AugCSE: Contrastive Sentence Embedding with Diverse Augmentations

Zilu Tang
Boston University
zilutang@bu.edu

Muhammed Yusuf Kocyigit
Boston University
kocyigit@bu.edu

Derry Wijaya
Boston University
wijaya@bu.edu

Abstract

Data augmentation techniques have been proven useful in many applications in NLP fields. Most augmentations are task-specific, and cannot be used as a general-purpose tool. In our work, we present AugCSE, a unified framework to utilize diverse sets of data augmentations to achieve a better, general purpose, sentence embedding model. Building upon the latest sentence embedding models, our approach uses a simple antagonistic discriminator that differentiates the augmentation types. With the finetuning objective borrowed from domain adaptation, we show that diverse augmentations, which often lead to conflicting contrastive signals, can be tamed to produce a better and more robust sentence representation. Our methods[1] achieve state-of-the-art results on downstream transfer tasks and perform competitively on semantic textual similarity tasks, using only unsupervised data.

1 Introduction

Data augmentation in NLP can be useful in many situations, from low resource data setting, domain adaptation (Wei et al., 2021), debiasing (Dinan et al., 2020), to improving generalization, robustness (Dhole et al., 2021). In the vision domain, Chen et al. (2020b) shows that a diverse set of augmentation can be used to learn a robust general-purpose representation with contrastive learning. Similar work in sentence embedding space (Gao et al. 2021; Chuang et al. 2022) has shown that a simple single augmentation such as dropouts from transformers (Devlin et al., 2019) can be used for contrastive objective. However, no previous work has thoroughly explored the impacts of a diverse set of augmentations with contrastive learning in the sentence embedding space. It is not straightforward to find the best augmentations that work for contrastive learning in different datasets or tasks (Gao et al., 2021). Single augmentation can instill invariance in models for a specific aspects of linguistic variability, while naively combining a diverse set of augmentations can lead to contradicting gradients, preventing models from generalizing well (Table 6)[2]. In this work, we present AugCSE (Figure 1), a general approach to select and unify a diverse set of augmentations for the purpose of building a general-purpose sentence embedding. During training, in addition to using contrastive loss, we randomly perturb sentences with different augmentations and use a discriminator loss to unify embeddings from diverse augmentations. In short, our work presents the following key contributions:

- We show simple data augmentation methods can be used to improve individual tasks, while degrading performance on other tasks (due to shifted domain distribution).
- We present our simple discriminator objective that achieves competitive results on sentence similarity task (STS) and transfer classification tasks against state-of-the-art methods.
- We demonstrate through ablation and visualization that our model can unify contrasting distribution from diverse augmentations and that simple rule-based augmentations are sufficient for achieving competitive results.

2 Background and Related Work

2.1 Contrastive learning

Contrastive learning is shown to provide a clear signal to improve the embedding space, which is crucial for downstream tasks. The goal of contrastive learning is to use similar or dis-similar datapoints to regularize the embedding representation, such that similar datapoints (by human, or pre-defined

[1] Our code and data can be found at https://github.com/PootieT/AugCSE

[2] Diverse augmentations have been shown to work without discriminator in vision (Chen et al., 2020b). We believe the difference resides in a much more structural distribution in natural language in comparison to images.

Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 375–398
November 20–23, 2022.

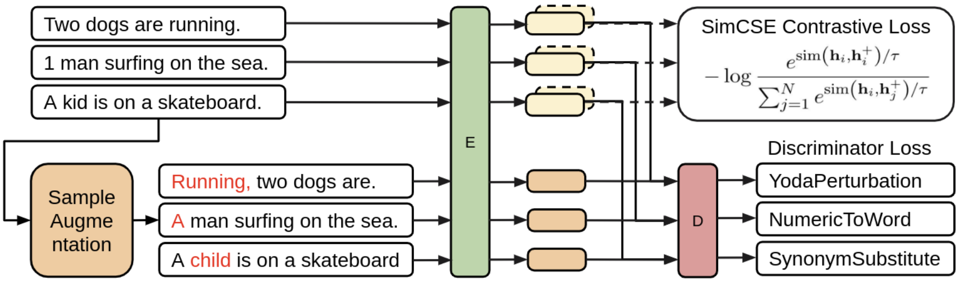

Two dogs are running.
1 man surfing on the sea.
A kid is on a skateboard.
Sample Augme ntation
Running, two dogs are.
A man surfing on the sea.
A child is on a skateboard
E
SimCSE Contrastive Loss
$-\log \frac{e^{\mathrm{sim}(\mathbf{h}_i,\mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}(\mathbf{h}_i,\mathbf{h}_j^+)/\tau}}$
Discriminator Loss
D
YodaPerturbation
NumericToWord
SynonymSubstitute

Figure 1: Overall framework of AugCSE. During training, each input sentence is randomly augmented with one of many augmentation methods. In addition contrastive loss from SimCSE, we add an antagonistic discriminator to predict the augmentation performed on the input example.

standards) are embedded closer than those datapoints that aren't similar. Recently, many works in vision use contrastive objectives to obtain SOTA performance on image tasks from classification, detection, to segmentation using ImageNet (Deng et al., 2009; Caron et al., 2018; Chen et al., 2020b; He et al., 2020; Caron et al., 2020; Grill et al., 2020; Zbontar et al., 2021; Chen and He, 2021; Bardes et al., 2022). Most similar to our work is SimCLR (Chen et al., 2020b), which uses a diverse set of augmentation as positive contrastive pairs. In SimCLR, however, the procedure to obtain the best performing augmentation distribution was not clearly documented. Further, no previous work has investigated whether such an idea would work in the language domain. Our work provides a parallel investigation in NLP, accessing the usefulness of diverse augmentations in improving sentence representations. We also propose methodical procedures and heuristics on how such set of augmentations can be obtained given an end task.

2.2 Sentence Embedding

Building a general purpose sentence embedding model is useful for many tasks (Wang et al., 2021a; Izacard et al., 2021; Gao and Callan, 2021; Gao et al., 2021; Chuang et al., 2022; Chang et al., 2021). SBERT (Reimers and Gurevych, 2019) pioneered the efforts to improve semantic similarities between sentence embeddings using a siamese network with BERT (Devlin et al., 2019). Fine-tuned with the natural language inference (NLI) dataset (Williams et al., 2018; Bowman et al., 2015), SBERT predicts whether a hypothesis sentence entails or contradicts the second sentence. To tackle anisotropicness of BERT embedding space (Ethayarajh, 2019), Li et al. (2020) and Su et al. (2021) learn projection layer which converts BERT embedding to a Gaussian or zero-mean fixed-variance space. Following contrastive learning literature in vision, few works investigate alternative positive and negatives: from using different layers (Zhang et al., 2020), different models (Carlsson et al., 2020), against frozen model (Carlsson et al., 2020), different parts of document (Giorgi et al., 2021), to next sentences (Neelakantan et al., 2022).

With simplicity in mind, unsupervised SimCSE (Gao et al., 2021) uses the same sentence with independent dropouts from transformers as positives and the rest of in-batch sentences as negatives, while supervised SimCSE uses NLI entailment sentence as positives, and contradiction as negatives. Lastly, the state-of-the-art method, DiffCSE (Chuang et al., 2022), proposes to add an additional discriminative loss similar to ones used in ELECTRA (Clark et al., 2019): the replaced token detection (RTD) loss to additionally increase the performance. The discriminator uses the original sentence embedding and a contextually perturbed sentence embedding to predict the token locations in which the two sentences differ. In contrast to DiffCSE, our discriminator predicts the augmentation type, a higher level task than predicting individual tokens. Additionally, our discriminator is in an antagonistic/adversarial relationship to our model, whereas the ELECTRA-like RTD objective is collaborative in nature.

2.3 NLP Augmentations

NLP augmentations are in more or less three flavors. Rule-based augmentations range from randomly deleting words, swap word orders (Wei and Zou, 2019), to more structurally-sounds, or semantically specific ones (Zhang et al., 2015; Logeswaran et al., 2018). These simple augmentations, however, have been found to be not particularly effective in higher resource domain for task-agnostic purposes (Longpre et al., 2020; Gao et al., 2021). The second kind of augmentations use pretrained language models (LM), to generate semantically similar examples. This area of work includes, but is not limited to back-translation (Li and Specia, 2019; Sugiyama and Yoshinaga, 2019), paraphrase models (Li et al., 2019, 2018; Iyyer et al., 2018), style transfer models (Fu et al., 2018; Krishna et al., 2020), contextually perturbed models (Morris et al., 2020; Jin et al., 2020), to large LM-base augmentation (Kumar et al., 2020; Yoo et al., 2021). Lastly, a few methods generate augmentations in the embedding space. These methods often perform interpolation (DeVries and Taylor, 2017; Chen et al., 2020a), noising (Kurata et al., 2016), and autoencoding (Schwartz et al., 2018; Kumar et al., 2019b) with embedded data points. However, due to the discreteness of NL (Bowman et al., 2016) and anisotropy (Ethayarajh, 2019), the introduced noise often outweighs the benefit of additional data.

Recently, NL-Augmenter (Dhole et al., 2021) collected over 100 augmentation methods, with the intention to provide robustness diagnostics for NLP models against different type of data perturbations[3]. In our work, we show that a diverse set of augmentations, even with simple rule-based augmentations, which are cheaper and more controllable than LM-based augmentations, can be used to learn robust general-purpose sentence embedding.

3 Motivation

3.1 Single augmentation is task specific

Augmentations, especially ones that exploit surface level semantics using simple rules, are task specific and have been used alone only if the augmentation aligns with the task objective for the dataset (Longpre et al., 2020). For instance, Dinan et al. (2020) changes gendered words in a sentence to instill gender invariance for bias mitigation. Inspired by hard negative augmentations in contrastive learning (Gao et al., 2021; Sinha et al., 2020), we use the following case studies to reinforce the conclusion from the perspective of negative data augmentation. In both scenarios, we use the negative augmentations ($\mathbf{h}_i^-$) loss (with positive examples $\mathbf{h}_i^+$) for contrastive objective (Gao et al., 2021):

$$-\log \frac{e^{\mathrm{sim}(\mathbf{h}_i,\mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}(\mathbf{h}_i,\mathbf{h}_i^+)/\tau} + e^{\mathrm{sim}(\mathbf{h}_i,\mathbf{h}_i^-)/\tau}} \qquad (1)$$

where sim is cosine similarity, τ is the temperature parameter controlling for the contrastive strength, and N is batch size. Since some augmentations do not have 100% perturbation rate, we remove datapoints that do not have a successful negative augmentation. For the remaining datapoints, we use original sentences as positives, and train with different augmentations as the negatives. In addition, we also present average transfer tasks (Conneau and Kiela, 2018) performance as a metric for embedding quality (**trans.**, detailed in Sec 5).

Augmentation	**CoLA**	**trans.**
$\mathrm{BERT}_{\mathrm{base}}$	75.93	84.66
Unsupervised $\mathrm{SimCSE}_{\mathrm{BERT}}$	71.91	**85.81**
RandomContextualWordAugmentation	**78.14**	80.51
SentenceSubjectObjectSwitch	76.80	80.31
Augmentation	**ANLI**	**trans.**
$\mathrm{BERT}_{\mathrm{base}}$	53.80	84.66
Unsupervised $\mathrm{SimCSE}_{\mathrm{BERT}}$	53.42	**85.81**
AntonymSubstitute	**58.78**	79.93
SentenceAdjectivesAntonymsSwitch	58.63	80.11

Table 1: Top negative augmentations for CoLA and ANLI, both measured in accuracy, with average transfer performance. See augmentation description in A.2

Case study 1: linguistic acceptability We first test embedding performance on CoLA (Warstadt et al., 2018), a binary sentence classification task predicting linguistically acceptability. If an augmentation frequently introduces grammatical errors, it should perform well as a negative.

Case study 2: contradiction vs. entailment Natural language inference (NLI) datasets (Bowman et al., 2015; Williams et al., 2018) provide triplets of sentences: an hypothesis, a sentence entailing, and a sentence in contradiction to the hypothesis. A good embedding should place the entailment sentence closer to the hypothesis than the contradiction sentence, and in fact, that is the exact hypothesis exploited by supervised SimCSE. We calculate the similarity between hypothesis and an entailment sentence and similarity between hypothesis and a contradiction sentence, and count how often is the former larger than the later in ANLI (Nie et al., 2020). If an augmentation can reverse the semantics of sentences, then it should perform well as a negative.

Insight: As expected (Table 1), augmentations known to introduce a lot of grammatical mistakes: RandomContextualWordAugmentation (Zang et al., 2020) performs the best in **CoLA** and those that reverse semantics: AntonymSubstitute, and SentenceAdjectivesAntonymsSwitch performs well in **ANLI**. However, single augmenta-

[3] https://github.com/GEM-benchmark/NL-Augmenter

Trial	STS-b
unsupervised SimCSE	81.18
supervised SimCSE	85.64
no contradiction	83.60
contradiction as pos	79.55
contradiction as pos, entailment as neg	67.16
supervised SimCSE w/ ANLI	75.99

Table 2: Alternative choices of positives and negatives with SimCSE. All results are reproduced by us.

tion significantly under-performs in **trans**fer tasks, reducing robustness. This suggests the need for diverse augmentations (Chen et al., 2020b; Ren et al., 2021).

3.2 Difficulty of selecting contrastive pairs

Gao et al. (2021) experimented with a combination of MNLI (Williams et al., 2018) and SNLI (Bowman et al., 2015) and found that using entailment as positives and contradictions as negatives performs well. In addition to this setting, we performed additional ablations to show that it is usually unclear which sentence pair dataset or augmentation would provide the best result as contrastive pairs (Table 2). Sometimes, non-intuitive pairs could yield decent results[4]. Together with the specificity of individual augmentations, this motivates for a general framework to select and combine multiple augmentations to achieve a robust, general-purpose embedding.

4 Methods

4.1 Augmentation Selection

Dhole et al. (2021) introduced 100+ augmentation methods. We also added non-duplicating augmentation methods from popular repositories: nlpaug, checklist, TextAugment, TextAttack, and TextAutoAugment (Ma 2019; Ribeiro et al. 2020; Marivate and Sefara 2020; Morris et al. 2020; Ren et al. 2021), including RandomDeletion, RandomSwap, RandomCrop, RandomWordAugmentation, RandomWordEmbAugmentation, and RandomContextualWordAugmentation[5].

To narrow down the augmentations we experiment with, we selected for single-sentence augmentations that are either labeled **highly meaning preserving**, **possible meaning alteration**, or **meaning alteration**. After preliminary filtering (Appendix A.3), Table 3 contains all augmentations we included in our experiments. To select for a diverse set of augmentation for main results in STS-b and transfer tasks, we trained models using single augmentation as positives, and pick augmentations that obtained top performance on STS-B and transfer tasks. For full single augmentation results see Appendix A.14.

4.2 Augmentation Sampling

To save computation and control for randomness, we augment the training dataset once for every augmentation and cache the results. Prior to training, augmentations are read from caches and uniformly sampled at each data point. Since not every augmentation perturbs the original sentence at every data point, we then correct augmentation label to "no augmentation" if the augmented sentence is the same as original sentence. This leads to a larger portion of the sentence having the label "no augmentation" than each individual augmentation[6].

4.3 Model Architecture

In our experiments, we train sentence embedding encoders using BERT- and RoBERTa-base for fair comparison to previous methods: SimCSE and DiffCSE. During training, we pass sentence representations through 2-layer projection layer with batchnorm, introduced by DiffCSE. We remove projection layers during inference and obtain sentence embeddings directly from the encoder. Formally, we train with contrastive loss, shown in the equation at the top right of Figure 1. We refer to this contrastive loss as $\mathcal{L}_{contrastive}$. We use the embedding corresponding to **[CLS]** token as sentence embedding in all experiments.

Contrastive loss regularizes on individual data pair level, which is a very strict constraint to resolve distributional shifts that augmentations introduce. To train sentence encoders that are invariant with respect to the shifts between diverse augmentations, we introduce an antagonistic discriminator. We pass the concatenated embeddings of original and augmented sentences into the discriminator (code in Appendix A.5) trained with the $\mathcal{L}_{discriminator}$ loss, defined as binary cross entropy between predicted and actual augmentations:

$$-\frac{1}{K}\sum_{i=1}^{K} y_i \log(p(y_i)) + (1-y_i)\log(1-p(y_i)) \qquad (2)$$

[4] See more discussion on negation in deep learning in A.15

[5] SimCSE tried RandomDeletion, RandomCrop; DiffCSE tried RandomDeletion, RandomInsertion, and their RTD is based on RandomContextualWordAugmentation.

[6] We also tried resampling augmentations between each epochs and found that to underperform fixed sampling.

Meaning Alteration	Possible Meaning Alteration	Highly Meaning Preserving
SentenceAdjectivesAntonymsSwitch, **SentenceAuxiliaryNegationRemoval**, ReplaceHypernyms, ReplaceHyponyms, **SentenceSubjectObjectSwitch**, **CityNamesTransformation** AntonymSubstitute	**ColorTransformation**,Summarization, **DiverseParaphrase***,**SentenceReordering**, TenseTransformation*,RandomDeletion, RandomCrop, **RandomSwap***, **RandomWordAugmentation**, RandomWordEmbAugmentation, RandomContextualWordAugmentation	**YodaPerturbation**, **ContractionExpansions***, **DiscourseMarkerSubstitution**, **Casual2Formal**, **GenderSwap**, GeoNamesTransformation, **NumericToWord**, SynonymSubstitution

Table 3: Final subsets of augmentations included in experiments. Augmentations in 16-Aug experiments are **bolded**, 12-Aug experiments are underlined, 8-Aug experiments are colored orange and 4-Aug experiments marked with asterisks(*). For full descriptions of augmentations, see Appendix A.2.

where K is the number of augmentation types (plus "no augmentation"), and $p(y_i)$ is the probability of augmentation type i predicted by the discriminator. To encourage augmentation-invariant encoder, the first layer of the discriminator uses a gradient reversal layer (Ganin and Lempitsky 2015; Zhu et al. 2015; Ganin et al. 2016) (code in Appendix A.4) that allows the gradient to be multiplied with a negative multiplier α in backward pass such that while discriminator is trained to minimize discriminator loss, the encoder is trained to maximize the discriminator loss all in one pass. We find this simple scheme to work well without having to deal with the instability around training adversarial networks (Creswell et al. 2018; Clark et al. 2019).

Finally, the overall loss of our model (AugCSE):

$$\mathcal{L} = \mathcal{L}_{contrastive} + \lambda * \mathcal{L}_{discriminator} \quad (3)$$

where λ is a coefficient that tunes the strength of discriminator loss.

5 Experiments

5.1 Evaluation Datasets

For fair comparison, we use the same dataset SimCSE used: 1M sentences randomly selected from Wikipedia. After training, we use frozen embeddings to evaluate our method on 7 semantic textual similarity (STS) tasks and 7 (SentEval) transfer tasks (Conneau and Kiela, 2018). STS tasks include **STS 2012 - 2016** (Agirre et al., 2016), **STS-Benchmark** (Cer et al.), and **SICK-Relatedness** (Marelli et al., 2014). In STS tasks, Spearman correlation is calculated between model's embedding similarity of the pair of sentences against human ratings (1-5). Transfer tasks are single sentence classification tasks from SentEval including **MR** (Pang and Lee, 2005), **CR** (Hu and Liu, 2004), **MPQA** (Wiebe et al., 2005), **MRPC** (Dolan and Brockett, 2005), **TREC** (Voorhees and Tice, 2000), **SST-2** (Socher et al., 2013), and **SUBJ** (Pang and Lee, 2004). We follow the standard evaluation setup from (Conneau and Kiela, 2018), training a logistic regression classifier on top of frozen sentence embeddings. See Appendix A.6 for details on hyperparameter search.

5.2 Evaluation Baselines

We include several levels of baselines. From word-averaged Glove embedding (Pennington et al., 2014), to BERT$_{\text{base}}$, using both average pooling as well as [CLS] token. We include post processing methods, **BERT-flow** (Li et al., 2020), and **BERT-whitening** (Su et al., 2021), as well as other more recent contrastive sentence embeddings: **CT-BERT** (Carlsson et al., 2020), **SG-OPT** (Kim et al., 2021), **SimCSE** (Gao et al., 2021), **DiffCSE** (Chuang et al., 2022). We also report results from **DeCLUTER** (Giorgi et al., 2021) and (Neelakantan et al., 2022) (**cpt-text-S**) as a comparison for what larger model and larger training data size would benefit. More specifically, DeCLUTER mines positives from documents, and cpt-text-S uses next sentence as positives.

5.3 STS Results

We show STS test results in Table 4. AugCSE performs competitively against SOTA methods, with both BERT and RoBERTa. AugCSE also outperforms larger models trained with more data (DeCLUTR and cpt-text-s). We discuss this in Sec 7.

5.4 Transfer Tasks Results

We show transfer tasks test set results in Table 5. With BERT$_{\text{base}}$ AugCSE outperforms DiffCSE in average transfer score and improve 4 out of 7 SentEval tasks. In RoBERTa$_{\text{base}}$, we still see competitive performance. Here, larger models with more training data outperform existing methods.

Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
GloVe embeddings (avg.) ♣	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
BERT$_{\text{base}}$ (first-last avg.) ♢	39.70	59.38	49.67	66.03	66.19	53.87	62.06	56.70
BERT$_{\text{base}}$-flow ♢	58.40	67.10	60.85	75.16	71.22	68.66	64.47	66.55
BERT$_{\text{base}}$-whitening ♢	57.83	66.90	60.90	75.08	71.31	68.24	63.73	66.28
SG-OPT-BERT$_{\text{base}}$ †	66.84	80.13	71.23	81.56	77.17	77.23	68.16	74.62
Unsupervised SimCSE-BERT$_{\text{base}}$ ♢.	68.40	82.41	74.38	80.91	78.56	76.85	**72.23**	76.25
DiffCSE-BERT$_{\text{base}}$ ♡	**72.28**	**84.43**	**76.47**	**83.90**	**80.54**	**80.59**	71.23	**78.49**
* AugCSE-BERT$_{\text{base}}$	71.40	83.93	75.59	83.59	79.61	79.61	72.19	77.98
RoBERTa$_{\text{base}}$ (first-last avg.) ♢	40.88	58.74	49.07	65.63	61.48	58.55	61.63	56.57
RoBERTa$_{\text{base}}$-whitening ♢	46.99	63.24	57.23	71.36	68.99	61.36	62.91	61.73
Unsupervised SimCSE-RoBERTa$_{\text{base}}$ ♢	**70.16**	81.77	73.24	81.36	80.65	80.22	68.56	76.57
DiffCSE-RoBERTa$_{\text{base}}$ ♡	70.05	**83.43**	**75.49**	**82.81**	**82.12**	**82.38**	**71.19**	**78.21**
* AugCSE-RoBERTa$_{\text{base}}$	69.30	82.17	73.49	81.82	81.40	80.86	68.77	76.83
Larger Training Data / Model Size								
DeCLUTR-RoBERTa$_{\text{base}}$ ♢	52.41	75.19	65.52	77.12	78.63	72.41	68.62	69.99
CPT-text-S ♠	62.1	60.0	62.0	71.8	73.7	-	-	-

Table 4: STS Test Set Performance (Spearman's correlation) from different sentence embedding models. ♣: results from (Reimers and Gurevych, 2019). ♢: results from (Gao et al., 2021). †: results from (Kim et al., 2021). ♡: results from (Chuang et al., 2022). Best results are **bolded**, second best results are underlined

5.5 Discriminator Objective Variations

In addition to predicting the augmentation type (**AugCSE**), we vary the discriminative objectives in Table 6. With **bool**, the discriminator predicts whether the second sentence is augmented or not (since not every augmentation is guaranteed 100% perturbation rate). With **positive**, we use augmented sentence as positives in the contrastive loss as well as using their augmentation types in the discriminator loss. For this setting, we use a symmetric loss similar to one in CLIP (Radford et al., 2021) to boost performance because contrasting two different distributions from augmented and natural text benefits from a symmetric regularization. In **no discriminator**, we use augmented sentence as positives in the contrastive loss but do not use a discriminator, which is the most naive way of using augmentation in contrastive learning (as in SimCLR(Chen et al., 2020b)). Empirically, we found that using augmentations only for the discriminative objective (**AugCSE**) performs the best and improves transfer results significantly over **no discriminator**. To understand such phenomenon, we can think of the discriminative objective as a weaker form of regularization, where we enforce invariance on the augmentation distribution level, rather than on individual augmented sentence level. The weaker constraint tolerates more noise in augmentation while distributionally improves the embedding space. Intuitively it make sense because the "noises" we introduce with augmentations do not impact the semantics of each sentence equally (e.g. randomly dropping an article in a sentence changes the semantics much less than dropping a verb). However, with the discriminative objective we do encourage that such noise be tolerated on a distributional level. This subtle difference is analogous to works in AI fairness, where antagonistic discriminator optimizes for group fairness (Chouldechova and Roth, 2020), while contrastive learning optimizes for individual fairness (Dwork et al., 2012).

We also experiment with different values of the α in gradient reversal layer in Table 7. Since α is a constant multiplied to the gradient from the discriminator and applied to downstream encoder, changing $\alpha = -1$ to $\alpha = 1$ is equivalent to changing discriminator from being antagonistic (AugCSE) to being collaborative (similar to DiffCSE). The magnitude determines how antagonistic or collaborative the discriminator is. We can see that the discriminator being antagonistic is crucial for our model performance (more detailed explorations and visualizations of the impact of α and on the embedding space are shown in Fig. 4 and 5 in the Appendix).

5.6 Augmentation ablation

We also vary the number of augmentation to determine the importance of diversity of augmentation for performance. For improving STS performance, we found 8 augmentations (Table 8) to be a sweet spot between including as diverse set of augmentations and keeping the augmentations relevant to the task. We see that including additional augmenta-

Model	MR	CR	SUBJ	MPQA	SST	TREC	MRPC	Avg.
GloVe embeddings (avg.) ♣	77.25	78.30	91.17	87.85	80.18	83.00	72.87	81.52
Avg. BERT embeddings ♣	78.66	86.25	94.37	88.66	84.40	**92.80**	69.54	84.94
BERT-[CLS]embedding ♣	78.68	84.85	94.21	88.23	84.13	91.40	71.13	84.66
SimCSE-BERT$_{\text{base}}$ ♢	81.18	86.46	94.45	88.88	85.50	89.80	74.43	85.81
w/ MLM	82.92	87.23	**95.71**	88.73	86.81	87.01	**78.07**	86.64
DiffCSE-BERT$_{\text{base}}$ ♡	82.69	87.23	95.23	89.28	86.60	90.40	76.58	86.86
* AugCSE-BERT$_{\text{base}}$	**82.88**	**88.19**	95.40	**89.43**	**87.15**	91.40	75.07	**87.07**
SimCSE-RoBERTa$_{\text{base}}$ ♢	81.04	87.74	93.28	86.94	86.60	84.60	73.68	84.84
w/ MLM	**83.37**	87.76	**95.05**	87.16	**89.02**	**90.80**	75.13	86.90
DiffCSE-RoBERTa$_{\text{base}}$ ♡	82.82	**88.61**	94.32	**87.71**	88.63	90.40	**76.81**	**87.04**
* AugCSE-RoBERTa$_{\text{base}}$	82.82	88.48	93.72	87.40	86.82	88.80	75.88	86.27
Larger Training Data / Model Size								
DeCLUTR-RoBERTa$_{\text{base}}$ †	85.16	90.68	95.78	88.52	90.01	93.20	74.61	88.28
CPT-text-S ♠	87.1	90.1	94.9	88.3	91.8	95.2	71.6	88.4

Table 5: SentEval Test Set Performance (accuracy) from different sentence embedding models. ♣: results from (Reimers and Gurevych, 2019). ♢: results from (Gao et al., 2021). †: results from (Giorgi et al., 2021). ♡: results from (Chuang et al., 2022). DeCLUTR was finetuned on 500K documents ♠: results from (Neelakantan et al., 2022). CPT-text-S models has 300M parameters and is trained on "Internet data".

discriminator	STS-b	Transfer
AugCSE	**85.25**	**85.80**
bool	84.52	85.44
positive	84.54	85.78
no discriminator	84.91	85.25

Table 6: Dev performance varying discriminator types.

α	STS-b	Transfer
100	60.47	85.68
10	72.33	85.67
1	80.85	85.78
-1 (AugCSE)	**85.25**	**85.80**
-10	84.68	85.68
-100	80.54	85.67

Table 7: Dev performance with various α values.

tion (16) can help further improve transfer results, but we use 8 augmentations in our main results for its simplicity. It is possible that we can improve our results further by including more diverse set of augmentations, we leave that for future studies.

5.7 Pretrained model based augmentation

LM-enabled augmentations could, in theory, beat the combination of all other augmentations by generating a diverse set of paraphrases using linguistic priors from training data. In 8, 12, and 16 augmentation setting, only **DiverseParaphrase** and **Casual2Formal** augmentations use pretrained model. To see how crucial LM-based augmentations are to our performance, we remove these augmentations and compare results with original settings. Without LM-based augmentations, we still see comparable results as before (Table 9). STS results actually **improve** across all trials.

Trial	STS-b	Transfer
4-Aug	84.97	85.79
8-Aug	**85.25**	85.80
12-Aug	84.63	85.73
16-Aug	84.83	**85.92**

Table 8: Ablation varying augmentations size.

Trial	STS-b (Δ)	Transfer (Δ)
8-2-Aug	85.31 (+0.06)	85.74 (-0.06)
12-2-Aug	84.83 (+0.20)	85.83 (+0.10)
16-2-Aug	84.84 (+0.01)	85.78 (-0.14)

Table 9: Performance after removing LM-based augmentations. Colored numbers indicate deltas compared to augmentation sets that include LM-based augs.

6 Analysis and Discussion

In our experiments, we selected subsets of top performing augmentations by looking at their individual finetuned performances. Such selection procedure may not be feasible due to resource constraints. In the following sections and in App. A.13, we discuss a few metrics that could be used to provide some signal in selecting the best augmentation (or dataset) for contrastive learning. We also discuss the broader impact of our work, advantages, and yet unresolved problems in the field.

6.1 Similarity and perplexity

One simple way of measuring point-wise distance between original and augmented sentences is using semantic similarity (approximated with cosine similarity between their SBERT embeddings[7]) and perplexity difference (calculated with GPT2 (Sanh et al., 2019)). Across all augmentations, similarities have positive correlation with STS-b and Transfer performance (Pearson correlation coefficients of 0.72 and 0.6, resp.) while perplexities difference have negative correlation with STS-b and Transfer performance (coefficients of -0.53 and -0.58, resp.) when augmentations are used as positives. This indicates that augmented sentences with higher similarities and lower perplexities differences to the originals may be useful as positive examples in contrastive learning. For more results and correlation with other metrics such as embedding isomorphism, see Appendix A.13 and A.14.

6.2 Domain shift in augmentation

In Figure 2 in the Appendix, we visualize the embedding distribution of sampled sentences pre- and post- augmentations, of pretrained BERT and AugCSE$_{\text{BERT}}$. We observe that augmentations do introduce distributional shift and that our discriminator can indeed unify distributions from diverse augmentations, along with evidence that α also impact unification (Figure 4 in the Appendix).

6.3 LM-based vs. rule-based augmentations

In our experiments, we observe that our model (AugCSE) performance does not depend on LM-based augmentations. AugCSE performance matches that of DiffCSE (that uses solely LM-based augmentation) and in many cases, removing LM-based augmentations even improves its performance (Table 9). This is an added advantage given that LM-based augmentations may be more expensive to run, are not as controllable as rule-based augmentations, and may contain bias learned from text in the wild that can reinforce undesirable properties in the sentence embedding. In comparison, rule-based models can precisely control for such behaviors, mitigate bias (Dinan et al., 2020), or introduce invariance in embedding space specific to the needs of the downstream tasks.

7 Conclusion

We present AugCSE, a general framework that combines diverse sets of augmentations to improve general sentence embeddings. In addition to the contrastive loss, we introduce an antagonistic discriminator that loosely constrain the model to become invariant to distributional shifts created from augmentations. In addition to outperforming previous methods, our framework is much more controllable, which has an added advantage of being able to mitigate undesirable properties from pretrained LMs, which inherit bias and toxicity from training data on the internet. Additionally, AugCSE can work with cheaper augmentations to run, resulting in a more resource-friendly approach to training generic sentence embedding models.

Limitations

Semantic textual similarity for evaluation. Sentence embedding literature has focused primarily on evaluating models using sentence semantic similarity tasks and SentEval transfer tasks. While transfer tasks may capture a wider range of desirable properties for a generic sentence embedding model, STS is often not a perfect indicator of sentence embedding quality. As noted by Neelakantan et al. (2022), STS tasks performance decreases as transfer task performance increases. This trend can also be observed in other robust models such as DeCLUTR. In future studies, we urge users to use STS tasks as only a subset of the transfer tasks when evaluating sentence embedding.

However, sentence semantic is still an important and difficult task that is not yet solved especially when considering the recursive structure, compositionality, and logics in sentences. In order to include the above more formally defined properties, additional data augmentation (Andreas, 2020; Akyürek et al., 2020) or architectural (Akyürek and Andreas, 2021) techniques may be needed.

Dense retrieval models and evaluations. Another downstream task relevant to sentence embedding is dense retrieval. Given sentences or documents, dense retrieval task aims to find the most relevant pairs within a corpus (Wang et al. 2021b,a; Thakur et al. 2021; Izacard et al. 2021; Liu and Shao 2022). Due to the way retrieval tasks are defined, models are trained with different data (Book Corpus, English Wikipedia (Gao and Callan 2021; Zhu et al. 2015)) and the objective encourages high scores given positive pairs, while (our) sentence embedding objective focuses on differentiating sentence semantics. Due to this subtle difference and project scope, we do not evaluate directly on retrieval tasks, and focus on comparing to previous works in the sentence embedding space.

[7] sentence-transformers/all-mpnet-base-v2

Choice of backbone models. We recognize that there have been many pretrained language models that have out-performed BERT. We used BERT and RoBERTa to make our evaluation comparable to previous works. Finetuning on additional models could lead to insights in trade-offs between pretraining objectives, data size and contrastive finetuning. We leave that for future studies.

Training data size and contrastive finetuning. Our method is able to produce SOTA results given a small fine-tuning dataset. However, we were unable to beat other methods that were trained/fine-tuned on much larger datasets. It is important to note, that Giorgi et al. (2021) reported RoBERTa$_{\text{base}}$ to score 87.31 on average transfer results. This indicates that finetuning RoBERTa with contrastive objective on wiki1m **reduces** the transfer performance (for SimCSE, DiffCSE, and AugCSE). One potential explanation for such behavior is that RoBERTa is trained on a much larger dataset with carefully designed next-sentence prediction objective, and has learned a robust sentence embedding already (given cpt-text-S was finetuned solely based on signals between neighboring sentences).

Language in concern During our study we limited our exploration to English only for better comparison to previous works. However, NLAugmentor does provide many augmentations that are focused on non-English, or multiple languages (which we filtered out for the scope of our project and training dataset). Nonetheless, our results could be extended to improving multi-lingual sentence embedding representations given the right training data and augmentation that can improve downstream multilingual tasks such as multilingual semantic textual similarity (Cer et al.), parallel corpus mining, a similar task to dense retrieval tasks in multilingual corpora (Zweigenbaum et al. 2017, 2018; Artetxe and Schwenk 2019; Reimers and Gurevych 2020; Jones and Wijaya 2021; Feng et al. 2022), machine translation (MT) and MT Quality Estimate (MTQE) that predicts the quality of the output provided by an MT system at test time when no gold-standard human translation is available (Fomicheva et al., 2020; Kocyigit et al., 2022). In fact, one of the main domains in which we believe our methods could come into use is in low-resource languages. Previous works have typically used backtranslation (Sennrich et al., 2016) and comparable corpora (recent works such as Rasooli et al. 2021 and Kuwanto and Akyürek that also uses code-switch data pre-train their MT encoder) to augment training data in low resource languages MT. In addition, in these settings we can incorporate augmentations that are linguistically rooted (created by language experts) or multi-lingual in nature, to improve neural representations of languages that are not as available as English.

Acknowledgements

We thank Dima Krotov and Yoon Kim for the early ideations of the project and Yoon Kim for helpful feedback before the final submission. We thank Dina Bashkirova, Andrea Burns, and Feyza Akyürek for guidance on experimentation methods. We thank Boston University for providing the computational resource and all anonymous reviewers for their insightful comments in the reviewing process. Zilu is supported by Lu Lingzi Scholarship from Boston University. This work is also supported in part by the U.S. NSF grant 1838193 and DARPA HR001118S0044 (the LwLL program). The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes. The views and conclusions contained in this publication are those of the authors and should not be interpreted as representing official policies or endorsements of DARPA and the U.S. Government.

References

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez Agirre, Rada Mihalcea, German Rigau Claramunt, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *SemEval-2016. 10th International Workshop on Semantic Evaluation; 2016 Jun 16-17; San Diego, CA. Stroudsburg (PA): ACL; 2016. p. 497-511.* ACL (Association for Computational Linguistics).

Ekin Akyürek, Afra Feyza Akyürek, and Jacob Andreas. 2020. Learning to recombine and resample data for compositional generalization. In *International Conference on Learning Representations*.

Ekin Akyürek and Jacob Andreas. 2021. Lexicon learning for few-shot neural sequence modeling. *arXiv preprint arXiv:2106.03993*.

II Alvin Grissom and Yusuke Miyao. 2012. Annotating factive verbs. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey*. Citeseer.

Jacob Andreas. 2020. Good-enough compositional data augmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7556–7566.

Mikel Artetxe and Holger Schwenk. 2019. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Adrien Bardes, Jean Ponce, and Yann Lecun. 2022. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR 2022-10th International Conference on Learning Representations*.

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.

Samuel Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21.

Fredrik Carlsson, Amaru Cuba Gyllensten, Evangelia Gogoulou, Erik Ylipää Hellqvist, and Magnus Sahlgren. 2020. Semantic re-tuning with contrastive tension. In *International Conference on Learning Representations*.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924.

D Cer, M Diab, E Agirre, I Lopez-Gazpio, and L Specia. Semeval-2017 task 1: semantic textual similarity-multilingual and cross-lingual focused evaluation. Association for Computational Linguistics.

Li-Hsin Chang, Iiro Rastas, Sampo Pyysalo, and Filip Ginter. 2021. Deep learning for sentence clustering in essay grading support. *arXiv preprint arXiv:2104.11556*.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020a. Mixtext: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020b. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758.

Alexandra Chouldechova and Aaron Roth. 2020. A snapshot of the frontiers of fairness in machine learning. *Communications of the ACM*, 63(5):82–89.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljačić, Shang-Wen Li, Wen-tau Yih, Yoon Kim, and James Glass. 2022. Diffcse: Difference-based contrastive learning for sentence embeddings. *arXiv preprint arXiv:2204.10298*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

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)*.

Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A. Bharath. 2018. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

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.

Terrance DeVries and Graham W Taylor. 2017. Dataset augmentation in feature space. *arXiv preprint arXiv:1702.05538*.

Kaustubh D Dhole, Varun Gangal, Sebastian Gehrmann, Aadesh Gupta, Zhenhao Li, Saad Mahamood, Abinaya Mahendiran, Simon Mille, Ashish Srivastava, Samson Tan, et al. 2021. Nl-augmenter: A framework for task-sensitive natural language augmentation. *arXiv preprint arXiv:2112.02721*.

Emily Dinan, Angela Fan, Adina Williams, Jack Urbanek, Douwe Kiela, and Jason Weston. 2020. Queens are powerful too: Mitigating gender bias in dialogue generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8173–8188, Online. Association for Computational Linguistics.

Bill Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005)*.

Haim Dubossarsky, Ivan Vulić, Roi Reichart, and Anna Korhonen. 2020. The secret is in the spectra: Predicting cross-lingual task performance with spectral similarity measures. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2377–2390.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226.

Steffen Eger, Gözde Gül Şahin, Andreas Rücklé, Ji-Ung Lee, Claudia Schulz, Mohsen Mesgar, Krishnkant Swarnkar, Edwin Simpson, and Iryna Gurevych. 2019. Text processing like humans do: Visually attacking and shielding nlp systems. 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 1634–1647.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. 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 55–65.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic bert sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Varun Gangal, Steven Y Feng, Malihe Alikhani, Teruko Mitamura, and Eduard Hovy. 2021. Nareor: The narrative reordering problem. *arXiv preprint arXiv:2104.06669*.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030.

Luyu Gao and Jamie Callan. 2021. Condenser: a pre-training architecture for dense retrieval. *arXiv preprint arXiv:2104.08253*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*.

John Giorgi, Osvald Nitski, Bo Wang, and Gary Bader. 2021. Declutr: Deep contrastive learning for unsupervised textual representations. 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)*, pages 879–895.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326, Dublin, Ireland. Association for Computational Linguistics.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

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*, pages 168–177.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Towards unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025.

Alexander Jones, William Yang Wang, and Kyle Mahowald. 2021. A massively multilingual analysis of cross-linguality in shared embedding space. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5833–5847, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alexander Jones and Derry Tanti Wijaya. 2021. Majority voting with bidirectional pre-translation for bitext retrieval. In *Proceedings of the 14th Workshop on Building and Using Comparable Corpora (BUCC 2021)*, pages 46–59.

Taeuk Kim, Kang Min Yoo, and Sang-goo Lee. 2021. Self-guided contrastive learning for bert sentence representations. 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)*, pages 2528–2540.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Muhammed Kocyigit, Jiho Lee, and Derry Wijaya. 2022. Better quality estimation for low resource corpus mining. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 533–543.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 737–762.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019a. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. 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 3609–3619, Minneapolis, Minnesota. Association for Computational Linguistics.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. In *Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems*, pages 18–26.

Varun Kumar, Hadrien Glaude, Cyprien de Lichy, and Wlliam Campbell. 2019b. A closer look at feature space data augmentation for few-shot intent classification. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 1–10.

Gakuto Kurata, Bing Xiang, and Bowen Zhou. 2016. Labeled data generation with encoder-decoder lstm for semantic slot filling. In *INTERSPEECH*, pages 725–729.

Garry Kuwanto and Afra Feyza Akyürek. Isidora chara tourni, siyang li, and derry wijaya. 2021. low-resource machine translation for low-resource languages: Leveraging comparable data, codeswitching and compute resources. *arXiv preprint arXiv:2103.13272*.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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)*, pages 9119–9130.

Zhenhao Li and Lucia Specia. 2019. Improving neural machine translation robustness via data augmentation: Beyond back-translation. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 328–336, Hong Kong, China. Association for Computational Linguistics.

Zichao Li, Xin Jiang, Lifeng Shang, and Hang Li. 2018. Paraphrase generation with deep reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3865–3878.

Zichao Li, Xin Jiang, Lifeng Shang, and Qun Liu. 2019. Decomposable neural paraphrase generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3403–3414.

Zheng Liu and Yingxia Shao. 2022. Retromae: Pretraining retrieval-oriented transformers via masked auto-encoder. *arXiv preprint arXiv:2205.12035*.

Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. 2018. Content preserving text generation with attribute controls. *Advances in Neural Information Processing Systems*, 31.

Shayne Longpre, Yu Wang, and Chris DuBois. 2020. How effective is task-agnostic data augmentation for pretrained transformers? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4401–4411.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. 2020. Gender bias in neural natural language processing. In *Logic, Language, and Security*, pages 189–202. Springer.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A sick cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223.

Vukosi Marivate and Tshephisho Sefara. 2020. Improving short text classification through global augmentation methods. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 385–399. Springer.

George A Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al. 2022. Text and code embeddings by contrastive pretraining. *arXiv preprint arXiv:2201.10005*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901.

V Păiș. 2019. *Contributions to semantic processing of texts; Identification of entities and relations between textual units; Case study on Romanian language*. Ph.D. thesis, PhD Thesis, Romanian Academy.

Bo Pang and Lillian Lee. 2004. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 271–es.

Bo Pang and Lillian Lee. 2005. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 115–124.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Emily Pitler, Mridhula Raghupathy, Hena Mehta, Ani Nenkova, Alan Lee, and Aravind Joshi. 2008. Easily identifiable discourse relations. In *Coling 2008: Companion volume: Posters*, pages 87–90, Manchester, UK. Coling 2008 Organizing Committee.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The penn discourse treebank 2.0. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Mohammad Sadegh Rasooli, Chris Callison-Burch, and Derry Tanti Wijaya. 2021. "wikily" supervised neural translation tailored to cross-lingual tasks. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1655–1670.

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.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525.

Shuhuai Ren, Jinchao Zhang, Lei Li, Xu Sun, and Jie Zhou. 2021. Text autoaugment: Learning compositional augmentation policy for text classification.

In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9029–9043.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS* $EMC^2Workshop$.

Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogerio Feris, Raja Giryes, and Alex Bronstein. 2018. Delta-encoder: an effective sample synthesis method for few-shot object recognition. *Advances in Neural Information Processing Systems*, 31.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Abhishek Sinha, Kumar Ayush, Jiaming Song, Burak Uzkent, Hongxia Jin, and Stefano Ermon. 2020. Negative data augmentation. In *International Conference on Learning Representations*.

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*, pages 1631–1642.

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

Amane Sugiyama and Naoki Yoshinaga. 2019. Data augmentation using back-translation for context-aware neural machine translation. In *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pages 35–44, Hong Kong, China. Association for Computational Linguistics.

Theodore R Sumers, Mark K Ho, Robert D Hawkins, Karthik Narasimhan, and Thomas L Griffiths. 2021. Learning rewards from linguistic feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6002–6010.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2021. Augmented sbert: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 296–310.

Ashwin Vijayakumar, Michael Cogswell, Ramprasaath Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search for improved description of complex scenes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Ellen M Voorhees and Dawn M Tice. 2000. Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 200–207.

Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021a. Tsdae: Using transformer-based sequential denoising auto-encoderfor unsupervised sentence embedding learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 671–688.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2021b. Gpl: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Benchmarking generalization via in-context instructions on 1,600+ language tasks. *arXiv preprint arXiv:2204.07705*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2018. Neural network acceptability judgments. *arXiv preprint arXiv:1805.12471*.

Daniel M Wegner and David J Schneider. 2003. The white bear story. *Psychological Inquiry*, 14(3-4):326–329.

Jason Wei, Chengyu Huang, Soroush Vosoughi, Yu Cheng, and Shiqi Xu. 2021. Few-shot text classification with triplet networks, data augmentation, and curriculum learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5493–5500.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122.

Sam Wiseman and Alexander M. Rush. 2016. Sequence-to-sequence learning as beam-search optimization. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1296–1306, Austin, Texas. Association for Computational Linguistics.

Kang Min Yoo, Dongju Park, Jaewook Kang, Sang-Woo Lee, and Woomyoung Park. 2021. Gpt3mix: Leveraging large-scale language models for text augmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2225–2239.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *ACL*.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR.

Jing Zhang, Bonggun Shin, Jinho D Choi, and Joyce C Ho. 2021. Smat: An attention-based deep learning solution to the automation of schema matching. In *European Conference on Advances in Databases and Information Systems*, pages 260–274. Springer.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Yan Zhang, Ruidan He, Zuozhu Liu, Kwan Hui Lim, and Lidong Bing. 2020. An unsupervised sentence embedding method by mutual information maximization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1601–1610.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2017. Overview of the second BUCC shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 60–67, Vancouver, Canada. Association for Computational Linguistics.

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2018. Overview of the third bucc shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of 11th workshop on building and using comparable corpora*, pages 39–42.

A Appendix

A.1 Ethics Statement

To our best knowledge, there is no outstanding ethical issue with our method of approach other than including potentially problematic augmentations (stereotype-reaffirming, toxic, etc) into the augmentation set. In fact, we believe one of the main advantage of our methods over previous methods is we can use rule-based augmentations to explicitly control for the type of invariances we want to instill within the sentence embedding, as opposed to propagating bias, stereotypes, and toxicity that exist in natural text and pre-trained LMs. NL-Augmenter includes many rule-based augmentations that tackle exactly such biases against country of origin, gender, geolocation, linguistic patterns, etc.

When considering computing resources and environmental impact, rule-based methods are much cheaper and more accessible to run, making our method a much more desirable approach for low-resource compute settings.

A.2 All Augmentations Descriptions in Experiments

In this section, we **word-by-word copy over** the descriptions of each of the augmentations we have mentioned in our paper from NL-Augmenter (Dhole et al., 2021), unless otherwise **noted**.

SentenceAdjectivesAntonymsSwitch This transformation switches English adjectives in a sentence with their WordNet (Miller, 1998) antonyms to generate new sentences with possibly different meanings and can be useful for tasks like Paraphrase Detection, Paraphrase Generation, Semantic Similarity, and Recognizing Textual Entailment.

Example: Amanda's mother was very beautiful → ugly .

SentenceAuxiliaryNegationRemoval This is a low-coverage transformation which targets sentences that contain negations. It removes negations in English auxiliaries and attempts to generate new sentences with the opposite meaning.

Example: Ujjal Dev Dosanjh was not → Ujjal Dev Dosanjh was the 1st Premier of British Columbia from 1871 to 1872.

ReplaceHypernyms / ReplaceHyponyms This transformation replaces common nouns with other related words that are either hyponyms or hypernyms. Hyponyms of a word are more specific in meaning (such as a sub-class of the word), eg: 'spoon' is a hyponym of 'cutlery'. Hypernyms are related words with a broader meaning (such as a generic category /super-class of the word), eg: 'colour' is a hypernym of 'red'. Not every word will have a hypernym or hyponym.

SentenceSubjectObjectSwitch This transformation switches the subject and object of English sentences to generate new sentences with a very high surface similarity but very different meaning. This can be used, for example, for augmenting data for models that assess semantic similarity

CityNamesTransformation This transformation replaces instances of populous and well-known cities in Spanish and English sentences with instances of less populous and less well-known cities to help reveal demographic biases (Mishra et al., 2020) prevelant in named entity recognition models. The choice of cities have been taken from the World Cities Dataset. [8]

AntonymSubstitute This transformation introduces semantic diversity by replacing an even number of adjective/adverb in a given text. We assume that an even number of antonyms transforms will revert back sentence semantics; however, an odd number of transforms will revert the semantics. Thus, our transform only applies to the sentence that has an even number of revertible adjectives or adverbs.We called this mechanism double negation.

Example: Steve is able → unable to recommend movies that depicts the lives of beautiful → ugly minds.

Note: To increase perturbation rate, and since we discovered that negations in semantics do not change sentence embeddings as much, we modified the original augmentations behavior by changing only odd number of antonyms. Hence, this augmentation changed from "Highly meaning preserving" to "Meaning Alteration". However, after we found out it was very similar to SentenceAjectivesAntonymsSwitch, we did not include it in main experiments for overlapping augmentation.

ColorTransformation This transformation augments the input sentence by randomly replacing mentioned colors with different ones from the 147

[8] https://www.kaggle.com/datasets/juanmah/world-cities

extended color keywords specified by the World Wide Web Consortium (W3C). Some of the colors include “dark sea green”, “misty rose”, “burly wood”.

Example: Tom bought 3 apples, 1 orange → misty rose , and 4 bananas and paid $10.

Summarization This transformation compresses English sentences by extracting subjects, verbs, and objects of the sentence. It also retains any negations. For example, “Stillwater is not a 2010 American liveaction/animated dark fantasy adventure film” turns into “Stillwater !is film”. (Zhang et al., 2021) used a similar idea to this transformation.

DiverseParaphrase This transformation generates multiple paraphrases of a sentence by employing 4 candidate selection methods on top of a base set of backtranslation models. 1) DiPS (Kumar et al., 2019a) 2) Diverse Beam Search (Vijayakumar et al., 2018) 3) Beam Search (Wiseman and Rush, 2016) 4) Random. Unlike beam search which generally focusses on the top-k candidates, DiPS introduces a novel formulation of using submodular optimisation to focus on generating more diverse paraphrases and has been proven to be an effective data augmenter for tasks like intent recognition and paraphrase detection (Kumar et al., 2019a). Diverse Beam Search attempts to generate diverse sequences by employing a diversity promoting alternative to the classical beam search (Wiseman and Rush, 2016).

SentenceReordering This perturbation adds noise to all types of text sources (paragraph, document, etc.) by randomly shuffling the order of sentences in the input text (Lewis et al., 2020). Sentences are first partially decontextualized by resolving coreference (Lee et al., 2018). This transformation is limited to input text that has more than one sentence. There are still cases where coreference can not be enough for decontextualization. For example, there could be occurences of ellipsis as demonstrated by (Gangal et al., 2021) or events could be mentioned in a narrative style which makes it difficult to perform re-ordering or shuffling (Kočiskỳ et al., 2018) while keeping the context of the discourse intact.

TenseTransformation This transformation converts English sentences from one tense to the other, for example simple present to simple past. This transformation was introduced by (Logeswaran et al., 2018).

RandomDeletion This augmentation randomly deletes a proportion of the words (Wei and Zou, 2019) and was added by us into the library of augmentations. Implementation uses nlpAug (Ma, 2019).

RandomCrop This augmentation randomly deletes a continuous span of words and was added by us into the library of augmentations. Implementation uses nlpAug (Ma, 2019).

RandomSwap This augmentation randomly swaps a proportion of the words and was added by us into the library of augmentations. Implementation uses nlpAug (Ma, 2019).

RandomWordAugmentation This augmentation transforms input by uniformly randomly select an augmentation from RandomDeletion, RandomCrop, and RrandomSwap. Implementation uses nlpAug (Ma, 2019).

RandomWordEmbAugmentation This augmentation substitute words with similar words defined by Glove embedding (Pennington et al., 2014). Implementation uses nlpAug (Ma, 2019).

RandomContextualWordAugmentation This augmentation randomly masks and fills words with pretrained BERT models. Similar ideas are often used in adversarial word embedding literature (Morris et al., 2020). Implementation uses nlpAug (Ma, 2019).

YodaPerturbation This perturbation modifies sentences to flip the clauses such that it reads like "Yoda Speak". For example, "Much to learn, you still have". This form of construction is sometimes called "XSV", where "the “X” being a stand-in for whatever chunk of the sentence goes with the verb", and appears very rarely in English normally. The rarity of this construction in ordinary language makes it particularly well suited for NL augmentation and serves as a relatively easy but potentially powerful test of robustness.

ContractionExpansions This perturbation substitutes the text with popular expansions and contractions, e.g., “I’m” is changed to “I am”and vice versa. The list of commonly used contractions expansions and the implementation of perturbation has been taken from Checklist (Ribeiro et al., 2020).

Example: He often does n't → not come to school.

DiscourseMarkerSubstitution This perturbation replaces a discourse marker in a sentence by a semantically equivalent marker. Previous work has identified discourse markers that have low ambiguity (Pitler et al., 2008). This transformation uses the corpus analysis on PDTB 2.0 (Prasad et al., 2008) to identify discourse markers that are associated with a discourse relation with a chance of at least 0.5. Then, a marker is replaced with a different marker that is associated to the same semantic class.

Example: It has plunged 13% since → inasmuch as July to around 26 cents a pound. A year ago ethylene sold for 33 cents

Casual2Formal This transformation transfers the style of text from formal to informal and vice versa. It uses the implementation of Styleformer[9].

Example: What you upto → currently doing ?

GenderSwap This transformation introduces gender diversity to the given data. If used as data augmentation for training, the transformation might mitigate gender bias, as shown in (Dinan et al., 2020). It also might be used to create a gender-balanced evaluation dataset to expose the gender bias of pre-trained models. This transformation performs lexical substitution of the opposite gender. The list of gender pairs (shepherd <–> shepherdess) is taken from (Lu et al., 2020). Genderwise names used from (Ribeiro et al., 2020) are also randomly swapped.

GeoNamesTransformation This transformation augments the input sentence with information based on location entities (specifically cities and countries) available in the GeoNames database[10]. E.g., if a country name is found, the name of the country is appended with information about the country like its capital city, its neighbouring countries, its continent, etc. Some initial ideas of this nature were explored in (Păiș, 2019).

NumericToWord This transformation translates numbers in numeric form to their textual representations. This includes general numbers, long numbers, basic math characters, currency, date, time, phone numbers, etc.

SynonymSubstitution This perturbation randomly substitutes some words in an English text with their WordNet (Miller, 1998) synonyms (Wei and Zou, 2019).

PigLatin This transformation translates the original text into pig latin. Pig Latin is a well-known deterministic transformation of English words, and can be viewed as a cipher which can be deciphered by a human with relative ease. The resulting sentences are completely unlike examples typically used in LM training. As such, this augmentation change the input into inputs which are difficult for a LM to interpret, while being relatively easy for a human to interpret.

PhonemeSubstitution This transformation adds noise to a sentence by randomly converting words to their phonemes.This transformation adds noise to a sentence by randomly converting words to their phonemes. Grapheme-to-phoneme substitution is useful in NLP systems operating on speech. An example of grapheme to phoneme substitution is "permit" → P ER0 M IH1 T'.

VisualAttackLetter This perturbation replaces letters with visually similar, but different, letters. Every letter was embedded into 576-dimensions. The nearest neighbors are obtained through cosine distance. To obtain the embeddings the letter was resized into a 24x24 image, then flattened and scaled. This follows the Image Based Character Embedding (ICES) (Eger et al., 2019). The top neighbors from each letter are chosen. Some were removed by judgment (e.g. the nearest neighbors for 'v' are many variations of the letter 'y') which did not qualify from the image embedding (Eger et al., 2019).

BackTranslation This transformation translates a given English sentence into German and back to English.This transformation acts like a light paraphraser. Multiple variations can be easily created via changing parameters like the language as well as the translation models which are available in plenty. Backtranslation has been quite popular now and has been a quick way to augment examples (Li and Specia 2019, ; Sugiyama and Yoshinaga 2019).

MultilingualBackTranslation This transformation translates a given sentence from a given language into a pivot language and then back to the original language. This transformation is a simple paraphraser that works on 100 different languages.

[9] https://github.com/PrithivirajDamodaran/Styleformer

[10] http://download.geonames.org/export/dump

Back Translation has been quite popular now and has been a quick way to augment (Li and Specia 2019; Sugiyama and Yoshinaga 2019; Fan et al. 2021).

Example: Being honest → Honesty should be one of our most important character traits → characteristics

FactiveVerbTransformation This transformation adds noise to all types if text source (sentence, paragraph, etc.) by adding factive verbs based paraphrases (Alvin Grissom and Miyao, 2012) Example: Peter published a research paper → Peter acknowledged that he published a research paper.

A.3 Narrowing down augmentations

we first filter for single sentence operations for unsupervised settings. We then remove augmentations that do not represent typical text distributions (PigLatin), or perturb based on audio (PhonemeSubstitution) or visual (VisualAttackLetter) similarities. Since semantic similarities between augmented and original sentence is important to our objective, we categorize all augmentations according to meaning preservation label provided by NL-Augmenter: **highly meaning preserving**, **possible meaning alteration**, and **meaning alteration**. Given not all augmentations were labeled, we manually label missing augmentations. Lastly, we filter out similar methods and only keep one from every type of augmentation (MultilingualBackTranslation, BackTranslation, etc.), and keep only augmentations that have relatively high perturbation rates (> 0.2). We then manually look through augmentation examples to filter out augmentations that produce repetitive artifacts that can be exploited by contrastive learning scheme (FactiveVerbTransformation).

A.4 Code for Gradient Reversal Layer

```python
from torch.autograd import Function

class GradReverse(Function):

  @staticmethod
  def forward(ctx, x, lambd, **kwargs: None):
    ctx.lambd = lambd
    return x.view_as(x)

  @staticmethod
  def backward(ctx, *grad_output):
    return grad_output[0] * -ctx.lambd, None
```

[11]

A.5 Code for Discrimimnator MLP

```python
class ProjectionMLP(nn.Module):
  def __init__(self, hidden_size, alpha=1.0):
    super().__init__()
    in_dim = hidden_size
    middle_dim = hidden_size * 2
    out_dim = hidden_size
    self.net = nn.Sequential(
        nn.Dropout(p=0.2),
        nn.Linear(in_dim, middle_dim),
        nn.Tanh(),
        nn.Dropout(p=0.2),
        nn.Linear(middle_dim, out_dim),
        nn.Tanh(),
    )
    self.alpha = alpha

  def forward(self, x):
    x = GradReverse.apply(x, self.alpha)
    return self.net(x)
```

A.6 Hyperparameter Selection

For main STS and transfer results, we follow similar search strategy as SimCSE and DiffCSE. For either tasks, we search for best performing dev runs in the hyperparmeter ranges (STS-b dev performance for STS test results; average transfer dev for transfer test results), and use that hyperparaemter set as the best performing set. The hyperparameter search range include: $\lambda \in \{1e-5, 5e-5, 1e-4, 5e-4, 1e-3, 5e-3, 1e-2\}$, learning rate $\in \{5e-6, 7e-6, 1e-5, 2e-5, 3e-5, 5e-5\}$ and batch size is fixed to 128. After obtaining the best hyperparameter for the task, we run the same trial with seed $\in \{1, 11, 42, 68, 421\}$ to obtain standard deviation and average. In the main result, we report maximum of the 5 seeds. In A.8, we report average and variance of 5 trials.

For all ablation experiments, we use the best hyperparameter main results (STS and transfer tasks separately), and search with different λ only for the best dev results for each ablation trial, and report the dev performances.

A.7 Best Hyperparameter for Main Results

See Table 10 and 11

A.8 Main Result Variance

See Table 12

[11] Implementation borrowed from https://zhuanlan.zhihu.com/p/263827804

hyperparameter	$\text{BERT}_{\text{base}}$	$\text{RoBERTa}_{\text{base}}$
λ	5e-3	1e-4
learning rate	2e-5	2e-5

Table 10: Best hyperparameters for main STS-B results.

hyperparameter	$\text{BERT}_{\text{base}}$	$\text{RoBERTa}_{\text{base}}$
λ	1e-4	1e-2
learning rate	2e-5	7e-6

Table 11: Best hyperparameter for main SentEval transfer results.

A.9 Reproducibility

All of our models are trained and inferenced on a single NVIDIA V100 GPU per trial. Training a single model for one epoch takes from 40 min to 5 hours, depending on the frequency of evaluation.

A.10 Model Size

See Table 13

A.11 Augmentation Unification

In Figure 2, we see AugCSE indeed can unify the distribution from different augmentations compare to baseline BERT. In Figure 3, we can see that in addition to contrastive objective from SimCSE (and baseline BERT), AugCSE brings distributions of augmentations vs. unperturbed sentences even closer together.

A.12 Importance of Gradient Reverse Multiplier

As seen in both training plots (Table 5, Figure 4), a positive alpha value (collaborative discriminator) results in embeddings that are easily classified by augmentations, whereas negative alpha values (antagonistic discriminator) results in unified embedding that is harder to pick out augmentation type. We use sklearn PCA module for all PCA results, and Multcore-TSNE [12] for ann TSNE plots.

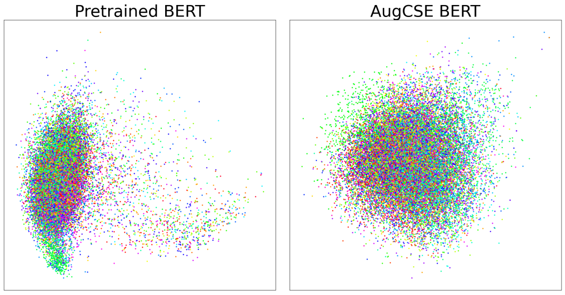


Figure 2: PCA of randomly sampled sentence embeddings from wiki1m dataset with various augmentations (27 augmentations) along with original sentence samples. Color indicates various augmentation types.

Mode	STS-b	Transfer
SimCSE /w MLM	76.25	86.64
DiffCSE	78.49	86.86
$\text{AugCSE}_{\text{BERT}}$	77.27 ± 0.63	86.74 ± 0.29
$\text{AugCSE}_{\text{RoBERTa}}$	75.54 ± 1.67	86.07 ± 0.21

Table 12: Main results with standard deviation

Model	Train	Inference
$\text{AugCSE}_{\text{BERT}}$	117M	110M
$\text{AugCSE}_{\text{RoBERTa}}$	132M	125M

Table 13: Model Sizes in our experiments

A.13 Embedding isomorphism

Different augmentations and datasets have been proposed as positive or negative pairs to learn sentence embedding. However, their performance differ drastically, despite many of them were created with the same original purpose, such as paraphrase. In search for what causes the difference in performance, we investigate further in NLI datasets, specifically ANLI (Nie et al., 2020), which was created with the same objective (entailment and contradiction) but with drastically different method. In ANLI, anchor sentences were provided, and entailment and contradictions were crowd-sourced for the purpose of fooling existing models. With such objective, sentences in contradiction and entailment may come from a different distribution as the anchor sentence.

We trained SimCSE using ANLI data only, and found ANLI-SimCSE to perform much worse than Supervised SimCSE (trained with MNLI and SNLI), even if we sample and adjust for dataset size difference (Table 14).

To measure some aspect of distributional shift in the embedding space, we used 3 embedding isomorphism measurements: harmonic mean of effective condition numbers **COND-HM**, singular value gap **SVG**, and Gromov-Hausdorff distance **GH** (Dubossarsky et al. 2020; Jones et al. 2021).

Seen in Table 15, for ANLI, entailment and con-

[12] https://github.com/DmitryUlyanov/Multicore-TSNE

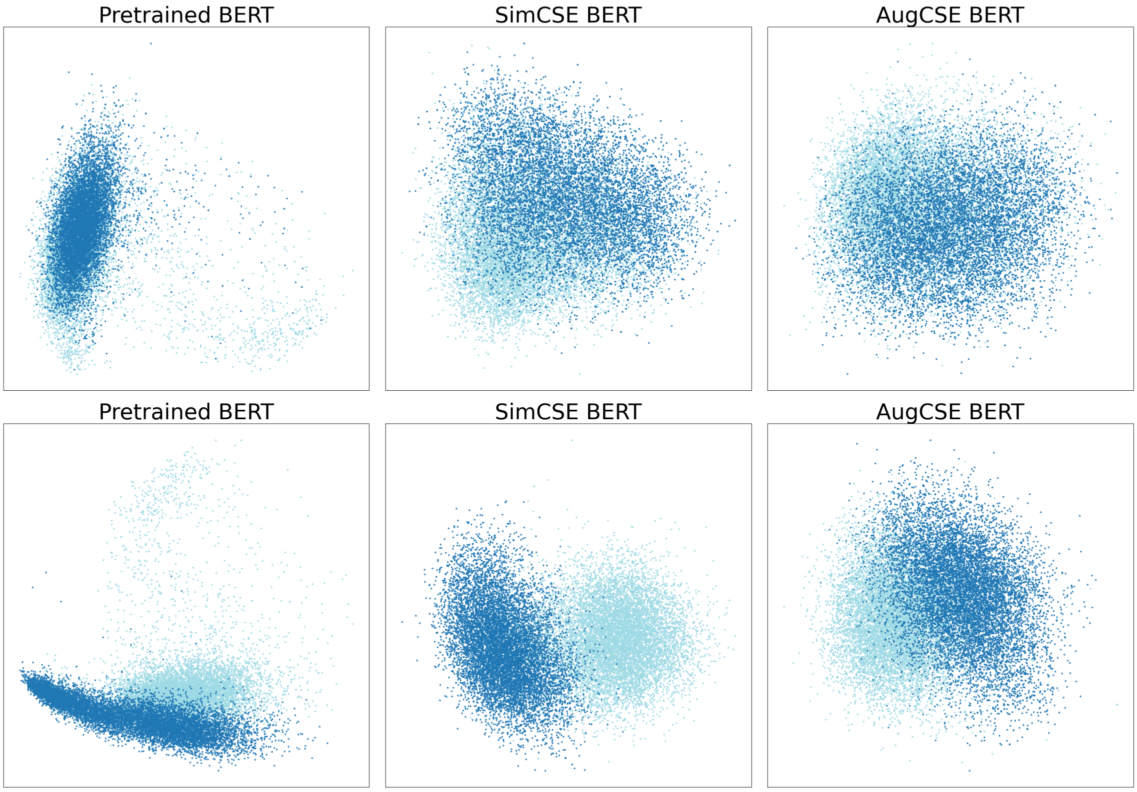


Figure 3: Embedding PCA plot with original sentences and augmented sentences. The augmentation in top row is SentenceAuxiliaryNegationRemoval, and in bottom row is Summarization

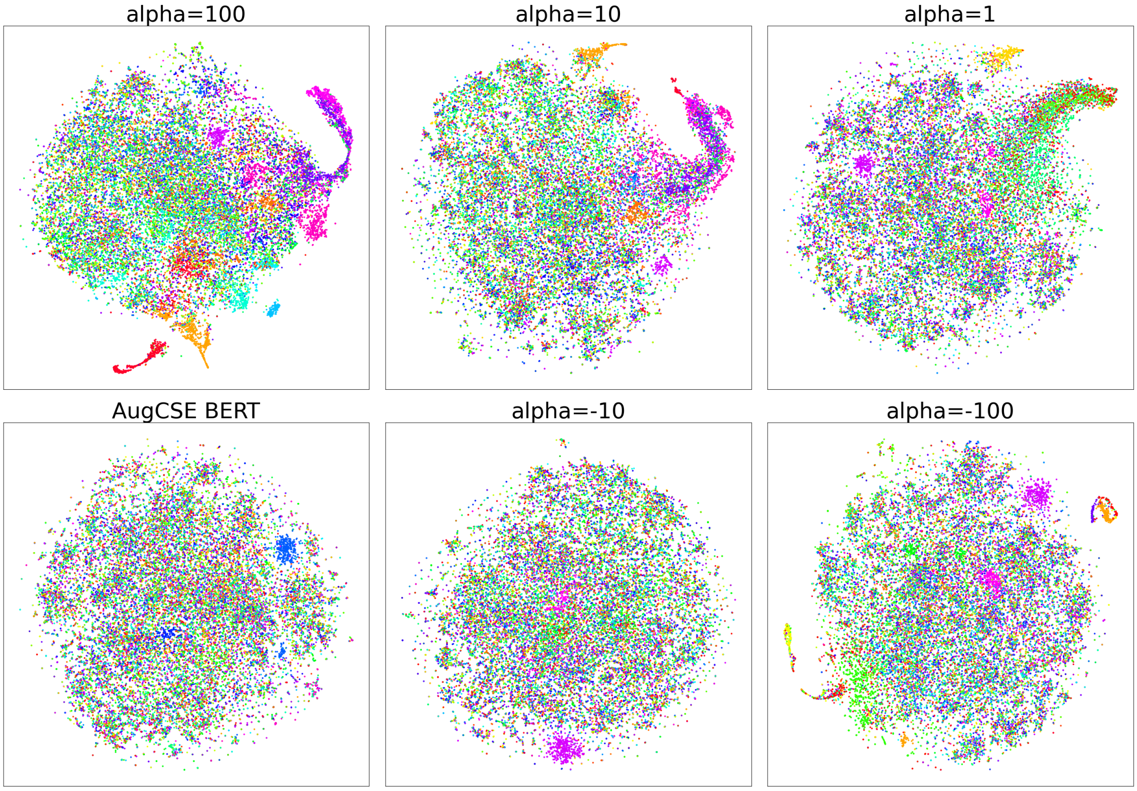


Figure 4: Embedding TSNE plot with different alphas. Colors indicate different augmentation types. Antagonistic discriminators (negative α) result in embedding spaces that are more invariant to augmentation types than collaborative discriminators (positive α).

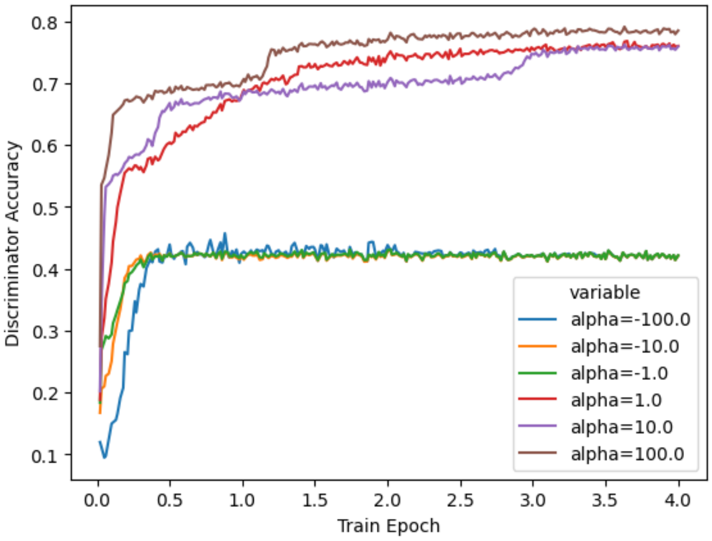


Figure 5: Discriminator accuracy over training with different alpha values.

Trial	STS-b
Unsupervised SimCSE	81.18
Supervised SimCSE	85.64
Supervised SimCSE (Sampled)	83.82
ANLI-SimCSE	75.99
ANLI-SimCSE w/o negatives	78.66

Table 14: Ablation experiments removing symmetric loss. All results are reproduced by us.

tradictions distributions were much more different from anchors than the that for NLI. We believe this difference could be one of the reason using ANLI examples do not work as well as NLI examples. In another word, in ANLI, perhaps because the embedding difference between contradiction and entailment sentences are so much smaller than both to anchor, that the contrasting signals from positives and negatives are conflicting rather than working together. This hypothesis can be confirmed with ANLI-SimCSE w/o negatives performing better than the trial with negatives.

In similar veins, we investigate whether the same measurement could be indicative of augmentation performance. However, were weren't able to find significant correlation. See the next section for more details.

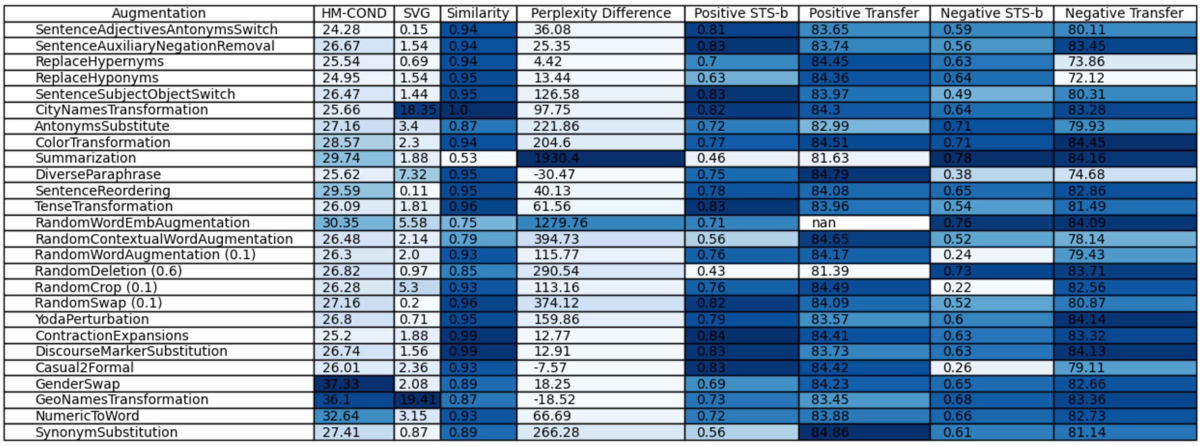
A.14 Single augmentation performance and embedding distance

For single augmentation experiments, we remove data points that are not transformed by the augmentation. We find this to work better than leaving some datapoints un-perturbed, which adds noise to the contrastive objective. In addition, we used symmetric contrastive loss similar to CLIP (Radford et al., 2021). This improves performance because augmentations introduce distributional shifts in the embedding space that benefits from a symmetric regularization.

Trial	A-E	A-C	E-C
MNLI + SNLI (sample)			
COND-HM	94.7	95.1	95.7
SVG	0.87	0.84	0.59
GD	0.31	0.29	0.05
ANLI			
COND-HM	96.0	95.7	91.54
SVG	0.86	0.82	0.29
GD	51.7	51.3	0.02

Table 15: Embedding isomorphism distance comparison between MNLI+SNLI to ANLI. A=Anchor, E=Entailment, C=Contradiction

In Figure 7, we can observe that similarity and perplexity difference are two measures most correlated feature with respect to all four metrics. Similarity is positively correlated with positive evaluations and perplexity difference is negatively correlated with positive evaluations. Both metrics relation with negative evaluations reverse directions but become much less strongly correlated. This is likely due to the nature of positive and negative aug-



Augmentation	HM-COND	SVG	Similarity	Perplexity Difference	Positive STS-b	Positive Transfer	Negative STS-b	Negative Transfer
SentenceAdjectivesAntonymsSwitch	24.28	0.15	0.94	36.08	0.81	83.65	0.59	80.11
SentenceAuxiliaryNegationRemoval	26.67	1.54	0.94	25.35	0.83	83.74	0.56	83.45
ReplaceHypernyms	25.54	0.69	0.94	4.42	0.7	84.45	0.63	73.86
ReplaceHyponyms	24.95	1.54	0.95	13.44	0.63	84.36	0.64	72.12
SentenceSubjectObjectSwitch	26.47	1.44	0.95	126.58	0.83	83.97	0.49	80.31
CityNamesTransformation	25.66	18.35	1.0	97.75	0.82	84.3	0.64	83.28
AntonymsSubstitute	27.16	3.4	0.87	221.86	0.72	82.99	0.71	79.93
ColorTransformation	28.57	2.3	0.94	204.6	0.77	84.51	0.71	84.45
Summarization	29.74	1.88	0.53	1930.4	0.46	81.63	0.78	84.16
DiverseParaphrase	25.62	7.32	0.95	-30.47	0.75	84.79	0.38	74.68
SentenceReordering	29.59	0.11	0.95	40.13	0.78	84.08	0.65	82.86
TenseTransformation	26.09	1.81	0.96	61.56	0.83	83.96	0.54	81.49
RandomWordEmbAugmentation	30.35	5.58	0.75	1279.76	0.71	nan	0.76	84.09
RandomContextualWordAugmentation	26.48	2.14	0.79	394.73	0.56	84.65	0.52	78.14
RandomWordAugmentation (0.1)	26.3	2.0	0.93	115.77	0.76	84.17	0.24	79.43
RandomDeletion (0.6)	26.82	0.97	0.85	290.54	0.43	81.39	0.73	83.71
RandomCrop (0.1)	26.28	5.3	0.93	113.16	0.76	84.49	0.22	82.56
RandomSwap (0.1)	27.16	0.2	0.96	374.12	0.82	84.09	0.52	80.87
YodaPerturbation	26.8	0.71	0.95	159.86	0.79	83.57	0.6	84.14
ContractionExpansions	25.2	1.88	0.99	12.77	0.84	84.41	0.63	83.32
DiscourseMarkerSubstitution	26.74	1.56	0.99	12.91	0.83	83.73	0.63	84.13
Casual2Formal	26.01	2.36	0.93	-7.57	0.83	84.42	0.26	79.11
GenderSwap	37.33	2.08	0.89	18.25	0.69	84.23	0.65	82.66
GeoNamesTransformation	36.1	19.41	0.87	-18.52	0.73	83.45	0.68	83.36
NumericToWord	32.64	3.15	0.93	66.69	0.72	83.88	0.66	82.73
SynonymSubstitution	27.41	0.87	0.89	266.28	0.56	84.86	0.61	81.14

Figure 6: Single augmentation as positive or negative pair in contrastive framework. No discriminator is used. When an augmentation is used as a negative augmentation, the corresponding positive augmentation is the original sentence itself with dropout (SimCSE). The float in parenthesis next to augmentation name indicates the rate of perturbation. **HM-COND**=harmonic mean of effective condition numbers between augmented and non-augmented sentence embedding samples. **SVG**=singular value gap between augmented and non-augmented sentence embedding samples. **Similarity**=cosine similarity of sentence embedding before and after augmentation. **Perplexity Difference**=perplexity of augmented sentence subtracted by perplexity of original sentence.

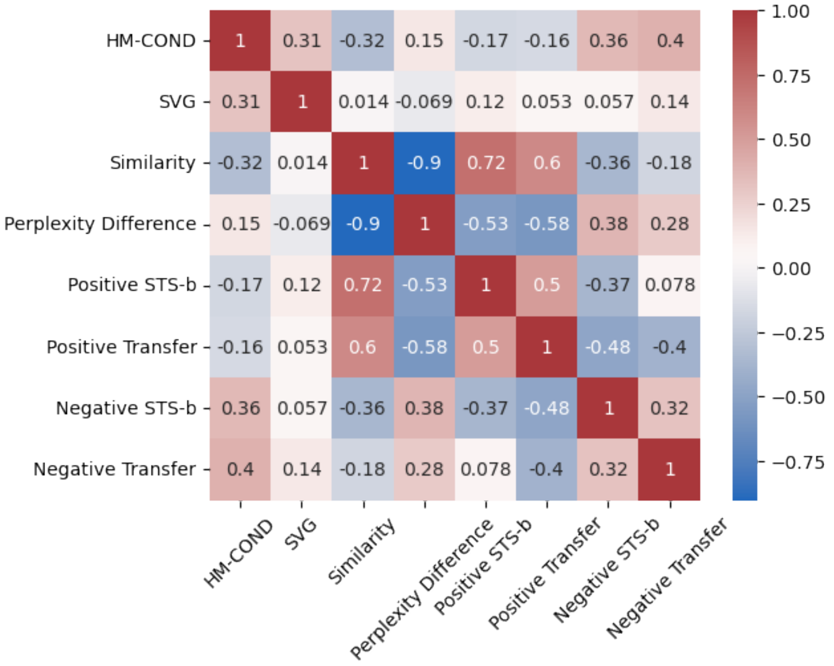


Figure 7: Pearson correlations between columns in Figure 6 across all single augmentation trials.

mentation usage in the contrastive objective. The negatives are aggregated along with rest of in-batch examples, lessen the effect. Additionally, the value of negatives is contextually dependent on positives, since the repulsion and attraction of negatives and positives conjointly defines the direction in which anchor embeddings go. HM-COND is also somewhat positively correlated with the with evaluation performance when using augmentation as negatives. It seems to suggest that the more isomorphic the embedding spaces are between augmented vs. original sentences, the better the augmentation is as a negative augmentation.

A.15 Negations in deep learning

As seen in Table 2, using contradiction as negatives obtains almost baseline performance, while being semantically entirely opposite. Similarly, in Appendix A.14, we have also observed that meaning preservation label (Table 3) has little indication of whether the augmentation performs well as a single positives. This is a particular interesting phenomenon that requires further study. While a sentence can represent semantically exactly opposite meaning, it is still discussing similar topics, and due to the symmetric nature of cosine similarity, it is difficult to use negation in deep learning. Negative examples do not help as much as in-context learning (Wang et al., 2022) or reinforcement learning rewards (Sumers et al., 2021), and negative natural language commands lead to exact opposite output from systems [13]. In toxicity NLP literature, this is related to the phenomenon that superficial textual token meanings are naively combined to yield sentence meaning, without taking to account of deeper structural relationships between entities mentioned (Hartvigsen et al., 2022). In the contrastive learning setting, providing a positive anchor (**SimCSE** in Table 2) helps direct the contrast to a specific direction against the positive examples, yet it is unclear how negatives can be used in other scenarios in deep learning. Such topic could also have interesting implications to "the white bear problem" (Wegner and Schneider, 2003), the phenomenon where "when someone is actively trying not to think of a white bear they may actually be more likely to imagine one." [14] in psychology, and whether failing to learn from negation in deep learning is a result of in-proper training methods or an indication that deep-learning models are aligned with human psychology, and to solve such problem may require human-centric strategies to deal with such short-comings.

[13] twitter.com/benjamin_hilton/status/1520469352008634373

[14] en.wikipedia.org/wiki/Ironic_process_theory