Injecting Wiktionary to improve token-level contextual representations using contrastive learning

Anna Mosolova[1,2], Marie Candito[1], Carlos Ramisch[2]
[1]Université Paris Cité, CNRS, LLF, Paris, France
[2]Aix Marseille Univ, CNRS, LIS, Marseille, France

Overview

Introduction

Problem:

- Contextualized token embeddings provide one representation per occurrence:
  - vectors of the same word sense are not close to each other [Ethayarajh, 2019]

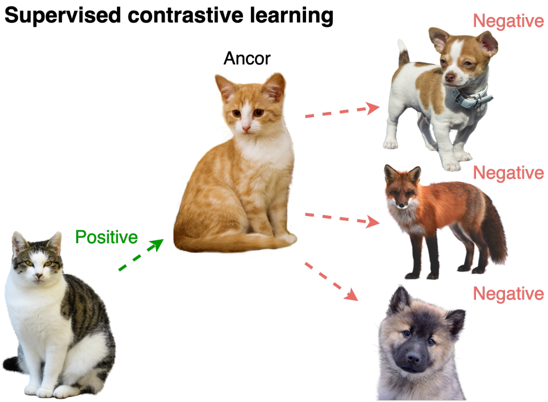
Our solution:

- **Tuning** of token-level contextual representations using **contrastive learning** with hand-crafted **lexicons**
- **Reducing** dimensions of the resulting embeddings

Contrastive learning

Contrastive learning main idea:

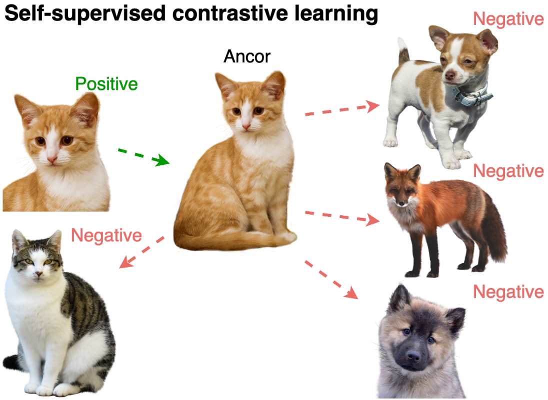
- bringing representations of **two** objects of the same class (or of an object and its augmented version) closer
- while pushing away all other objects



Contrastive learning

Contrastive learning main idea:

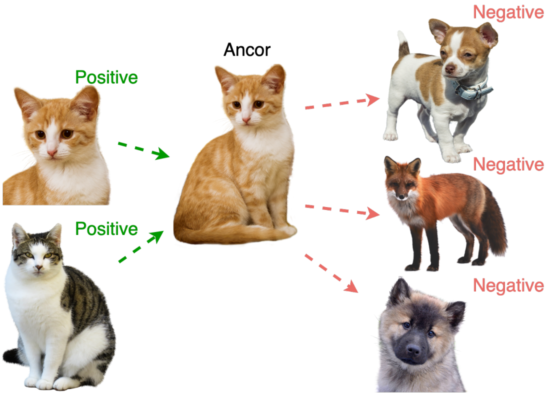
- bringing representations of **two** objects of the same class (or of an object and its augmented version) closer
- while pushing away all other objects



Supervised contrastive learning

In CV, Supervised CL with **multiple** positives was proposed [Khosla et al., 2020]:

- bringing representations of **all objects** of the same class closer
- while pushing away all other objects



Self-supervised contrastive learning in NLP

Positive examples in NLP come from self-supervision mainly

Self-augmentation methods for sentence representations:

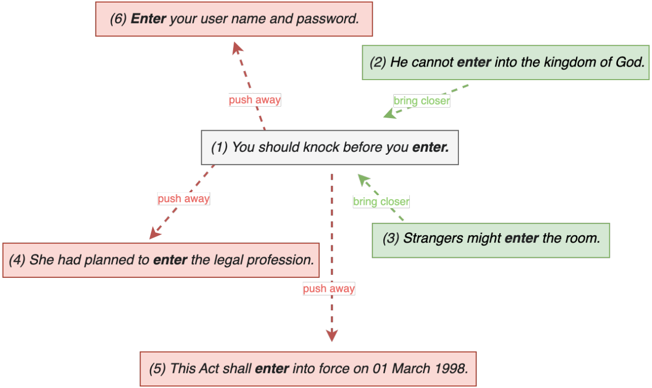
- back translation [Fang et al., 2020]
- text corruption [Liu et al., 2021a]
- dropout [Gao et al., 2021, Chuang et al., 2022]

Self-augmentation methods for token representations:

- masking random words in context [Liu et al., 2021a, Liu et al., 2021b]
- dropout [Liu et al., 2021a, Liu et al., 2021b]

Injecting lexicon sense examples through CL

- Supervised contrastive learning with multiple positives
- Wiktionary: example sentences for each sense
    - Examples for the same sense: same class (positive examples)
    - Other examples for the same lemma: other class (negative examples)



PLM fine-tuning experiments: some details

Training dataset - examples from Wiktionary:

- All verbs having from 1 to 10 senses[1]
- Divided into 90/5/5% (train, dev, test)

Examples	Verbs	Senses
68,271	13,118	26,398

- Mean nb of examples per sense: 2.59
- Mean nb of senses per verb: 2.01
- Mean nb of examples per verb: 5.21

Model for fine-tuning - bert-base-uncased

[1] Except verbs having a single sense with a single example and multiword verbs

Intrinsic evaluation: Word-in-Context task

Word-in-Context task [Pilehvar and Camacho-Collados, 2019]:

- Predict whether one target word in two sentences is used in the same sense or not
- Example:
    - **Kill** the engine.
    - He **killed** the ball.
    - Answer: False

Intrinsic evaluation: Word-in-Context task

Motivation:

- To tune the hyperparameters:
  - Training: learning rate, epochs, loss parameter τ
  - Dimensionality reduction (PCA): number of components, whitening application
- To evaluate if fine-tuning works
  - Compare ourselves to the previous SoTA: MirrorWiC [Liu et al., 2021b]
    - MirrorWiC: CL with self-augmentation on Wikipedia examples

Algorithm:

- Unsupervised approach: Threshold-based classifier on the cosine similarity between the target token embeddings

Word-in-Context datasets

Three WiC datasets for the evaluation:

- Original WiC
- New WiC datasets:
  - Wiktionary WiC
    - Development and test parts of the Wiktionary dataset
  - Framenet WiC
    - Predict whether one target word in two sentences evokes the same frame or not

Dataset	Dev	Test
Original WiC	638	1400
Wiktionary WiC	1200	1200
Framenet WiC	1800	1700

Unsupervised WiC results on dev and test set

Model	Wikt WiC	Frame WiC	Orig WiC
BERT	58.0	70.9	67.9
BERT+PCA	58.9	73.9	69.6
BERT+FT	**65.1**(±0.3)	73.6(±0.4)	72.2(±0.8)
BERT+FT+PCA	64.8(±0.5)	**76.0**(±0.2)	**73.5**(±0.5)
MirrorWiC	-	-	71.9

Table: Accuracy results on the development sets.

Model	Wikt WiC	Frame WiC	Orig WiC
BERT	55.9	67.3	65.4
BERT+PCA	59.6	72.4	68.4
BERT+FT	70.0(±0.9)	69.6(±0.4)	69.6(±0.6)
BERT+FT+PCA	**70.5**(±0.8)	**73.1**(±0.4)	**71.4**(±0.2)
MirrorWiC	-	-	69.6

Table: Accuracy results on the test sets.

→ PCA application **improves** the results even before fine-tuning

→ Major **improvements** on all datasets after fine-tuning

→ New **SoTA** on the original WiC dataset in unsupervised settings

→ Same tendencies after fine-tuning RoBERTA, BERT large, CamamBERT and FlauBERT

Extrinsic evaluation: Frame induction

Frame induction: identification of semantic classes (frames) that group senses of different lemmas

- Example:
  - IBM has **opted** for the mouse stick in the middle of the keyboard.
  - Greek islanders **chose** to leave rather than live in poverty and terror.
  - Frame: Choosing

Frame induction algorithm

Dataset and **algorithm** (with modifications) are coming from [Yamada et al., 2021]:

- Two-step clustering:
  - 1st step: Clustering instances of the same verb
  - 2nd step: Clustering across all verbs using clusters from the 1st step
- Instances are represented as contextualized embeddings of the target lemma

Results on frame induction dev and test sets

- **Purity** - "cleanliness" of each cluster
- **B-Cubed** - average precision and recall of each item

Model	F-Purity	F-B-Cubed
BERT	76.3	70.3
BERT+PCA	75.4	69.3
BERT+FT	**80.7**	**75.4**
BERT+FT+PCA	80.3	74.8

Table: Results on the development set.

Model	F-Purity	F-B-Cubed
BERT	69.8	61.3
BERT+PCA	68.6	58.3
BERT+FT	70.2	61.3
BERT+FT+PCA	**71.7**	**62.1**

Table: Results on the test set.

→ **Improvements** on the dev and test sets after fine-tuning

Results analysis

Model	Purity	Inv. Purity	F-Purity	B-Cubed Precision	B-Cubed Recall	F-B-Cubed
BERT	72.2	80.8	76.3	65.7	75.5	70.3
BERT+PCA	71.9	79.1	75.4	65.4	73.5	69.3
BERT+FT	**80.2**	**81.2**	**80.7**	**74.9**	**75.8**	**75.4**
BERT+FT+PCA	79.4	81.1	80.3	73.8	75.7	74.8

Table: Detailed results on the development set.

→ Purity and B-Cubed Precision **increase the most** after fine-tuning

→ Resulting clusters contain more same class items

Conclusion

Contributions:

- New approach for fine-tuning token-level representation of PLMs:
  - using contrastive learning with multiple positives
  - leveraging examples from the crowd-sourced lexicon (Wiktionary)
  - which can be extended to other languages (having a large Wiktionary)
- New SoTA result on the WiC test set in the unsupervised setting
- Gains on two new WiC test sets with different sense inventories
- Improvements on WiC tasks after fine-tuning other models (RoBERTa, BERT large) and other languages (French)
- Some improvements on the frame induction task

Thank you!

References I

Chuang, Y.-S., Dangovski, R., Luo, H., Zhang, Y., Chang, S., Soljacic, M., Li, S.-W., Yih, S., Kim, Y., and Glass, J. (2022).
DiffCSE: Difference-based contrastive learning for sentence embeddings.
In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States. Association for Computational Linguistics.

Ethayarajh, K. (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*

References II

Processing (EMNLP-IJCNLP), pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Fang, H., Wang, S., Zhou, M., Ding, J., and Xie, P. (2020).
Cert: Contrastive self-supervised learning for language understanding.
arXiv preprint arXiv:2005.12766.

Gao, T., Yao, X., and Chen, D. (2021).
Simcse: Simple contrastive learning of sentence embeddings.
arXiv preprint arXiv:2104.08821.

References III

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. (2020).
Supervised contrastive learning.
Advances in neural information processing systems, 33:18661–18673.

Liu, F., Vulić, I., Korhonen, A., and Collier, N. (2021a).
Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders.
In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1442–1459, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

References IV

Liu, Q., Liu, F., Collier, N., Korhonen, A., and Vulić, I. (2021b).
MirrorWiC: On eliciting word-in-context representations from pretrained language models.
In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 562–574, Online. Association for Computational Linguistics.

Pilehvar, M. T. and Camacho-Collados, J. (2019).
Wic: the word-in-context dataset for evaluating context-sensitive meaning representations.
In *Proceedings of NAACL-HLT*, pages 1267–1273.

References V

Yamada, K., Sasano, R., and Takeda, K. (2021).
Semantic Frame Induction using Masked Word Embeddings and Two-Step Clustering.

In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 811–816, Online. Association for Computational Linguistics.

Hyperparameters tuning on the WiC task

LR	E	τ	N comp.	Whitening	Macro-Accuracy	Orig-WiC	Framenet-WiC	Wikt-WiC
bert-base-uncased			-	-	65.6	67.9	70.9	58.0
bert-base-uncased			100	True	67.5	69.6	73.9	58.9
5e-6	2	0.5	100	True	**71.4**($\pm$0.1)	73.5($\pm$0.5)	76.0($\pm$0.2)	64.8($\pm$0.5)
5e-6	3	0.5	100	True	71.4($\pm$0.2)	73.7($\pm$0.4)	75.8($\pm$0.2)	64.8($\pm$0.3)
5e-6	3	0.5	300	True	71.4($\pm$0.4)	72.0($\pm$0.7)	77.6($\pm$0.4)	64.4($\pm$0.4)
5e-6	2	0.5	300	False	71.3($\pm$0.2)	**73.9**($\pm$0.4)	74.6($\pm$0.2)	65.3($\pm$0.4)
5e-6	2	0.5	300	True	71.3($\pm$0.4)	71.9($\pm$0.6)	**77.8**($\pm$0.3)	64.1($\pm$0.6)
5e-6	3	0.5	400	True	71.2($\pm$0.4)	72.0($\pm$0.8)	77.5($\pm$0.4)	64.1($\pm$0.5)
5e-6	3	0.5	200	True	71.2($\pm$0.2)	72.6($\pm$0.5)	76.7($\pm$0.2)	64.3($\pm$0.4)
5e-6	2	0.5	200	False	71.2($\pm$0.3)	73.5($\pm$0.5)	74.6($\pm$0.3)	**65.4**($\pm$0.3)
MirrorWiC			-	-	-	71.9	-	-