Contrastive Learning

Learn efficient representation through semantically close samples being pulled together and non-similar samples being pushed apart.

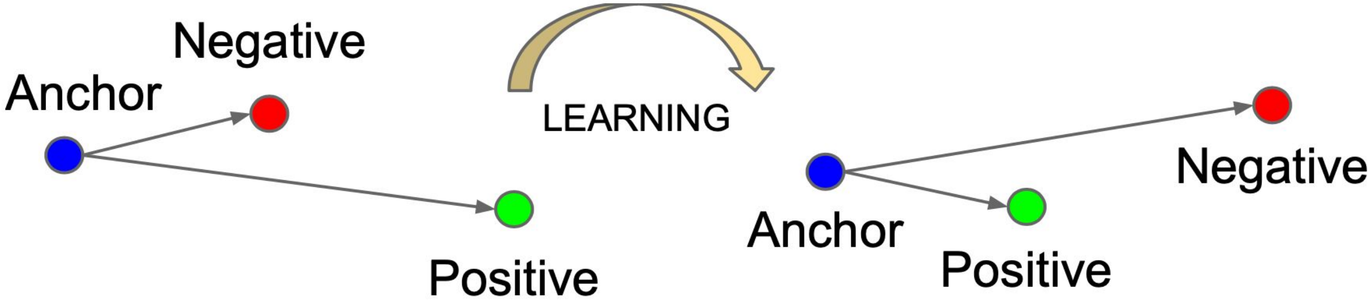


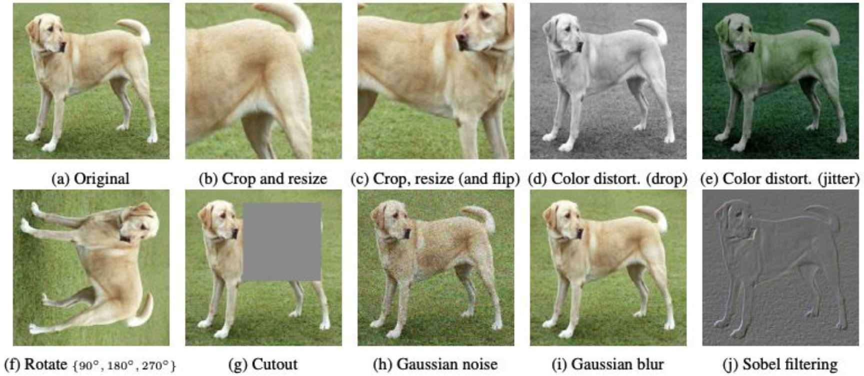
Image Source: Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

Contrastive Learning

Data Augmentation: Construct positive sample of an anchor, (x_i, x_i^+)

Unsupervised (a.k.a, self-supervised) Approach

Visual: Image Transformation



Language: Dropout Masks

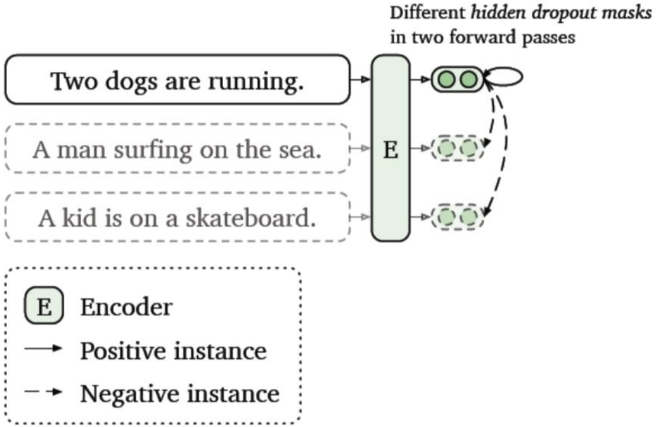


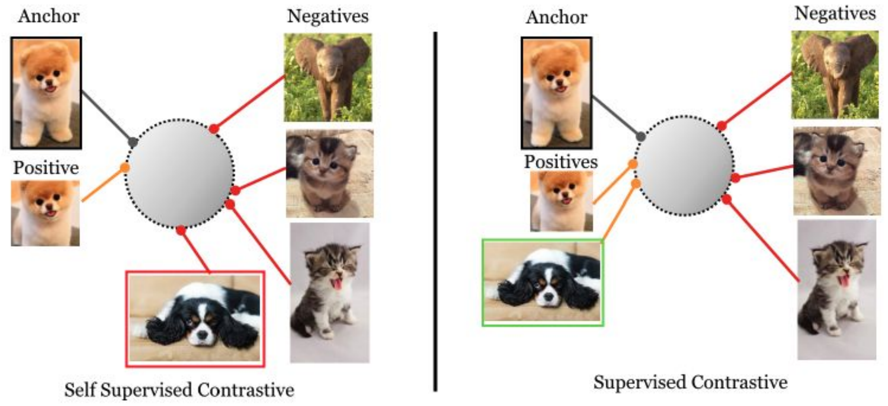
Image Source: Chen, Ting, et al. "A simple framework for contrastive learning of visual representations." *International conference on machine learning*. PMLR, 2020.

Image Source: Gao, Tianyu, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings." *arXiv preprint arXiv:2104.08821* (2021).

Contrastive Learning

Data Augmentation: Construct positive pairs by **supervised** data

Visual: ImageNet



Language: Sentiment Analysis

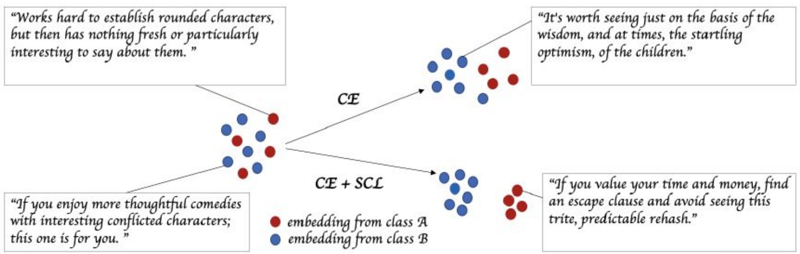


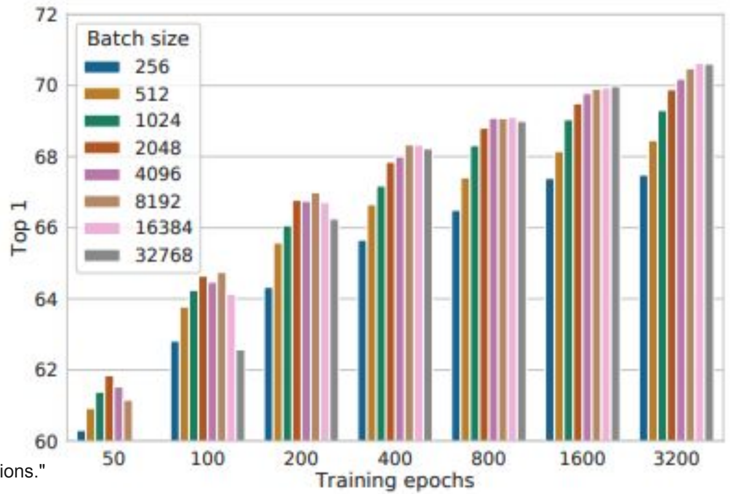
Image Source: Khosla, Prannay, et al. "Supervised Contrastive Learning." Advances in Neural Information Processing Systems 33 (2020).

Image Source: Gunel, Beliz, et al. "Supervised Contrastive Learning for Pre-trained Language Model Fine-tuning." International Conference on Learning Representations. 2020.

Contrastive Learning

- Large Batch Size:

  Most contrastive learning models rely on in-batch negatives. A large batch contains diverse negative samples and help the model distinguish different samples.



Linear evaluation (top-1) of ResNet-50 trained with different batch sizes and longer epochs

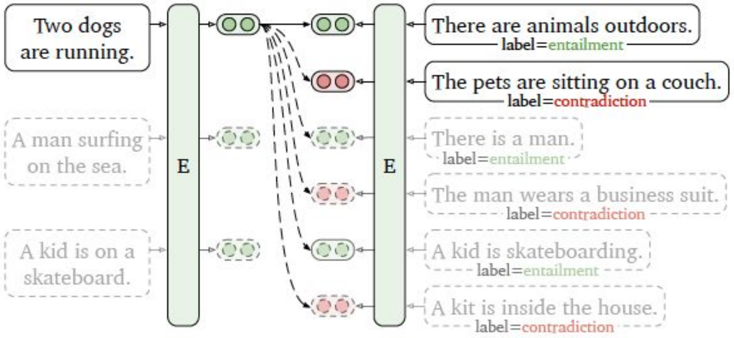
Image Source: Chen, Ting, et al. "A simple framework for contrastive learning of visual representations." *International conference on machine learning*. PMLR, 2020.

Contrastive Learning

- Hard Negative Mining: Identify task-specific hard negatives, (x_i, x_i^+, x_i^-)

  Hard negative samples refers to the samples that close to the anchor in the embedding space but have different labels from the anchor.

Visual

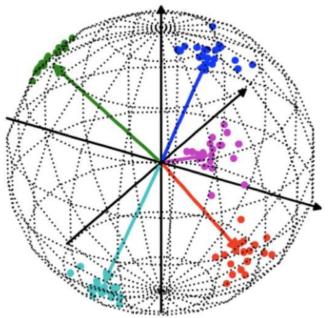


Language: Natural Language Inference

Image source:Robinson, Joshua David, et al. "Contrastive Learning with Hard Negative Samples." *International Conference on Learning Representations*. 2020.

Image Source: Gao, Tianyu, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings." *arXiv preprint arXiv:2104.08821* (2021).

Contrastive Learning



Two key properties to measure the quality of representation:

Alignment: Given a distribution of positive pairs p_{pos}, alignment calculates expected distance between embeddings of the paired instances. **Positive** instances should stay **close.**

$$\ell_{\text{align}} \triangleq \mathop{\mathbb{E}}_{(x,x^+)\sim p_{\text{pos}}} \|f(x) - f(x^+)\|^2.$$

Uniformity: Uniformity measures how well the embeddings are uniformly distributed. **Random** instances should **scatter** on the hypersphere.

$$\ell_{\text{uniform}} \triangleq \log \mathop{\mathbb{E}}_{x,y \overset{i.i.d.}{\sim} p_{\text{data}}} e^{-2\|f(x)-f(y)\|^2},$$

Image Source: https://zhuanlan.zhihu.com/p/389064413

Mirror-BERT

Fast, Effective, and Self-Supervised: Transforming Masked Language Models into Universal Lexical and Sentence Encoders

Fangyu Liu, Ivan Vulić, Anna Korhonen, Nigel Collier
Language Technology Lab, TAL, University of Cambridge
{fl399, iv250, alk23, nhc30}cam.ac.uk

Assumptions

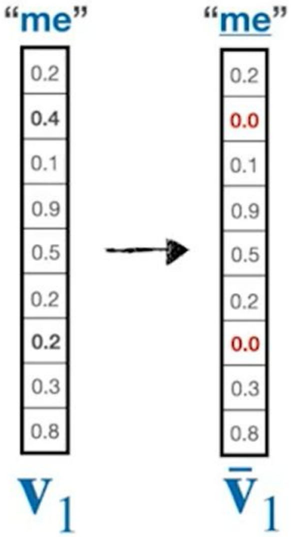
Assumption 1: *randomly masking a tiny part of a text should not change much of its semantics* (since human brains can usually reconstruct it based on context).

Econ[MASK]

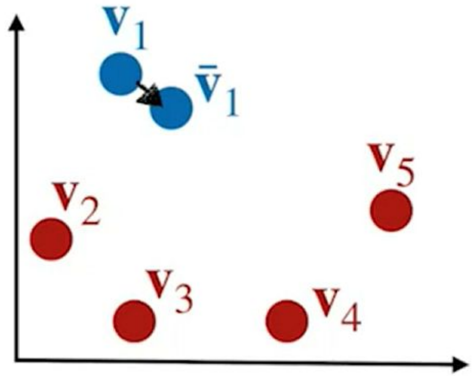
Econ[MASK] Paul Krugman mainly works on trade models.

Assumptions

Assumption 2: *In vectorised distributed text representations, erasing/changing a small set of elements of a vector should not change much of its semantics.*



When part(s) of "me" changed, am I still *me*?



Method

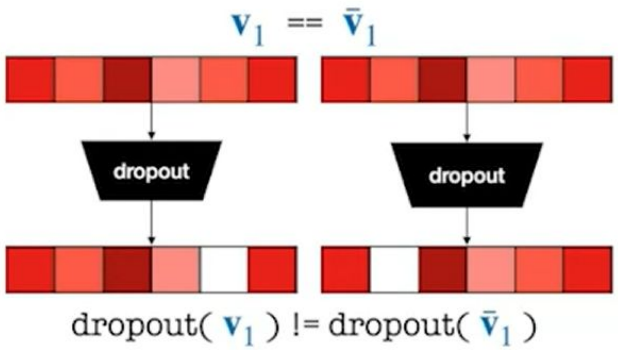
Technique 1: random span masking (data augmentation on the input space)

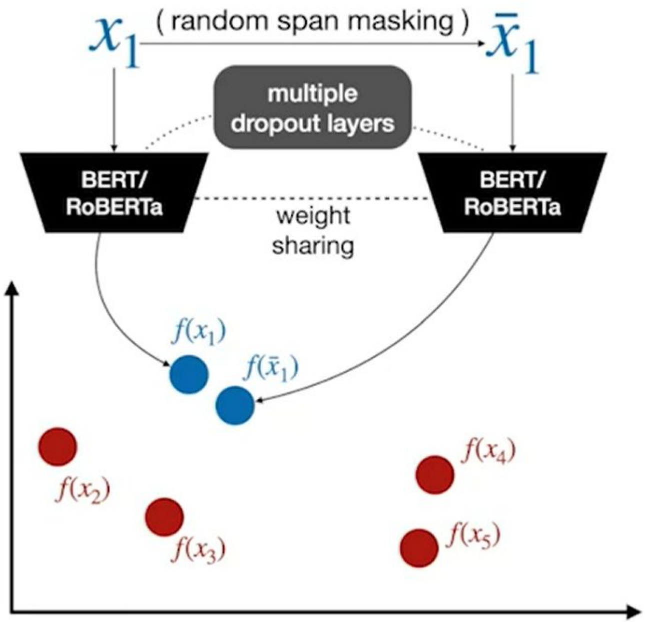
x_1: Economist Paul Krugman mainly works on trade models.

$\bar{x}_1$: Econ [MASK] Paul Krugman mainly works on trade models.

Method

Technique 2: dropout (data augmentation on the feature space)





dist($f(x_1), f(\bar{x}_1)$) < dist($f(x_1)/f(\bar{x}_1), f(x_2)/f(x_3)/\ldots$)

Method

Learning objective: InfoNCE

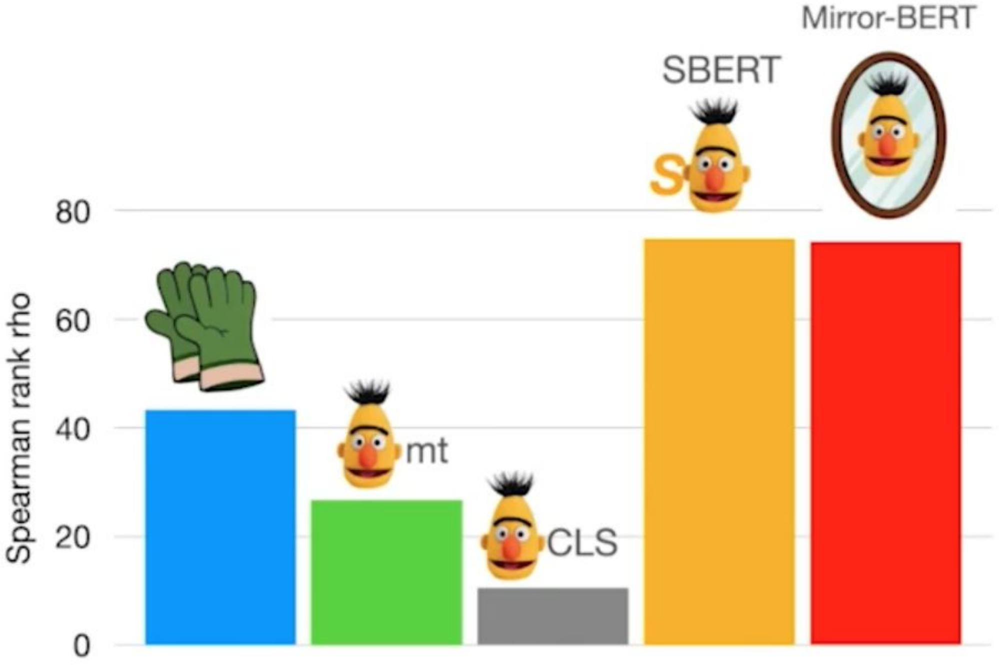
similarity of a positive pair

$$\mathcal{L}_b = -\sum_{i=1}^{|\mathcal{D}_b|} \log \frac{\boxed{\exp(\cos(f(x_i), f(\overline{x}_i))/\tau)}}{\boxed{\sum_{x_j \in \mathcal{N}_i} \exp(\cos(f(x_i), f(x_j))/\tau)}}$$

sum of similarity of negative pairs

Main Results

Average performance on STS benchmarks

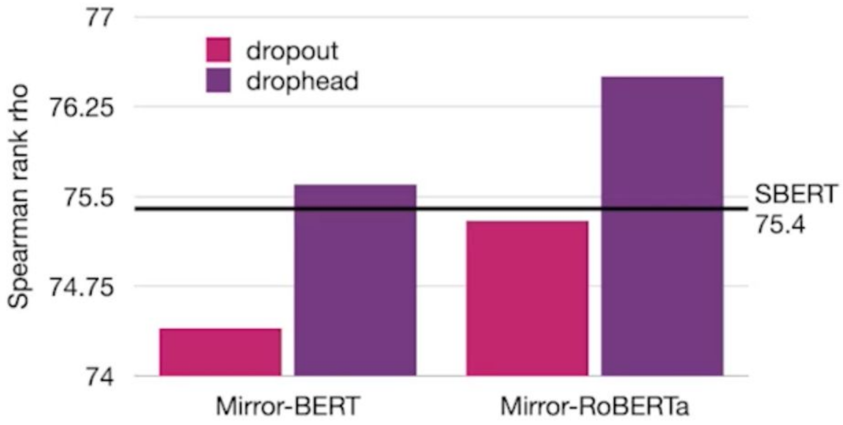


{Multi-text-granularity} x {Multi-domain} x {Multilingual}

{Multi-text-granularity}	{Multi-domain}	{Multilingual}
Word: Word similarity; Bilingual lexicon induction	Generic	English French
Phrase: Biomedical entity linking	Biomedical	Hebrew Italian
Sentence: Semantic textual similarity; Question-answer entailment	Social media	Spanish Turkish
	QA	Arabic Russian
		French Estonian
		Chinese Polish

Other Types of Augmentations?

Randomly dropping Transformer heads (Zhou et al., 2020), instead of neurons

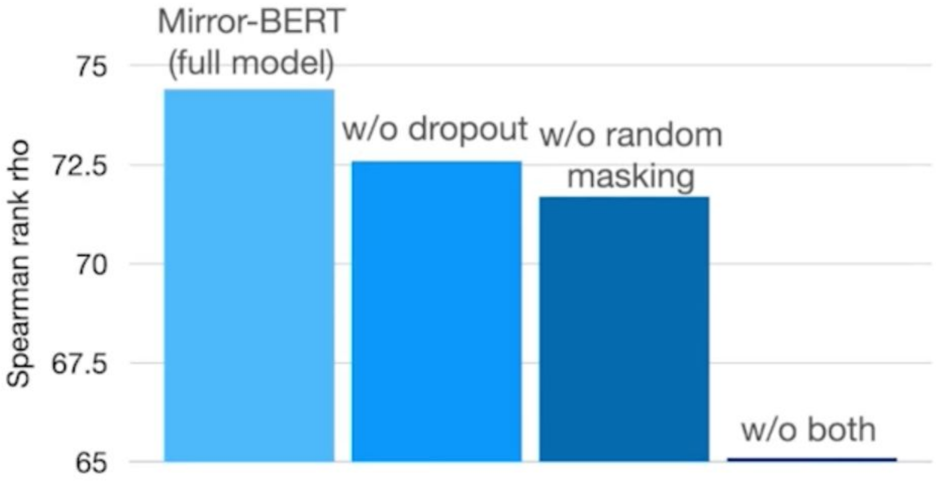


Future work:

- Other heuristic-based augmentations
- Virtual Adversarial Training (Miyato et al., 2018)

Ablation Studies

The synergistic effect between random masking and dropout



Ablation Studies

Learning from negatives alone is still beneficial

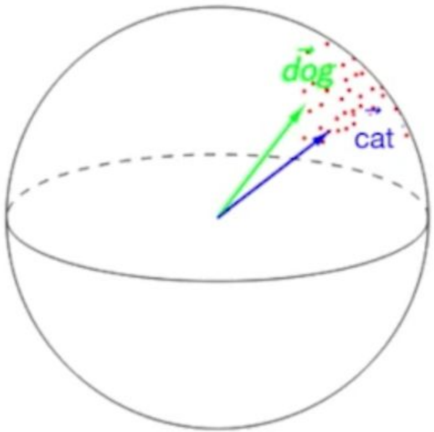
When r.m. (❌) and dropout (❌)

$$\mathcal{L}_b = -\sum_{i=1}^{|\mathcal{D}_b|} \log \frac{\overset{=\text{constant}}{\boxed{\exp(\cos(f(x_i), f(\overline{x}_i))/\tau)}}}{\boxed{\sum_{x_j \in \mathcal{N}_i} \exp(\cos(f(x_i), f(x_j))/\tau)}}$$

gradients solely come from the negative pairs

Interpretation

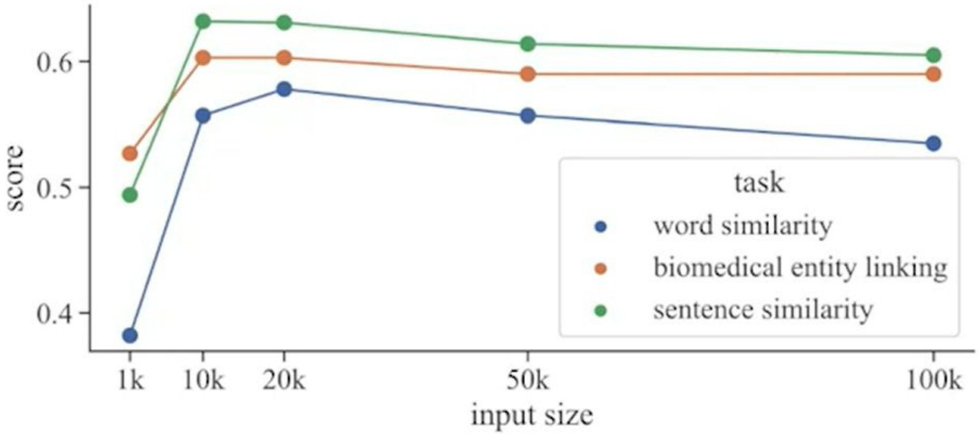
The anisotropy of BERT



“In all layers of BERT, ELMo, and GPT-2, the representations of all words are anisotropic: they occupy a narrow cone in the embedding space instead of being distributed throughout.”

Interpretation

Learning new knowledge or exposing existing knowledge in BERT?



More training data don't help.

Interpretation

An experiment of ‘zero-semantics’ random string tuning

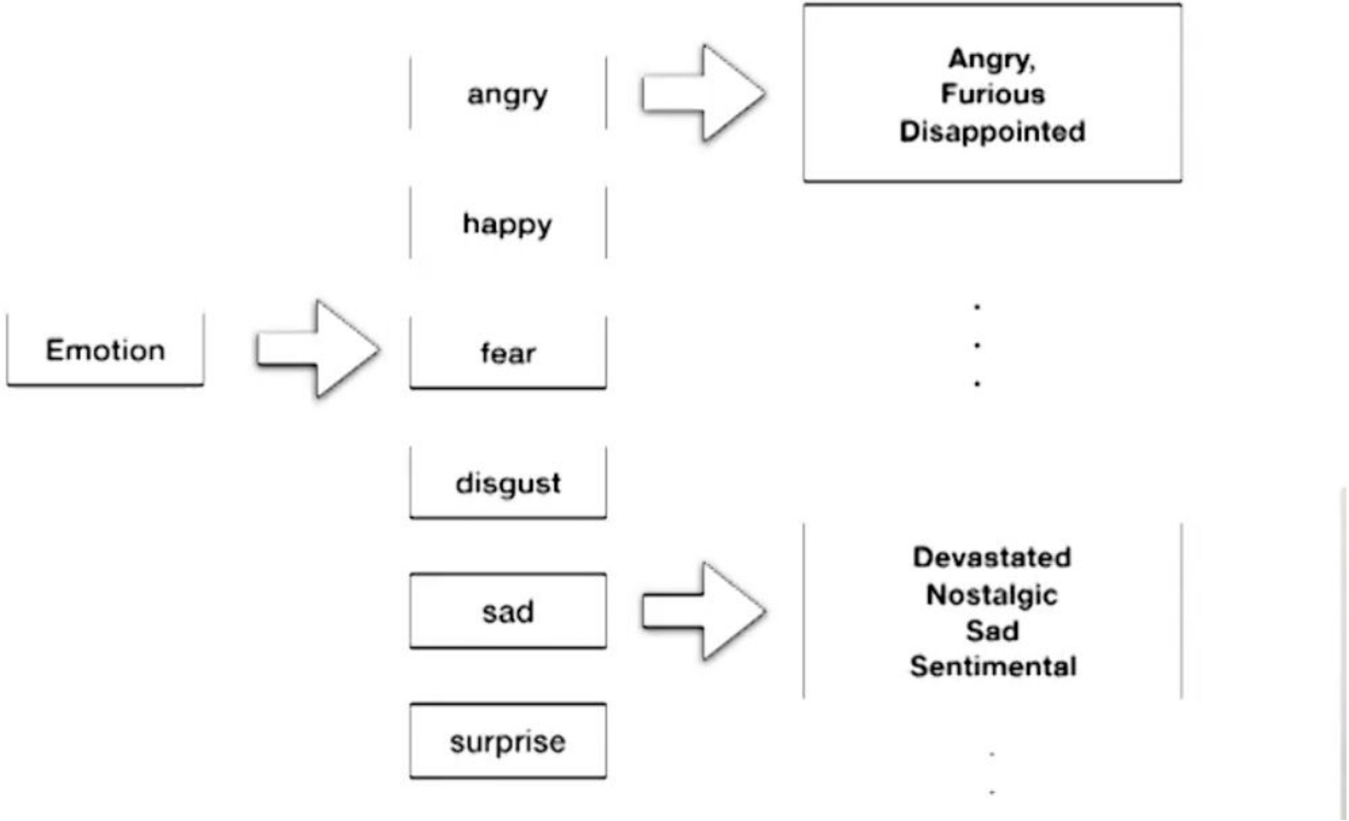
model	ρ
fastText	.434
BERT	.267
+ Mirror	.556
+ Mirror (random string, lr 5e−5)	.481

Not All Negatives are Equal:
Label-Aware Contrastive Loss for Fine-grained Text Classification

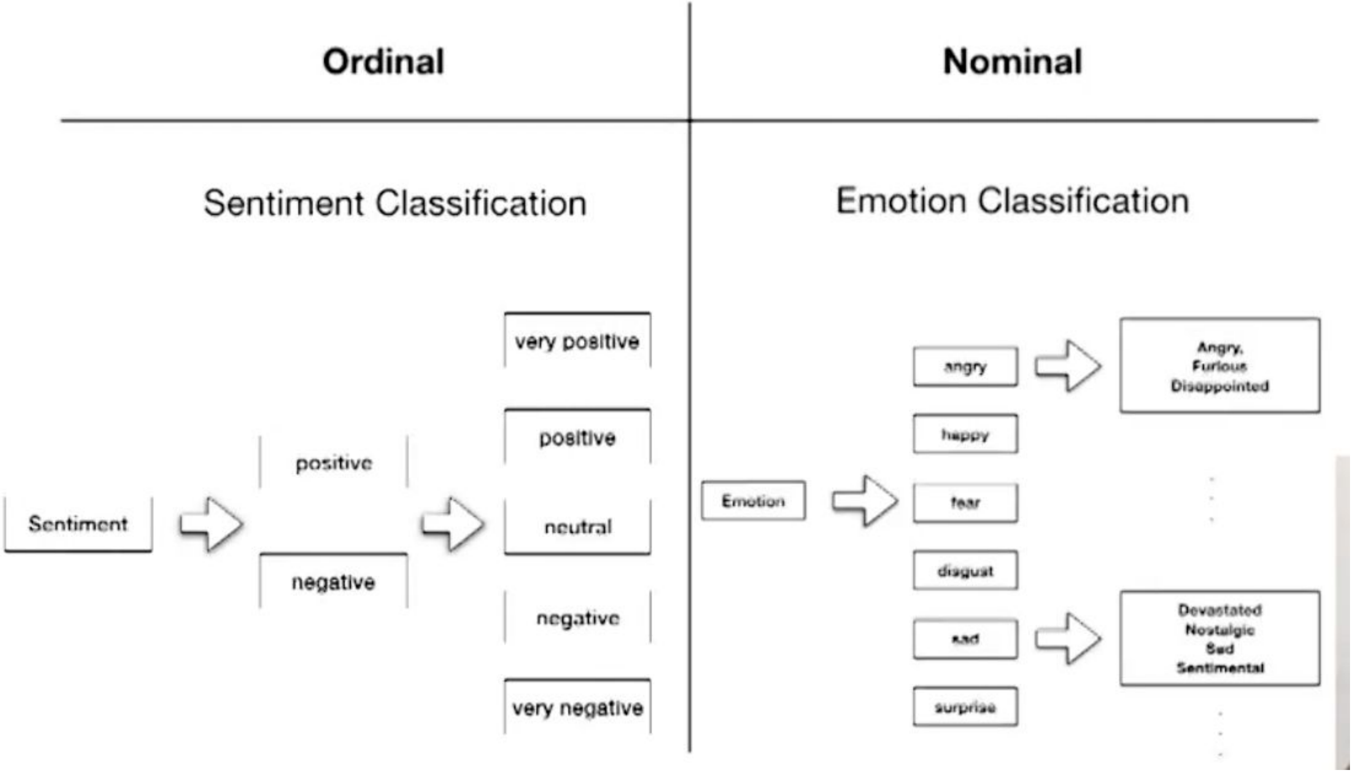
Varsha Suresh
Dept. of Computer Science
National University of Singapore
varshasuresh@u.nus.edu

Desmond C. Ong
Dept. of Information Systems and Analytics
National University of Singapore,
& Institute of High Performance
Computing, A*STAR
dco@comp.nus.edu.sg

Fine Grained Text Classification

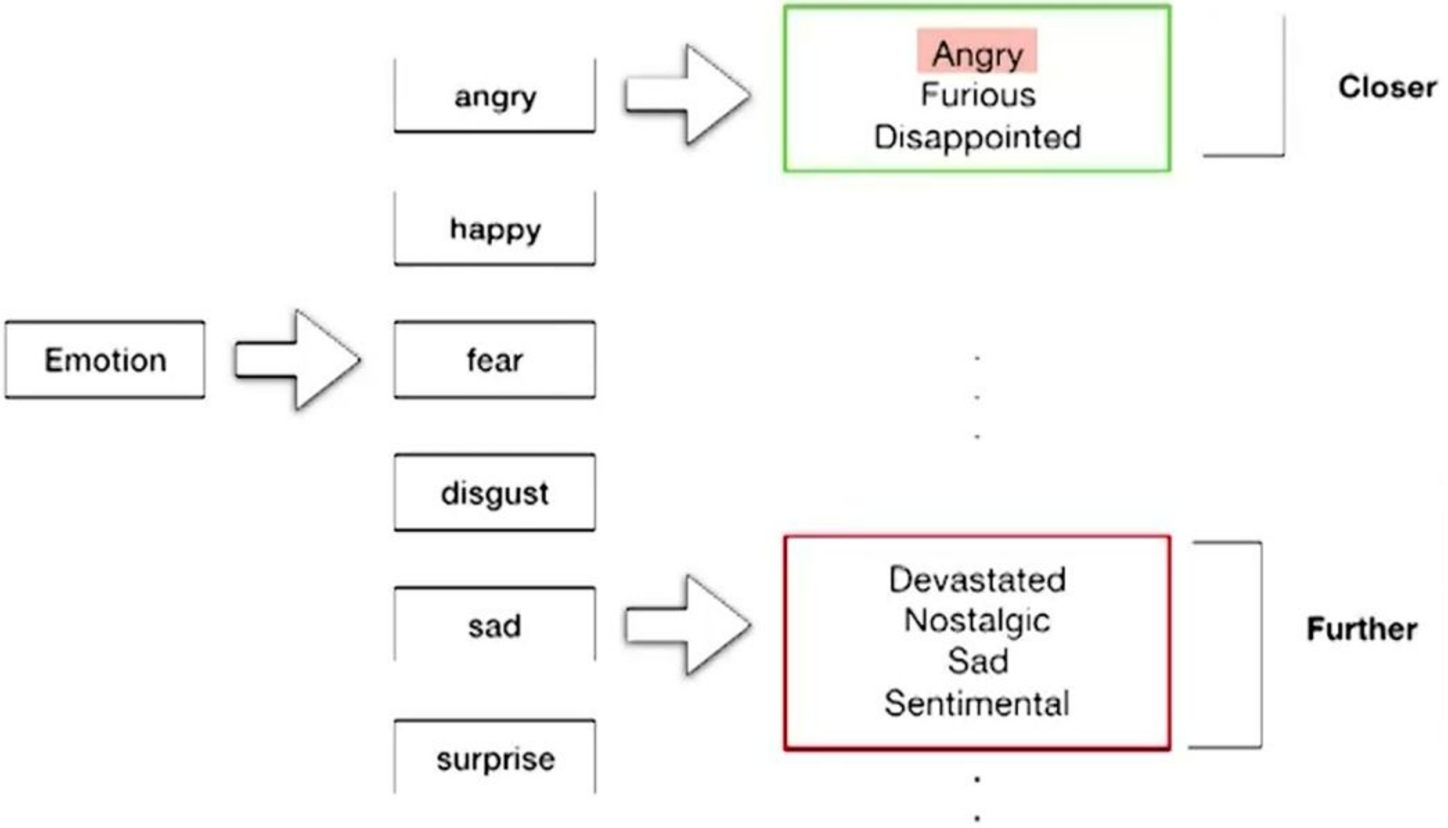


Examples of Fine-grained Classes



Pre-trained language models

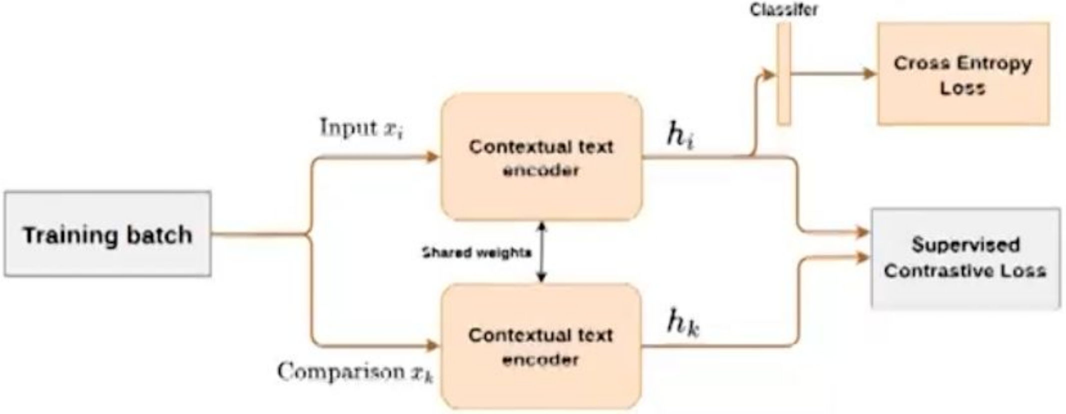
- Pre-trained language models are the current de-facto for text classification.
- Fine-grained text classification using pre-trained language models
  - Multi-task training (Balikas et al.,2017)
  - Adding external knowledge such as emotion lexicons (Khanpour et al., 2018; Suresh et al., 2021)
  - Sentiment specific pre-training objectives (Yin et al., 2020; Tian et al., 2020)



Emotion
angry
happy
fear
disgust
sad
surprise
Angry
Furious
Disappointed
Closer
Devastated
Nostalgic
Sad
Sentimental
Further

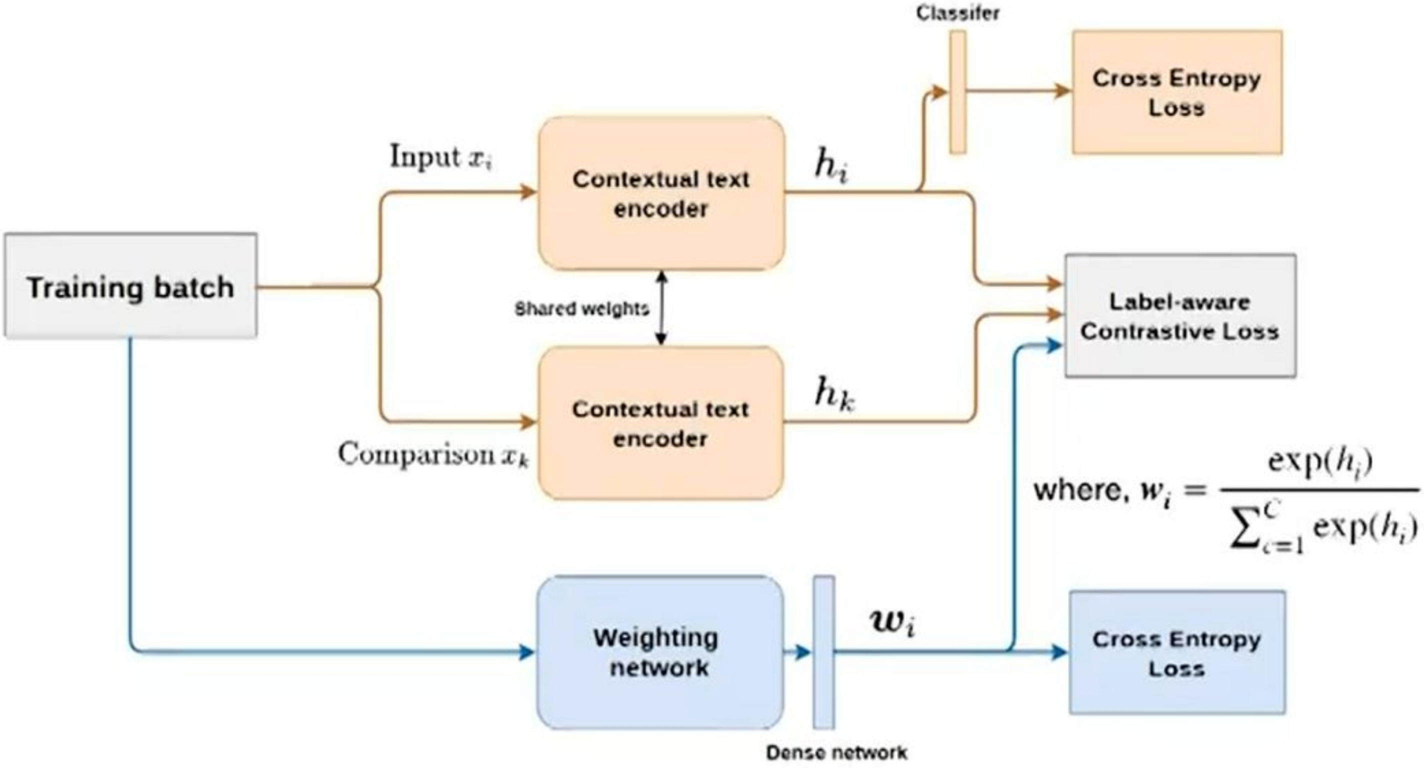
Contrastive fine-tuning

- Contrastive fine-tuning of pre-trained language models (Gunel et al., 2021)



Supervised Contrastive Loss
(Gunel et al., 2020)

$$L_{SCL} = \sum_{i=1}^{2K} \frac{-1}{|\mathscr{P}|} \sum_{p \in \mathscr{P}} \log \frac{\exp(h_i \cdot h_p / \tau)}{\sum_{k \in \mathscr{I}/i} \exp(h_i \cdot h_k / \tau)}$$



Classifer
Cross Entropy Loss
Input x_i
Contextual text encoder
h_i
Training batch
Shared weights
Label-aware Contrastive Loss
Contextual text encoder
h_k
Comparison x_k
where, $w_i = \frac{\exp(h_i)}{\sum_{c=1}^{C} \exp(h_i)}$
Weighting network
w_i
Cross Entropy Loss
Dense network

Supervised Contrastive Loss
(Gunel et al., 2020)

$$L_{SCL} = \sum_{i=1}^{2K} \frac{-1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \log \frac{\exp(h_i \cdot h_p / \tau)}{\sum_{k \in \mathcal{I}/i} \exp(h_i \cdot h_k / \tau)}$$

Label-aware Contrastive Loss
(Ours)

$$\mathcal{L}_i = \sum_{p \in \mathcal{P}} \log \frac{w_{i,y_i} \cdot \exp(h_i \cdot h_p / \tau)}{\sum_{k \in \mathcal{I} \setminus i} w_{i,y_k} \cdot \exp(h_i \cdot h_k / \tau)}$$

Results

Fine-tuning Strategy	Fine-grained	Coarse-grained
	SST - 5 [1]	SST - 2[1]
	# of labels	
	5	2
Cross Entropy	57.1	94.4
Supervised Contrastive	57.4	94.3
Label-aware Contrastive	**58.5**	**94.5**

Task —— Sentiment Classification

Contextual Encoder, Weighting Network: ELECTRA (Clark et al., 2019)

Results

Fine-tuning Strategy	Fine-grained		Coarse-grained	
	Empathetic Dialogues[1]	GoEmotions[2]	ISEAR[3]	EmoInt[4]
	# of labels			
	32	27	7	4
Cross Entropy	58.3	64.8	71.4	85.5
Supervised Contrastive	58.5	64.3	70.5	85.7
Label-aware Contrastive	**60.1**	**65.5**	**72.4**	**86.6**

Task —> Emotion Classification

Effect of number of classes

Fine-tuning Strategy	Number of classes*			
	32	16	8	4
Cross Entropy	58.1	68.8	78.0	**89.2**
Supervised Contrastive	58.6	67.9	77.0	88.8
Label-aware Contrastive	**60.1**	**69.6**	**78.7**	88.8

4-easy subset : Anger, Fear, Sad, Happy

Hard label sets

Fine-tuning Strategy	Anticipating, Excited, Hopeful, Guilty	Angry, Ashamed, Furious, Guilty	Devastated, Nostalgic, Sad, Sentimental	Anxious, Apprehensive, Afraid, Terrified
Cross Entropy	67.4	54.3	63.2	56.1
Supervised Contrastive	68.1	53.3	63.7	55.4
Label-aware Contrastive	**69.5**	**55.6**	**64.2**	**57.5**

Model confidence comparison

- Model confidence for the fine-grained classification.

$$\text{Entropy}_k = -\sum_k s_k \cdot \log_2(s_k)$$