

FedCSIS_2023_paper_2799.pdf

Distance Metric Learning Loss Functions in Few-Shot Scenarios of Supervised Language Models Fine-Tuning

Witold Sosnowski

0000-0002-2241-9588

Polish-Japanese

Academy of Information Technology

Warsaw, Poland

Email: witold.sosnowski.dokt@pw.edu.pl

Karolina Seweryn

0000-0003-0617-7301

Warsaw University of Technology

NASK - National Research Institute, Poland

Warsaw, Poland

Email: karolina.seweryn@nask.pl

Anna Wróblewska, Piotr Gawrysiak

0000-0002-3407-7570

0000-0002-9647-6761

Warsaw University of Technology

Warsaw, Poland

Email: anna.wroblewska1@pw.edu.pl

p.gawrysiak@ii.pw.edu.pl

Abstract—This paper analyses the influence of the Distance Metric Learning (DML) loss functions on the supervised fine-tuning of the language models for classification tasks. We experimented with seven datasets from SentEval Transfer Tasks for three different scenarios: 10, 100, and 1,000 training samples.

Our study results indicate that applying the DML loss function can increase performance on downstream classification tasks of RoBERTa-large models in few-shot scenarios. Models fine-tuned with the use of *SoftTriple* loss can achieve better results than models with a standard *categorical cross-entropy* loss function by about 2.89 percentage points from 0.04 to 13.48 percentage points depending on the training dataset. *SoftTriple* is also superior to the *SupCon* loss by average 1.62 percentage points from -0.22 to 9.19 percentage points (it is better for 19 out of 21 scenarios). In addition, we conducted a comprehensive case study analysis on the sentiment analysis task with the MR dataset. The analysis of the accuracy differences between embedding groups in relation to the central embedding representation proved that our method outperforms others for all analysed groups. Moreover, our approach achieved much better accuracy for samples close to the central embedding than cross-entropy loss. Also, Shapley values and Geval methods were used to assess the models' reliability and explain their results.

I. INTRODUCTION

The development of new techniques in the Natural Language Processing (NLP) field has been studied over the last few years. It resulted in a few breakthroughs, which constantly stimulated waves of interest in the text processing area. The most recent and sophisticated discoveries are based on the new Transformer [1] architecture that enabled capturing the semantic meaning of the sentence. The Transformer architecture facilitated the discovery of the BERT encoder [2] that nowadays is massively used to solve most NLP tasks by adapting it in the fine-tuning process.

Unfortunately, fine-tuning of pre-trained models has a number of flaws. First of all, it is not designed to perform well when the number of observations is limited, so large training sets are consistently required for models to perform well [3]. Secondly, the fine-tuning process happens to be very unstable across different runs with different seeds, even though just a few minor components of the learning process are dependent

on random seeds [4]. The lack of stability is even more exacerbated in the case of few-shot learning scenarios.

On the other hand, pre-trained models are fine-tuned for a specific task using a *cross-entropy* objective function that focuses on learning class-specific features rather than their class representations. In other words, it only encourages inter-class distances and is not taking care of minimizing intra-class distances that would result in learning discriminative features [5]. This leads to a poor generalization of the model and thus causes problems with noisy or outlier data [6].

In response to these challenges, recent research has explored the application of contrastive learning strategies, a subset of the Distance Metric Learning (DML) function family, to improve the fine-tuning process. For example, [7] and [8] explored contrastive learning for text classification with limited annotations, showing promising results. In contrast, [9] introduced a simple contrastive learning framework called SimCSE, for sentence embeddings, which proved superior to traditional sentence augmentations in their framework. [10] proposed a contrastive framework for self-supervised sentence representation transfer ConSERT that showed significant improvements in performance. Additionally, the DeCLUTR model [11] combined contrastive learning with a masked language model objective, yielding improved text representations. MoCoSE [12] applied different contrastive learning frameworks and data augmentation techniques, showing significant improvements in sentence embeddings. PT-BERT [13] introduced a pseudo token embedding layer for managing sentence length impacts, effectively mitigating the impacts of sentence length. DiffCSE [14] learned sentence embedding differences with contrastive learning. DCLR [15] proposed an instance weighting method to penalize false negatives in sentence embedding, creating a more uniform representation space.

The prevailing body of research has primarily centered on the application of contrastive learning methods in NLP, adopting a self-supervised approach. Contrary to this, our investigation has been geared towards understanding how DML

losses function in supervised settings. Additionally, much of the existing literature has employed traditional contrastive learning methodologies, whereas our approach has led us to delve into the more sophisticated proxy-based approach. As a result, our focus has shifted towards studying the impact of incorporating loss functions, from the DML family, into the supervised fine-tuning process of a BERT-based encoder for downstream tasks, specifically in the context of few-shot learning situations.

Our main contributions are the following:

- 1) We apply *SoftTriple* loss in the few-shot scenarios for supervised fine-tuning language model
- 2) We examine the effect of using loss functions from the DML family (i.e. *SoftTriple*, *SupCon*) on the supervised fine-tuning of the RoBERTa-large language model.
- 3) We establish that *SoftTriple* loss is more efficient than Supervised Contrastive loss for the supervised fine-tuning RoBERTa-large language model.
- 4) We show that applying the DML loss to the RoBERTa-large language model is more viable the smaller the training set.
- 5) We check the reliability of models with comprehensive analysis and explainability techniques.

The following section is dedicated to a brief overview of the DML methods (Section II). Our new method is outlined in Section III. The next Section IV provides a performance analysis of the models and investigates their behavior. Finally, a summary of the experiments is described in the last section.

II. METHODOLOGICAL BACKGROUND

The DML loss can be applied whenever embedded representations of input observations are learned [16]. It is most often utilized in image processing, where the high-quality image embeddings are formed into the downstream tasks, such as k-nearest neighbours classification [17], clustering or image retrieval [18]. The main goal of DML is to minimize distances between observations from the same class while maximizing inter-class distances.

A. Contrastive Loss

One of the standard DML methods is Contrastive Loss [19]. It considers pairs of observations and minimizes their distances if they belong to the same class, otherwise, it maximizes them.

The Contrastive Loss is given in Equation 1.

$$\ell = (1 - y)\frac{1}{2} (D(z_1, z_2))^2 + (y)\frac{1}{2} \{\max(0, m - D(z_1, z_2))\}^2 \quad (1)$$

where z_1, z_2 denotes the representation (embedding) pair of the sample x_2 . y denotes the label assigned to this pair. If x_1 and x_2 belong to the same class, $y = 1$, otherwise, $y = 0$. D denotes learnable distance function between pair of points x_1, x_2 as the euclidean distance between their embeddings. $m > 0$ is a margin. Unsimilar points affect the loss if the distance between their embeddings is no greater than m .

B. Triplet Loss

Triplet Loss is another standard DML method that is an extension of Contrastive Loss [20]. It focuses on triples consisting of an anchor, a positive observation (an observation from the anchor class), and a negative observation (an observation from outside the anchor class). The distance between the anchor and the positive observation is minimized, while that from the anchor to the negative example is maximized. The Triplet Loss is given in Equation 2.

$$\ell = \sum_{i=1}^N \left[\|z_i^a - z_i^p\|_2^2 - \|z_i^a - z_i^n\|_2^2 + m \right]_+ \quad (2)$$

where i denotes the index of the triplet from the set of all triplets $\mathcal{T}$ in the training set with the cardinality N . z represents the embedding of an example x . x_i^a is an anchor, x_i^p (positive) is the positive observation, x_i^n (negative) denotes a negative observation, α denotes a margin imposed between positive and negative examples.

C. Supervised Contrastive Learning Loss

Supervised Contrastive Learning (*SupCon*) is one of the modern DML approaches which outperforms traditional methods such as Triplet Loss or Contrastive Loss. The *SupCon* introduces temperature regularization [21] and batch processing which means that the loss is not calculated just for a single triplet but is the average of all possible triplets from a given batch. The *SupCon* loss extends the self-supervised batch DML approach to the fully-supervised setting, which provides the ability to use the label information [22]. Instead of contrasting one positive example for an anchor with all other observations from the batch, *SupCon* contrasts all examples from the same class (as positives) with all other observations from the batch as negatives. The most critical issue with this approach is that as the number of observations in the batch grows, the number of triplets grows cubically.

The *SupCon* loss has been previously applied in the supervised fine-tuning of the RoBERTa-large language model [23], and its formula is as follows:

$$\ell_{SupCon} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z_i \cdot z_p / \tau)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a / \tau)}, \quad (3)$$

where $i \in I \equiv \{1, \dots, N\}$ denotes the index of an anchor observation x_i , $P(i)$ is the set of observations from the same class that x_i belongs to, $A(i) \equiv I \setminus \{i\}$, z_i denotes the representation of the x_i observation that is outputted by the encoder. x_p denotes observation from the same class as x_i (positive class), while x_n denotes the observations from different class as i . $\tau \in \mathcal{R}^+$ is a scalar temperature.

D. SoftTriple Loss

SoftTriple loss is another modern DML approach that extends the batch supervised DML approach [24]. It employs proxies – artificial embeddings that are learned during the training procedure – to represent classes from the dataset. In other words, each class has its own additional set of points

that best approximates the position of all observations from the class. The number of proxies is given as a hyperparameter and should reflect the global distribution of features from the class, which may have more than one focal point. The loss is calculated based on the possible triplets in the batch. It is the same as the *SupCon* loss. The difference is in calculating the single triplet. It is composed of the anchor (observation), positive proxy (a proxy from the anchor class) and negative proxy (a proxy from a class other than the anchor). The number of all possible triples is linear with respect to the number of original examples, while it grows cubically (as stated earlier) for the *SupCon* or other non-proxy losses. Therefore, this approach consumes much fewer resources and has higher performance because the proxy provides better resistance to outliers.

The following formulas define the *SoftTriple* loss:

$$\ell_{SoftTriple} = -\frac{1}{N} \sum_{i \in I} \log \frac{\exp(\lambda(S'_{i,y_i} - \delta))}{\exp(\lambda(S'_{i,y_i} - \delta)) + \sum_{j \neq y_i} \exp(\lambda S'_{i,j})} \quad (4)$$

$$S'_{i,c} = \frac{\exp\left(\frac{1}{\gamma} \mathbf{E}(\mathbf{x}_i)^\top \mathbf{w}_c^k\right)}{\sum_{k \in K} \sum_{c \in C} \exp\left(\frac{1}{\gamma} \mathbf{E}(\mathbf{x}_i)^\top \mathbf{w}_c^k\right)} \mathbf{E}(\mathbf{x}_i)^\top \mathbf{w}_c^k, \quad (5)$$

where $i \in I \equiv \{1, \dots, N\}$ denotes the index of an anchor observation x_i , y_i its corresponding label, $c \in C$ denotes the class index, C denotes the class number, $k \in K$ denotes the proxy index of class c , K denotes the number of proxies per class, δ denotes a minimum interclass margins, λ scaling factor that reduces the outliers' impact, γ scaling factor for the entropy regularizer, x_i denotes a i th observation' embedding, $E(\cdot) \in \mathbf{R}^d$ denotes an encoder and $\mathbf{w}_c^k$ are weights that represent embeddings of the class c .

III. OUR APPROACH

We analyze the impact of the DML loss function on the supervised training of the pre-trained language model by extending the loss function with *SupCon* loss or *SoftTriple* loss as a continuation of our earlier research [24]. In general, the new loss function contains two elements; the first one is *categorical cross-entropy* loss applied to the output of the classification layer, and the DML loss is applied directly to the output of the encoder. The second part of our loss function maximizes distances between observations from different classes and minimizes distances between observations from the same class.

The goal function is given by the following formula:

$$\mathcal{L} = (\beta) \ell_{CCE} + (1 - \beta) \ell_{DML}, \quad (6)$$

$$\ell_{CCE} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{i,c} \cdot \log \hat{y}_{i,c} \quad (7)$$

where N is the number of observations, C denotes the class number, $y_{i,c}$ denotes the label of the class c for the i th observation, $p_{i,c}$ denotes the models's predicted probability of

the i th observation for the c th class β denotes the scaling factor that tunes the influence of both parts of the loss, ℓ_{DML} denotes either ℓ_{SupCon} or $\ell_{SoftTriple}$. ℓ_{CCE} is the *categorical cross-entropy*.

A. Model Architecture

We compare the loss functions for supervised fine-tuning of the RoBERTa-large [25] encoder provided by the *huggingface* library as the pre-trained model *roberta-large*. The standard supervised fine-tuning procedure starts by input text tokenization with the use of WordPiece tokenizer [26]. The tokenized text is represented as an array of tokens with the *[CLS]* token at the beginning, the *[EOS]* at the end, and the *[SEP]* separating sentences. The array of tokens is then passed to the RoBERTa model $E(\cdot)$. The encoder output is an array of embeddings corresponding to the array of input tokens. In the standard setup, the embedding array is passed to a fully connected layer from which as many neurons as we have classes are output. The output of this layer is then passed to the softmax function and then to the *categorical cross-entropy* function, which, while fed with labels, calculates the loss.

Our experiments involve modifying the loss function by adding the part calculated from the DML loss. In particular, we analyze the effect of introducing an additional *SupCon* loss or a *SoftTriple* loss.

1) *CCE + SupCon*: The first experimental setting includes fine-tuning a model operating on the loss function consisting of the *categorical cross-entropy* loss and *SupCon* loss according to Formula 6. The *SupCon* objective calculates the loss on the x_i representation outputted by the encoder $E(\cdot)$. In our case, the encoder is RoBERTa-large, and a single observation is represented as the vector associated with the *[CLS]* token. A single loss is calculated based on the x_i representations from the batch. Such architecture was previously proposed by [23] and outperformed a baseline in a few-shot learning setting.

2) *CCE + SoftTriple*: The second experimental setting consists of the *SoftTriple* loss function built according to Formula 6, where the ℓ_{DML} part is represented by the $\ell_{SoftTriple}$. Similarly, as in the case of *SupCon* loss, the *SoftTriple* objective calculates the loss based on the batch of x_i representations outputted by the encoder $E(\cdot)$. Here we also use RoBERTa-large as the encoder, but instead of treating a single *[CLS]* vector as the observation representation, we use the entire encoder output. Such implementation can be considered as the development of the idea where the *SupCon* loss was applied only to the representation of the *[CLS]* token [23]. Feeding the DML loss to the entire model output may provide better embedding generalization but is less resource-efficient. The *SoftTriple* loss uses a proxy to compute the loss, which reduces and balances the resource requirements of the entire procedure. This is the first time this architecture is studied in the few-shot learning settings.

B. Training Procedure

We conducted experiments operating on 40-fold cross-validation; each fold was run 40 times. Thus, each result is

an average F1-score of 40 runs from 40-fold cross-validation. The training was done in few-shot learning scenarios, so the required training set sizes were limited to 20 and 100 observations. We also conducted experiments on datasets limited to 1,000 observations for comparison purposes. The 40-fold cross-validation was introduced to tackle the problem of high variance, which is natural in the few-shot learning settings [27].

For each task, the baseline result was obtained separately based on the hyperparameter sweep with the same batch size = 64, learning rates $\in \{1e5, 2e5, 3e5\}$, epochs number $\in \{8, 16, 64, 128\}$, linear warmup for the first 6% of steps and weight decay coefficient = 0.01. The best hyperparameter set for each task includes a learning rate of $1e-5$ and 8 epochs for 1,000 elements datasets, 64 for 100-element datasets and 128 for 20-element datasets.

TABLE I
SupCon HYPERPARAMETERS.

Loss	# train	β	τ
SupCon	20	0.9	0.6
SupCon	100	0.9	0.7
SupCon	1,000	0.9	0.7

TABLE II
SoftTriple HYPERPARAMETERS.

Loss	# train	k	γ	λ	δ	β
SoftTriple	20	25	0.1	9	0.7	0.4
SoftTriple	100	2000	0.1	4	0.7	0.8
SoftTriple	1,000	2000	0.1	7	0.9	0.9

The hyperparameter search for models with the DML loss function includes additional elements. For the SupCon loss, the grid search was performed on the following parameters: $\tau \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $\beta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. In the case of SoftTriple loss, we searched the following parameters: $k \in \{5, 25, 1000, 2000\}$, $\gamma \in \{0.01, 0.03, 0.05, 0.07, 0.1\}$, $\lambda \in \{1, 3, 3.3, 4, 6, 8, 10\}$, $\delta \in \{0.1, 0.3, 0.5, 0.7, 0.9, 1\}$ and $\beta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. In most cases, the same hyperparameter set was present across datasets with the same size, and for the SupCon, it is shown in the Table I while for the SoftTriple, it is referred in the Table II.

C. Datasets

The datasets on which we tested the models come from the well-known SentEval Transfer Task, which includes classification and textual entailment tasks [28], giving a fair picture of overall model performance (see Table VI).

IV. EXPERIMENTAL RESULTS

A. Result Analysis

Tables III to V present our results separately for models trained on 20, 100 and 1,000 observations. The results show a baseline performance where the model was trained using only CCE loss compared to models trained on CCE + SupCon loss and CCE + SoftTriple loss.

1) 20-element dataset: Table III shows the performance of models trained on datasets limited to 20 elements. In every case except the MPQA set, the SupCon loss models outperformed the baseline, resulting in an average improvement of 3.59% over the baseline. Furthermore, models with the SoftTriple loss outperformed the baseline and models with the SupCon loss in each case, giving an average gain of 6.23 percentage points over baseline and 2.61 percentage points over the SupCon approach. The most significant improvement over baseline was observed for the MR dataset and the SoftTriple loss model, almost 13.5 percentage points. For the models with the SupCon loss, the most considerable gain is also marked on the MR dataset and is about 9 percentage points. On the other hand, the lowest improvement from baseline for the SoftTriple loss models is 2.29 percentage points for the MPQA dataset, while the SupCon showed a deterioration in performance from baseline on this dataset. In addition, analysis of descriptive statistics showed that all results for SoftTriple have p -value less than 0.05, while for SupCon, the results for the SST2, MR, and MRPC datasets have p -value less than 0.05.

2) 100-element dataset: Table IV details the performance of models trained on datasets limited to 100-element. The SupCon loss models are better than the baseline in every case except the MPQA and TREC datasets, which yield an average improvement of 0.79 percentage points. The SoftTriple loss models are better than the baseline model in every case, resulting in an average improvement of 1.85 percentage points. Furthermore, the SoftTriple loss models outperform the SupCon loss models in every case, except for the SST2 dataset, where the SupCon model outperforms the SoftTriple by 0.35 percentage points. The largest improvement over baseline was observed for the MR set and the SoftTriple loss model and is 2.91 percentage points. For the models with the SupCon loss, the largest gain is also observed on the MR dataset and is 2.67 percentage points. On the other hand, the lowest improvement from baseline for the SoftTriple loss models is 0.67 percentage points for the SUBJ dataset. At the same time, the SupCon showed a deterioration in performance from baseline on the MPQA and TREC datasets. We observed that for SoftTriple, the p -value smaller than 0.05 was obtained for all datasets except MRPC and TREC, while for SupCon, the p -value less than 0.05 was found for SST2 and MR datasets.

3) 1,000-element dataset: Table V details the performance of models trained on datasets limited to 1,000 elements. The SupCon loss models are better than the baseline in every case except the MPQA dataset, which yields an average improvement of 0.44 percentage points. The SoftTriple loss models are better than the baseline model in every case, resulting in an average improvement of 0.58 percentage points. Furthermore, the SoftTriple loss models outperform the SupCon loss models in every case, except for the TREC dataset, while in the case of SST2, the performance was the same. The largest improvement over baseline was observed for the MRPC set and the SoftTriple loss model and is 1.9 percentage points. For the models with the SupCon loss, the largest improvement is also observed on the MRPC dataset and is 1.36 percentage

1

TABLE III

F1 SCORE OF ROBERTA-LARGE (RL) VS ROBERTA-LARGE WITH SUPERVISED CONTRASTIVE LEARNING (*SupCon*) LOSS VS ROBERTA-LARGE WITH *SoftTriple* LOSS TRAINED ON THE 20-ELEMENT DATASETS.

Model	Loss	SST2	MR	MPQA	SUBJ	TREC	CR	MRPC	Average
RoBERTa-large	CCE	53.71±8.74	56.55±8.67	65.66±5.01	85.54±6.38	41.48±9.46	65.03±8.26	63.30±7.71	61.61
RoBERTa-large	CCE + SupCon	60.04±8.98	65.74±9.69	64.92±4.91	87.66±4.76	42.81±10.55	66.59±6.64	68.85±5.42	65.23
	p-value	0.002	<0.001	0.5	0.096	0.555	0.355	<0.001	
RoBERTa-large	CCE + SoftTriple	62.35±7.44	70.03±8.16	67.96±4.72	89.48±4.62	47.21±9.44	68.55±6.91	69.32±4.71	67.84
	p-value	<0.001	<0.001	0.038	0.002	0.008	0.042	<0.001	

TABLE IV

F1 SCORE OF ROBERTA-LARGE (RL) VS ROBERTA-LARGE WITH SUPERVISED CONTRASTIVE LEARNING (*SupCon*) LOSS VS ROBERTA-LARGE WITH *SoftTriple* LOSS TRAINED ON THE 100 ELEMENT DATASETS.

Model	Loss	SST2	MR	MPQA	SUBJ	TREC	CR	MRPC	Average
RoBERTa-large	CCE	85.87±5.50	82.57±5.94	82.50±5.65	93.91±1.47	82.72±4.73	89.43±3.65	72.13±4.24	84.16
RoBERTa-large	CCE + SupCon	88.47±1.38	85.24±3.30	81.18±6.16	94.39±1.43	81.82±4.51	90.73±3.02	72.85±4.41	84.95
	p-value	0.006	0.016	0.321	0.143	0.386	0.087	0.459	
RoBERTa-large	CCE + SoftTriple	88.12±1.83	85.49±3.14	85.26±4.12	94.57±1.39	83.51±4.37	91.16±3.41	73.95±4.34	86.01
	p-value	0.018	0.008	0.015	0.042	0.44	0.031	0.062	

1

TABLE V

F1 SCORE OF ROBERTA-LARGE (RL) VS ROBERTA-LARGE WITH SUPERVISED CONTRASTIVE LEARNING (*SupCon*) LOSS VS ROBERTA-LARGE WITH *SoftTriple* LOSS TRAINED ON THE 1,000-ELEMENT DATASETS.

Model	Loss	SST2	MR	MPQA	SUBJ	TREC	CR	MRPC	Average
RoBERTa-large	CCE	91.59±0.69	89.73±2.20	90.16±1.25	96.04±1.30	89.89±2.87	93.16±2.60	77.65±3.63	89.75
RoBERTa-large	CCE + SupCon	91.98±0.70	89.91±2.17	89.94±1.95	96.23±1.16	90.34±2.66	93.90±2.63	79.01±4.21	90.19
	p-value	0.014	0.714	0.550	0.492	0.469	0.209	0.126	
RoBERTa-large	CCE + SoftTriple	91.98±0.81	90.02±2.00	90.60±1.74	96.26±1.20	89.93±2.45	94.00±2.28	79.55±4.12	90.33
	p-value	0.023	0.539	0.198	0.434	0.947	0.129	0.032	

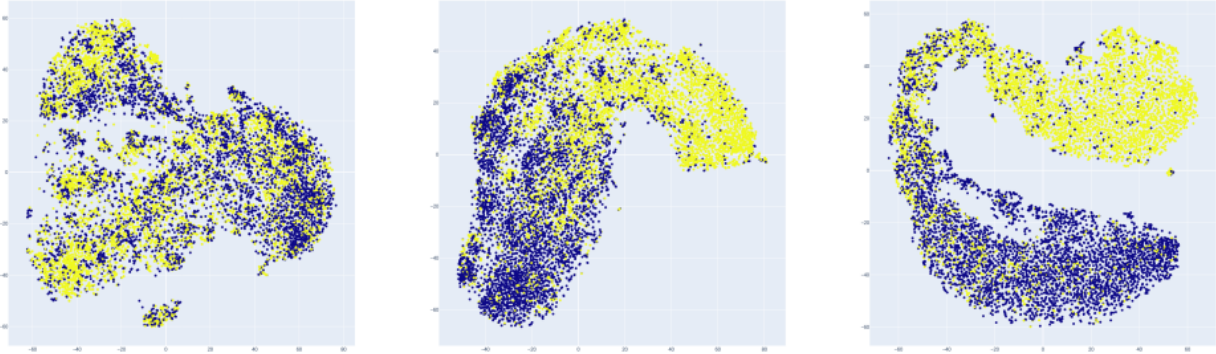


Fig. 1. t-SNE visualization of [CLS] embeddings for few-shot models trained on 20 examples from the MR dataset with different loss functions. Color represents sentiment: blue - negative, and yellow - positive. Left: *cross-entropy* (CCE), Middle: CCE + *SupCon* Loss, Right: CCE + *SoftTriple* Loss.

1

TABLE VI

SENTEVAL TRANSFER TASK DATASETS USED FOR OUR EXPERIMENTS.

Dataset	# Sentences	# Classes	Task
SST2	67k	2	Sentiment (movie reviews) [29]
MR	11k	2	Sentiment (movie reviews) [30]
MPQA	11k	2	Opinion polarity [31]
SUBJ	10k	2	Subjectivity status [32]
TREC	5k	6	Question-type classification [30]
CR	4k	2	Sentiment (product review) [33]
MRPC	4k	2	Paraphrase detection [34]

baseline for the *SoftTriple* loss models is 0.67 percentage points for the SUBJ dataset. At the same time, the *SupCon* showed a deterioration in performance from baseline on the MPQA and TREC datasets. We found that except for SST2 and MRPC datasets for *SoftTriple* and SST2 dataset for *SupCon*, the *p-value* for all the results was below 0.05.

1

The average performance of the models trained using CCE loss combined with *SoftTriple* loss is higher than the baseline or models trained using CCE loss and *SupCon* loss.

1

points. On the other hand, the lowest improvement from

1

TABLE VII

DETAILED METRICS FOR A FEW-SHOT LEARNING SCENARIO WITH 20 TRAINING SAMPLES FROM THE MR DATASET.

Model	F1 score	Accuracy	Recall	Precision
CCE	56.55 $\pm$ 8.67	59.98 $\pm$ 5.84	61.73 $\pm$ 26.19	63.98 $\pm$ 13.30
CCE + SupCon	65.74 $\pm$ 9.69	67.22 $\pm$ 8.30	62.10 $\pm$ 20.48	71.74 $\pm$ 11.16
CCE + SoftTriple	70.03 $\pm$ 8.16	71.01 $\pm$ 7.08	71.3 $\pm$ 16.59	73.48 $\pm$ 10.71

1

B. Comprehensive Analysis

Among all the results, the most significant percentage improvement of the F1 score compared to the baseline (CCE loss function) is for the MR dataset (+24% of relative difference, +13.48 of absolute difference). This section investigates sentiment analysis models trained in the few-shot scenarios on the smallest analyzed training dataset: 20 samples from the MR dataset. We conducted a comprehensive investigation to compare the models' results for those observations.

Table VII summarizes various metrics proving that the proposed method (CCE + SoftTriple) outperforms other approaches (CCE, CCE + SupCon) on the MR dataset. Particularly, considerable progress has been made regarding the recall. In this case, CCE + SupCon achieves similar results to the base model (CCE), while values for CCE + SoftTriple differ significantly. Moreover, smaller values of standard deviation indicate that models with CCE + SoftTriple are more stable.

Figure 2 presents Receiver Operating Characteristic (ROC) curve showing the trade-off between sensitivity (TPR) and specificity (1 - FPR). The curve of our method is the closest to the top-left corner, which indicates a better performance. The AUC scores are equal to 63.8, 223.60, 89.95 for CCE, CCE + SupCon, CCE + SoftTriple, respectively.

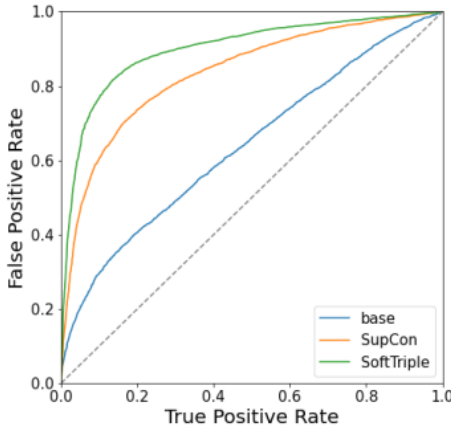


Fig. 2. Receiver Operating Characteristic (ROC) curve for few-shot setting using MR dataset.

We utilized the t-SNE algorithm to visualize sentence embeddings [35]. Figure 20 illustrates a 2-dimensional representation of input texts in the few-shot learning setting of having 20 examples in the training dataset. The baseline model

(CCE) is not able to separate the different classes. Much better results are achieved with additional SupCon loss. However, the observations in the middle of the plot are not decoupled. SoftTriple loss helps almost to isolate two classes. Even 20 observations for fine-tuning the model are enough to achieve satisfactory results. Figure 1 provides further evidence that using distance metrics in the training procedure improves the model's ability to separate different classes from each other and thus enhances its quality.

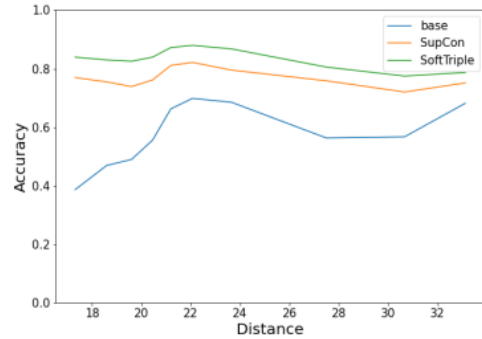


Fig. 3. Relationship between accuracy and Euclidean distance between sentence embedding and mean embedding computed by baseline model with cross-entropy loss.

Figure 3 shows a significant difference in models' performance depending on the chosen loss function. We defined a mean sentence embedding based on [CLS] representation from the baseline model with cross-entropy loss. Then, we computed the Euclidean distance between the given observation and the mean embedding. The farther from the central (mean) embedding, the greater the chance of considering the observation as an outlier. Next, the observations were sorted according to the distance from the central embedding and divided into groups of equal size. The models' accuracy was calculated in designated groups and presented on the chart. These results prove that models trained using our proposed loss function outperform others for all distance groups. Interestingly, for small values of distance, the accuracy of the base model was found to be surprisingly low. The analysis of t-SNE provides additional support that observations in the middle are more likely to be misclassified by this model. Moreover, Figure 3 illustrates the advantage of using distance-based losses during training. Both models with additional SupCon and SoftTriple losses achieve significantly better results than the base model not using these methods. Results of models

trained using distance-metric learning show a similar trend, but the *SoftTriple* is always more accurate than the *SupCon* loss.

Explainable AI techniques such as LIME [36], SHAP values [37], and integrated gradients [38] allowed us to determine whether the same features were relevant for analyzed classifiers. Moreover, such an analysis allows to indicate the weaknesses of the model and its compliance with intuition. Below we present three interesting examples of explainability with Shapley values. They all show models trained in the few-shot learning setting on 20 samples. Red shades mean that a word increases the likelihood of positive sentiment, while blue shades mean, on the contrary, an increase in the probability of predicting a negative class.

Example 1

For some samples, the models predict the same correct classes consistently. Moreover, an explanation of these models indicates that the same words increase the probability of the predicted class for all analyzed models. An example of such a situation is presented below. All models predict correct negative sentiment for this observation. Intuitively, the word *terrible* has a big influence on this prediction. Surprisingly, the word *painful* increases the probability of negative sentiment only for model with *CCE* + *SoftTriple* loss. Other models treat this word as neutral.

- ground-truth label: negative,
- prediction *cross-entropy*: positive,
it's a terrible movie in every regard, and utterly painful to watch.
- prediction *SupCon*: negative,
it's a terrible movie in every regard, and utterly painful to watch.
- prediction *SoftTriple*: negative,
it's a terrible movie in every regard, and utterly painful to watch.

Example 2

The figures below explain sentiment classification for randomly chosen observations from the test dataset. In this case, two of the models incorrectly predicted the sentiment of the review. Interestingly, the sentence contains the expression *terrible*, which, according to the explanation using the SHAP values method, does not affect the predicted class in the case of the *cross-entropy* model. On the contrary, this word increases the probability of a negative review for the other models.

- ground-truth label: negative,
- prediction *cross-entropy*: positive,
a terrible movie that some people will nevertheless find moving.
- prediction *SupCon*: positive,
a terrible movie that some people will nevertheless find moving.
- prediction *SoftTriple*: negative,
a terrible movie that some people will nevertheless find moving.

Example 3

Below, we present an observation that all analyzed models classified incorrectly as positive sentiment. Interestingly, the statement consists of the word *puzzling* whose sentiment can be interpreted differently depending on context. The model trained with *SupCon* loss function treats this word as positive

while the sentiment of this word measured with other models is slightly more negative.

- ground-truth label: negative,
- prediction *cross-entropy*: positive,
a puzzling experience.
- prediction *SupCon*: positive,
a puzzling experience.
- prediction *SoftTriple*: positive,
a puzzling experience.

Additionally, we performed test set analysis with *geval* method [39]. This method searches for features that lower the evaluation metric in such a way that it cannot be ascribed to a chance – as measured by their p-values of the Mann-Whitney rank U test. Table VIII shows a few groups of input observations, so-called buckets, that are particularly difficult for our models. For example, one bucket contains all the observations from the test set that contain the "but" word. Here, in sentiment analysis modelling, even one word "but" can change the whole sentence sense and the model result, e.g. "Its premise is smart, but the execution is pretty weary." Still, even for those hard groups of input examples (buckets), the *SubCon* is better than the baseline (*CCE*), and the *SoftTriple* achieved the best performance.

TABLE VIII
RESULTS OF THE THREE TESTED LOSS FUNCTIONS IN MODEL TRAINING: *CCE*, *CCE+SubCon*, *CCE+SoftTriple*, FOR BUCKETS – GROUPS OF INPUT EXAMPLES WITH A GIVEN CHARACTERISTIC, E.G. ALL THE INPUT EXAMPLES CONTAINING A WORD "BUT".

bucket	# examples	CCE	CCE+SubCon	CCE+SoftTriple
"n"	884	0.52	0.71	0.78
"but"	1613	0.55	0.72	0.76
"if"	524	0.52	0.70	0.74
"you"	880	0.52	0.66	0.72
"more"	685	0.54	0.72	0.79
"will"	349	0.53	0.69	0.73
Accuracy in the test set		0.58	0.77	0.83

V. CONCLUSION

This paper investigated the impact of using DML loss functions on supervised fine-tuning language models by comparing the results of RoBERTa-large trained with raw *cross-entropy* loss function and training procedure enriched with distance metrics: *SupCon* loss and *SoftTriple* loss. The performance of the models was tested based on 40-fold cross-validation over multiple datasets from the SentEval Transfer Tasks. We found that the use of these functions improves the performance of the model, and the improvement is greater the smaller the dataset is. Applying *SoftTriple* loss increases the performance over baseline on average by about 2.89 percentage points, from 0.04 to 13.48 percentage points which is superior comparing to the average 1.62 percentage points from -0.22 to 9.19 percentage points for the *SupCon*. We would like to acknowledge that most of the results for which *p-value* is less than 0.05 belong to few-shot learning scenarios (20-element datasets, 100-element datasets), which is consistent with the results from previous work [23].

The comprehensive analysis of few-shot models' performance on sentiment analysis task from the MR dataset revealed that enriching training procedure with DML methods (both *SupCon* and *SoftTriple*) cause an increase in model accuracy. The model fitted with *SoftTriple* loss achieved higher scores for all analysed metrics (F1 score, accuracy, recall, precision, and AUC). Moreover, t-SNE visualizations point towards the idea that our method separates classes better than other models, even for 20 training observations.

REFERENCES

- [1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
- [2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [3] T. Bansal, R. Jha, and A. McCallum, "Learning to few-shot learn across diverse natural language classification tasks," *arXiv preprint arXiv:1911.03863*, 2019.
- [4] T. Zhang, F. Wu, A. Katiyar, K. Weinberger, and Y. Artzi, "Revisiting few-sample bert fine-tuning," *arXiv preprint arXiv:2006.05987*, 2020.
- [5] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European conference on computer vision*. Springer, 2016, pp. 499–515.
- [6] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," *Advances in neural information processing systems*, vol. 32, 2019.
- [7] Y. Du, T. Ma, L. Wu, F. Xu, X. Zhang, B. Long, and S. Ji, "Constructing contrastive samples via summarization for text classification with limited notations," *arXiv preprint arXiv:2104.05094*, 2021.
- [8] Y. Yu, S. Zuo, H. Jiang, W. Ren, T. Zhao, and C. Zhang, "Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach," *arXiv preprint arXiv:2010.07835*, 2020.
- [9] T. Gao, X. Yao, and D. Chen, "Simcse: Simple contrastive learning of sentence embeddings," *arXiv preprint arXiv:2104.08821*, 2021.
- [10] Y. Yan, R. Li, S. Wang, F. Zhang, W. Wu, and W. Xu, "Consert: A contrastive framework for self-supervised sentence representation transfer," *arXiv preprint arXiv:2105.11741*, 2021.
- [11] C. Zong, F. Xia, W. Li, and R. Navigli, "Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: Long papers)," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2024.
- [12] R. Cao, Y. Wang, Y. Liang, L. Gao, J. Zheng, J. Ren, and Z. Wang, "Exploring the impact of negative samples of contrastive learning: A case study of sentence embeddin," *arXiv preprint arXiv:2202.13093*, 2022.
- [13] H. Tan, W. Shao, H. Wu, K. Yang, and L. Song, "A sentence is worth 128 pseudo tokens: A semantic-aware contrastive learning framework for sentence embeddings," *arXiv preprint arXiv:2203.05877*, 2022.
- [14] Y.-S. Chuang, R. Dangovski, H. Luo, Y. Zhang, S. Chang, M. Soljačić, S.-W. Li, W.-t. Yih, Y. Kim, and J. Glass, "Diffcse: Difference-based contrastive learning for sentence embeddings," *arXiv preprint arXiv:2204.10298*, 2022.
- [15] K. Zhou, B. Zhang, W. X. Zhao, and J.-R. Wen, "Debiased contrastive learning of unsupervised sentence representations," *arXiv preprint arXiv:2205.00656*, 2022.
- [16] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin, "Softtriple loss: Deep metric learning without triplet sampling," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 650–6458.
- [17] Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of machine learning research*, vol. 10, no. 2, 2009.
- [18] E. Xing, M. Jordan, S. J. Russell, and A. Ng, "Distance metric learning with application to clustering with side-information," *Advances in neural information processing systems*, vol. 15, 2002.
- [19] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1735–1742.
- [20] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

65%

SIMILARITY INDEX

PRIMARY SOURCES

1	deepai.org Internet	3487 words — 53%
2	arxiv.org Internet	134 words — 2%
3	annals-csis.org Internet	105 words — 2%
4	Qian Wang, Weiqi Zhang, Tianyi Lei, Yu Cao, Dezhong Peng, Xu Wang. "CLSEP: Contrastive learning of sentence embedding with prompt", Knowledge-Based Systems, 2023 Crossref	91 words — 1%
5	www.epfl.ch Internet	64 words — 1%
6	Zahra Ebrahimian, Ramin Toosi, Mohammad Ali Akhaee. "Multinomial Emoji Prediction Using Deep Bidirectional Transformers and Topic Modeling", 2022 30th International Conference on Electrical Engineering (ICEE), 2022 Crossref	62 words — 1%
7	www.researchgate.net Internet	34 words — 1%

8 Kanghao Chen, Weixian Lei, Ping Hu, Shen Zhao, Wei-Shi Zheng, Ruixuan Wang. "PCCT: Progressive Class-Center Triplet Loss for Imbalanced Medical Image Classification", IEEE Journal of Biomedical and Health Informatics, 2023 32 words — < 1%

[Crossref](#)

9 Chaoming Liu, Wenhao Zhu, Xiaoyu Zhang, QiuHong Zhai. "Exploiting Syntactic Information to Boost the Fine-tuning of Pre-trained Models", 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), 2022 29 words — < 1%

[Crossref](#)

10 Hyung-Il Kim, Kimin Yun, Yong Man Ro. "Face Shape-Guided Deep Feature Alignment for Face Recognition Robust to Face Misalignment", IEEE Transactions on Biometrics, Behavior, and Identity Science, 2022 29 words — < 1%

[Crossref](#)

11 Haoyang Ma, Zeyu Li, Hongyu Guo. "A Contrastive Framework to Enhance Unsupervised Sentence Representation Learning", ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2023 24 words — < 1%

[Crossref](#)

12 Vinícius Cleves de Oliveira Carmo. "A framework for closed domain question answering systems in the low data regime.", Universidade de Sao Paulo, Agencia USP de Gestao da Informacao Academica (AGUIA), 2022 24 words — < 1%

[Crossref](#) [Posted Content](#)

13 Mohamad Khajezade, Milad Ramezankhani, Fatemeh Hendijani Fard, Mohamed S Shehata, Abbas Milani. "Toward Using Few-Shot Learning for Prediction of Complex In-Service Defects of Composite Products: A case 21 words — < 1%

-
- 14 www.mdpi.com 19 words — < 1 %
Internet
-
- 15 G. Iwanski, W. Koczara. "Sensorless stand alone
variable speed system for distributed
generation", 2004 IEEE 35th Annual Power Electronics
Specialists Conference (IEEE Cat. No.04CH37551), 2004 14 words — < 1 %
Crossref
-
- 16 Taihua Shao, Fei Cai, Jianming Zheng, Mengru
Wang, Honghui Chen. "Pairwise contrastive
learning for sentence semantic equivalence identification with
limited supervision", Knowledge-Based Systems, 2023 14 words — < 1 %
Crossref
-
- 17 Peng Peng, Jiaxun Lu, Shuting Tao, Ke Ma, Yi
Zhang, Hongwei Wang, Heming Zhang. "Progressively-Balanced Supervised Contrastive Representation
Learning for Long-Tailed Fault Diagnosis", IEEE Transactions on
Instrumentation and Measurement, 2022 11 words — < 1 %
Crossref
-
- 18 pure.rug.nl 10 words — < 1 %
Internet
-
- 19 repository.kulib.kyoto-u.ac.jp 10 words — < 1 %
Internet
-
- 20 www.arxiv-vanity.com 10 words — < 1 %
Internet
-
- 21 R.Z. Morawski. "Cauchy-filter-based algorithms for
reconstruction of absorption spectra", Proceedings 9 words — < 1 %

- 22

bmcneurol.biomedcentral.com
Internet

9 words — < 1%
- 23

["Medical Image Computing and Computer Assisted Intervention – MICCAI 2019", Springer Science and Business Media LLC, 2019](#)
Crossref

8 words — < 1%
- 24

[Xiaokai Xia, Zhiqiang Fan, Gang Xiao, Fangyue Chen, Yu Liu, Yiheng Hu. "Learning Representative Features by Deep Attention Network for 3D Point Cloud Registration", Sensors, 2023](#)
Crossref

8 words — < 1%
- 25

export.arxiv.org
Internet

8 words — < 1%

EXCLUDE QUOTES OFF
EXCLUDE BIBLIOGRAPHY OFF

EXCLUDE SOURCES OFF
EXCLUDE MATCHES OFF