Text Classification with Negative Supervision

Sora Ohashi[†], **Junya Takayama**[†], **Tomoyuki Kajiwara**[‡], **Chenhui Chu**[‡], **Yuki Arase**[†]
[†] Graduete School of Information Science and Technology, Osaka University
[‡] Institute of Datability Science, Osaka University
[†] {ohashi.sora, takayama.junya, arase}@ist.osaka-u.ac.jp
[‡] {kajiwara, chu}@ids.osaka-u.ac.jp

Abstract

Advanced pre-trained models for text representation have achieved state-of-the-art performance on various text classification tasks. However, the discrepancy between the semantic similarity of texts and labelling standards affects classifiers, *i.e.* leading to lower performance in cases where classifiers should assign different labels to semantically similar texts. To address this problem, we propose a simple multitask learning model that uses *negative supervision*. Specifically, our model encourages texts with different labels to have distinct representations. Comprehensive experiments show that our model outperforms the state-of-the-art pre-trained model on both single- and multi-label classifications, sentence and document classifications, and classifications in three different languages.

1 Introduction

Text classification generally consists of two processes: an encoder that converts texts to numerical representations and a classifier that estimates hidden relations between the representations and class labels. The text representations are generated using N-gram statistics (Wang and Manning, 2012), word embeddings (Joulin et al., 2017; Wang et al., 2018), convolutional neural networks (Kalchbrenner et al., 2014; Zhang et al., 2015; Shen et al., 2018), and recurrent neural networks (Yang et al., 2016, 2018). Recently, powerful pre-trained models for text representations, *e.g.* Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019), have shown state-of-the-art performance on text classification tasks using only the simple classifier of a fully connected layer.

However, a problem occurs when a classification task is adversarial to text encoders. Encoders aim to represent the meanings of texts; hence, semantically similar texts tend to have closer representations. Meanwhile, a classifier should distinguish subtle differences that lead to different label assignments, although the texts are semantically similar. Table 1 shows an example of classification results using BERT for the MedWeb dataset (Wakamiya et al., 2017). This task requires the labelling of a disease contracted by the writer of a text. Although both texts in Table 1 refer to the common cold, only the second example implies that the writer had a cold. BERT mistakenly labelled both texts as `Cold`[1], likely owing to their semantic relatedness. When the standard of class label assignments disagrees with the semantic similarity, the classifier tends to be error-prone owing to the excessive effects of the semantic similarity.

Sentence	Label	BERT
A cold is a legit disease.	–	`Cold`
Oh my god! I caught a cold!	`Cold`	`Cold`

Table 1: Examples of BERT classification for labelling a disease contracted by a writer. Both sentences are about the common cold. Only the second example indicates that the writer had a cold. BERT misclassified the first sentence.

To address this problem, we propose utilizing negative examples, *i.e.* texts with different labels, to enable *negative supervision* of the encoder for generating distinct representations for each class. In this study, we design a simple multitask learning model that trains two models simultaneously with a shared text encoder. The first model learns an ordinary classification task (herein referred to as the main task). Meanwhile, the second model encourages representations with different class labels to be distinct (herein referred to as the auxiliary

[1] We use the `typewriter font` to indicate a class label throughout this paper.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 351–357

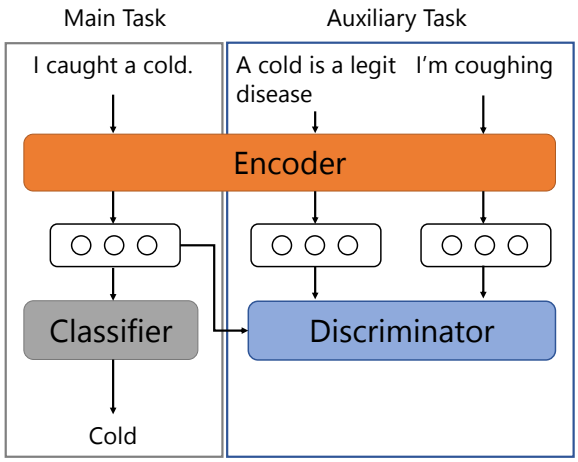

Figure 1: Our model consists of a classifier, discriminator, and shared text encoder. The main task learns classification, while the auxiliary task gives negative supervision to generate distinct representations for sentences with different labels.

task).

We empirically show the effectiveness of our model using the following standard benchmarks of five single-label and four multi-label classification datasets. This study has two main contributions.

- Our multi-tasking learning model consistently outperforms the state-of-the-art model in terms of both single and multi-label classifications, sentence and document classifications, and classifications in three languages.
- Our model is simple and easily applicable to any text encoders and classifiers.

2 Multitask Learning Framework

Figure 1 shows an overview of our multitask learning framework that consists of main and auxiliary tasks. Herein, we refer to the model for the main task as a classifier and the model for the auxiliary task as a discriminator. The overall loss function $\mathcal{L}$ sums the loss of the main task $\mathcal{L}_m$ and that of the auxiliary task $\mathcal{L}_a$:

$$\mathcal{L} = \mathcal{L}_m + \mathcal{L}_a.$$

The classifier and discriminator share and jointly optimize the text encoder, which encodes an input text into a d-dimensional vector $\boldsymbol{v} \in \mathbb{R}^d$. In this paper, we use the terms of text and representation interchangeably when the intention is obvious from the context.

2.1 Main Task

The main task is the primary classification task to optimize. We use a simple classifier as employed in BERT. The classifier takes an input vector $\boldsymbol{v}^m$ and calculates probabilities $\boldsymbol{p} \in \mathbb{R}^{|C|}$ to assign a set of class labels C:

$$\boldsymbol{p} = g(\boldsymbol{W}\boldsymbol{v}^m + \boldsymbol{b}),$$

where $\boldsymbol{W} \in \mathbb{R}^{|C| \times d}$ and $\boldsymbol{b} \in \mathbb{R}^{|C|}$ are parameters of the classifier, in which $|\cdot|$ counts the number of elements in a set.

For g, we employ a softmax function for single-label classification and a sigmoid function for multi-label classification. In both cases, $\mathcal{L}_m$ is a negative log-likelihood of predictions.

2.2 Auxiliary Task

The auxiliary task aims to give negative supervision to encourage distinct representations of texts with different labels. The discriminator samples a set of n texts $\boldsymbol{v}^a_1, \ldots, \boldsymbol{v}^a_n$ from the same batch as $\boldsymbol{v}^m$, all of which have different labels from $\boldsymbol{v}^m$.

To encourage these texts to have distinct representations, we designed the loss function $\mathcal{L}_a$ as

$$\mathcal{L}_a = \frac{1}{n}\sum_i s^m_i, \quad s^m_j = 1 + \mathrm{cossim}(\boldsymbol{v}^m, \boldsymbol{v}^a_j),$$

where the cossim function computes the cosine similarity between the representations. This loss function intuitively encourages the negative examples to have smaller cosine similarities.

3 Experiments

We conducted a comprehensive evaluation to investigate the performance of our model in terms of (a) single- and multi-label classifications, (b) sentence- and document-level classification, and (c) different languages. We collected the standard evaluation datasets from heterogeneous sources, as summarised in Table 2.

3.1 Single-Label Classification

As datasets assigned single labels to sentences, we used the following datasets from the SentEval (Conneau and Kiela, 2018)[2] benchmark.

MR Binary classification of sentiment polarity of movie reviews.

[2]https://github.com/facebookresearch/SentEval

	Input	Language	$\|C\|$	# of train data	# of validation data	# of test data
MR	sentence	English	2	6,823	1,706	2,133
CR	sentence	English	2	2,416	604	755
SST-5	sentence	English	5	8,544	1,101	2,210
TREC	sentence	English	6	4,361	1,090	500
SUBJ	sentence	English	2	6,400	1,600	2,000
MedWeb	sentence	Japanese	8	1,536	384	640
	sentence	English	8	1,536	384	640
	sentence	Chinese	8	1,536	384	640
arXiv	document	English	40	38,188	9,548	11,935

Table 2: Statistics on the datasets. The upper group is single-label classification tasks, whereas the bottom group is multi-label classification tasks.

CR Binary classification of sentiment polarity of product reviews.

SST-5 Multi-class classification of the fine-grained sentiment polarity of movie reviews. Labels are `Positive`, `Somewhat Positive`, `Neutral`, `Somewhat Negative`, and `Negative`.

TREC Multi-class classification of question types.[3]

SUBJ Binary classification of subjectivity.

Because the MR, CR, and SUBJ datasets do not separate validation and test sets, we split 20% of each dataset for testing and 20% of the remainder for validation. The evaluation metric for these single-label classification tasks is accuracy.

3.2 Multi-Label Classification

We used the NTCIR-13 MedWeb (Wakamiya et al., 2017) and arXiv datasets (Yang et al., 2018) for multi-label classification.

MedWeb Assigning disease labels that a writer of a sentence contracted.[4]

arXiv Classification of areas of abstracts extracted from papers in the computer science field.[5]

Because the arXiv dataset released by Yang et al. (2018) removed all line breaks, we created one ourselves. We collected abstracts and categories of papers submitted to arXiv from January 1st, 2019 to June 4th, 2019 using arXiv API.[6]

The evaluation metric for multi-label classification is Exact-Match.

$$\text{ExactMatch} = \frac{1}{M}\sum_{i=1}^{M} I(\boldsymbol{y}_i = \hat{\boldsymbol{y}}_i),$$

where $\boldsymbol{y}_i$ and $\hat{\boldsymbol{y}}_i$ are one-hot vectors of gold and predicted labels, respectively, and $I(x)$ takes 1 when x is true and takes 0 otherwise. M is the size of a test set.

3.3 Settings

As a text encoder, we employed BERT and a Hierarchical Attention Network (HAN) (Yang et al., 2016) for generating sentence and document representation, respectively. For BERT, we used the pre-trained BERT-base[7] ($d = 768$). We implemented the HAN following Yang et al. (2016) who used the bi-directional Gated Recurrent Unit as the encoder with the hidden size of 50 ($d = 50$). The embedding layer of the HAN was initialised using CBOW (Mikolov et al., 2013) embeddings (with dimensions of 200), which were trained using negative sampling on the training and development sets of each task.

For systematic comparison, we investigated the performance of the following models. As a baseline, we compared models that conduct only the main task (referred to as Baseline), which corresponds to the fine-tuned BERT-base for sentence classification and the original HAN for document classification. Note that this BERT baseline significantly outperforms previous state-of-the-art methods, which were also compared in the experiment. To investigate the effects of negative supervision at

[3] All question types are in the appendix.

[4] http://research.nii.ac.jp/ntcir/permission/ntcir-13/perm-ja-MedWeb.html

[5] The labels of these two tasks are in the appendix.

[6] https://arxiv.org/help/api

[7] https://github.com/google-research/bert

	MR	CR	SST-5	TREC	SUBJ	MedWeb			arXiv
						Ja	En	Zh	
SOTA	83.5	86.3	52.4	96.4	95.5	82.5	79.5	80.9	-
Baseline	86.5	89.2	**54.0**	97.0	96.5	86.1	83.1	86.9	36.0
ACE	86.3	88.8	53.2	97.0	96.5	86.2	82.8	86.8	35.8
AM	86.4	89.1	52.9	**97.2**	96.3	86.5	83.2	**87.1**	36.3
AAN	**86.8**	**89.4**	53.0	96.9	**96.6**	**87.1**	**83.6**	86.4	**36.4**

Table 3: Evaluation results. The best scores are presented in the **bold** font, and scores higher than the Baseline are underlined. Our models consistently outperform the baseline and ACE, which indicates the effectiveness of negative supervision through the auxiliary task. Previous SOTA results are reported by Du et al. (2019) (MR), Zhou et al. (2016) (CR, SST-5), Howard and Ruder (2018) (TREC), Zhao et al. (2015) (SUBJ) and Iso et al. (2017) (MedWeb).

the auxiliary task, we compared our model to one that predicts a sentence with the same label. Accurately, this model conducts classification given cosine similarities using cross entropy loss (referred to as ACE (the auxiliary task with cross entropy loss)).

Furthermore, we evaluated two variations of our model. The first purely gives negative supervision, *i.e.*, the auxiliary task only encourages the generation of distinct representation to negative examples, as described in Section 2.2 (referred to as AAN (the auxiliary task using all negative examples)). The second uses the following margin-based loss as $\mathcal{L}_a$ with a positive example as well as negative examples:

$$\mathcal{L}_a = \max\left(0, \delta - s_k^m + \frac{1}{n-1}\sum_{i \neq k} s_i^m\right),$$

where the k-th sample is selected to have the same label as the input $\boldsymbol{v}^m$ to the main task and δ is the margin empirically set to 0.4 (referred to as AM (the auxiliary task with the margin-based loss)). The intuition is that texts with the same label should have more similar representations than negative examples.

We set the batch size of the main task to 16 and set n to four in the auxiliary task, which performed best on the validation set of the MR task. We used early stopping to cease training when the validation score did not improve for 10 epochs. The optimization algorithm used was Adam (Kingma and Ba, 2015) with $\beta_1 = 0.999$ and $\beta_2 = 0.9$. For each task, we selected the best learning rate among $1\mathrm{e}-5$, $3\mathrm{e}-5$, and $5\mathrm{e}-5$ using the validation set. To alleviate randomness owing to initialization, we reported average scores of 10 time trials excluding the best and worst results.

3.4 Results

Table 3 shows the performance of all compared methods as well as the performance of the previous state-of-the-art methods (referred to as SOTA). The results in Table 3 indicate that our models of AM and AAN consistently outperform the strong Baselines on both single-label and multi-label classifications, sentence and document classifications, and classifications in different languages. Most notably, our models are effective even for multi-label classification, which is more challenging than its single-label counterpart. In general, AAN achieved greater performance than AM. However, their effectiveness turned out to be task-dependent.

Unlike AM and AAN, ACE degraded the performance of the Baseline except for the MedWeb Japanese task. This result shows that simple multitask learning is ineffective and that our design using negative supervision is crucial.

SST-5 is an exception wherein our models degraded the performance of the Baseline. We hypothesise that this is because its class labels are gradational, *e.g.* `Somewhat Negative` is closer to `Negative` rather than `Positive` sentences. AM and AAN treat all negative examples equally, disregarding variables, such as relations between class labels. Future work should focus on the semantic relations among class labels in the auxiliary task.

4 Related Work

Multitask learning has been employed to improve the performance of text classification (Liu et al.,

2019; Xiao et al., 2018). Previous studies aimed to improve multiple tasks; hence, they required multiple sets of annotated datasets. In contrast, our method does not require any extra labelled datasets and is easily applicable to various classification tasks.

The methods proposed in Arase and Tsujii (2019) and Phang et al. (2018) improved the BERT classification performance by further training the pre-trained model using natural language inference and paraphrase recognition. Similar to multitask learning, both methods require an additional large-scale labelled dataset. Furthermore, these previous studies revealed that the similarity of tasks in training affects the models' final performance (Xiao et al., 2018; Arase and Tsujii, 2019). Our method achieved consistent improvements across tasks, indicating its wider applicability.

5 Conclusion

In this paper, we proposed a simple multitask learning model that uses negative supervision to generate distinct representations for texts with different labels. Comprehensive evaluation empirically confirmed that our model consistently outperformed BERT and HAN models on single- and multi-label classifications, sentence and document classifications, and classifications in different languages. Our multitask learning model provides a general framework that is easily applicable to existing text classification models.

In future work, we will examine semantic relations between class labels in the auxiliary task. Moreover, we will adapt our model to text generation tasks. We expect that our model will encourage a generation model to generate texts with different labels, such as styles, have distinct representations, which will result in class specific expressions.

Acknowledgement

This work was supported by JST AIP-PRISM Grant Number JPMJCR18Y1, Japan.

References

Yuki Arase and Jun'ichi Tsujii. 2019. Transfer fine-tuning: A BERT case study. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 5393–5404.

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*, pages 1699–1704.

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*, pages 4171–4186.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, Jianxin Liao, Chun Wang, and Bing Ma. 2019. Investigating capsule network and semantic feature on hyperplanes for text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 456–465.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 328–339.

Hayate Iso, Camille Ruiz, Taichi Murayama, Katsuya Taguchi, Ryo Takeuchi, Hideya Yamamoto, Shoko Wakamiya, and Eiji Aramaki. 2017. NTCIR13 MedWeb Task: multi-label classification of tweets using an ensemble of neural networks. In *Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies*, pages 56–61.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of Tricks for Efficient Text Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 427–431.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A Convolutional Neural Network for Modelling Sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 655–665.

Diederik Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, pages 1–15.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, Workshop Track Proceedings*.

Jason Phang, Thibault Févry, and Samuel R. Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks.

Dinghan Shen, Yizhe Zhang, Ricardo Henao, Qinliang Su, and Lawrence Carin. 2018. Deconvolutional Latent-Variable Model for Text Sequence Matching. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 5438–5445.

Shoko Wakamiya, Mizuki Morita, Yoshinobu Kano, Tomoko Ohkuma, and Eiji Aramaki. 2017. Overview of the NTCIR-13: MedWeb Task. In *Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies*, pages 40–49.

Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. Joint Embedding of Words and Labels for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2321–2331.

Sida Wang and Christopher Manning. 2012. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 90–94.

Liqiang Xiao, Honglun Zhang, Wenqing Chen, Yongkun Wang, and Yaohui Jin. 2018. Learning what to share: Leaky multi-task network for text classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2055–2065.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: Sequence Generation Model for Multi-label Classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3915–3926.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 649–657.

Han Zhao, Zhengdong Lu, and Pascal Poupart. 2015. Self-adaptive hierarchical sentence model. In *Proceedings of the 24th International Conference on Artificial Intelligence*, page 4069–4076.

Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, and Bo Xu. 2016. Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling. In *Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3485–3495.

A Appendix

A.1 Labels in TREC Dataset

Table 4 lists all the labels defined in the TREC dataset, which is a classification task of question types.

ABBREVIATION	ENTITY
DESCRIPTION	HUMAN
LOCATION	NUMERIC

Table 4: Labels in TREC dataset

A.2 Labels in MedWeb Dataset

Table 5 lists all the labels defined in the MedWeb dataset. The same label set was used for all Japanese, English, and Chinese tasks. The MedWeb task requires to estimate if a writer of text contracted diseases and had symptoms in Table 5. When the writer does not have any of these, the text is allowed to have no label.

Runnynose	Cough
Influenza	Diarrhea
Hayfever	Fever
Headache	Cold

Table 5: Labels in MedWeb dataset

A.3 Labels in arXiv Dataset

Table 6 lists labels used in our arXiv dataset, which are sub-areas in the computer science field. The arXiv is a document level and multi-label classification task. It requires predicting all areas that a paper belongs from its abstract.

cs.AI	cs.AR	cs.CC	cs.CE	cs.CG
cs.CL	cs.CR	cs.CV	cs.CY	cs.DB
cs.DC	cs.DL	cs.DM	cs.DS	cs.ET
cs.FL	cs.GL	cs.GR	cs.GT	cs.HC
cs.IR	cs.IT	cs.LG	cs.LO	cs.MA
cs.MM	cs.MS	cs.NA	cs.NE	cs.NI
cs.OH	cs.OS	cs.PF	cs.PL	cs.RO
cs.SC	cs.SD	cs.SE	cs.SI	cs.SY

Table 6: Labels in arXiv dataset