JBNU-CCLab at SemEval-2022 Task 7: DeBERTa for Identifying Plausible Clarifications in Instructional Texts

Daewook Kang, Sung-Min Lee, Eunhwan Park, Seung-Hoon Na
Computer Science and Engineering, Jeonbuk National University, South Korea
{dwkng, cap1232, judepark, nash}@jbnu.ac.kr

Abstract

In this study, we examine the ability of *contextualized representations* of pretrained language model to distinguish whether sequences from instructional articles are plausible or implausible. Towards this end, we compare the BERT, RoBERTa, and DeBERTa models using simple classifiers based on the sentence representations of the [CLS] tokens and perform a detailed analysis by visualizing the representations of the [CLS] tokens of the models. In the experimental results of Subtask A: *Multi-Class Classification*, DeBERTa exhibits the best performance and produces a more distinguishable representation across different labels. Submitting an ensemble of 10 DeBERTa-based models, our final system achieves an accuracy of 61.4% and is ranked fifth out of models submitted by eight teams. Further in-depth results suggest that the abilities of pretrained language models for the plausibility detection task are more strongly affected by their model structures or attention designs than by their model sizes.

1 Introduction

WikiHow[1] is the largest how-to website with more than 300,000 articles and over 2.5M registered users that help user improve their knowledge of specific areas. However, these instructional articles have grammatical errors and ambiguous content that cause misunderstandings. To enhance the clarity of instructional texts, *clarification* is required as a revision that makes implicit elements explicit, resolves ambiguities, or replaces underspecified phrases with a clearer and more precise expressions.

SemEval-2022 Task 7 (Roth et al., 2022) evaluates the ability of an NLP system to distinguish between plausible and implausible clarifications of an instruction. The task is formulated as a CLOZE task in which clarification is presented as a *filler* in a blanked sentence. Given a context with a filler option (i.e., a sentence X and filler option O), the system should determine the plausibility of a sequence, that is whether a sequence is "plausible," "neutral," and "implausible."

Our work is motivated by the recent successes of pretrained language models such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), and DeBERTa (He et al., 2021a,b), which have effectively induced *contextualized representations*, achieving remarkable fine-tuning performance on downstream tasks.

To explore the plausibility detection of clarifications in SemEval-2022 Task 7, we compare the plausibility detection abilities of three pretrained language models (BERT, RoBERTa, and DeBERTa) in Subtask A, *Multi-Class Classification*. Our main observations of the development set are highlighted as follows.

- Among the three models, DeBERTa exhibits the best performance in the plausibility classification task[2].
- Visualization analysis of the representations of the [CLS] tokens of BERT, RoBERTa, and DeBERTa, reveals that the best distinguishable representations among different classes are achieved with DeBERTa.
- A comparison of base and large models confirms that large models are always better than base models for all the types: BERT, RoBERTa and DeBERTa.
- Given the comparative results among various models, we hypothesize that the abilities of pretrained language models for the plausibility detection task are more strongly affected

[1] https://www.wikihow.com/Main-Page

[2] Note that we used BERT-{base, large}, RoBERTa-{base, large}, and DeBERTa-{base, large}. DeBERTa V3 model of (He et al., 2021a) was used for DeBERTa models.

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1050 - 1055
July 14-15, 2022 ©2022 Association for Computational Linguistics

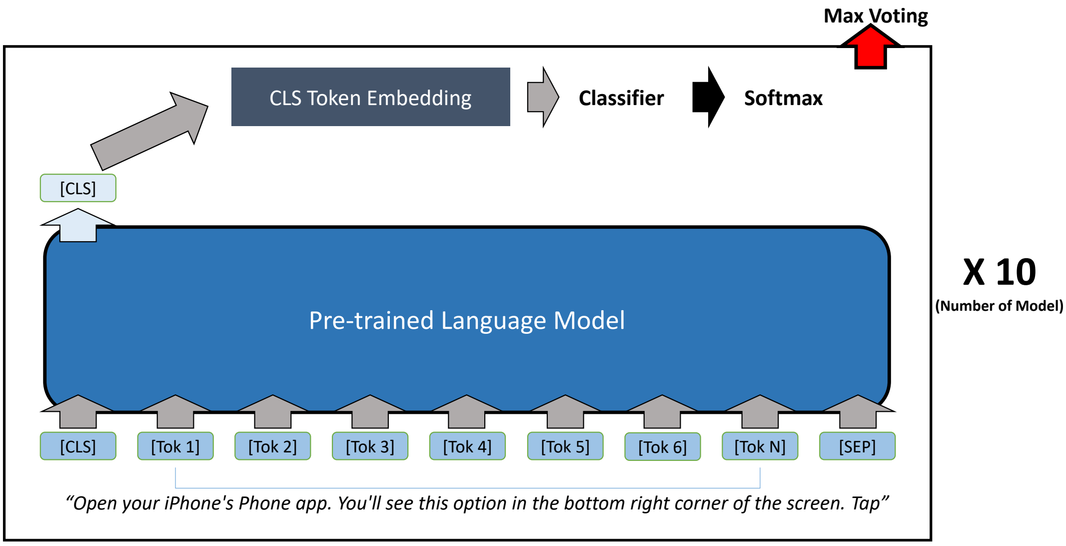


Figure 1: Architecture of the proposed system for plausibility classification using pretrained language models

by their model structures or attention designs rather than their parameter sizes.

By ensembling 10 different DeBERTa-based models, our final submitted system achieves an accuracy of 61.4% on the test data and is ranked the 8th place among 21 systems[3], that is, 5^th th place among the 8 teams who submitted their papers on Subtask A.

The remainder of this paper is organized as follows: Section 2 presents related work. Section 3 describes the system architecture in detail. Section 4 describes the experimental settings, results, and analyses. Our concluding remarks and a description of future work are presented in Section 5.

2 Related work

Since the success of BERT (Devlin et al., 2019), it has been used for numerous natural language processing (NLP) tasks and has inspired the emergence of many other pretrained models. In RoBERTa (Liu et al., 2019), dynamic masking in the masked language modeling (MLM) objective dynamically while revisiting the next sentence prediction owing to its uncertain effectiveness has achieved promising results on GLUE (Wang et al., 2018), RACE (Lai et al., 2017), and SQuAD (Rajpurkar et al., 2016). DeBERTa (He et al., 2021b) further advanced BERT based on two major extensions, *disentangled attention* and *enhanced mask decoder*, by combining both the relative and absolute positions of words. DeBERTa V3 (He et al., 2021a) replaces the MLM objective in DeBERTa with the replaced token detection (RTD) objective proposed by ELECTRA (Clark et al., 2020) and further proposes *gradient-disentangled embedding sharing* to alleviate the *tug-of-war* problem between the generator and discriminator [4] as an improvement of the embedding sharing method used in ELECTRA.

3 System description

Figure 1 presents the architecture of our system, which uses pretrained language models equipped with ensemble inference.

3.1 Methods with pretrained language models

Let X^p, X^c and X^n be the previous, main, and follow-up context, respectively. As in Section 4.1, we concatenate these sentences for the i-th training example as follows: $X_i = X^p \oplus X^c \oplus X^n$, where X_c is unmasked by filling each possible filler option O_{ij}. We use $Y_i \in \{0, 1, 2\}$ to refer to the ground truth of the i-th example, where Y_0, Y_1, and Y_1 refer to *implausible*, *neutral*, and *plausible* sequences, respectively. Finally, we denote a training set as $\mathcal{D} = \{X_i, Y_i\}_{i=1}^N$, where N is the total number of training examples obtained by unmasking all the main sentences with their possible filler options.

We feed X_i into a pretrained language model denoted as LM to encode contextualized represen-

[3] https://competitions.codalab.org/competitions/35210

[4] Because the generator and discriminator have different objectives, they tend to pull shared word embeddings in different directions, resulting in degradation of the training speed.

tations $\mathbf{M}_i \in \mathbb{R}^{|X_i| \times d}$ as follows:

$$\mathbf{M}_i = \mathsf{LM}(X_i)$$

where $|X_i|$ is the length of the concatenated sentence X_i and d is the dimensionality of the hidden representation of LM. As mentioned, we use BERT, RoBERTa, and DeBERTa for LM.

Let $\mathbf{M}_{i,[CLS]}$ be the representation of the [CLS] token of X_i. To perform a plausibility prediction, we feed $\mathbf{M}_{i,[CLS]}$ to a linear layer as follows:

$$f(X_i) = \mathbf{W}^T \mathbf{M}_{i,[CLS]} + \mathbf{b}, \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d \times 3}$ and $\mathbf{b} \in \mathbb{R}^3$ are task-specific parameters of the linear layer. The loss function $\mathcal{L}$ used to optimize our system is formulated as follows:

$$\mathcal{L} = - \sum_{(X_i, Y_i) \in \mathcal{D}} \mathbf{y}_i \cdot \log \mathrm{softmax}(f(X_i)) \tag{2}$$

where $\mathbf{y}_i \in \{0, 1\}^3$ is the one-hot vector for Y_i.

4 Experiments

Team Name (or User Name)	**Accuracy**
X-PuDu	68.9
HW-TSC	66.1
PALI	65.4
Nowruz	62.4
DuluthNLP	53.3
Stanford MLab	46.6
niksss	44.2
JBNU-CCLAB	**61.40**

Table 1: Results of our system on the test dataset (Official Leaderboard)

4.1 Experimental setting

Dataset We used the data and labels from SemEval-2022 Task 7 (Roth et al., 2022). The data consists of the article title, sub-heading, masked sentence, previous and follow-up context, and possible filler options with corresponding labels (Subtask A) and ratings (Subtask B). During our preprocessing step, the placeholder in the main sentence is filled with each possible filler option O_i.

Preprocessing We concatenate the previous context, main context, and follow-up context without using the article name and section header; any parenthesis and the content inside, special characters, and redundant whitespaces are removed. No truncation is applied because none of the concatenated data exceeds the maximum token length of the pretrained language models.

For example, suppose that a masked sentence is given as *"You'll see this ______ in the bottom right corner of the screen."* and is filled with each filler text "*option*." Assume that the sentence that precedes it is *"1. Open your iPhone's Phone app. (...),"* and the sentence that follows it *"3. Tap ."* The concatenated input looks like *"Open your iPhone's Phone app. You'll see this option in the bottom right corner of the screen. Tap"*.

Training We fine-tune the model on the training data with a batch size of 32 and 20 of epochs. We use the AdamW optimizer (Loshchilov and Hutter, 2019) and a cosine scheduler with warm-up steps for the initial 5% of the total steps at a learning rate of $1e-5$.

We also create a DeBERTa-based ensemble model using ten models trained on different seeds to obtain the final submission result.

4.2 Official results

For each model, we select the checkpoint with the best accuracy on the validation data as the fine-tuned model.

As mentioned in Section 4.1, we ensemble 10 finetuned DeBERTa-v3-large models by *max-voting* their outputs to further improve our DeBERTa-based model.

Table 1 shows the official results of our DeBERTa-based ensemble model compared with the other participants' systems.

4.3 Analysis

4.3.1 Comparison of the results on the development set

Table 2 presents the results of the validation data using three different pretrained language models without an ensemble [5]

As shown in Table 2, DeBERTa-large outperforms the baseline models by a decent margin.

Furthermore, Figure 2 shows the detailed confusion matrix of the finetuned DeBERTa-large model for the development set. As shown in Figure 2, DeBERTa-large distinguishes between plausible and implausible sequences reasonably well but has difficulty identifying neutral sequences. In our ex-

[5] While the official evaluation only measures the accuracy of the system, Table 2 lists the precision, recall and f1 score for analysis in detail.

Model	Parameters	Metric			
		Accuracy	**Precision**	**Recall**	**F1-score**
BERT-base	110M	45.36	43.32	43.06	42.67
BERT-large	340M	48.00	43.06	43.32	38.89
RoBERTa-base	125M	51.48	50.20	48.22	46.32
RoBERTa-large	355M	53.12	49.39	49.50	48.28
DeBERTa-v3-base	86M	55.92	49.19	50.84	48.36
DeBERTa-v3-large	304M	**59.88**	**52.92**	**54.20**	**50.81**

Table 2: Comparative results of BERT, RoBERTa, and DeBERTa on the validation dataset. The precision, recall, and F1-score are calculated via macro-average.

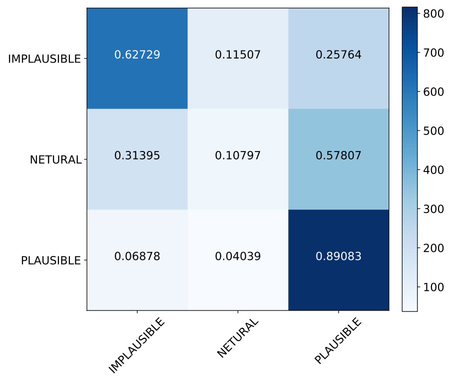


Figure 2: Illustration of the confusion matrix of our DeBERTa-based model on the validation dataset

periment, a similar tendency was also observed on BERT- and RoBERTa-based models.

Overall, from these results, we hypothesize that the ability of a pretrained language model to contextualize representations for the plausibility detection task is more strongly affected by its model structures, such as attention design or refining positional embeddings, rather than its parameter size.

4.3.2 Visualization of [CLS] representation

Figure 3 shows the visualization of the representations of [CLS] tokens using T-SNE (van der Maaten and Hinton, 2008) to compare the abilities of pretrained language models to distinguish between plausible, neutral and implausible sequences. As shown in Figure 3, the context representation distributions of the two DeBERTa models are more coherent and distinctive that those of BERT and RoBERTa. In contrast, no significant differences are observed between the BERT and RoBERTa models.

5 Conclusion

In this study, we compare BERT, RoBERTa and DeBERTa in SemEval-2022 Task 7 Subtask A: Multi-Class Classification. The results show that DeBERTa presents the best performances with improved distinguishable representations. We assume that the substantial changes made to the model structure of DeBERTa, such as disentangled attention, enhanced mask decoder, and RTD objective, would give DeBERTa a significant advantage in the addressed task.

Our final submission, based on an ensemble model comprising 10 fine-tuned DeBERTa-based models, achieved an accuracy of 61.4% on the test data. Our proposed model is ranked fifth out of eight models of teams who reported their papers.

In future studies, it would be worthwhile to explore other pretrained language models, such as ELECTRA, or models resulting from task-specific pretraining.

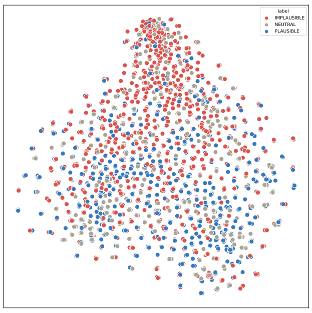
Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2021-0-02068, Artificial Intelligence Innovation Hub)

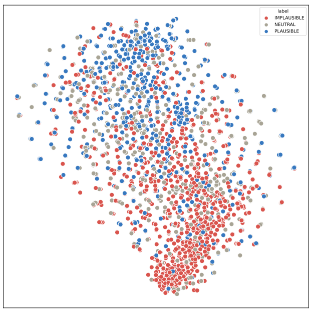
References

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

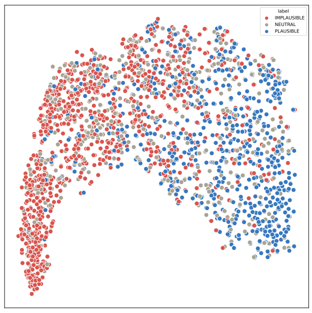
Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Tech-*



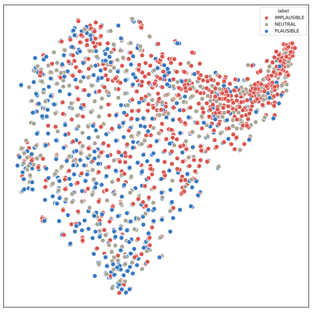
(a) BERT-base



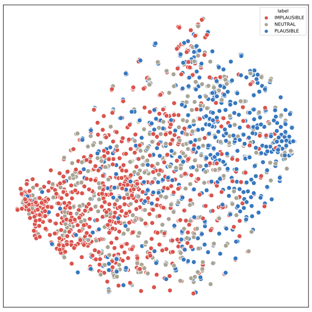
(b) RoBERTa-base



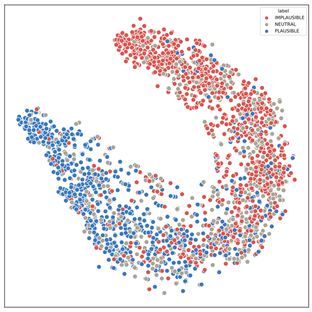
(c) DeBERTa-base



(d) BERT-large



(e) RoBERTa-large



(f) DeBERTa-large

Figure 3: Visualization of the representations of the [CLS] token using T-SNE among BERT, RoBERTa, DeBERTa.

nologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021b. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Michael Roth, Talita Anthonio, and Anna Sauer. 2022. SemEval-2022 Task 7: Identifying plausible clarifications of implicit and underspecified phrases in instructional texts. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

A Hyperparameters

Table 3 shows the setup of hyper-parameters of our models.

epochs	20
total batch size	32
accumulation steps	4
learning rate	1e-5
optimizer	AdamW
warm-up proportion	0.05
weight decay	0.01

Table 3: Hyperparameters