Contrastive Learning for Natural Language Processing

Rui Zhang, Penn State University
https://ryanzhumich.github.io/
August 4, 2022 @ Amazon

PennState

Contrastive Learning

Learning embeddings such that similar data sample pairs are close while dissimilar sample pairs stay far apart (Chopra et al., 2005)

$$\mathrm{sim}(f(\boldsymbol{x}), f(\boldsymbol{x}^+)) \gg \mathrm{sim}(f(\boldsymbol{x}), f(\boldsymbol{x}^-))$$

f : encoder, e.g., neural networks

sim : similarity measure, e.g., inner product

$\boldsymbol{x}$: anchor

$\boldsymbol{x}^+$: positive example

$\boldsymbol{x}^-$: negative example

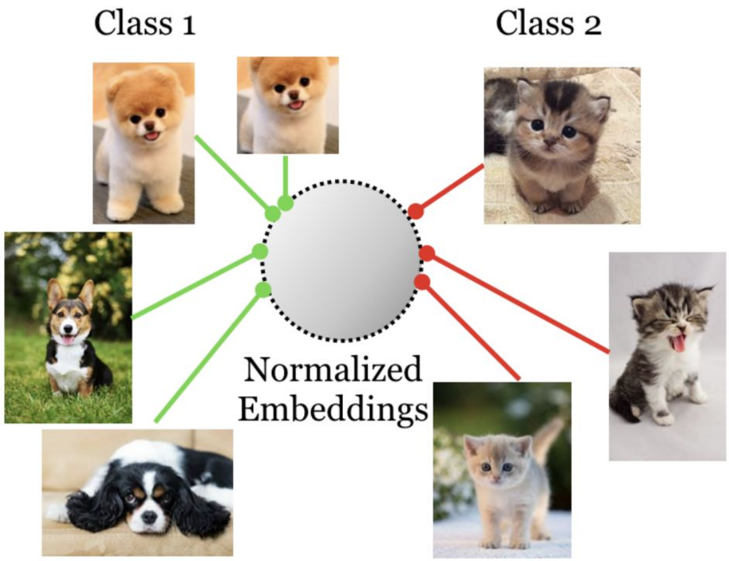
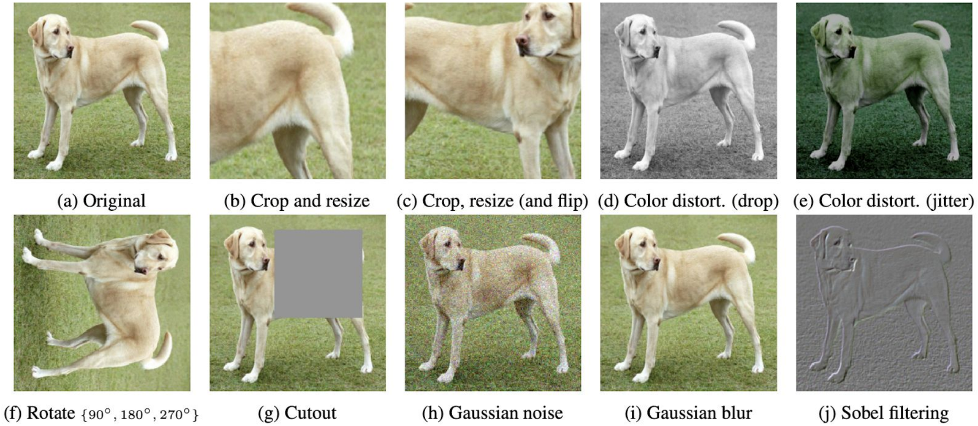


Figure from Khosla et al., 2020

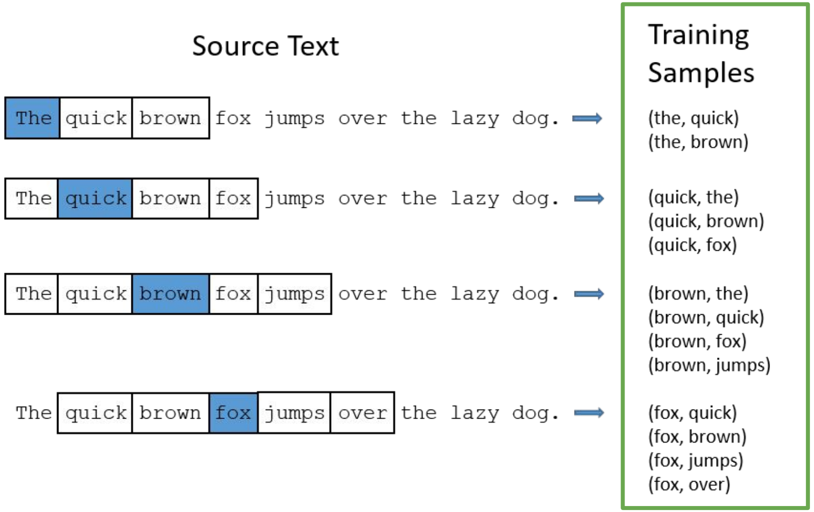
Contrastive Learning in Computer Vision

[SimCLR (Chen et al., 2020)]



Most Successful Example of Contrastive Learning for NLP

word2vec (Mikolov et al., 2013) for word embeddings



word2vec's skip-gram model. Figure from Chris McCormick

$$\boxed{\log \sigma({v'_{w_O}}^\top v_{w_I})} + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} \boxed{\left[\log \sigma(-{v'_{w_i}}^\top v_{w_I})\right]}$$

f : word embeddings

sim : inner product

$\boldsymbol{x}$: current word

$\boldsymbol{x}^+$: context word

$\boldsymbol{x}^-$: random word by negative exampling

Agenda

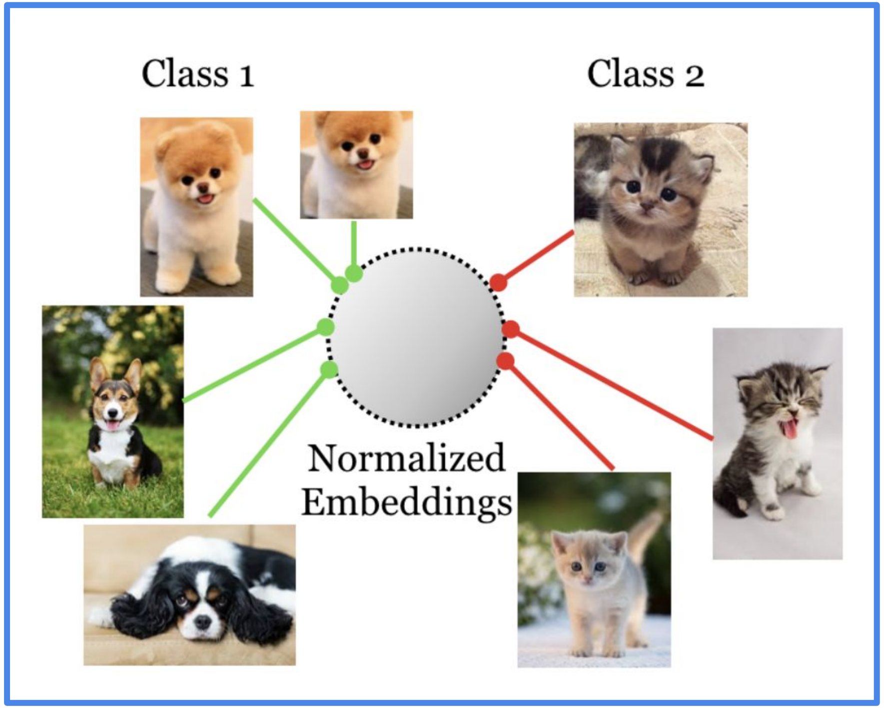
- Part 1: Foundations of Contrastive Learning
- Part 2: Contrastive Learning for NLP: A Case Study in Named Entity Recognition

Part 1.

Foundations of Contrastive Learning

Two Elements of Contrastive Learning

Contrastive Learning = Contrastive Data Creation + Contrastive Objective Optimization



$$\mathrm{sim}(f(\boldsymbol{x}), f(\boldsymbol{x}^+)) \gg \mathrm{sim}(f(\boldsymbol{x}), f(\boldsymbol{x}^-))$$

f : encoder, e.g., neural networks

sim : similarity measure, e.g., inner product

$\boldsymbol{x}$: anchor

$\boldsymbol{x}^+$: positive example

$\boldsymbol{x}^-$: negative example

Part 1.1

Contrastive Learning Objectives

Noise Contrastive Estimation (NCE)

Use Logistic Regression with cross-entropy loss to differentiate positive samples (i.e., target distribution) and negative samples (i.e., noise distribution).

$\ell(\boldsymbol{x})$	Logit function of a sample from the target distribution
$\sigma(\ell(\boldsymbol{x}))$	Probability a sample from the target distribution

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{x}^+, \boldsymbol{x}^-) &= -\left[\log \sigma(\ell(\boldsymbol{x}^+)) + \log(1 - \sigma(\ell(\boldsymbol{x}^-)))\right] \\
&= -\left[\log \sigma(\ell(\boldsymbol{x}^+)) + \log \sigma(-\ell(\boldsymbol{x}^-))\right]
\end{aligned}
$$

Noise-contrastive estimation: A new estimation principle for unnormalized statistical models (Gutmann and Hyvärinen, 2010)

InfoNCE

Use softmax loss to differentiate a positive sample from a set of noise examples.

$\boldsymbol{c}$ Context Vector, e.g., anchor point

$X = \{x_1, \ldots, x_N\}$ N samples with 1 positive sample and N-1 negative samples

$$\mathcal{L} = -\log \frac{\boxed{f(\boldsymbol{x}, \boldsymbol{c})}}{\boxed{\sum_{\boldsymbol{x}' \in X} f(\boldsymbol{x}', \boldsymbol{c})}}$$

1 positive sample

1 positive sample + N-1 negative samples

[Representation Learning with Contrastive Predictive Coding (van den Oord et al., 2018)]

Normalized Temperature-scaled Cross-Entropy (NT-Xent)

$$\mathcal{L} = -\log \frac{\boxed{\exp(\mathrm{sim}(\boldsymbol{x}, \boldsymbol{x}^{+})/\tau)}}{\boxed{\exp(\mathrm{sim}(\boldsymbol{x}, \boldsymbol{x}^{+})/\tau)} + \boxed{\sum_{j=1}^{N-1} \exp(\mathrm{sim}(\boldsymbol{x}, \boldsymbol{x}_j^{-})/\tau)}}$$

Cosine Similarity

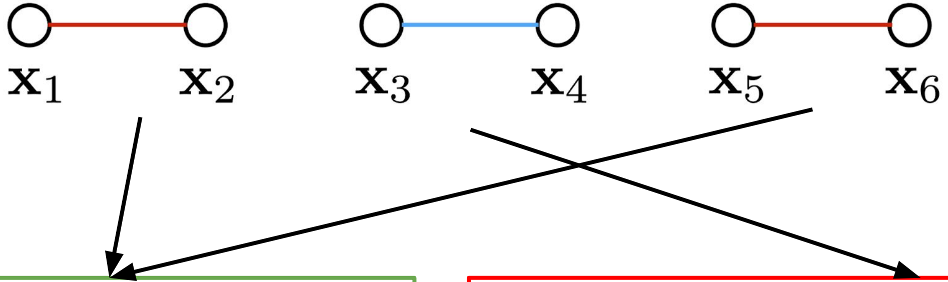
Normalized Embeddings

Temperature controls the relative importance of the distances between point pairs

- At low temperatures, the loss is dominated by the small distances.
- At high temperatures, the loss is dominated by the large distances.

[A Simple Framework for Contrastive Learning of Visual Representations. (Chen et al., 2020)](#)

Contrastive Loss



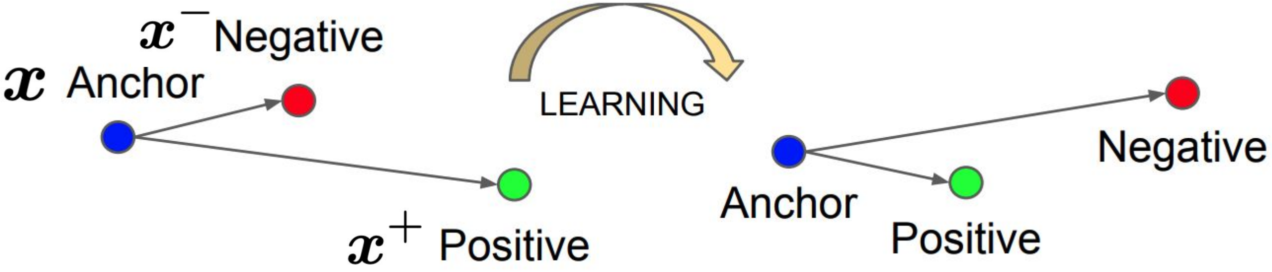
$$\mathcal{L}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boxed{\mathbb{1}[(y_i = y_j)] |f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j)|^2} + \boxed{\mathbb{1}[(y_i \neq y_j)] \max(0, m - |f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j)|)^2}$$

minimizes the embedding distance when they are from the same class

maximizes the embedding distance when they are from the same class

[Learning a Similarity Metric Discriminatively, with Application to Face Verification (Chopra et al., 2005)]

Triplet Loss

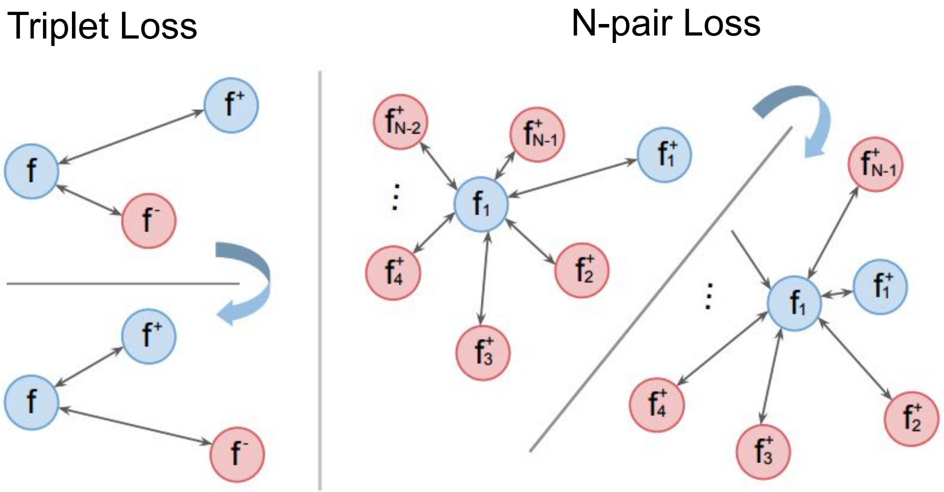


$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{x}^+, \boldsymbol{x}^-) = \max(0, \boxed{m + ||f(\boldsymbol{x}) - f(\boldsymbol{x}^+)||_2^2} - \boxed{||f(\boldsymbol{x}) - f(\boldsymbol{x}^-)||_2^2})$$

We push the the distance between positive and anchor + margin to be smaller than the distance between negative and anchor.

[FaceNet: A Unified Embedding for Face Recognition and Clustering (Schroff et al., 2015)]

N-pair Loss



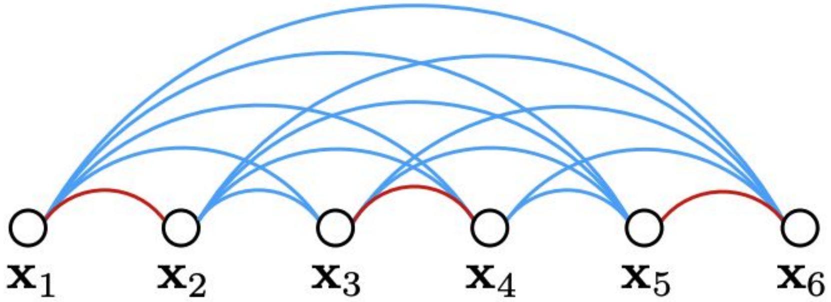
$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_i^-\}_{i=1}^{N-1}) = \log\left(1 + \sum_{i=1}^{N-1} \exp(f(\boldsymbol{x})^\top f(\boldsymbol{x}_i^-) - f(\boldsymbol{x})^\top f(\boldsymbol{x}^+))\right)$$

- Extend to N-1 negative examples
- Inner product similarity + softmax loss
- Similar to multi-class classification

[Improved Deep Metric Learning with Multi-class N-pair Loss Objective (Sohn, 2016)]

Lifted Structured Loss

Lifted Structured Loss explicitly takes into account all pairwise edges within the batch.



(c) Lifted structured embedding

Illustration for a training batch with six examples.
Red edges: similar examples.
Blue edges: dissimilar examples.

$$\mathcal{L}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \max\left(0, d_{i,j} + \log\left(\sum_{(i,k)} \exp(m - d_{i,k}) + \sum_{(j,l)} \exp(m - d_{j,l})\right)\right)^2$$

[Deep Metric Learning via Lifted Structured Feature Embedding (Song et al., 2016)]

Summary of Contrastive Learning Objectives

Loss Function	Paper	Contrast Unit			Number of Examples		Used In
		Pair	Triplet	Set	# of positive	# of negative	
Contrastive Loss	(Chopra et al., 2005)	✓			0/1	0/1	
Triplet Loss	(Schroff et al., 2015)		✓		1	1	
N-pair Loss	(Sohn, 2016)			✓	1	$N-1$	
NCE	(Gutmann and Hyvärinen, 2010)	✓			0/1	0/1	
Negative Sampling	(Mikolov et al., 2013)			✓	1	$N-1$	word2vec
InfoNCE	(van den Oord et al., 2018)			✓	1	$N-1$	
NT-Xent	(Chen et al., 2020)			✓	1	$N-1$	simCLR,simCSE,CLIP
Soft-Nearest Neighbors Loss	(Frosst et al., 2019)			✓	M	N	
Lifted Structured Loss	(Oh Song et al., 2016)			✓	M	N	

Part 1.2

Contrastive Data Sampling and Augmentation Strategies

Self-Supervised Contrastive Learning

Positive: Data Augmentation

Negative: Random, e.g., In-batch Negatives

The Biggest Advantage: No label is required!

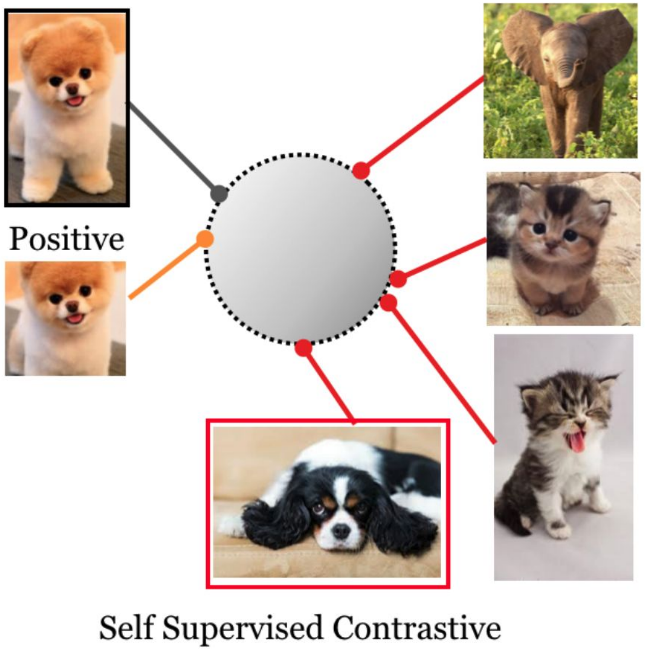
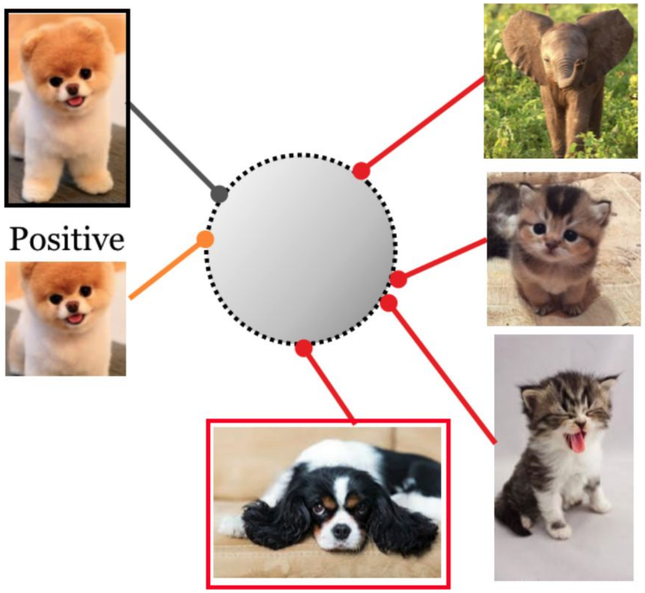


Figure from (Khosla et al., 2020)

Four Challenges of Self-Supervised Contrastive Learning

1. Non-trivial Data Augmentation
2. Risk of “Sampling Bias” (i.e., False Negative)
3. Hard Negative Mining
4. Large Batch Size



Self Supervised Contrastive

Figure from [(Khosla et al., 2020)](#)

Data Augmentation for Text

Text Space

- Lexical Editing (token-level)
- Back-Translation (sentence-level)

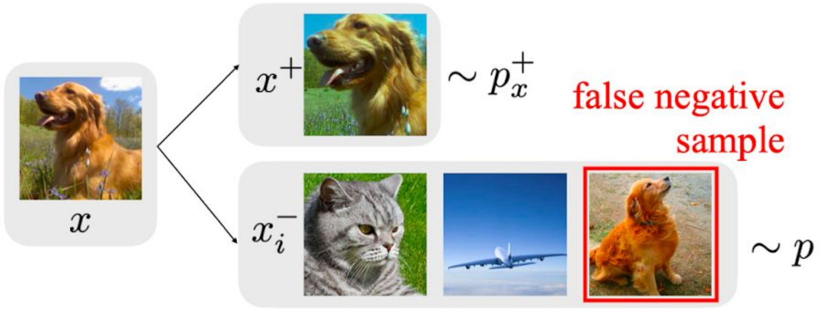
Embedding Space

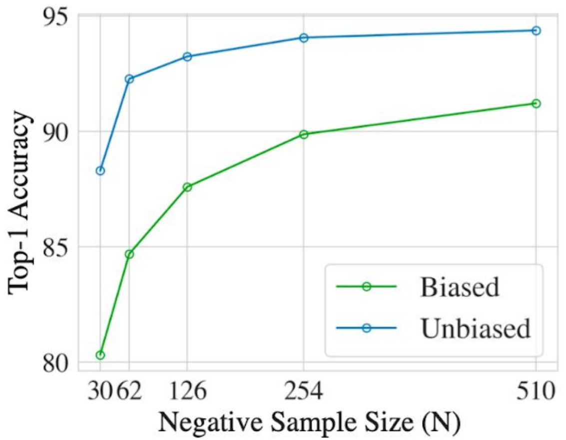
- Dropout
- Cutoff
- Mixup

Manual

EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. (Wei and Zhou, 2019)
Conditional BERT Contextual Augmentation (Wu et al., 2018)
Improving Neural Machine Translation Models With Monolingual Data (Sennrich et al., 2016)
CERT: Contrastive Self-supervised Learning for Language Understanding (Fang et al., 2020)
SimCSE: Simple Contrastive Learning of Sentence Embeddings. (Gao et al., 2021)
A Simple but Tough-to-Beat Data Augmentation Approach for Natural Language Understanding and Generation. (Shen et al., 2020)
mixup: Beyond Empirical Risk Minimization. (Zhang et al., 2017)
NL-Augmenter A Framework for Task-Sensitive Natural Language Augmentation (Dhole et al., 2021)

Sampling Bias





Problem: Because we don't know the label, we may accidentally create false negative by sampling examples from the same class.

[Debiased Contrastive Learning (Chuang et al., 2020)]

Debiased Contrastive Learning

Key Idea: Assume a prior probability between positive and negative, then approximate the distribution of negative examples to debias the loss.

$$p(x') = \tau^+ p_x^+(x') + \tau^- p_x^-(x')$$

Then samples N samples (may contain positive and negative) and M positive samples

$$\text{replace } p_x^- \text{ in } L^N_{\text{Unbiased}} \text{ with } p_x^-(x') = (p(x') - \tau^+ p_x^+(x'))/\tau^-$$

$$-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + N g\left(x, \{u_i\}_{i=1}^N, \{v_i\}_{i=1}^M\right)}$$

[Debiased Contrastive Learning (Chuang et al., 2020)]()

Hard Negative Mining

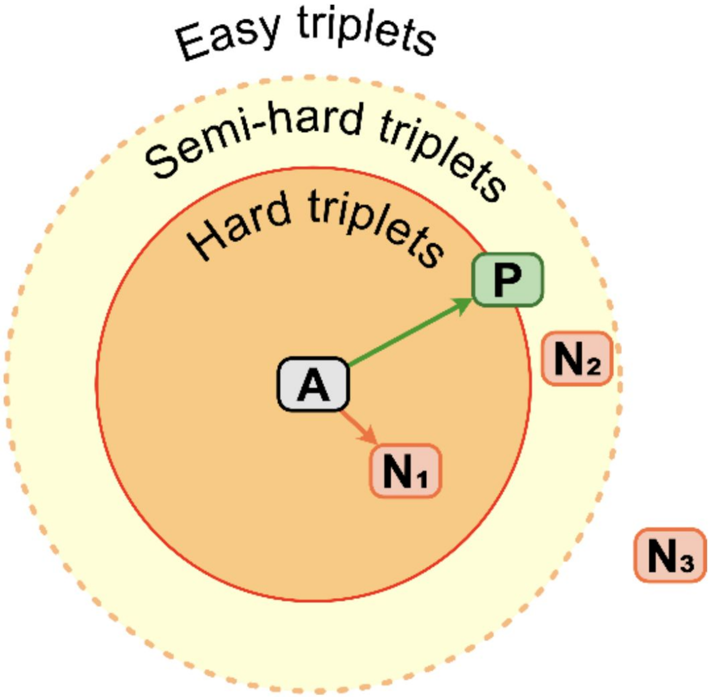


Figure from [Kurowski et al., 2021]

A: Anchor. P: Positive. N: Negative

We want to AN is greater than AP, at least by the margin.

Hard Negative Mining: Find hard negatives

Hard Negative Mining by Importance Sampling

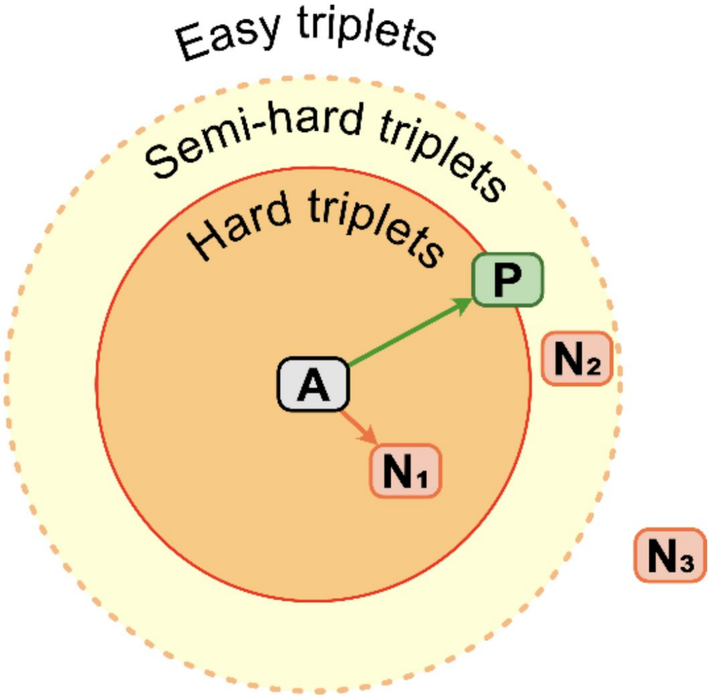


Figure from Kurowski et al., 2021

$$\boxed{q_\beta(x^-)} \propto \boxed{e^{\beta f(x)^\top f(x^-)}} \cdot \boxed{p(x^-)}$$

new sampling probability

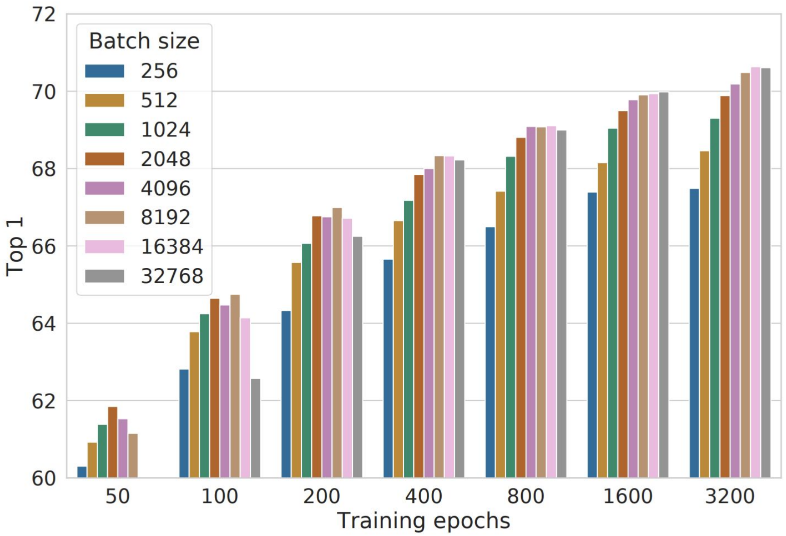
similarity

original sampling probability

Key Idea: If this negative sample is close to the anchor sample, then we up-weight its probability of being selected.

Contrastive Learning with Hard Negative Samples (Robinson et al., 2021)

Large Batch Size



SimCLR of ResNet-50 trained with different batch sizes and epochs.

“*We train with larger batch size (up to 32K) and longer (up to 3200 epochs).*”

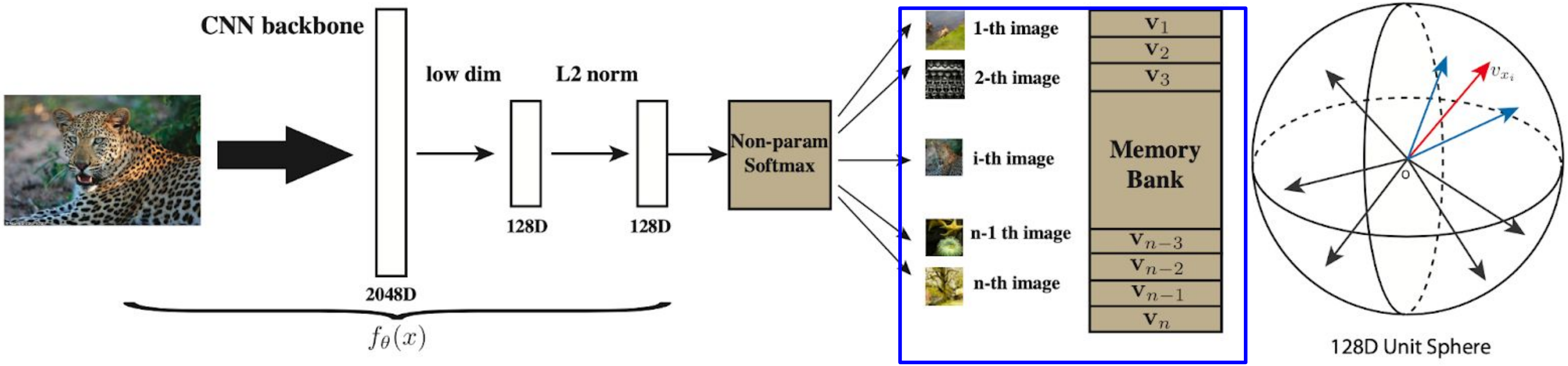
— Chen et al., SimCLR

“*We use a very large minibatch size of 32,768.*”

— Radford et al., CLIP

Memory Bank to Reduce Computation

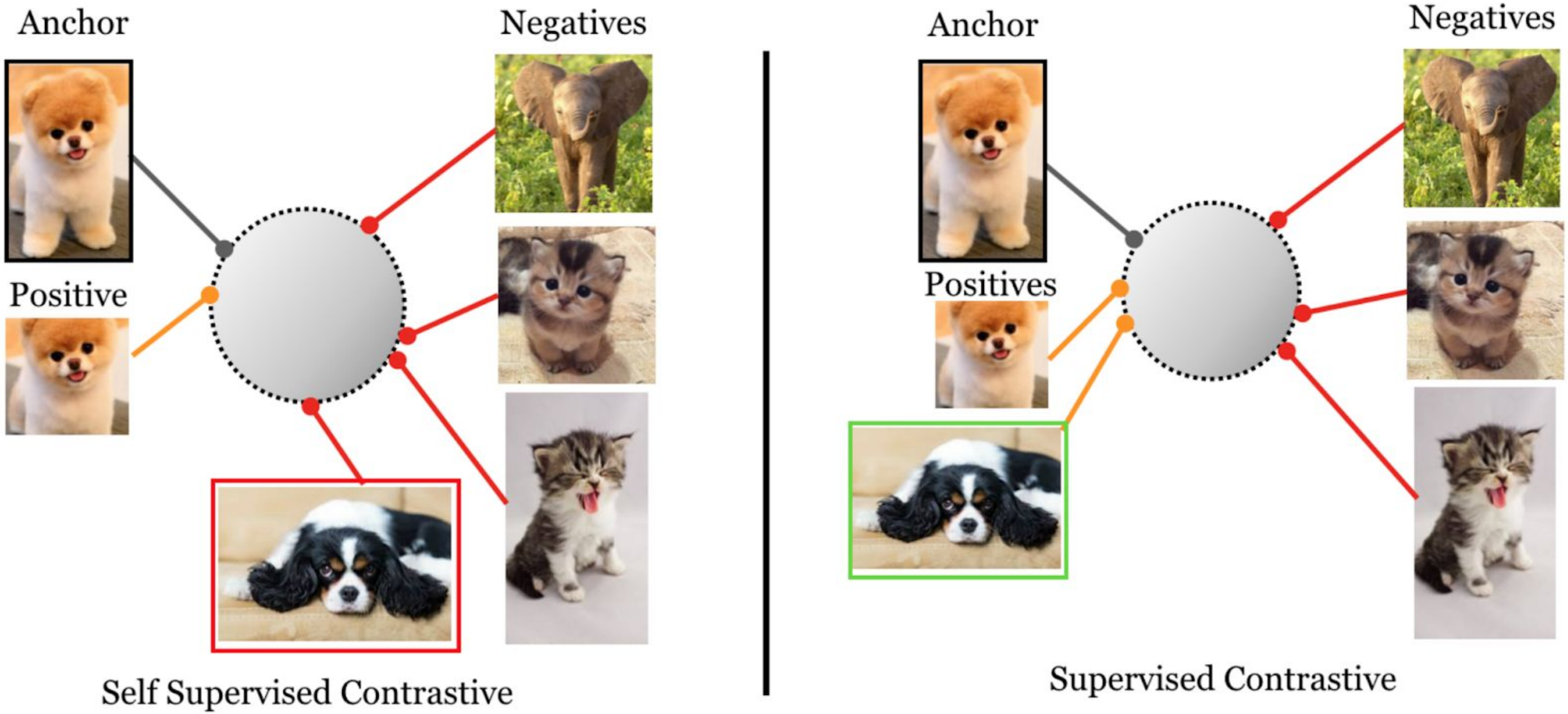
Memory Bank: Compute and store the representations in advances, instead of computing embeddings for all examples in a batch.



Instance-level discrimination uses contrastive learning to maximally scatter the features of training samples over the 128-dimensional unit sphere. Embeddings are stored in a Memory Bank.

[Unsupervised Feature Learning via Non-Parametric Instance Discrimination. (Wu et al., 2018)]

From Self-Supervised to Supervised Contrastive Learning



[Supervised Contrastive Learning (Khosla, et al., 2020)]

Supervised Contrastive Learning

Positive: Same Class

Negative: Different Class

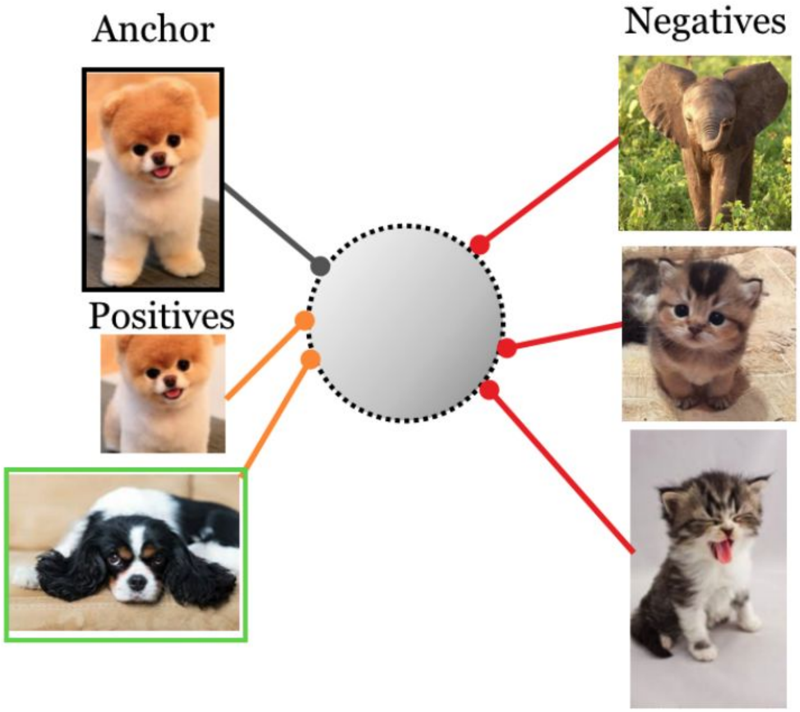
Pros

- No Need for Data Augmentation
- No Risk of “False Negative”
- No Need for Large Batch Size

Cons

- Need Label

Sentence-BERT, SimCSE, DPR, CLIP



Supervised Contrastive

Supervised Contrastive Learning (Khosla, et al., 2020)

Part 2.

Contrastive Learning for NLP: A Case Study in Named Entity Recognition

Contrastive Learning for NLP

(Smith and Eisner, 2005): The first NLP paper introducing “contrastive estimation” as an unsupervised training objective for log-linear models.

Contrastive Estimation: Training Log-Linear Models on Unlabeled Data*

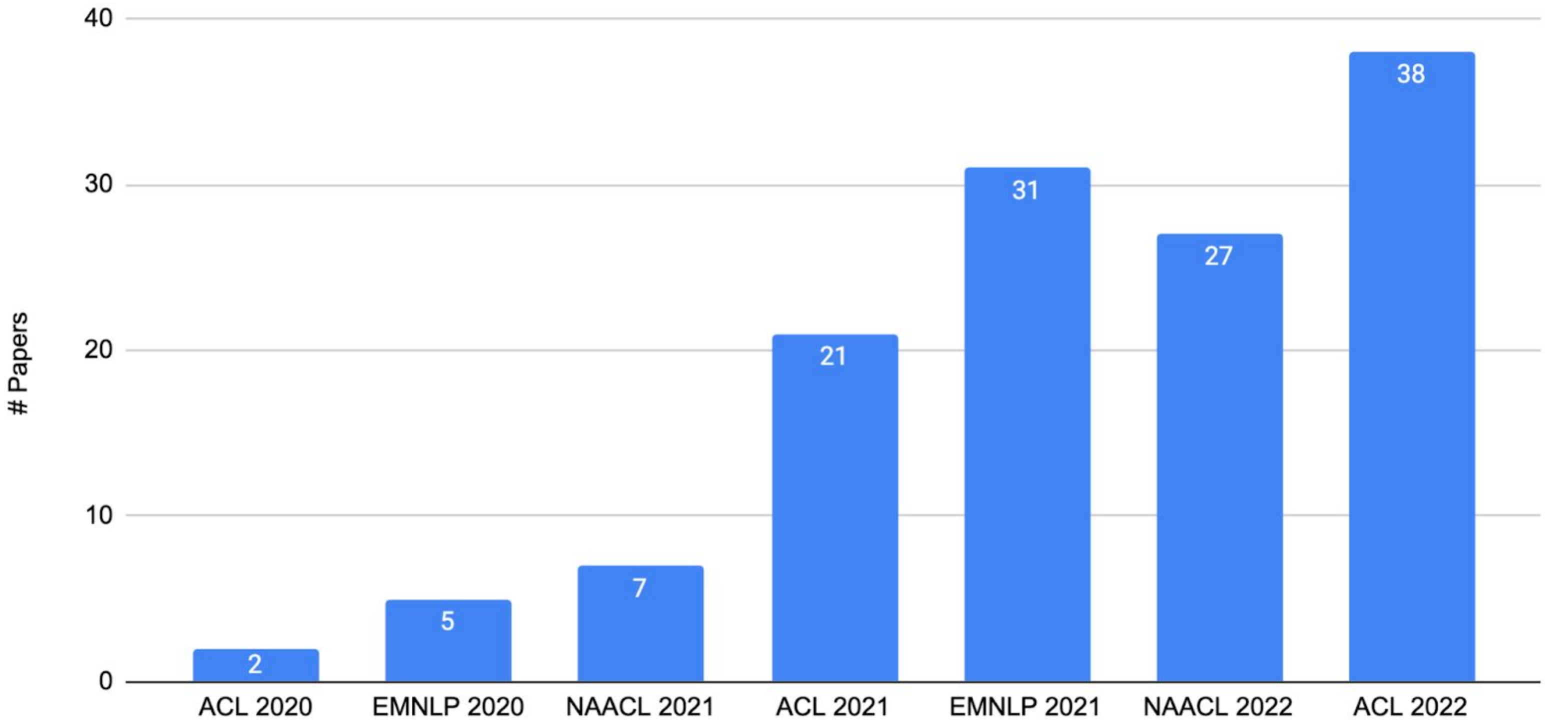
Noah A. Smith and **Jason Eisner**
Department of Computer Science / Center for Language and Speech Processing
Johns Hopkins University, Baltimore, MD 21218 USA

$$\prod_i p\left(X_i = x_i \mid X_i \in \boxed{\mathcal{N}(x_i)}, \vec{\theta}\right)$$

“neighborhood” N(xi) is a set of implicit negative examples plus the example xi itself.

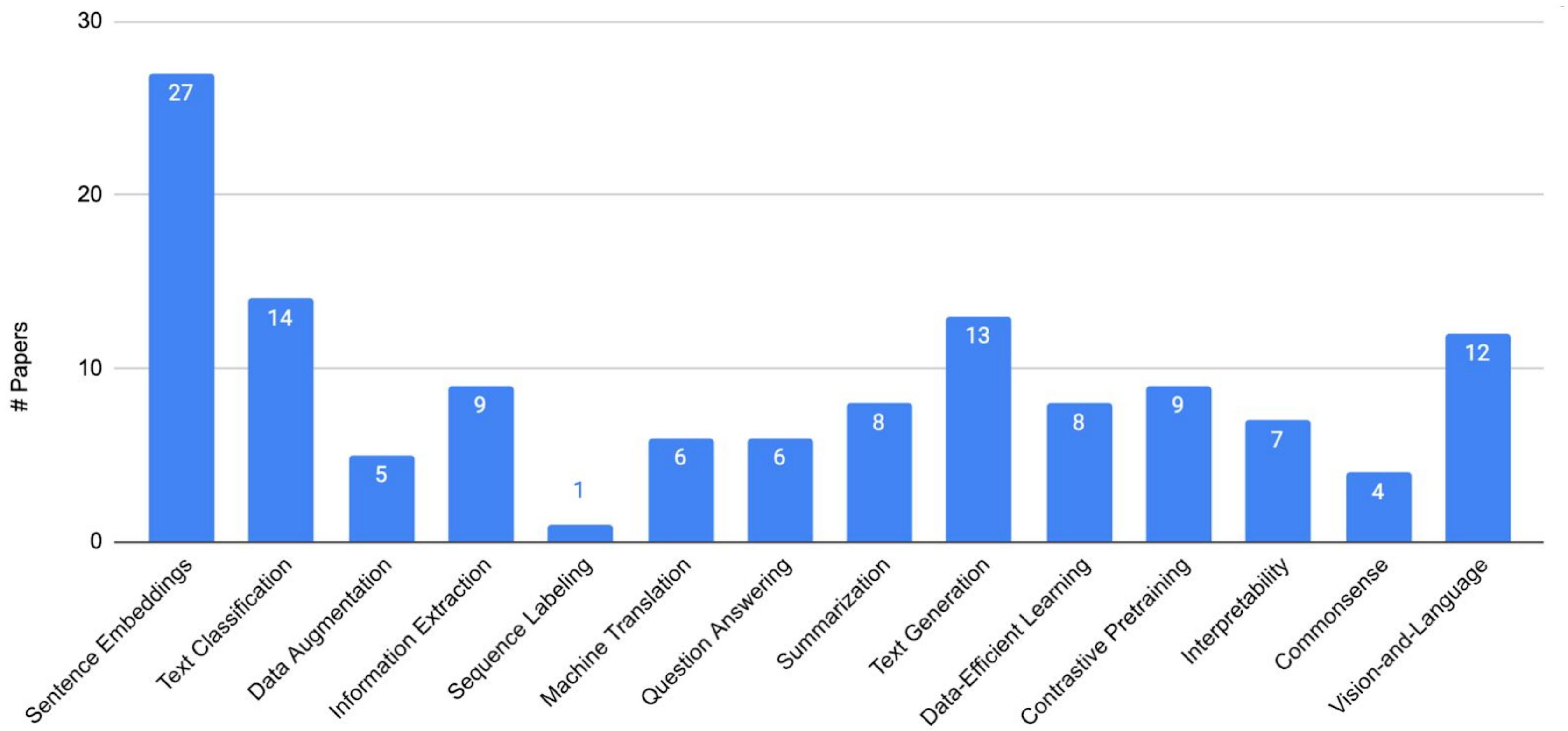
Why Talk about Contrastive Learning for NLP Today?

Number of papers with titles containing “contrastive learning” in recent NLP conferences



Why Talk about Contrastive Learning for NLP Today?

Number of papers with titles containing “contrastive learning” in recent NLP conferences



Contrastive Learning for NLP

- word embeddings ⇨ sentence representations ⇨ various tasks.
  - Classification: Text Classification, Information Extraction
  - Reasoning: Commonsense Reasoning, Question Answering, Fact Verification
  - Generation: Summarization, Machine Translation, Text Generation
  - Multimodal Learning: Vision-and-Language

- performance improvements ⇨ desired characteristics
  - Task-agnostic Sentence Representation
  - Data-efficient Learning in Zero-shot and Few-shot settings
  - Interpretability and Robustness
  - Faithful Text Generation

Contrastive Learning for NLP

- word embeddings ⇨ sentence representations ⇨ various tasks.
  - Classification: Text Classification, Information Extraction
  - Reasoning: Commonsense Reasoning, Question Answering, Fact Verification
  - Generation: Summarization, Machine Translation, Text Generation
  - Multimodal Learning: Vision-and-Language

- performance improvements ⇨ desired characteristics
  - Task-agnostic Sentence Representation
  - **Data-efficient Learning in Zero-shot and Few-shot settings**
  - Interpretability and Robustness
  - Faithful Text Generation

CONTaiNER: Few-shot Named Entity Recognition Using Contrastive Learning

Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca J. Passonneau, Rui Zhang

ACL 2022

Named Entity Recognition (NER)

Barack Obama (born August 4, 1961) is an American attorney and politician who served as the 44th President of the United States from January 20, 2009 to January 20, 2017.

Person Date Location Other ("O")

Few-Shot Named Entity Recognition

Barack Obama (born August 4, 1961) is an American attorney and politician who served as the 44th President of the United States from January 20, 2009 to January 20, 2017.

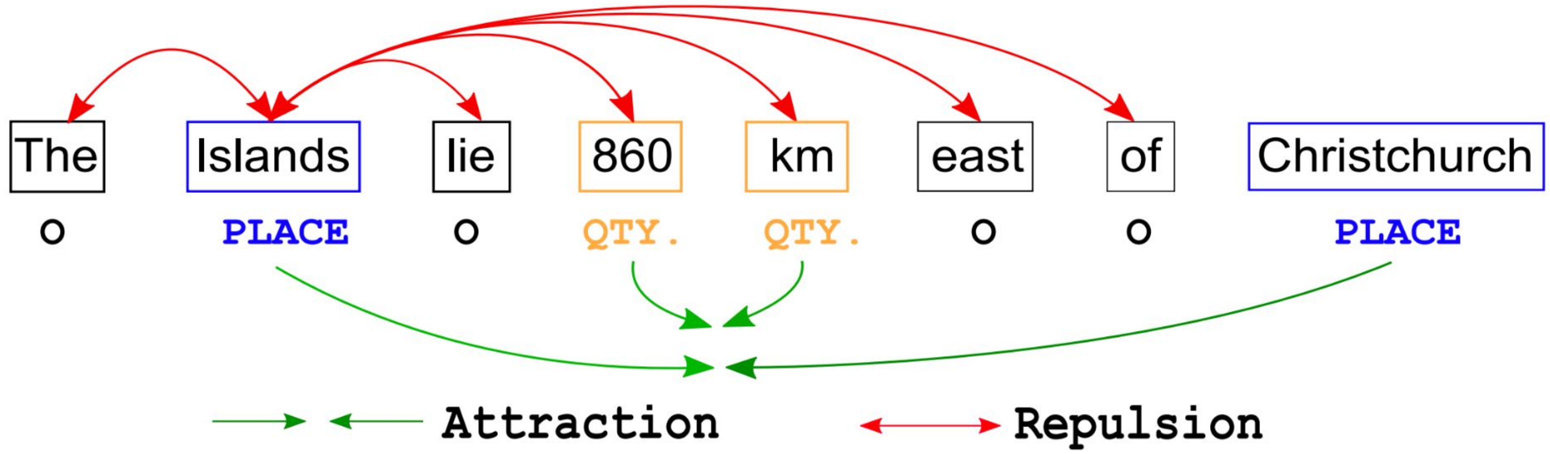
Person Date Location Other (“O”)

Traditionally, we have a training dataset with sufficient examples for each of the categories (e.g., OntoNotes).

Entities from low resource domains (e.g., Medical) suffer from the scarcity of data.

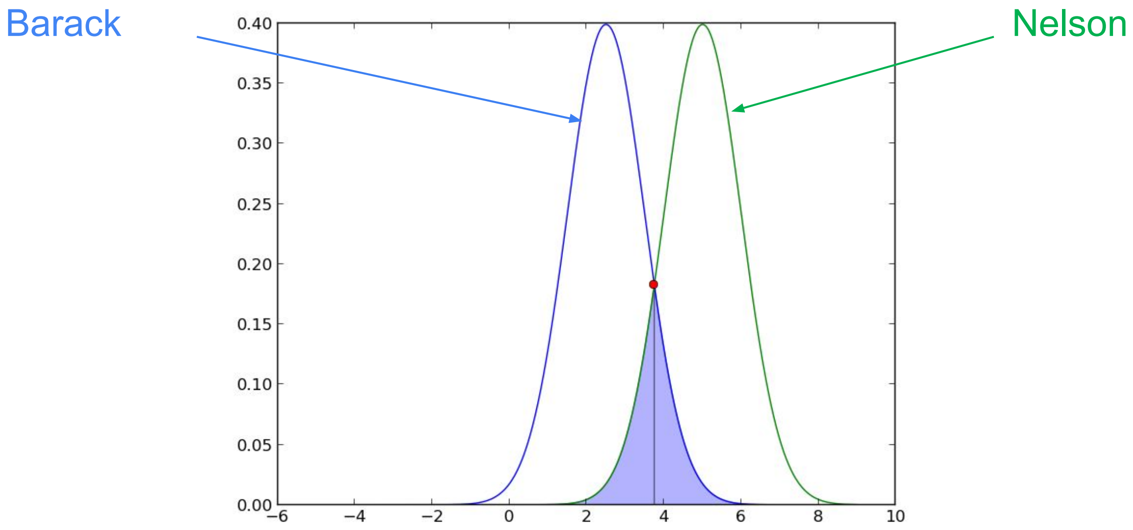
Can we learn NER models on new domains / new categories with only a few examples?

Main Idea: Contrastive Learning over Token Representations

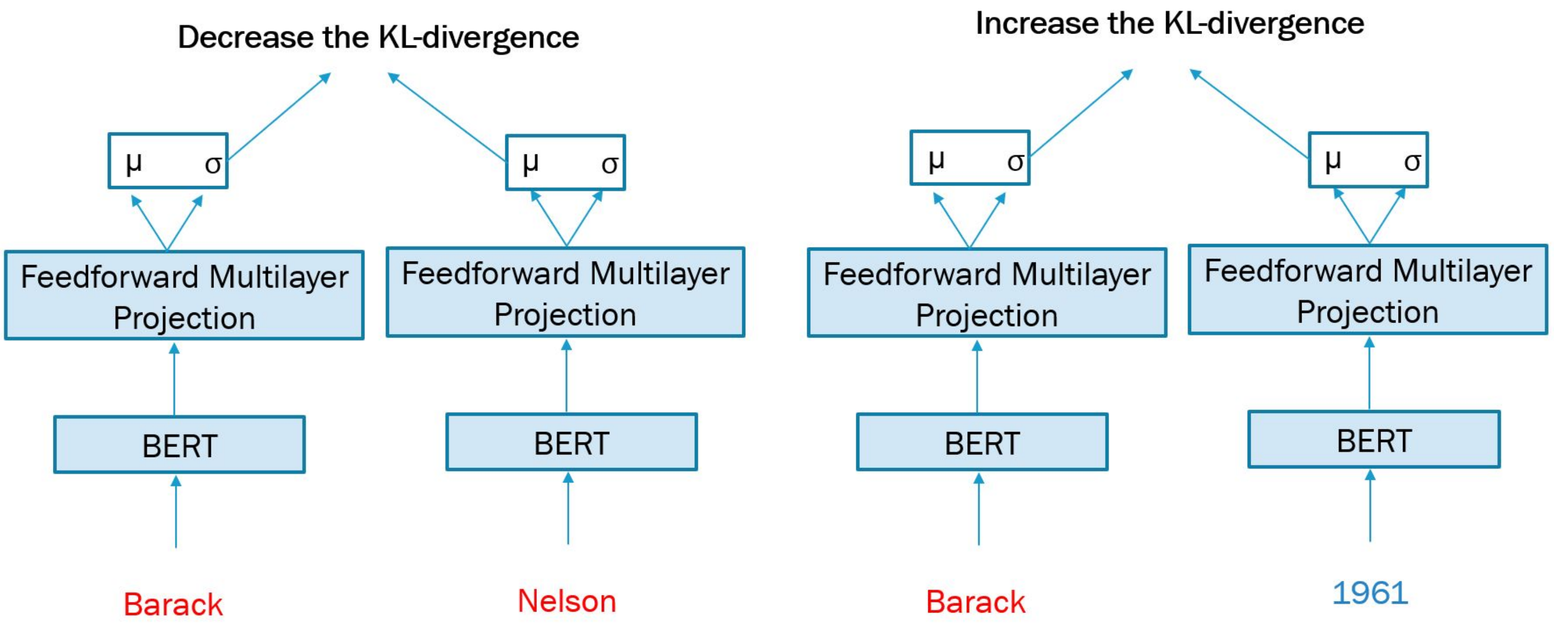


Idea 2: Gaussian Token Embeddings

- Model embeddings as distributions instead of points
- Word Representations via Gaussian Embedding [(Vilnis and McCallum, ICLR 2015)]
  - The token embeddings follow some Gaussian Distribution (mean μ, diagonal covariance σ).



Contrastive Learning with Gaussian Embeddings



Idea 3: Few-shot Learning from Source to Target

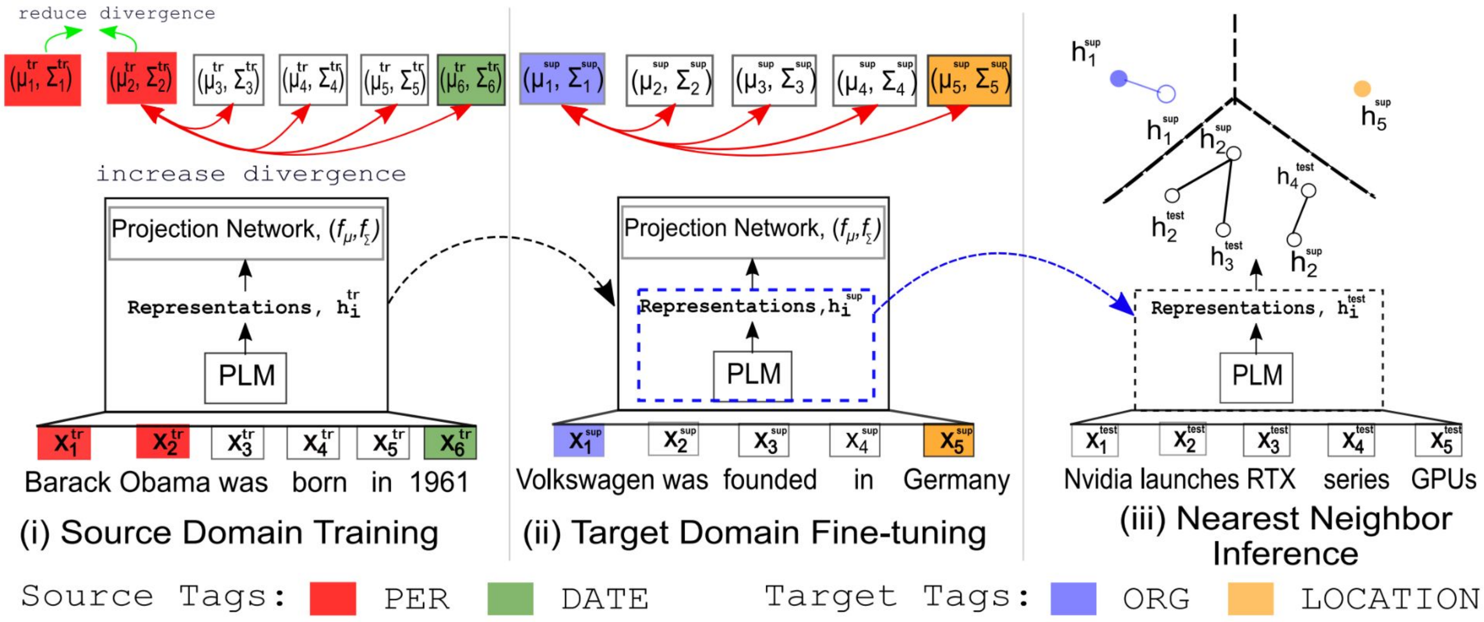
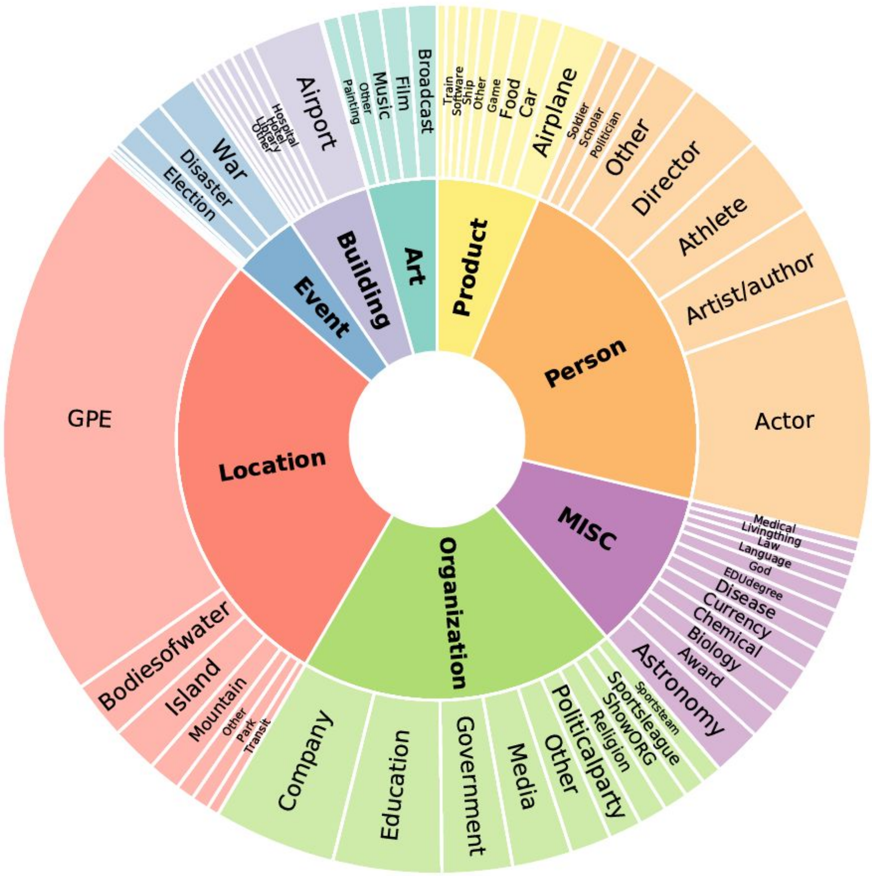


Figure 2: Illustration of our proposed CONTaiNER framework based on Contrastive Learning over Gaussian Embedddings: (i) Training in source domains using training NER labels PER and DATE, (ii) Fine-tuning to target domains using target NER labels ORG and LOCATION, (iii) Assigning labels to test samples via Nearest Neighbor support set labels.

Datasets

Dataset	Domain	# Class	# Sent
OntoNotes	General	18	76K
I2B2'14	Medical	23	140K
CoNLL'03	News	4	20K
WNUT'17	Social	6	5K
GUM	Mixed	11	3.5K
FEW-NERD	Wikipedia	66	188K

Table 1: Summary Statistics of Datasets



Few-NERD (Ding et al., 2021)

Generalization to Unseen Tags

Model	1-shot				5-shot			
	Group A	**Group B**	**Group C**	**Avg.**	**Group A**	**Group B**	**Group C**	**Avg.**
Proto	19.3 ± 3.9	22.7 ± 8.9	18.9 ± 7.9	20.3	30.5 ± 3.5	38.7 ± 5.6	41.1 ± 3.3	36.7
NNShot	28.5 ± 9.2	27.3 ± 12.3	21.4 ± 9.7	25.7	44.0 ± 2.1	51.6 ± 5.9	47.6 ± 2.8	47.7
StructShot	30.5 ± 12.3	28.8 ± 11.2	20.8 ± 9.9	26.7	47.5 ± 4.0	53.0 ± 7.9	48.7 ± 2.7	49.8
CONTaiNER	**32.2 ± 5.3**	**30.9 ± 11.6**	**32.9 ± 12.7**	**32.0**	**51.2 ± 5.9**	**55.9 ± 6.2**	**61.5 ± 2.7**	**56.2**
+ Viterbi	**32.4 ± 5.1**	**30.9 ± 11.6**	**33.0 ± 12.8**	**32.1**	**51.2 ± 6.0**	**56.0 ± 6.2**	**61.5 ± 2.7**	**56.2**

Table 2: F1 scores in Tag Set Extension on OntoNotes. Group A, B, C are three disjoint sets of entity types.

Generalization to Unseen Domains

Model	1-shot					5-shot				
	I2B2	CoNLL	WNUT	GUM	Avg.	I2B2	CoNLL	WNUT	GUM	Avg.
Proto	13.4 ± 3.0	49.9 ± 8.6	17.4 ± 4.9	17.8 ± 3.5	24.6	17.9 ± 1.8	61.3 ± 9.1	22.8 ± 4.5	19.5 ± 3.4	30.4
NNShot	15.3 ± 1.6	61.2 ± 10.4	22.7 ± 7.4	10.5 ± 2.9	27.4	22.0 ± 1.5	74.1 ± 2.3	27.3 ± 5.4	15.9 ± 1.8	34.8
StructShot	21.4 ± 3.8	**62.4 ± 10.5**	24.2 ± 8.0	7.8 ± 2.1	29.0	30.3 ± 2.1	74.8 ± 2.4	30.4 ± 6.5	13.3 ± 1.3	37.2
CONTaiNER	16.4 ± 1.7	57.8 ± 10.7	24.2 ± 2.9	17.9 ± 1.8	29.1	24.1 ± 1.9	72.8 ± 2.0	27.7 ± 2.2	24.4 ± 2.2	37.3
+ Viterbi	**21.5 ± 1.7**	61.2 ± 10.7	**27.5 ± 1.9**	**18.5 ± 4.9**	**32.2**	**36.7 ± 2.1**	**75.8 ± 2.7**	**32.5 ± 3.8**	**25.2 ± 2.7**	**42.6**

Table 3: F1 scores in Domain Extension with OntoNotes as the source domain.

Results on Few-NERD

Few-NERD (Inter): train and test classes can share coarse grained types

Model	5-way		10-way		Avg.
	1∼2 shot	5∼10 shot	1∼2 shot	5∼10 shot	
StructShot	**57.33**	57.16	**49.46**	49.39	53.34
ProtoBERT	44.44	58.80	39.09	53.97	49.08
NNShot	54.29	50.56	46.98	50.00	50.46
CONTaiNER	55.95	**61.83**	48.35	**57.12**	**55.81**
+ Viterbi	56.1	**61.90**	48.36	**57.13**	**55.87**

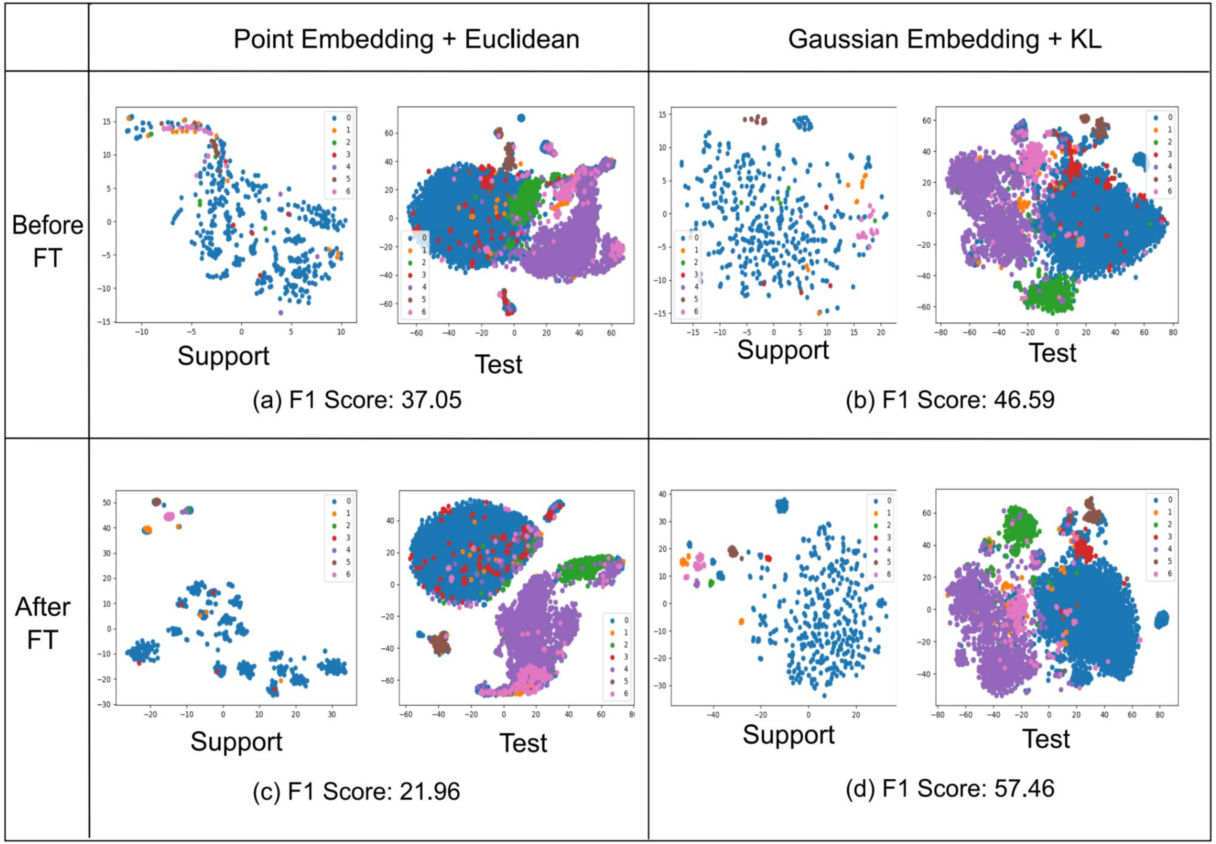
Table 4: F1 scores in FEW-NERD (INTER).

Few-NERD (Intra): Train and Test classes **do not share their coarse-grained types,** which makes it more challenging.

Model	5-way		10-way		Avg.
	1∼2 shot	5∼10 shot	1∼2 shot	5∼10 shot	
StructShot	35.92	38.83	25.38	26.39	31.63
ProtoBERT	23.45	41.93	19.76	34.61	29.94
NNShot	31.01	35.74	21.88	27.67	29.08
CONTaiNER	**40.43**	**53.70**	**33.84**	**47.49**	**43.87**
+ Viterbi	**40.40**	**53.71**	**33.82**	**47.51**	**43.86**

Table 3: F1 scores in FEW-NERD (INTRA).

Point Embeddings vs Gaussian Embeddings



Future Work

Few-shot Sequence Labelling

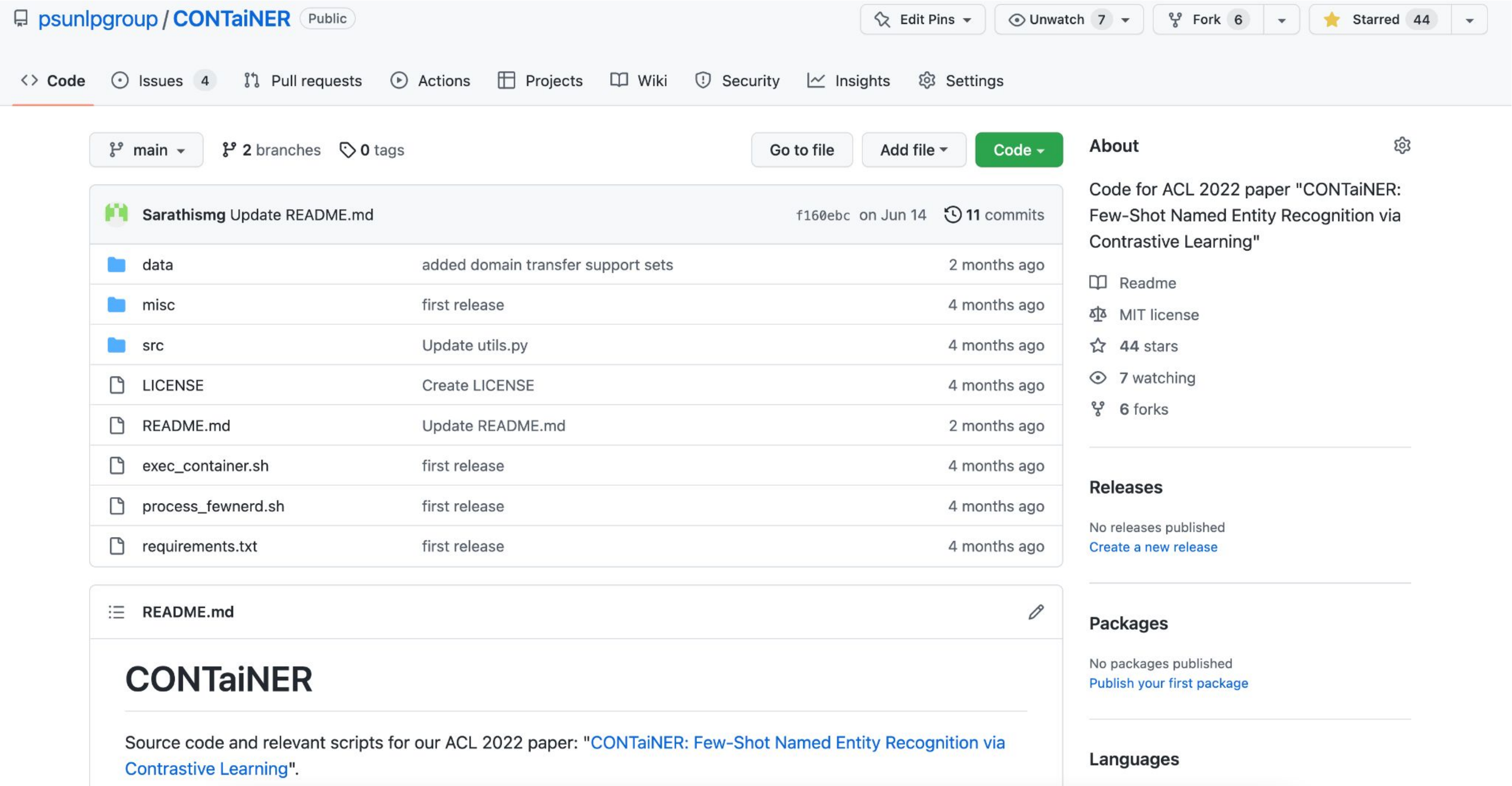
- How can we create few-shot learning models for structured predictions?
- NER -> Joint entity and relation extraction, Semantic role labeling, Coreference resolution, Semantic Parsing, ……

Contrastive Pretraining

- Is MLM the best pretraining strategy?
- Contrastive Pretraining based on sentence embeddings is very slow.
- How can we use contrastive learning on tokens to do pretraining?

CONTaiNER Code

https://github.com/psunlpgroup/CONTaiNER



Full Version of Tutorial

https://contrastive-nlp-tutorial.github.io/

Contrastive Data and Learning for Natural Language Processing

Tutorial at NAACL 2022 at Seattle, WA. July 10 - July 15, 2022

Tutorial Time and Location

Location: Columbia A + Zoom
Time: 2:00-5:30pm PDT, July 10, 2022
Zoom Q&A sessions: 1:30 - 2:00pm, 6:00 - 6:45pm PDT, July 10, 2022

Tutorial Materials

1. Tutorial abstract in the conference proceeding [PDF]
2. Tutorial slides [slides]
3. Tutorial video [video]
4. Paper reading list of contrastive learning for NLP [Github]

Thanks! Any Questions?

Contact

- https://ryanzhumich.github.io/
- rmz5227@psu.edu
- https://github.com/psunlpgroup

PennState