Holistic Sentence Embeddings for Better Out-of-Distribution Detection

Sishuo Chen[1], Xiaohan Bi[1], Rundong Gao[1], Xu Sun[2]
[1]Center for Data Science, Peking University
[2]MOE Key Laboratory of Computational Linguistics, School of Computer Science,
Peking University
{chensishuo,xusun}@pku.edu.cn {bxh,gaord20}@stu.pku.edu.cn

Abstract

Detecting out-of-distribution (OOD) instances is significant for the safe deployment of NLP models. Among recent textual OOD detection works based on pretrained language models (PLMs), distance-based methods have shown superior performance. However, they estimate sample distance scores in the last-layer CLS embedding space and thus do not make full use of linguistic information underlying in PLMs. To address the issue, we propose to boost OOD detection by deriving more holistic sentence embeddings. On the basis of the observations that token averaging and layer combination contribute to improving OOD detection, we propose a simple embedding approach named *Avg-Avg*, which averages all token representations from each intermediate layer as the sentence embedding and significantly surpasses the state-of-the-art on a comprehensive suite of benchmarks by a 9.33% FAR95 margin. Furthermore, our analysis demonstrates that it indeed helps preserve general linguistic knowledge in fine-tuned PLMs and substantially benefits detecting background shifts. The simple yet effective embedding method can be applied to fine-tuned PLMs with negligible extra costs, providing a free gain in OOD detection. Our code is available at https://github.com/lancopku/Avg-Avg.

1 Introduction

Pretrained language models have achieved remarkable performance on various NLP tasks under the assumption that the train and test samples are drawn from the same distribution (Wang et al., 2019). However, in real-life applications such as dialogue systems and clinical text processing, it is inevitable for models to make predictions on out-of-distribution (OOD) samples, which may result in fatally unreasonable predictions (Hendrycks et al., 2020). Therefore, it is crucial for fine-tuned PLMs to automatically detect OOD inputs.

Among recent works on textual OOD detection, distance-based methods have received much attention due to their superior performance (Podolskiy et al., 2021; Zhou et al., 2021). They calculate the sample distance to the training-data distribution as the uncertainty measure for OOD detection. In these approaches, the distance scores are usually calculated in the space of the last-layer CLS vectors (i.e., the inputs to the classification head) produced by fine-tuned PLMs. As known, the CLS embedding space is optimized for the in-distribution classification task during fine-tuning, thus not necessarily optimal for OOD detection.

In this paper, we investigate how to derive sentence embeddings suitable for OOD detection from fine-tuned PLMs. Motivated by the token averaging and layer combination techniques proposed to enhance unsupervised sentence embeddings (Su et al., 2021; Huang et al., 2021b), we apply them to OOD detection and make two intriguing empirical findings: (1) averaging all token representations outperforms the standard practice only using the CLS vector; (2) combining token representations from all intermediate layers brings further improvements. These observations lead to an extremely simple yet effective pooling technique: averaging all token representations in each intermediate layer as the sentence embedding for OOD detection.

We name the all-layer-all-token pooling technique *Avg-Avg* and demonstrate that it consistently uplifts the OOD detection performance of BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) models on a comprehensive suite of textual OOD detection benchmarks. Further investigations into the rationales behind the improvement show that *Avg-Avg* effectively helps reserve general linguistic information in the feature space and benefits detecting background shifts. In summary, our proposal serves as a plug-and-play post-processing technique to improve the capability of fine-tuned

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 6705 - 6715
December 7-11, 2022 ©2022 Association for Computational Linguistics

PLMs to detect OOD instances and reveals that it is a promising direction to boost textual OOD detection via deriving more holistic representations.

2 *Avg-Avg*: Holistic Sentence Embedding for Better OOD Detection

2.1 Preliminaries

Modern pretrained language models have been developed based on the Transformer (Vaswani et al., 2017) architecture. Given a sentence $S = \{t_1, t_2, \ldots, t_n\}$ as the input, an L-layer Transformer-based PLM yields a series of hidden vectors $\mathbb{H} = \{H_0, H_1, \ldots, H_L\}$, where $H_i = \left[h_i^1, h_i^2, \ldots, h_i^n\right] (0 < i \leq L)$ are the embedding vectors for each token in S in the i-th Transformer layer and H_0 denotes the static token embeddings.

2.2 Methodology

In the pretraining-finetuning paradigm, the CLS token is usually put at the beginning of S, and the corresponding vector produced by the last Transformer layer h_L^1 is fed into the classification head for fine-tuning. In existing works, the CLS vector h_L^1 is regarded as the sentence representation, and OOD detection is conducted in the corresponding embedding space (Podolskiy et al., 2021; Zhou et al., 2021). Such a practice does not fully exploit linguistic information contained in $\mathbb{H}$. Consequently, we resort to two pooling strategies to derive more holistic sentence representations:

- *Intra-Layer Token Averaging*: For the i-th layer, we average hidden vectors for all tokens as the pooled representation P_i, i.e., $P_i = \frac{1}{n}\sum_{j=1}^{n} h_i^j$, to replace the default $P_i = h_i^1$.
- *Inter-Layer Combination*: For given intermediate pooled representations $P_1, P_2, \ldots, P_L$, we perform layer combination to obtain the final pooled sentence representation P for OOD detection: $P = \frac{1}{|M|}\sum_i P_i, i \in M$, where $M \subseteq \{1, 2, \ldots, L\}$ denotes the subset of intermediate layers for combination.

In our embedding approach *Avg-Avg*, token averaging is performed for intra-layer pooling; all layers are chosen for layer combination, in other words, $M = \{1, 2, \ldots, L\}$. Table 1 shows the rationality of our choice: for a RoBERTa-based model fine-tuned on the SST-2 sentiment analysis dataset, *Avg-Avg* significantly outperforms other pooling strategies for detecting 20 Newsgroup (20NG) samples as OOD data, including the default last-layer CLS pooling and the *first-last-avg* pooling used for unsupervised sentence embedding (Su et al., 2021).

Intra-Layer	Inter-Layer	AUROC%	Remark
CLS	L12	90.48	Default
Avg	L12	93.17	-
CLS	All Layers	94.34	-
Avg	L1+L12	98.65	*first-last-avg*
Avg	All Layers	99.99	*Avg-Avg* (Ours)

Table 1: The performance of different pooling strategies on the SST-2 v.s. 20NG benchmark. Mahalanobis distance (Lee et al., 2018) is the OOD detection method. Avg denotes token average pooling and L12 denotes the 12th layer (the last) of the RoBERTa model. These results are exploratory and the superiority of *Avg-Avg* will be further confirmed by following experiments.

3 Experiments

3.1 Experimental Setup

Benchmarks Following Zhou et al. (2021), we choose four datasets corresponding to three tasks as the in-distribution (ID) datasets: SST-2 (Socher et al., 2013) and IMDB (Maas et al., 2011) for sentiment analysis, TREC-10 (Li and Roth, 2002) for question classification, and 20 Newsgroups (Lang, 1995) for topic classification. Among the four, any pair of datasets coming from different tasks is regarded as an ID-OOD. Besides, we use four additional datasets as OOD test data for each ID dataset: WMT-16 (Bojar et al., 2016), Multi30k (Elliott et al., 2016), RTE (Dagan et al., 2005), and SNLI (Bowman et al., 2015). More details of these datasets can be found in Appendix A.1.

Model Configuration We build text classifiers by fine-tuning the RoBERTa-base model (Liu et al., 2019) (110M parameters) in main experiments. Our implementation is based on Hugging Face's Transformers library (Wolf et al., 2020). Please refer to Appendix B for more details.

Evaluation Protocol For OOD detection performance, we report AUROC and FAR95 following Zhou et al. (2021). Higher AUROC and lower FAR95 values indicate better OOD detection performance (specific definitions in Appendix C).

Baselines for Comparison We reimplement a series of existing OOD detection methods for comparison: MSP (Hendrycks and Gimpel, 2017), Energy Score (Liu et al., 2020), LOF (Lin and Xu, 2019), Mahalanobis distance (MD for short) (Lee et al.,

AUROC↑ / FAR95↓	SST-2	IMDB	TREC-10	20NG	Avg.
Baselines					
MSP (Hendrycks and Gimpel, 2017)	88.10 / 70.00	96.36 / 22.43	94.28 / 24.15	87.96 / 49.95	91.68 / 41.63
LOF (Lin and Xu, 2019)	78.63 / 66.29	89.36 / 54.19	96.37 / 22.11	92.56 / 36.81	89.23 / 44.85
Energy (Liu et al., 2020)	87.47 / 72.43	95.83 / 23.95	95.50 / 19.68	90.37 / 32.31	92.29 / 37.09
MD Baseline (Podolskiy et al., 2021)	91.88 / 48.82	99.19 / 2.86	99.12 / 2.25	96.75 / 15.75	96.74 / 17.42
MD + $\mathcal{L}_{\text{scl}}$ (Zhou et al., 2021)	92.16 / 49.04	98.86 / 4.08	98.59 / 4.96	95.47 / 21.56	96.27 / 19.91
MD + $\mathcal{L}_{\text{margin}}$ (Zhou et al., 2021)	95.35 / 29.43	**99.93 / 0.15**	99.36 / 1.72	96.51 / 18.51	97.79 / 12.45
Ours					
token=AVG, layer=L12	93.14 / 41.24	99.67 / 1.09	98.29 / 2.88	97.07 / 14.07	97.04 / 14.82
token=CLS, layer=ALL	93.93 / 34.73	99.53 / 1.35	99.33 / 1.29	96.73 / 15.67	97.38 / 13.26
token=AVG, layer=ALL (*Avg-Avg*)	**97.75 / 10.67**	**99.87 / 0.21**	**99.66 / 0.23**	**99.70 / 1.35**	**99.25 / 3.12**

Table 2: The AUROC / FAR95 results of previous OOD detection methods and ours on four benchmarks. ↑ indicates larger is better and ↓ indicates lower is better. For each ID dataset, we report the macro average of AUROC / FAR95 values on all corresponding OOD datasets. All values are percentages averaged over five times with different random seeds, and the best results are highlighted in **bold**. $\mathcal{L}_{\text{scl}}$ and $\mathcal{L}_{\text{margin}}$ denote the contrastive and margin-based auxiliary targets proposed by Zhou et al. (2021), respectively.

AUROC ↑	SST-2	IMDB	TREC-10	20NG	Avg.
ALBERT-base	95.43 (+9.62)	99.51 (+1.10)	98.82 (+0.98)	99.55 (+7.02)	98.33 (+4.68)
DistillRoBERTa-base	97.80 (+7.17)	99.87 (+1.16)	99.17 (+0.93)	99.94 (+4.15)	99.20 (+3.36)
BERT-base-uncased	95.78 (+2.22)	99.51 (+0.45)	98.81 (-0.45)	99.79 (+1.69)	98.45 (+0.95)
RoBERTa-base	97.75 (+5.87)	99.87 (+0.68)	99.66 (+0.54)	99.70 (+2.35)	99.25 (+2.51)
RoBERTa-large	97.49 (+5.13)	99.94 (+1.24)	99.42 (+0.16)	99.97 (+2.88)	99.21 (+2.38)

Table 3: The improvements brought by *Avg-Avg* compared to the MD baseline (Podolskiy et al., 2021) for different PLMs. AUROC values are reported (the number in the bracket is the improvement).

2018; Podolskiy et al., 2021), and MD combined with contrastive targets (Zhou et al., 2021). See Appendix D for the introduction and implementation details of these baseline methods.

3.2 Overall Results

Table 2 gives main results. Except the contrastive-based tuning method (Zhou et al., 2021), all methods use the same model vanilla fine-tuned on the ID training set. Our methods use the Mahalanobis distance to obtain OOD scores, following Podolskiy et al. (2021) (the only difference lies in the embedding space). We find that compared to the baseline calculating MD in the last-layer CLS embedding space, both token averaging and layer combination bring improvements on almost all benchmarks. When the two techniques are combined, namely *Avg-Avg* is applied, the performance continues to grow and exceeds the previous state-of-the-art (Zhou et al., 2021) that needs extra contrastive targets in the fine-tuning stage, by a considerable margin of 9.33% FAR95 averaged on four benchmarks. Further experiments on other PLM backbones also substantiate the enhancement brought by our method, supported by the results in Table 3.

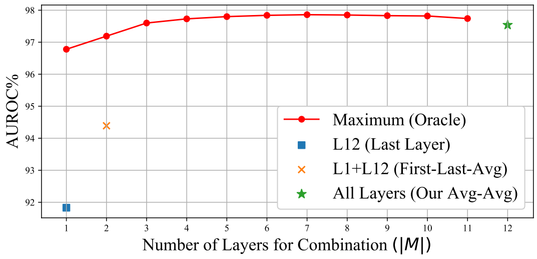


Figure 1: Maximum AUROC (averaged over 6 OOD datasets) values corresponding to sentence embeddings from a RoBERTa model fine-tuned on SST-2 with different numbers of combining layers. The maximum values are results searched on the test data. Token averaging is performed for intra-layer pooling.

3.3 Analysis

The Impact of Layer Choice To verify the rationality of choosing all intermediate layers for inter-layer combination, we show the maximum AUROC values corresponding to different numbers of intermediate layers to derive sentence embeddings in Figure 1. As the number of layers grows, the AUROC metric first increases and then remains relatively stable when more than four layers are chosen. Notably, the peak appears when 7 layers are combined, only 0.3% higher than our *Avg-Avg*. Since searching for the best combination of layers is infeasible due to the unavailability of OOD data, using all layers is a sensible choice.

Probing Analysis Given that intermediate layers of PLMs contain a rich hierarchy of linguistic information (Jawahar et al., 2019), a plausible explanation of the performance lift is that *Avg-Avg* leads to an embedding space containing more gen-

Intra-Layer	Inter-Layer	Surface		Syntactic			Semantic					Average
		SentLen	WC	TreeDepth	TopConst	BShift	Tense	SubjNum	ObjNum	SOMO	CoordInv	
CLS	L12	48.63	10.84	26.20	49.31	74.92	83.99	76.68	72.77	57.77	62.17	56.33
CLS	AVG	67.74	5.55	27.86	49.53	78.88	85.43	79.75	76.32	57.67	63.45	59.22
AVG	L12	61.69	12.22	30.20	51.89	79.21	85.12	78.55	76.68	59.71	62.62	59.79
AVG	AVG	**91.31**	**17.42**	**41.24**	**74.38**	**88.42**	**88.46**	**84.54**	**84.54**	**63.77**	**69.14**	**70.32**

Table 4: Probing task performance for representations corresponding to different pooling strategies. All values are percentages averaged over five RoBERTa models fine-tuned with different random seeds.

Dominant Shift	ID	OOD	AUROC	
			Baseline	Ours
Background	SST-2	IMDB	69.86	97.57 (+27.70)
		CR	75.46	82.26 (+6.80)
Semantic	News Top-5	News Rest	83.41	83.83 (+0.42)
	CLINC	CLINC$_{\text{OOD}}$	97.58	97.88 (+0.30)

Table 5: Performance (AUROC) on different kinds of distribution shifts, corresponding to the MD baseline and our proposed *Avg-Avg*. All values are percentages areraged over five different random seeds.

Method	SST-2	IMDB	TREC	20NG	Avg.
SBERT	21.02	2.40	19.03	3.63	11.52
SBERT$_{\text{ft}}$	35.02	5.65	0.78	12.13	13.40
unsup-SimCSE	42.02	9.51	62.68	**0.00**	28.55
unsup-SimCSE$_{\text{ft}}$	44.43	2.30	2.69	19.56	17.25
sup-SimCSE	32.37	3.16	35.42	0.03	17.75
sup-SimCSE$_{\text{ft}}$	38.91	0.59	1.30	15.14	13.99
vanilla+*last-cls*	48.82	2.86	2.25	15.75	17.42
SBERT$_{\text{ft}}$+*Avg-Avg*	9.70	0.33	0.11	1.67	3.03
unsup-SimCSE$_{\text{ft}}$+*Avg-Avg*	11.36	0.42	0.40	1.13	3.33
sup-SimCSE$_{\text{ft}}$+*Avg-Avg*	**7.68**	**0.18**	**0.14**	0.77	**2.19**
vanilla+*Avg-Avg*	10.67	0.21	0.23	1.35	3.12

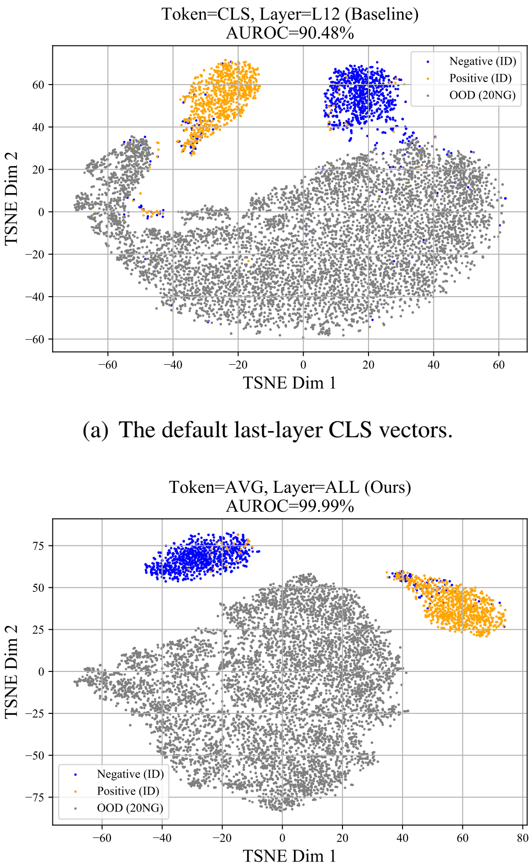
Table 6: The OOD detection performance (FAR95) of different embedding approaches (lower FAR95 values indicate better detection performance). The "ft" subscript denotes that the embedding model is fine-tuned on the in-distribution data for classification.

eral linguistic information, where ID and OOD data are more sharply separated. To verify this, we evaluate the sentence embeddings produced by the RoBERTa model fine-tuned on SST-2 corresponding to different pooling strategies on the probing tasks proposed by Conneau et al. (2018) (details in Appendix A.3). As shown in Table 4, our proposed method consistently raises the probing accuracies of surface, syntactic, and semantic level probing tasks, suggesting that we obtain more holistic embeddings by integrating intermediate hidden states.

Detecting Different Kinds of Shifts OOD texts can be categorized by whether they exhibit a background shift or a semantic shift (Arora et al., 2021). In previous main experiments, ID and OOD data come from different tasks and both kinds of shifts exist. To explore the source of the performance growth, we conduct ablation experiments by evaluating our method in settings where background or semantic shifts dominate. For the semantic shift setting, we use the News Category (Misra, 2018) and CLINC (Larson et al., 2019a) datasets (ID and OOD parts share the same background distribution, but belong to different classes); for the background shift setting, we regard SST-2 as ID and IMDB, Customer Reviews (CR for short) (Hu and Liu, 2004) as OOD (they all belong to the sentiment analysis task but differ in background features, e.g, the length and style). Refer to Appendix A.2 for dataset details. As shown in Table 5, our method drastically strengthens the capability of detecting background shifts; in contrast, it only slightly improves detecting semantic shifts, which indicates that the performance gain mainly comes from the task-agnostic general linguistic information in the holistic embeddings obtained by our pooling technique, in line with the probing analysis.

3.4 Comparison with Universal Sentence Embedding Approaches

Here we further show the advantage of our method *Avg-Avg* over two representative universal sentence embedding approaches, SentenceBERT (SBERT) (Reimers and Gurevych, 2019) and SimCSE (Gao et al., 2021) on OOD detection. For SBERT, we test the model trained on NLI (natural language inference) data (last-layer mean pooling is adopted as recommended in the original work); for SimCSE, we test the unsupervised model and the supervised model trained on NLI data (last-layer CLS pooling is adopted). The backbone model is RoBERTa-base in all methods. We also fine-tune the pre-trained models on the ID data and use the default pooling ways and our *Avg-Avg* to obtain embeddings from fine-tuned models for thorough comparison on OOD detection. As results in Table 6, when *Avg-Avg* is applied, it brings consistent improvements and beats both pre-trained and fine-tuned sentence embedding models using the default pooling way. These results corroborate the advantage of *Avg-Avg* as a specialized embedding method for OOD detection.



(a) The default last-layer CLS vectors.

(b) Our *Avg-Avg* embeddings.

Figure 2: Visualization of the representations obtained for positive, negative instances in SST-2 and OOD ones (20 Newsgroups).

3.5 Embedding Visualization

To demonstrate the influence of the studied pooling strategies on the embedding space, we fine-tune the RoBERTa-base model on SST-2 and visualize instance embeddings corresponding to different pooling strategies from the SST-2 test set (ID) and an OOD test set (20 Newsgroups) using t-SNE (Van der Maaten and Hinton, 2008). As plotted in Figure 2, in the representation space produced by *Avg-Avg* (Figure 2(b)) where is almost no overlap between ID and OOD instances, ID and OOD samples are more sharply separated than in the space of default last-layer CLS embeddings (Figure 2(a)). This further supports our claim that *Avg-Avg* is better suited for OOD detection.

4 Related Works

4.1 Textual OOD Detection

OOD detection aims to detect abnormalities that come from a different distribution from the training set (Hendrycks and Gimpel, 2017). Compared with the widely studied OOD image detection problem (Liang et al., 2018; Lee et al., 2018; Liu et al., 2020; Huang et al., 2021a; Fort et al., 2021; Yang et al., 2021), textual OOD detection remains underexplored. Hendrycks et al. (2020) first showed that pretrained Transformers improved OOD detection using the maximum softmax probability (Hendrycks and Gimpel, 2017). Afterward, Podolskiy et al. (2021) used the Mahalanobis distance approach (Lee et al., 2018) for PLM-based OOD detection and achieved superior performance. Following this, Zhou et al. (2021) further raised the performance by utilizing contrastive auxiliary targets in the fine-tuning stage.

4.2 Unsupervised Sentence Embedding

Unsupervised sentence embedding is a well-established area (Kiros et al., 2015; Pagliardini et al., 2017; Li et al., 2020; Reimers and Gurevych, 2019; Gao et al., 2021). Relevant to our work, Su et al. (2021) and Huang et al. (2021b) proposed to obtain better sentence embeddings via averaging token representations, layer combination, and a whitening operation. It is noteworthy that these embedding approaches are mainly studied for sentence matching and retrieval tasks. As far as we know, we are the first to study novel embedding ways to replace the default last-layer CLS pooling for boosting textual OOD detection.

5 Conclusion

In this work, we focus on how to derive sentence embeddings suitable for OOD detection from fine-tuned PLMs. Specifically, we introduce token averaging and layer combination to derive more holistic representations and substantially improve the capability of PLMs to detect OOD inputs. Moreover, our analysis shows that our approach helps preserve general linguistic information and benefits detecting background shifts. Overall, our work points out a new perspective that textual OOD detection can be enhanced by obtaining high-quality sentence embeddings, and we hope to extend this idea to training-time methods in future work.

Limitations

The contemporary solution *Avg-Avg* is primarily motivated by empirical observations and its effectiveness is confirmed by extensive experiments on different PLMs and benchmarks. Currently, its superiority lacks strict theoretical justifications and there is still a small performance gap between our method and the ideal upper bound as shown in Figure 1. In future work, we plan to explore theory-guaranteed embedding approaches to further boost the OOD detection ability of PLMs.

Ethical Considerations

Our work presents an efficient embedding method to enhance the OOD detection ability of NLP models. We believe that our proposal will help reduce security risks resulting from OOD inputs to NLP models deployed in the open-world environment. In addition, all experiments in this work are conducted on open datasets and our code is publicly available. While we do not expect any direct negative consequences to the work, we hope to continue exploring more efficient and robust sentence embedding approaches for textual OOD detection in future work.

Acknowledgement

We sincerely thank all the anonymous reviewers for their constructive comments and suggestions. This work is supported by Natural Science Foundation of China (NSFC) No. 62176002. Xu Sun is the corresponding author of this paper.

References

Udit Arora, William Huang, and He He. 2021. Types of out-of-distribution texts and how to detect them. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 10687–10701. Association for Computational Linguistics.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, et al. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 131–198.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. 2000. Lof: identifying density-based local outliers. In *ACM sigmod record*, volume 29, pages 93–104. ACM.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single \$&!#* vector: Probing sentence embeddings for linguistic properties. 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 2126–2136. Association for Computational Linguistics.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

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

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30k: Multilingual english-german image descriptions. *arXiv preprint arXiv:1605.00459*.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. 2021. Exploring the limits of out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34:7068–7081.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. *arXiv preprint arXiv:2004.06100*.

Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. 2020. Generalized ODIN: detecting out-of-distribution image without learning from out-of-distribution data. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10948–10957. IEEE.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth*

ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA, August 22-25, 2004, pages 168–177. ACM.

Haiwen Huang, Zhihan Li, Lulu Wang, Sishuo Chen, Xinyu Zhou, and Bin Dong. 2021a. Feature space singularity for out-of-distribution detection. In *Proceedings of the Workshop on Artificial Intelligence Safety 2021 (SafeAI 2021) co-located with the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2021), Virtual, February 8, 2021*, volume 2808 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Junjie Huang, Duyu Tang, Wanjun Zhong, Shuai Lu, Linjun Shou, Ming Gong, Daxin Jiang, and Nan Duan. 2021b. WhiteningBERT: An easy unsupervised sentence embedding approach. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 238–244, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

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

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning, Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, California, USA, July 9-12, 1995*, pages 331–339. Morgan Kaufmann.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019a. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316, Hong Kong, China. Association for Computational Linguistics.

Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, et al. 2019b. An evaluation dataset for intent classification and out-of-scope prediction. *arXiv preprint arXiv:1909.02027*.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 7167–7177.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9119–9130. Association for Computational Linguistics.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Shiyu Liang, Yixuan Li, and R. Srikant. 2018. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Ting-En Lin and Hua Xu. 2019. Deep unknown intent detection with margin loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5491–5496, Florence, Italy. Association for Computational Linguistics.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. 2020. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, volume 33, pages 21464–21475. Curran Associates, Inc.

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. *arXiv preprint arXiv:1907.11692*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 142–150. The Association for Computer Linguistics.

Rishabh Misra. 2018. News category dataset.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2017. Unsupervised learning of sentence embeddings using compositional n-gram features. *arXiv preprint arXiv:1703.02507*.

Alexander Podolskiy, Dmitry Lipin, Andrey Bout, Ekaterina Artemova, and Irina Piontkovskaya. 2021. Revisiting mahalanobis distance for transformer-based out-of-domain detection. In *Proceedings of the AAAI Confer-*

ence on Artificial Intelligence, volume 35, pages 13675–13682.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

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, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1631–1642. ACL.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. 2021. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*.

Wenxuan Zhou, Fangyu Liu, and Muhao Chen. 2021. Contrastive out-of-distribution detection for pretrained transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 1100–1111. Association for Computational Linguistics.

Dataset	**#Classes**	**#Train**	**#Dev**	**#Test**	**L**
SST-2	2	6920	872	1821	19
IMDB	2	23000	2000	25000	230
TREC-10	6	4907	545	500	10
20 Newsgroups	20	10182	1132	7532	289

Table 7: Statistics of in-distribution text datasets. **L** denotes the average length of samples.

Dataset	**#Test**	**L**
Multi30k	1014	13
WMT16	2000	22
RTE	3000	48
SNLI	2000	21

Table 8: Statistics of out-of-distribution text datasets. **L** denotes the average length of samples.

A Dataset Statistics and Introduction

A.1 Datasets Used in Main Experiments

The statistics of in-distribution (ID) and out-of-distribution (OOD) textual datasets in main experiments (Section 3.1 and 3.2), including the number of classes, the dataset size, and the average length of samples, are given in Table 7 and 8, respectively. Here is a brief introduction to these datasets: Multi30k (Elliott et al., 2016) and WMT16 (Bojar et al., 2016) are parts of the English side data of English-German machine translation datasets; RTE (Dagan et al., 2005) and SNLI (Bowman et al., 2015) are the concatenations of the precise and respective hypotheses from NLI datasets.

A.2 Datasets Used In the Distribution Shift Analysis

Arora et al. (2021) categorized the distribution shifts in natural language data into two main types: background shifts and semantic shifts. We follow their division and study OOD detection performance under the setting where either kind of shift dominates in Section 3.3. The statistics of extra datasets used in the distribution shift analysis are given in Table 9. Here is a brief introduction to these datasets.

Background Shift Setting. Background shifts refer to the shift of background features (e.g., formality) that do not depend on the label. We consider domain shifts in sentiment classification datasets.

Dataset	# Classes	# Train	# Dev	# Test	L
Customer Reviews (OOD)	2	-	-	1000	20
News Top-5 (ID)	5	68859	8617	8684	30
News Rest (OOD)	36	-	-	11402	29
CLINC (ID)	150	15000	3000	4500	8
CLINC$_{\text{OOD}}$ (OOD)	-	-	-	1000	9

Table 9: Statistics of extra datasets introduced for the distribution shift analysis. **L** denotes the average length of each sample.

SST-2 contains short movie reviews by the audience, while IMDB contains longer and more professional movie reviews. Customer Reviews (Hu and Liu, 2004) contains reviews for different kinds of commercial products on the web, representing a domain shift from SST-2. So the IMDB and Customer Reviews test data can be regarded as OOD samples for the model fine-tuned on SST-2.

Semantic Shift Setting. In this setting, OOD data are from the same task as ID data and share similar background characteristics, but belong to classes unseen during training. We use the News Category (Misra, 2018) and the CLINC (Larson et al., 2019b) datasets to create two ID/OOD pairs under the setting. Following Arora et al. (2021), we use the data from the five most frequent classes of the News Category as ID (News Top-5) and the data from the remaining 36 classes as OOD (News Rest). In the CLINC dataset for intent classification, there is a 150-class ID subset and an OOD test set CLINC$_{\text{OOD}}$ composed of utterances belonging to actions not supported by existing ID intents.

A.3 Probing Benchmarks

To probe the linguistic information contained in sentence embeddings, we use the probing tasks proposed by Conneau et al. (2018), which are grouped into three categories. For surface information, we use *SentLen* (sentence length) and *WC* (the presence of words); for syntactic information, we use *BShift* (sensitivity to word order), *TreeDepth* (the depth of the syntactic tree), and *TopConst* (the sequence of top-level constituents); for semantic information, we use *Tense* (tense), *SubjNum* and *ObjNum* (the subject/direct object number in the main clause), *SOMO* (the sensitivity to random replacement of a noun/verb), and *CoordInv* (the random swapping of coordinated clausal conjuncts). Each probing dataset contains 100k training samples, 10k validation samples, and 10k test samples. We use the SentEval toolkit (Conneau and Kiela, 2018) along with the recommended hyperparameter space to search for the best probing classifier according to the validation accuracy and report test accuracies.

B Details of Pretrained Language Model Fine-tuning

B.1 Vanilla Fine-tuning

We use the RoBERTa-base pretrained model (Liu et al., 2019) as the backbone to build text classifiers by fine-tuning it on the ID training data. We use a batch size of 16 and fine-tune the model for 5 epochs. The model is optimized with the Adam (Kingma and Ba, 2015) optimizer using a learning rate of 2e-5. We evaluate the model on the ID development set after every epoch and choose the best checkpoint as the final model. The setting is the same for other pretrained Transformers studied in the paper (RoBERTa-large, BERT-base-uncased, DistilRoBERTa-base, and ALBERT-base). DistilRoBERTa (Sanh et al., 2019) is a light distilled RoBERTa and ALBERT (Lan et al., 2019) is a lite BERT with factorized embedding parameterization and cross-layer parameter sharing.

B.2 Contrastive Auxiliary Targets

Zhou et al. (2021) introduced two alternatives of contrastive loss to boost textual OOD detection, i.e., the supervised contrastive loss and the margin-based contrastive loss. For a classification task with C classes, given a batch of training examples $\{x_i, y_i\}_{i=1}^{M}$, where x_i is the input and y_i is the label, the supervised contrastive loss term L_{scl} and the final optimization target $\mathcal{L}$ can be formulated as:

$$\begin{aligned}\mathcal{L}_{\text{scl}} &= \sum_{i=1}^{M} \frac{-1}{M|P(i)|} \sum_{p \in P(i)} \log \frac{e^{\boldsymbol{z}_i^{\top} \boldsymbol{z}_p / \tau}}{\sum_{a \in A(i)} e^{\boldsymbol{z}_i^{\top} \boldsymbol{z}_a / \tau}}, \\ \mathcal{L} &= \mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{scl}}\end{aligned} \quad (1)$$

where $A(i) = \{1, ..., M\} \smallsetminus \{i\}$ is the set of all anchor instances, $P(i) = \{p \in A(i) : y_i = y_p\}$ is the set of anchor instances from the same class as i, τ is a temperature hyper-parameter, $\boldsymbol{z}$ is the L2-normalized CLS embedding before the softmax layer, $\mathcal{L}_{\text{ce}}$ is the cross entropy loss, and λ is a positive coefficient. Following Zhou et al. (2021), we use $\tau = 0.3$ and $\lambda = 2$.

The margin-based loss term $\mathcal{L}_{\text{margin}}$ and the final optimization target $\mathcal{L}$ can be formulated as:

$$
\begin{aligned}
\mathcal{L}_{\text{pos}} &= \sum_{i=1}^{M} \frac{1}{|P(i)|} \sum_{p \in P(i)} \left\| \boldsymbol{h}_i - \boldsymbol{h}_p \right\|^2 , \\
\mathcal{L}_{\text{neg}} &= \sum_{i=1}^{M} \frac{1}{|N(i)|} \sum_{n \in N(i)} \left(\xi - \left\| \boldsymbol{h}_i - \boldsymbol{h}_n \right\|^2 \right)_+ , \\
\mathcal{L}_{\text{margin}} &= \frac{1}{dM} \left(\mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{neg}} \right) , \\
\xi &= \max_{i=1}^{M} \max_{p \in P(i)} \left\| \boldsymbol{h}_i - \boldsymbol{h}_p \right\|^2 , \\
\mathcal{L} &= \mathcal{L}_{\text{ce}} + \lambda \mathcal{L}_{\text{margin}} .
\end{aligned}
\tag{2}
$$

Here $N(i) = \{n \in A(i) : y_i \neq y_n\}$ is the set of anchor instances from other classes than y_i, $\boldsymbol{h} \in \mathbb{R}^d$ is the unnormalized CLS embedding before the softmax layer, ξ is the margin, d is the number of dimensions of $\boldsymbol{h}$, and λ is a positive coefficient. We use $\lambda = 2$ following Zhou et al. (2021).

Except for the loss term, we use the same hyperparameters for these two tuning methods as vanilla tuning. Table 10 gives test accuracies on four ID datasets for the RoBERTa models tuned with vanilla cross-entropy loss ($\mathcal{L}_{\text{ce}}$), supervised contrastive loss ($\mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{scl}}$), and margin-based contrastive loss ($\mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{margin}}$), where are not significant differences.

B.3 Hardware Requirements

All the experiments (fine-tuning and inference) in this paper are conducted on a single NVIDIA TITAN RTX GPU, except that the fine-tuning of the RoBERTa-large model needs 4 TITAN RTX GPUs.

C Definition of Evaluation Metrics for OOD Detection

For an input instance $\mathbf{x}$, the output of an OOD detector is the confidence score $S(\mathbf{x})$ (a higher confidence score). A higher confidence score indicates that the detector tends to regard $\mathbf{x}$ as a normal ID sample. In real applications, system users need to choose a threshold γ and treat the OOD detection module as a binary classifier:

$$
G(\mathbf{x}) = \begin{cases} \text{in}, & \text{if } S(\mathbf{x}) \geq \gamma \\ \text{out}, & \text{if } S(\mathbf{x}) < \gamma \end{cases} \tag{3}
$$

Following previous works (Hendrycks and Gimpel, 2017; Zhou et al., 2021), we use the following two threshold-free metrics for evaluation:

Loss	SST-2	IMDB	TREC	20NG
$\mathcal{L}_{\text{ce}}$	93.96	94.56	95.88	84.52
$\mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{scl}}$	94.23	94.53	96.36	84.65
$\mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{margin}}$	93.69	94.21	95.76	84.63

Table 10: Test accuracies on four ID datasets for RoBERTa-base models tuned with three different fine-tuning strategies. All values are percentages averaged over five times with different random seeds.

AUROC is short for the area under the receiver operating curve. It plots the true positive rate (TPR) against the false positive rate (FPR) and can be interpreted as the probability that the model ranks a random positive(ID) example more highly than a random negative (OOD) example. A higher AUROC indicates better OOD detection performance.

FAR95 is the probability for a negative example (OOD) to be mistakenly classified as positive (ID) when the TPR is 95%. A lower value indicates better detection performance.

D OOD Detection Baselines

D.1 MSP

MSP (Hendrycks and Gimpel, 2017) is a classical baseline using the maximum softmax probability in the prediction outputs of the classifier as the confidence score, i.e., $S(\mathbf{x}) = \max_{y \in \Upsilon} p_y(\mathbf{x})$.

D.2 Energy Score

Liu et al. (2020) proposed using free energy as a scoring function for OOD detection. For a classification problem with C classes, a multi-class classifier $f(\mathbf{x}) : \mathcal{X} \to \mathbb{R}^C$ can be interpreted from an energy-based perspective by viewing the logit output $f_{y_i}(\mathbf{x})$ corresponding to class y_i as an energy function $E(\mathbf{x}, y_i) = -f_{y_i}(\mathbf{x})$. The free energy function $E(\mathbf{x})$ for an input $\mathbf{x}$ is $E(\mathbf{x}) = \sum_{i=1}^{C} e^{f_{y_i}(\mathbf{x})}$, and $S(\mathbf{x}) = -E(\mathbf{x})$.

D.3 LOF

Lin and Xu (2019) proposed identifying unknown user intents by feeding feature vectors to the density-based novelty detection algorithm, local outlier factor (LOF) (Breunig et al., 2000). We use the last-layer CLS vectors produced by the fine-tuned RoBERTa models as the input and train a LOF model following the implementation details of Lin and Xu (2019) on the ID training set and use the local density output as $S(\mathbf{x})$.

AUROC↑ / FAR95↓	**SST-2**	**IMDB**	**TREC-10**	**20NG**	**Avg**
Maha Baseline	91.88 / 48.82	99.19 / 2.86	99.12 / 2.25	96.75 / 15.75	96.74 / 17.42
SE, token=CLS	94.68 / 26.09	**99.94 / 0.05**	**99.75 / 0.19**	99.47 / 2.62	98.45 / 7.24
SE, token=AVG	97.19 / 13.58	99.84 / 0.33	99.50 / 0.38	**99.82 / 0.83**	99.09 / 3.78
Avg-Avg (Ours)	**97.75 / 10.67**	99.87 / 0.21	99.66 / 0.23	99.70 / 1.35	**99.25 / 3.12**

Table 11: Comparison between score ensemble (SE) and *Avg-Avg*.The setting of backbone models and ID/OOD benchmarks is the same as that in Table 2.

D.4 Mahalanobis Distance

Mahalanobis distance score (Lee et al., 2018) is a representative distance-based OOD detection algorithm, which uses the sample distance to the nearest ID class in the embedding space as the OOD uncertainty measure. For a given feature extractor ψ, the Mahalanobis distance score is defined as: $S(\mathbf{x}) = \max_{c\in\Upsilon} -(\psi(\mathbf{x})-\mu_c)^T\Sigma^{-1}(\psi(\mathbf{x})-\mu_c)$, where $\psi(\mathbf{x})$ is the embedding vector of the input $\mathbf{x}$, μ_c is the class centroid for a class c, and Σ is the covariance matrix. The estimation of μ_c and Σ is defined as follows:

$$\begin{aligned}\mu_c &= \frac{1}{N_c}\sum_{\mathbf{x}\in\mathcal{D}_{\text{in}}^c}\psi(\mathbf{x}),\\ \Sigma &= \frac{1}{N}\sum_{c\in\Upsilon}\sum_{\mathbf{x}\in\mathcal{D}_{i\text{in}}^c}(\psi(\mathbf{x})-\mu_c)(\psi(\mathbf{x})-\mu_c)^T,\end{aligned} \tag{4}$$

where $\mathcal{D}_{\text{in}}^c = \{\mathbf{x} \mid (\mathbf{x},y)\in\mathcal{D}_{\text{in}}, y=c\}$ denotes the training samples belonging to the class c, N is the size of the training set, and N_c is the number of training instances belonging to class c.

E Comparison with Score Ensemble

Apart from the layer combination technique studied in the paper, there is another way to utilize intermediate representations for OOD detection: estimating the sample distance score in the embedding space of each intermediate layer and taking their sum as the final OOD score. For the Mahalanobis distance score, the final ensemble score $S(\mathbf{x})$ is defined as:

$$\begin{aligned}S^\ell(\mathbf{x}) &= \max_{c\in\Upsilon} -\left(\psi^\ell(\mathbf{x})-\mu_c^\ell\right)^T\Sigma_\ell^{-1}\left(\psi^\ell(\mathbf{x})-\mu_c^\ell\right),\\ S(\mathbf{x}) &= \sum_\ell \alpha_\ell S^\ell(\mathbf{x}),\end{aligned} \tag{5}$$

where $\psi^\ell(\mathbf{x})$ denotes the output features at the ℓth-layer of neural networks, and μ^ℓ and Σ_ℓ are the class mean and the covariance matrix, correspondingly. The layer-wise weighting hyperparameter is α_ℓ. In the original work (Lee et al., 2018), α_ℓ is tuned on a small validation set containing both ID and OOD for each OOD dataset, which is impractical in the setting of unsupervised OOD detection followed by recent works (OOD data is not available). Following Hsu et al. (2020), we use uniform weighting, i.e., $S(\mathbf{x}) = \sum_\ell S^\ell(\mathbf{x})$, in the baselines for comparison.

We compare the performance of SE (score ensemble) and *Avg-Avg* and show the results in Table 11. We observe that SE also brings consistent improvements over the baseline using only last-layer CLS vectors. Without token averaging, SE slightly surpasses *Avg-Avg* on IMDB and TREC-10, but underperforms *Avg-Avg* significantly on SST-2 and 20NG; when token averaging is performed, SE only beats *Avg-Avg* on 20NG but underperforms on other three benchmarks, especially remarkably on SST-2. In view of the average performance on the four benchmarks, we can get *Avg-Avg* > SE (AVG) > SE (CLS). Considering that the class mean μ^ℓ and the inverse of covariance matrix Σ_ℓ^{-1} need to be estimated and stored for each layer in SE, *Avg-Avg* is also more convenient for deployment. So compared with SE, *Avg-Avg* enjoys both simplicity and advantages in performance.