directly. The evaluation metric is the uncentered Pearson correlation coefficient; thus, we can use any range in \( \mathbb{R} \) as well.

We take a two-stage approach; (i) we first solve Subtask 2 by predicting \( s_1 \) and \( s_2 \) directly, and (ii) we second solve Subtask 1 by calculating \( d \) from the predicted \( s_1 \) and \( s_2 \).

4 Model

As we mentioned in the above, we explore similarity measures in Transformer’s representation space exhaustively, which will represent the human sense of word similarity well. We introduce a cosine similarity-based measure and then take a “layer-wise” exploration strategy.

Transformer Similarity

Because the input contexts are plain text, we apply two-level tokenization (i.e., word-level tokenization and subword-level tokenization) for each context \( c_1 \) and \( c_2 \) and then feed the subword-level tokens to a Transformer-based language model to get contextual word vectors:

\[
\begin{align*}
\mathbf{v}_{11}^{(\tau,\lambda)} &= \mathbf{e}^{(\tau,\lambda)}(w_1, c_1), & \mathbf{v}_{21}^{(\tau,\lambda)} &= \mathbf{e}^{(\tau,\lambda)}(w_2, c_1), \\
\mathbf{v}_{12}^{(\tau,\lambda)} &= \mathbf{e}^{(\tau,\lambda)}(w_1, c_2), & \mathbf{v}_{22}^{(\tau,\lambda)} &= \mathbf{e}^{(\tau,\lambda)}(w_2, c_2),
\end{align*}
\]

where \( \mathbf{e}^{(\tau,\lambda)}(w, c) \) represents the representation vector of word \( w \) in context \( c \), taken from the \( \lambda \)-th layer of the given Transformer-based language model \( \tau \). To get a word-level token representation \( w \), we take an average of all the representation vectors of the corresponding subword-level tokens.

To calculate the similarity between two words, we take cosine-similarity (written as “sim”) between the corresponding word vectors. Cosine-similarity scores between the contextualized vectors are repre-
sent as the following operations.

\[ s_1^{(\tau,\lambda)} = \text{sim} \left( \mathbf{v}_{11}^{(\tau,\lambda)}, \mathbf{v}_{21}^{(\tau,\lambda)} \right), \]
\[ s_2^{(\tau,\lambda)} = \text{sim} \left( \mathbf{v}_{12}^{(\tau,\lambda)}, \mathbf{v}_{22}^{(\tau,\lambda)} \right). \]

For **Subtask 1**, we predict the similarity score difference \( d^{(\tau,\lambda)} \) by:

\[ d^{(\tau,\lambda)} = s_2^{(\tau,\lambda)} - s_1^{(\tau,\lambda)} \]

For **Subtask 2**, we simply exploit the above similarity scores \( s_1^{(\tau,\lambda)} \) and \( s_2^{(\tau,\lambda)} \) as our predictions.

**Exploration on Representation Space**

As described in the above, each combination of \((\tau, \lambda)\) induces a similarity measure. Therefore, we define **Exploration Space** \( \Theta = \{(\tau, \lambda) | \tau \in \text{Transformers}, \lambda \in \text{Layers}(\tau) \} \), where \text{Transformers} represents a set of Transformer-based language model types, and \text{Layers}(\tau) represents the set of layer indices that Transformer-based language model \( \tau \) contains. We investigate \( \Theta \) to find the one that approximates gold similarity the best. Details on \text{Transformers} and \text{Layers} are given in Section 5.

**Rank-Weighted Voting for English**

Relatively larger number of pieces of annotated data are available in the English task. This enables us to tune a slightly more complicated system for better predicting gold labels. Therefore, we decided to build a special system for the English task that utilizes the multiple predictions made from different similarity measures in \( \Theta \) for more robust predictions.

First, we sort the predictions of different similarity measures in \( \Theta \) in the order of the overall performance on the development data, that is, in descending order of the Pearson coefficients between the predictions and gold labels. Let \( y_r \) denote the prediction on a given sample made by the \( r \)-ranked similarity measure. Concretely, \( y_r \) corresponds to \( d^{(\tau_r,\lambda_r)} \) in Subtask 1, \( s_1^{(\tau_r,\lambda_r)} \) and \( s_2^{(\tau_r,\lambda_r)} \) in Subtask 2. We calculate the rank-decayed weighted average of the predictions:

\[ y = \sum_r \omega(r) F(y_r), \quad \omega(r) = \frac{\exp(-r/R)}{\sum_r \exp(-r/R)}, \]

where \( R \) is a tunable hyperparameter representing the rank decay rate, and \( F \) is a non-linear transformation function. Note that the non-linearity of \( F \) is a significant property. Using a linear transformation function is equivalent to only taking weighted \( n \)-best predictions, which have a smaller Pearson coefficient.\(^1\)

**5 Experiments**

**Setup**

We employed six types of Transformer-based language models as shown in Table 2. For the non-English languages, we used multilingual models, namely, multilingual BERT (Devlin et al., 2019) and XLM-RoBERTa (Conneau et al., 2019). For English, we employed BERT (monolingual/multilingual) (Devlin et al., 2019), GPT-2 (Radford et al., 2019), Transformer-XL (Dai et al., 2019), XLNet (Yang et al., 2019), and XLM-RoBERTa (Conneau et al., 2019).

\(^1\)Let \( y \) denote a linear combination of uncorrelated stochastic variables: \( y = \sum_r \omega(r)y_r \). Let \( l \) denote another stochastic variable. In our case, \( y_r \) is the prediction of the \( r \)-th ranked similarity measure and \( l \) the gold label. By simple calculation, we can show that the Pearson coefficient \( \text{Pearson}(y, l) \) takes the max when the \( \omega(r) \) is taken as follows.

\[ \omega(r) = \begin{cases} 1 & (r = r_0) \\ 0 & (\text{otherwise}) \end{cases} \]

\[ \text{Pearson}(y_{r_0}, l) \geq \text{Pearson}(y, l) \]

Although, in our case, \( y_r \) does have correlations if taken from a different layer of the same Transformer, the correlations may originate from rather trivial degeneracy, which we do not want the system to rely on.
Table 2: Provided Transformer-based language models. ✓ shows models used in our experiments.

| Transformers              | Layers | type            | English | Hungarian | Finnish | Slovenian |
|---------------------------|--------|-----------------|---------|-----------|---------|-----------|
| BERT (Devlin et al., 2019) | 24     | large-uncased   | ✓       |           |         |           |
| BERT(multilingual) (Devlin et al., 2019) | 12     | base-cased      | ✓ ✓ ✓ ✓ | ✓         | ✓         | ✓         |
| GPT-2 (Radford et al., 2019) | 24     | medium          | ✓       |           |         |           |
| Transformer-XL (Dai et al., 2019) | 18     | wt103           | ✓       |           |         |           |
| XLNet (Yang et al., 2019)   | 24     | large-cased     | ✓       |           |         |           |
| XLM-RoBERTa (Conneau et al., 2019) | 24     | large           | ✓ ✓ ✓ ✓ | ✓         | ✓         | ✓         |

Table 3: Submitted similarity measures. m-B represents multilingual-BERT.

| model layer index | English | Subtask1 | Hungarian | Subtask2 | Slovenian |
|-------------------|---------|----------|-----------|----------|-----------|
|                   | weighted-voting | m-B | m-B | m-B | weighted-voting | m-B | m-B | m-B |
| (a) Subtask1 - multilingual-BERT | - | 8 | 8 | 8 | - | 5 | 12 | 6 |
| (b) Subtask1 - XLM-RoBERTa | - | 7 | 13 | 19 | 25 | - | 7 | 13 | 25 |
| (c) Subtask2 - multilingual-BERT | - | 1 | 4 | 7 | 10 | 13 | - | 1 | 4 | 7 | 10 | 13 |
| (d) Subtask2 - XLM-RoBERTa | - | 1 | 7 | 13 | 19 | 25 | - | 1 | 7 | 13 | 19 | 25 |

Figure 1: Pearson coefficient heatmaps for each Transformer layer in multi-lingual models (i.e., multilingual-BERT) and XLM-RoBERTa. Brighter color means higher value. “average” shows layer-wise average over three languages.

We employed the log function as the scaling function $F$ for English, which performed best on the development data.

All of the experimental code was implemented with PyTorch (Paszke et al., 2019) and jiant (Pruskachatkun et al., 2020). jiant is a recently developed transfer learning framework, which in turn utilizes Hugging Face’s library (Wolf et al., 2019) for Transformer-based language models and their tokenizers.

Results

Table 4 and Table 5 present the official ranking of the subtasks. We submitted the similarity measures described in Table 3, which performed the best on the test data among the fixed number of trials.2 The similarity measures that performed the best on the development data were selected for the trials.

Interestingly, for all the non-English language tasks that employed multilingual-BERT and XLM-RoBERTa, multilingual-BERT outperformed XLM-RoBERTa. Furthermore, for Subtask 1, the 8th-layer outperformed the other layers submitted for the trials.

Which Layer Approximates the Human Sense of Similarity Better?: Figure 1 shows heatmaps of Pearson coefficients between the gold labels and the predictions made by two of the Transformer-based language models (i.e., multilingual-BERT and XLM-RoBERTa), calculated on the development data. We also show the layer-wise averages over the languages. Note that the Finnish results are not shown.

---

2Trials on the test set were permitted up to 9 times.
Table 4: Official ranking of Subtask 1. Values shown are Pearson coefficients.

| Subtask1 |          |          |          |
|----------|----------|----------|----------|
| **English** | **Croatian** | **Slovenian** | **Finnish** |
| Ferryman | 0.774 | BabelEnconding | 0.74 | Hitachi (ours) | 0.654 | will_go | 0.772 |
| will_go | 0.768 | Hitachi (ours) | 0.681 | BRUMS | 0.648 | Ferryman | 0.745 |
| MultiSem | 0.76 | BRUMS | 0.664 | BabelEnconding | 0.646 | BabelEnconding | 0.726 |
| LMMS | 0.754 | Ferryman | 0.634 | CITIUS-NLP | 0.624 | BRUMS | 0.671 |
| BRUMS | 0.754 | LMMS | 0.616 | Ferryman | 0.606 | CITIUS-NLP | 0.671 |
| Hitachi (ours) | 0.749 | will_go | 0.597 | will_go | 0.603 | MultiSem | 0.593 |
| BabelEnconding | 0.73 | CITIUS-NLP | 0.587 | LMMS | 0.56 | Hitachi (ours) | 0.574 |
| CITIUS-NLP | 0.721 | MineriaUNAM | 0.374 | MineriaUNAM | 0.328 | MineriaUNAM | 0.389 |
| MineriaUNAM | 0.544 | MultiSem | 0 | MultiSem | 0 | LMMS | 0.36 |

Table 5: Official ranking of Subtask 2. Values shown are Pearson coefficients.

| Subtask2 |          |          |          |
|----------|----------|----------|----------|
| **English** | **Croatian** | **Slovenian** | **Finnish** |
| MineriaUNAM | 0.723 | BabelEnconding | 0.658 | BabelEnconding | 0.579 | BRUMS | 0.645 |
| LMMS | 0.72 | Hitachi (ours) | 0.616 | BRUMS | 0.573 | BRUMS | 0.611 |
| AlexU-Aux-Bert | 0.719 | MineriaUNAM | 0.613 | CITIUS-NLP | 0.538 | MineriaUNAM | 0.597 |
| MultiSem | 0.718 | LMMS | 0.565 | will_go | 0.516 | MultiSem | 0.492 |
| BRUMS | 0.715 | BRUMS | 0.545 | AlexU-Aux-Bert | 0.516 | Ferryman | 0.357 |
| will_go | 0.695 | CITIUS-NLP | 0.496 | Hitachi (ours) | 0.514 | LMMS | 0.354 |
| Hitachi (ours) | 0.695 | AlexU-Aux-Bert | 0.402 | MineriaUNAM | 0.487 | will_go | 0.35 |
| CITIUS-NLP | 0.687 | will_go | 0.402 | LMMS | 0.483 | Hitachi (ours) | 0.335 |
| BabelEnconding | 0.634 | Ferryman | 0.397 | Ferryman | 0.345 | CITIUS-NLP | 0.289 |
| Ferryman | 0.437 | MultiSem | 0 | MultiSem | 0 | AlexU-Aux-Bert | 0.289 |

because no development data was distributed.

We can see from Figure 1 that better approximations of word sense similarity can be induced in the middle to upper layers of Transformers for most of the languages. This is also consistent with the intended design of the multi-layered self-attention mechanism, which aims to obtain more contextualized word representations on the upper layers.

Looking more into detail, there are different characteristics between the multilingual-BERT and XLM-RoBERTa. For multilingual-BERT, it seems that the deeper the layer is, the higher the performance is. For XLM-RoBERTa, the middle layers tend to perform better than the other layers. This implies that different Transformer language models capture word similarity differently.

6 Conclusion

In this paper, we proposed a model for the task of capturing the effects of context on word similarity. We employed similarity measures induced by the hidden layer representation vectors of pre-trained Transformer-based language models. We explored all the layers of the models to find the one that matches human perception the best.

Our experimental results show that the multi-layered self-attention mechanism of Transformer-based language models successfully captures the human sense of context-dependent word similarity. The results also revealed a universal language characteristic, that is, for all the Transformer-based language models, the middle to upper layers perform better on the task than the others.

Acknowledgments

Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used. We thank Dr. Masaaki Shimizu for the convenience of computational resources.

References

Carlos S. Armendariz, Matthew Purver, Senja Pollak, Nikola Ljubešić, Matej Ulčar, Marko Robnik-Šikonja, Ivan Vulić, and Mohammad Taher Pilehvar. 2020. SemEval-2020 task 3: Graded word similarity in context (GWSC). In Proceedings of the 14th International Workshop on Semantic Evaluation.
