Towards Efficient, General, and Robust Entity Disambiguation Systems

Lihu Chen[1], Gaël Varoquaux[2] and Fabian M. Suchanek[1]

[1] LTCI, Télécom Paris, Institut Polytechnique de Paris, France

[2] Soda, Inria Saclay, Université Paris-Saclay, France

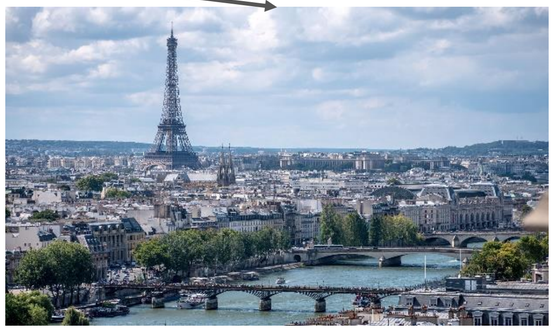
Background

Entity Disambiguation is the task of mapping entity mentions in text documents to standard entities in a given knowledge base

Paris is the son of King Priam

?

Paris (mythology), a prince of Troy in Greek mythology



Paris, the capital of France

Background

Questions

The field of Entity Disambiguation is very vibrant with many novel work popping up. However, there are several questions that are underexplored by prior work:

- *Can we use a small model to approach the performance of a big model?* → **Efficiency**
- *How to develop a single disambiguation system adapted to multiple domains?* → **Generalizability**
- *Are existing systems robust to out-of-vocabulary problems?* → **Robustness**

Lihu Chen

Gaël Varoquaux

Fabian Suchanek

Outline

Q: Can we use a small model to approach the performance of a big model? → Efficiency

Q: How to develop a single disambiguation system adapted to multiple domains? → Generalizability

Q: Are existing systems robust to out-of-vocabulary problems? → Robustness

Outline

Q: Can we use a small model to approach the performance of a big model? → Efficiency

Q: How to develop a single disambiguation system adapted to multiple domains? → Generalizability

Q: Are existing systems robust to out-of-vocabulary problems? → Robustness

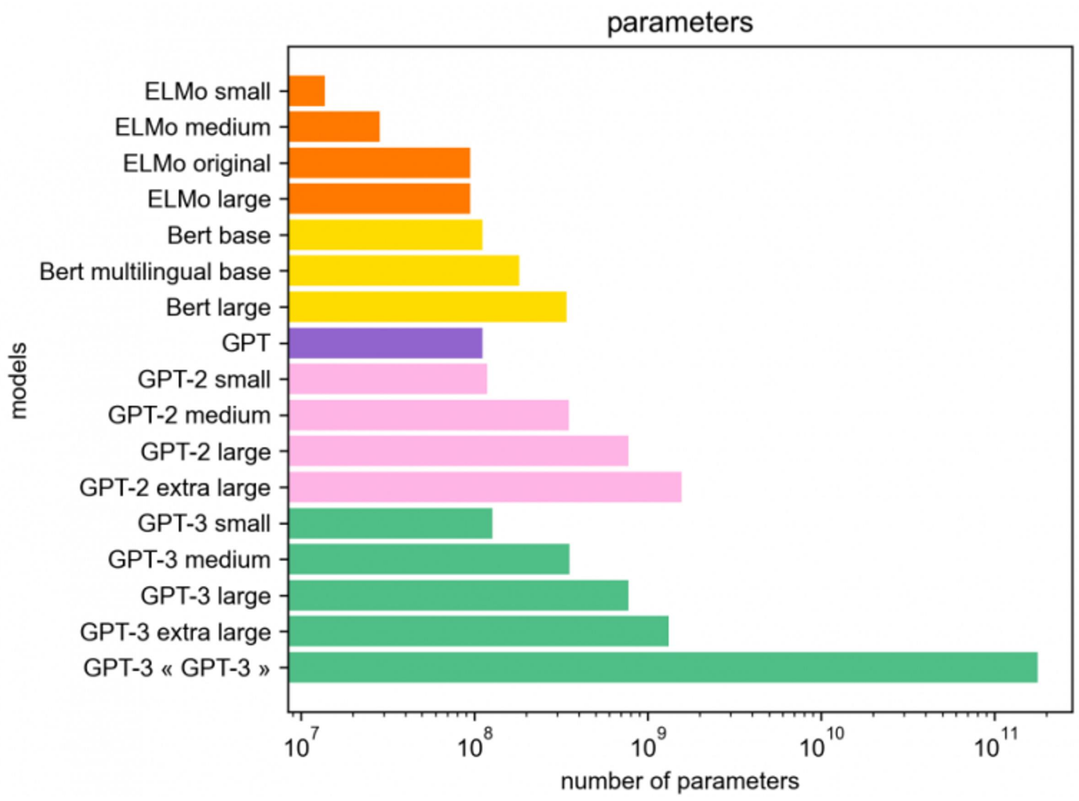
Biomedical Entity Disambiguation

In the biomedical domain, entity disambiguation maps mentions of diseases, drugs, and measures to normalized entities in standard vocabularies

Alstrom syndrome is a rare disorder characterized by retinal degeneration and type 2 diabetes.

Alstrom syndrome	*retinal degeneration*	*type 2 diabetes*
1. Alström–Hallgren syndrome	1. Retinal Degeneration	1. Type 1 Diabetes
2. Alsing Syndrome	2. Late-Onset Retinal Degeneration	2. Diabetes Mellitus, Type 2
3. ……	3. ……	3. ……

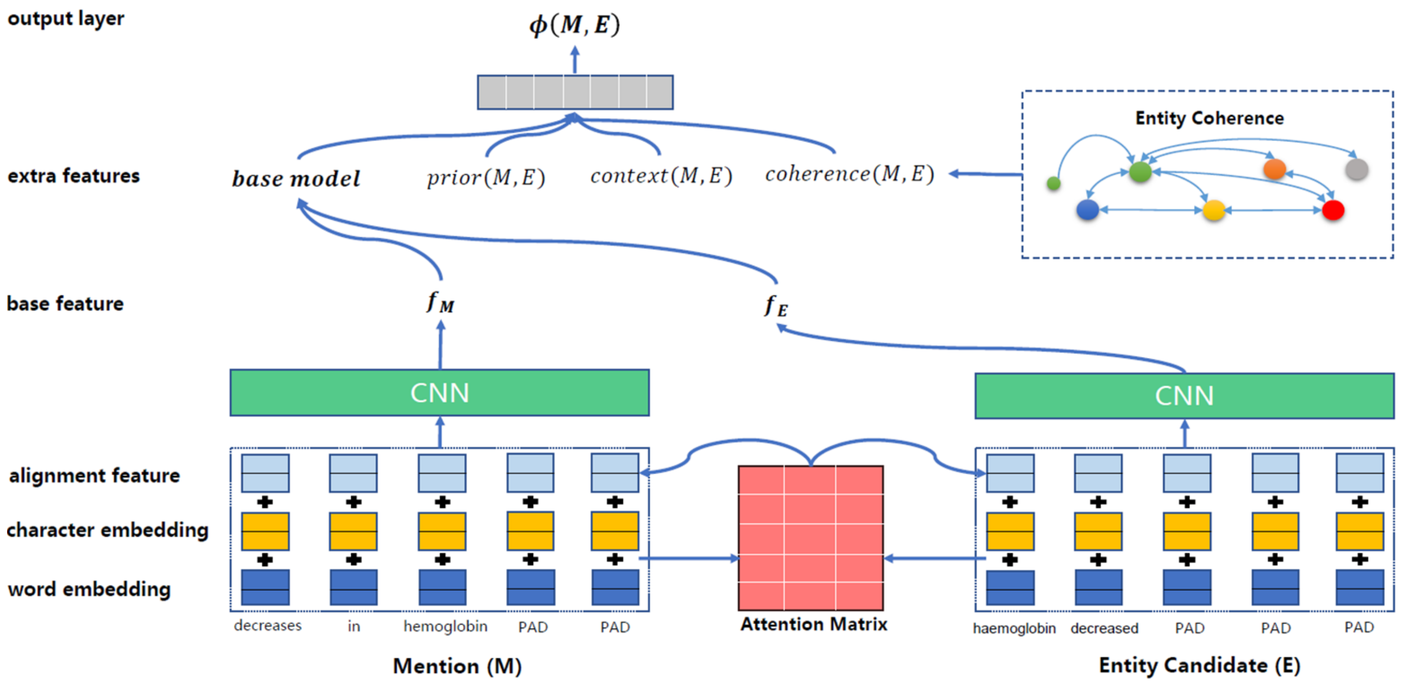
Motivation



source

Can we use a small model to approach the performance of a big model?

Our Lightweight Model



Model Architecture

Experimental Results

Model	ShARe/CLEF	NCBI	ADR
DNorm (Leaman, Islamaj Doğan, and Lu 2013)	-	82.20±3.09	-
UWM (Ghiasvand and Kate 2014)	89.50±1.02	-	-
Sieve-based Model (D'Souza and Ng 2015)	90.75±0.96	84.65±3.00	-
TaggerOne (Leaman and Lu 2016)	-	88.80±2.59	-
Learning to Rank (Xu et al. 2017)	-	-	92.05±0.84
CNN-based Ranking (Li et al. 2017)	90.30±1.00	86.10±2.79	-
BERT-based Ranking (Ji, Wei, and Xu 2020)	**91.06±0.96**	89.06±2.63	**93.22±0.79**
Our Base Model	90.10±1.00	89.07±2.63	92.63±0.81
Our Base Model + Extra Features	90.43±0.99	**89.59±2.59**	92.74±0.80

Model	Parameters	ShARe/CLEF		NCBI		ADR		Avg	Speedup
		CPU	GPU	CPU	GPU	CPU	GPU		
BERT (large)	340M	2230s	1551s	353s	285s	2736s	1968s	1521s	12.3x
BERT (base)	110M	1847s	446s	443s	83s	1666s	605s	848s	6.4x
TinyBERT$_6$	67M	1618s	255s	344s	42s	2192s	322s	796s	6.0x
MobileBERT (base)	25.3M	1202s	330s	322s	58s	1562s	419s	649s	4.7x
ALBERT (base)	12M	836s	**129s**	101s	24s	1192s	170s	409s	2.6x
Our Base Model	4.6M	**181s**	131s	**38s**	**22s**	**196s**	**116s**	**114s**	-

Table 5: Number of model parameters and observed inference time

Our model achieves similar results while is much smaller (4.6M VS 110M)

Conclusion

- We propose a **simple** and **lightweight** neural model for biomedical entity disambiguation
- Our model achieve a performance that is **statistically indistinguishable** from BERT-based models
- Our model is **23x smaller** and **6.4x faster** than BERT-based models

Lihu Chen

Gaël Varoquaux

Fabian Suchanek

Paper & Code

Chen, Lihu, Gaël Varoquaux, and Fabian M. Suchanek. "A lightweight neural model for biomedical entity linking." Proceedings of the AAAI 2021.

Outline

Q: Can we use a small model to approach the performance of a big model? → Efficiency

Q: How to develop a single disambiguation system adapted to multiple domains? → Generalizability

Q: Are existing systems robust to out-of-vocabulary problems? → Robustness

Acronym Disambiguation

An acronym is an abbreviation formed from the initial letters of a longer name. Acronym Disambiguation (AD) is the task of mapping a given acronym in a given sentence to the intended long form.

This is the product's first true AI version, and it understands your voice instantly → **Artificial Intelligence**

In the United States, the AI for potassium for adults is 4.7 grams. → **Adequate Intake**

An example for the acronym "AI"

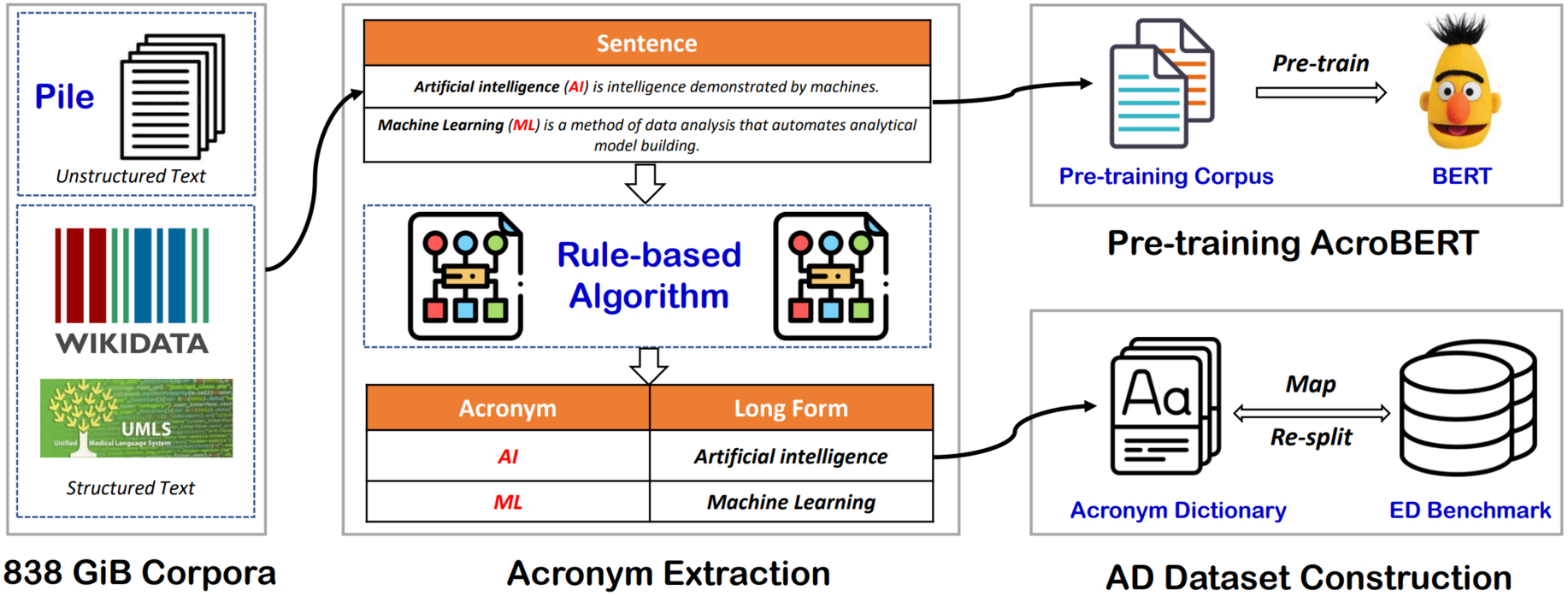
Motivation

ID	Long Form	Popularity	Domain
1	***Artificial Intelligence***	★★★★★	Computer Science
2	*Adequate Intake*	★★★★	Food and Nutrition
3	*Aromatase Inhibitor*	★★★	Chemistry
4	*Apoptotic Index*	★★★	Biomedicine
5	*Asynchronous Irregular*	★★★	Neuroscience
6	*Amnesty International*	★★	Organization
7	*Anterior Insula*	★★	Biomedicine
8	*Air India*	★★	Organization
9	***Article Influence***	★★	Science
.....			
2243	*Agricultural Implement*	★	Agriculture

Existing acronym disambiguation benchmarks and tools are limited to specific domains, and the size of prior benchmarks is rather small

Constructing GLADIS

To accelerate the research on acronym disambiguation, we construct a new benchmark named GLADIS (a **G**eneral and **L**arge **A**cronym **DIS**ambiguation benchmark) with three components and a pre-trained model named AcroBERT



Data Source

Subset	Domain	Size (GiB)
Pile-CC	Web Archive files	227.12
Books3	Books	100.96
Github	Open-source codes	95.16
PubMed Central	Biomedical articles	90.27
OpenWebText2	Reddit submissions	62.77
ArXiv	Research papers	56.21
FreeLaw	Legal proceedings	51.15
Stack Exchange	Question-answer texts	32.20
USPTO Backgrounds	Patents	22.90
PubMed Abstracts	Biomedical abstracts	19.26
OpenSubtitles	Subtitles	12.98
Gutenberg (PG-19)	Western literatures	10.88
DM Mathematics	mathematical problems	7.75
Wikipedia (en)	Wikipedia pages	6.38
BookCorpus2	Books	6.30
Ubuntu IRC	Chatlog data	5.52
EuroParl	Proceedings	4.59
HackerNews	Comments of social news	3.90
YoutubeSubtitles	YouTube subtitles	3.73
PhilPapers	Philosophy publications	2.38
NIH ExPorter	Awarded applications	1.89
Enron Emails	Emails	0.88
Wikidata Alias	Alias Table	11.00
UMLS Concept	Biomedical Vocabulary	1.96
Total	-	838.14

Acronym	Long Form	Provenance
ELEC	*Election Law Enforcement Commission*	*Christie, some legislators and the state Election Law Enforcement Commission (**ELEC**), have joined the comptroller in voicing support for the elimination of the loophole.*
ISR	*in-stent restenosis*	*Although conventional stents are routinely used in clinical procedures, clinical data shows that these stents are not capable of completely preventing in-stent restenosis (**ISR**) or restenosis caused by intimal hyperplasia.*
IL-6	*interleukin-6*	*Consistent blood markers in afflicted patients are normal to low white cell counts and elevated interleukin-6 (**IL-6**) levels which, among its many activities, signal the liver to increase synthesis and secretion of CRP.*
PCP	*Planar cell polarity*	*Establishment of photoreceptor cell polarity and ciliogenesis Planar cell polarity (**PCP**)-associated Prickle genes (Pk1 and Pk2) are tissue polarity genes necessary for the establishment of PCP in Drosophila.*
DEP	*dielectrophoretic*	*They included: a particle counter, trypan blue exclusion (Cedex), an in situ bulk capacitance probe, an off-line fluorescent flow cytometer, and a prototype dielectrophoretic (**DEP**) cytometer.*
AQP3	*aquaporin3*	*The laxative effect of bisacodyl is attributable to decreased aquaporin-3 expression in the colon induced by increased PGE2 secretion from macrophages.The purpose of this study was to investigate the role of aquaporin3 (**AQP3**) in the colon in the laxative effect of bisacodyl.*

Schwartz and Hearst, 2002[11]

We use 838GiB data to construct our new acronym dictionary

Constructing Datasets

Artificial Intelligence is intelligence demonstrated by machines

Entity Disambiguation Dataset

AI is intelligence demonstrated by machines

WikiData

WikilinksNED

General

MedMentions

Biomedical

SciAD

Scientific

We adapt the existing Entity Disambiguation datasets by replacing the long form of entity with the acronym

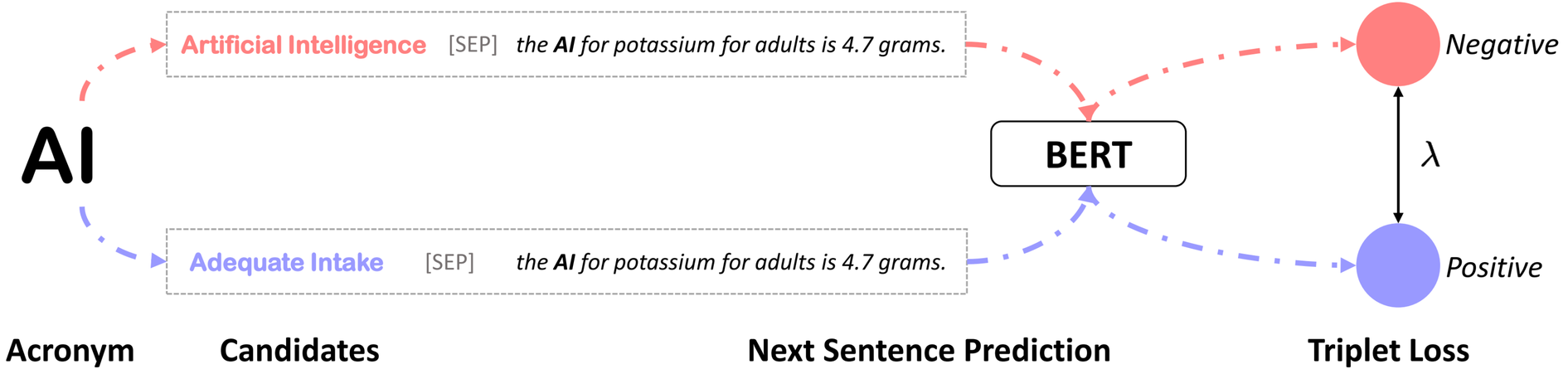
Components in GLADIS

	Source	Desc
Acronym Dictionary	Pile (MIT license), Wikidata, UMLS	1.6 million acronyms and 6.4 million long forms
Three Datasets	WikilinksNED Unseen, SciAD(CC BY-NC-SA 4.0), Medmentions(CC0 1.0)	three AD datasets that cover general, scientific, biomedical domains
A Pre-training Corpus	Pile (MIT license)	160 million sentences with acronyms
AcroBERT	BERT-based model	the first pre-trained language model for general acronym disambiguation

Statistics of our GLADIS benchmark

AcroBERT

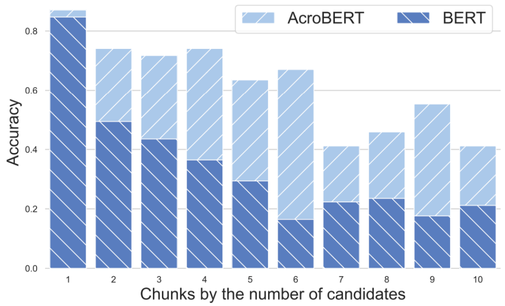
Currently, there is no other pre-trained model for general disambiguation. Our approach is the first that capitalizes on large-scale corpora and pre-training



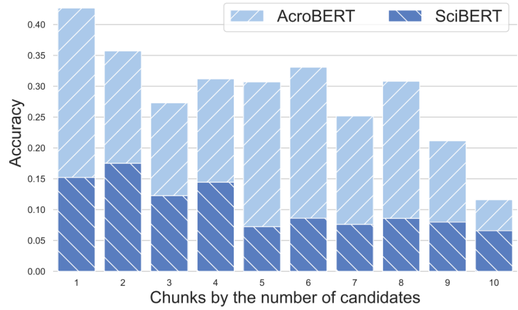
AcroBERT is pre-trained by using triplet loss

Experimental results

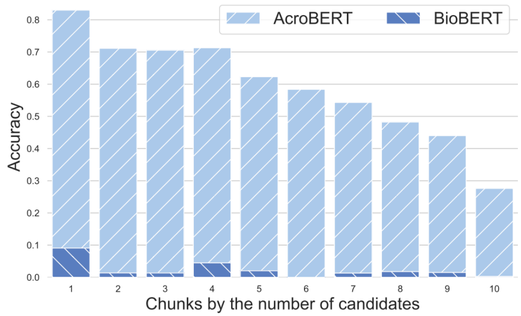
Model	General				Scientific				Biomedical				Avg	
	Dev		Test		Dev		Test		Dev		Test			
	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
BM25 (1995)	29.9	32.6	35.5	25.8	14.1	5.4	17.1	10.7	13.1	8.3	17.0	14.3	21.1	16.2
FastText (2017)	11.3	12.9	18.7	12.7	3.3	0.9	5.7	2.5	0.2	0.1	1.3	0.7	6.8	5.0
MadDog (2021)	28.1	11.7	29.9	23.1	17.8	15.5	22.4	17.9	33.8	19.3	41.2	35.9	28.9	20.6
BERT (2018)	32.3	32.5	37.7	28.2	15.1	5.8	17.6	9.3	3.1	1.3	3.5	2.1	18.2	13.2
Popularity-Ours	35.2	39.1	39.0	43.2	5.5	22.9	4.9	12.3	46.0	61.3	49.9	54.0	30.1	38.8
AcroBERT	**74.7**	**78.8**	**70.0**	**72.0**	**26.9**	**36.6**	**28.8**	**27.4**	**58.4**	**66.0**	**59.9**	**61.4**	**53.1**	**57.0**



General



Scientific



Biomedical

AcroBERT significantly outperforms existing systems across multiple domains

Conclusion

- We have presented GLADIS, a challenging and large benchmark for AD
- We have also proposed AcroBERT, the first pre-trained model for general AD

*More often than not, **PR** is a preemptive process. Celebrity publicists are paid lots of money to keep certain stories out of the news.*

→ **AcroBERT** → [(PR, Public Relations)]

Input

Output

Paper

GLADIS

AcroBERT

The 17th Conference of the European Chapter of the Association for Computational Linguistics

European Chapter | EACL

Chen, Lihu, Gaël Varoquaux, and Fabian M. Suchanek. "GLADIS: A General and Large Acronym Disambiguation Benchmark" In Proceedings of the EACL 2023.

Outline

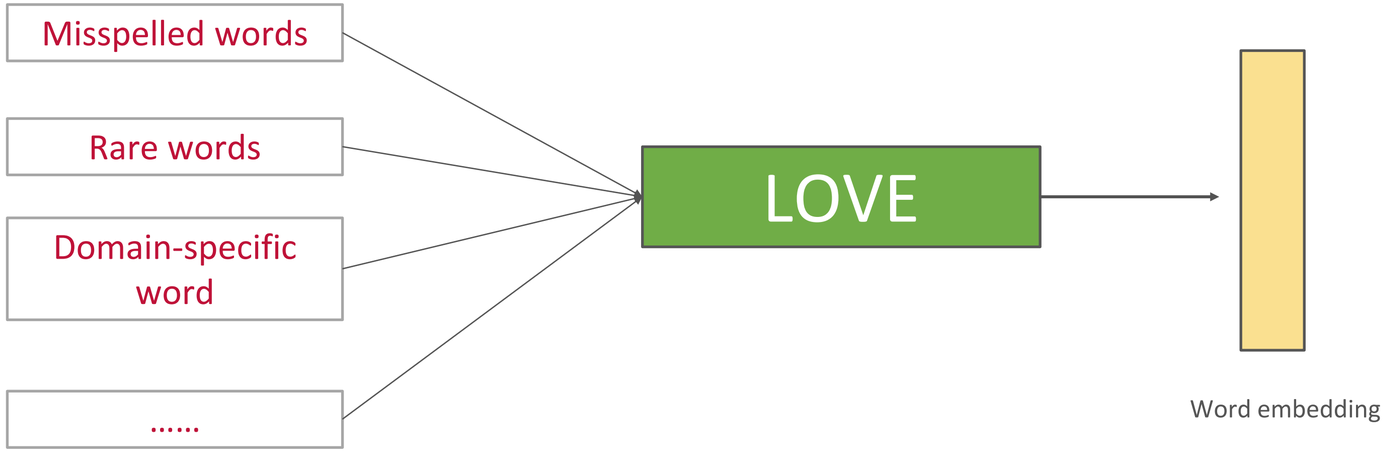
Q: Can we use a small model to approach the performance of a big model? → Efficiency

Q: How to develop a single disambiguation system adapted to multiple domains? → Generalizability

Q: Are existing systems robust to out-of-vocabulary problems? → Robustness

LOVE

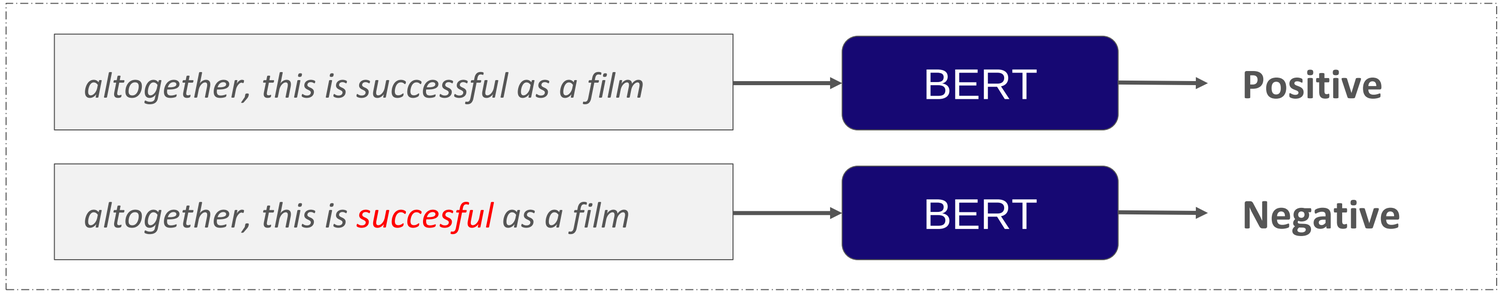
LOVE means **L**earning **O**ut-of-**V**ocabulary **E**mbeddings.



LOVE can generate embeddings for arbitrary words

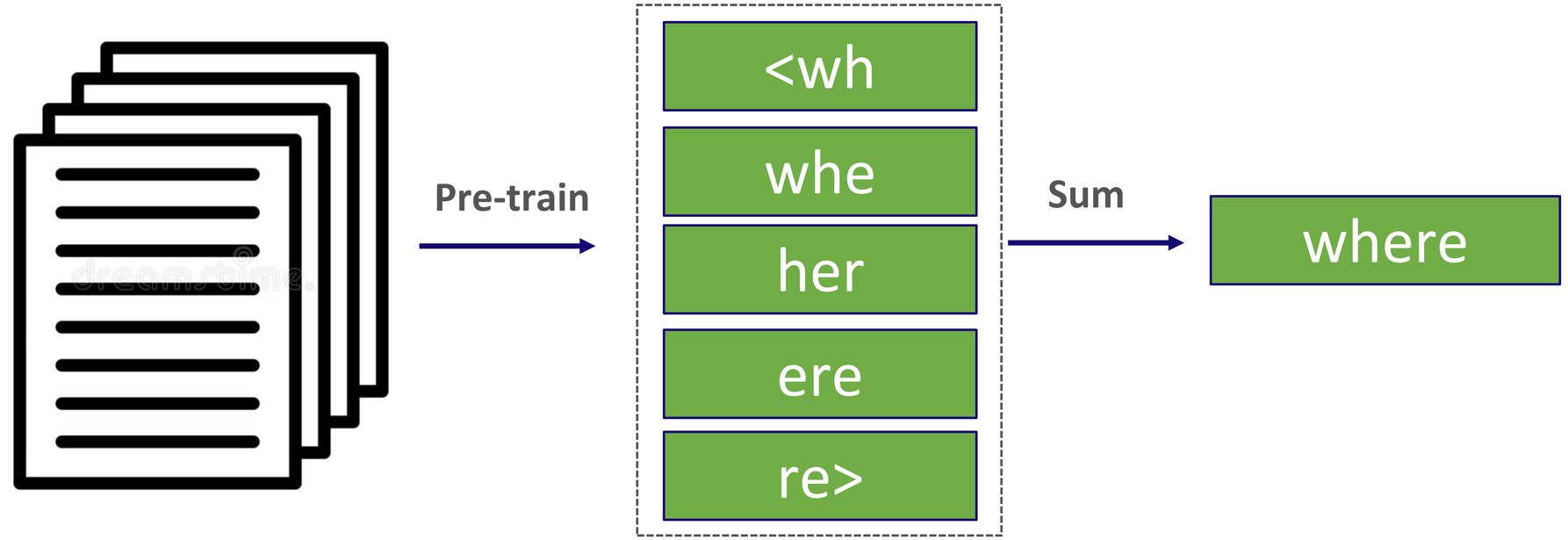
Motivation

State-of-the-art NLP systems rely on pre-trained language models, but these are brittle when faced with Out-of-Vocabulary (OOV) words.



Minor character perturbations can flip the prediction of a model!

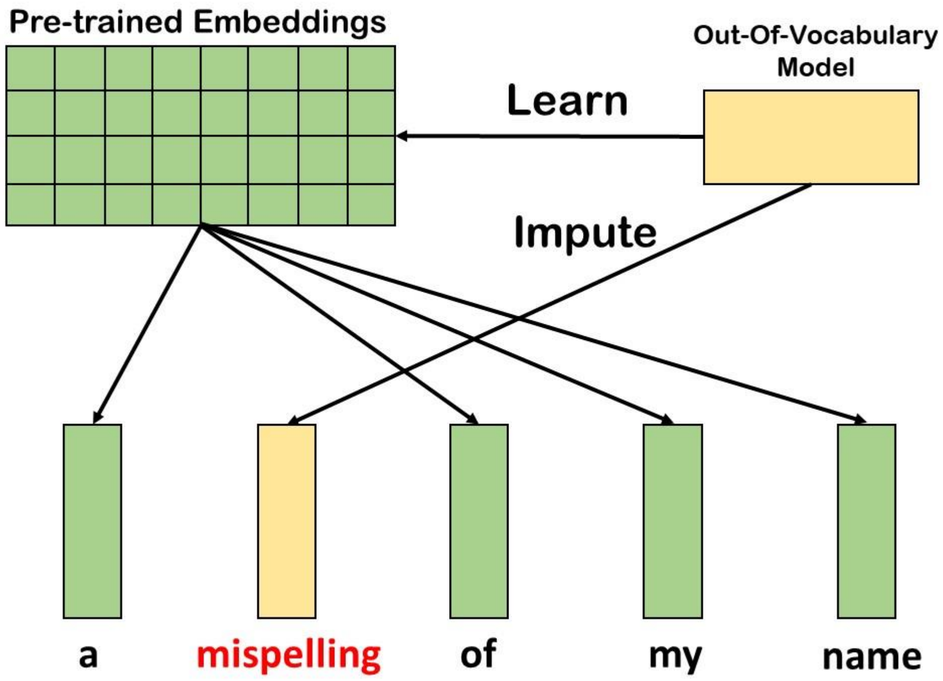
Existing Work



Large-scale corpora | N-gram characters | Word vector

Pretraining word embeddings with morphological features:
FastText [4], CharacterBERT [8], CharBERT [9]

Existing Work



	Input	Encoder	Loss
MIMICK (2017)	character sequence {s,p,e,l,l}	RNNs	$\mathcal{L}_{\text{dis}}$
BoS (2018)	n-gram subword {spe,pel,ell}	SUM	$\mathcal{L}_{\text{dis}}$
KVQ-FH (2019)	adapted n-gram subword {spe,pel,ell}	Attention	$\mathcal{L}_{\text{dis}}$

Table 1: Details of different mimick-like models, with the word `spell` as an example.

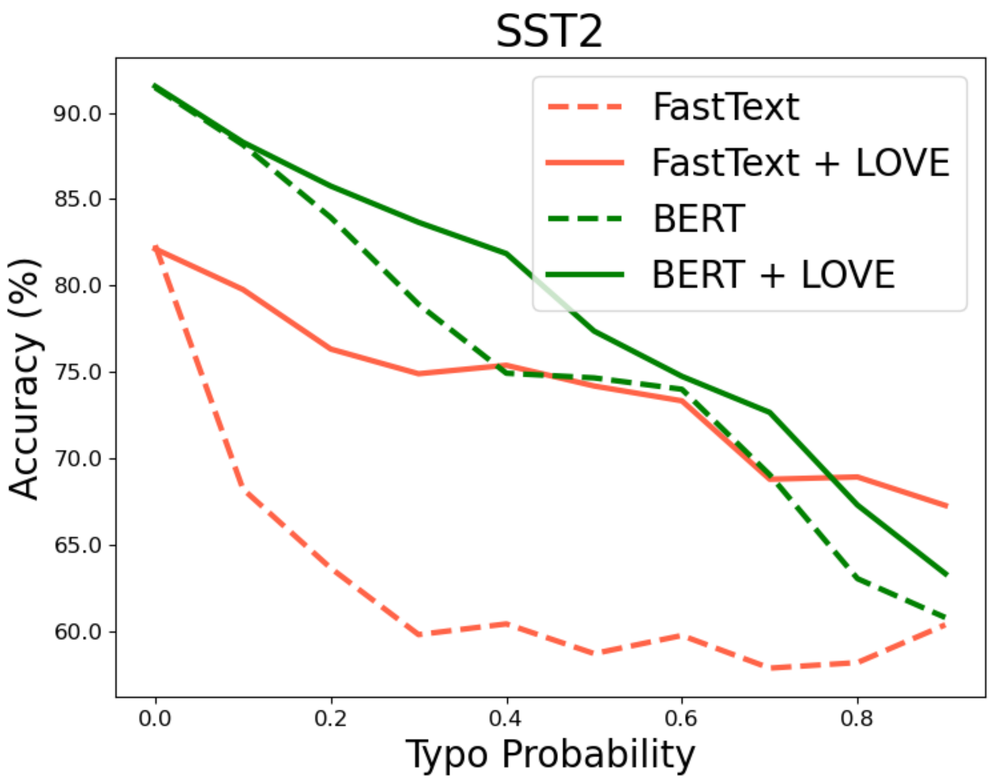
Mimicking the behavior of pre-trained embeddings using only the surface form: MIMICK [5], Bos [6], KVQ-FH [7].

Existing Work

The limitations of existing mimic-like work

- Remain bound in the trade-off between complexity and performance (FastText ~900M, BoS ~500M)
- Cannot be used with existing pre-trained language models such as BERT

A First Glance of LOVE



FastText: 900M

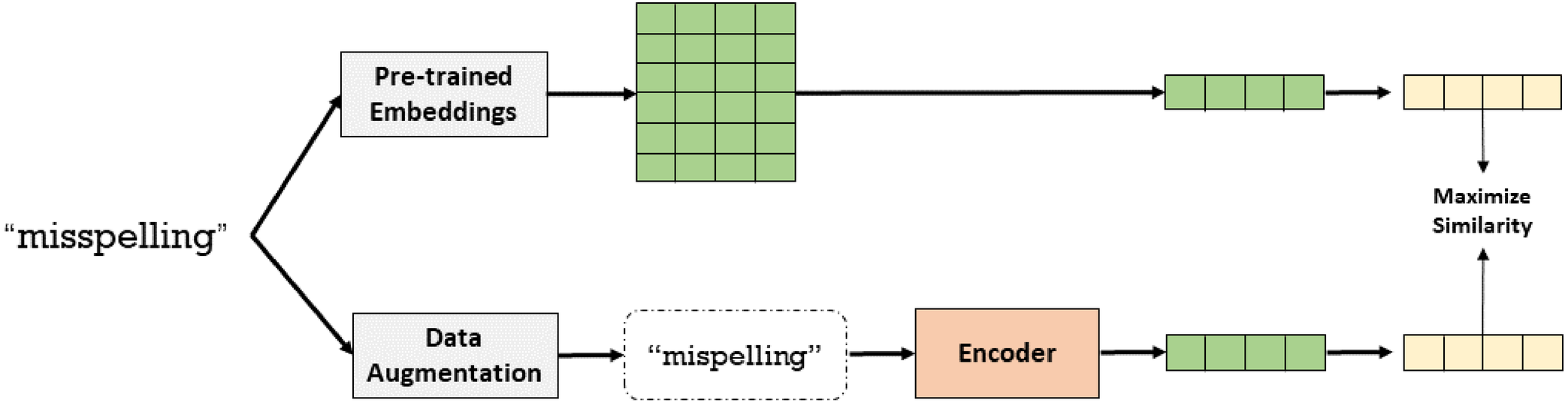
LOVE: 6.5M

LOVE makes language models more robust with little cost!

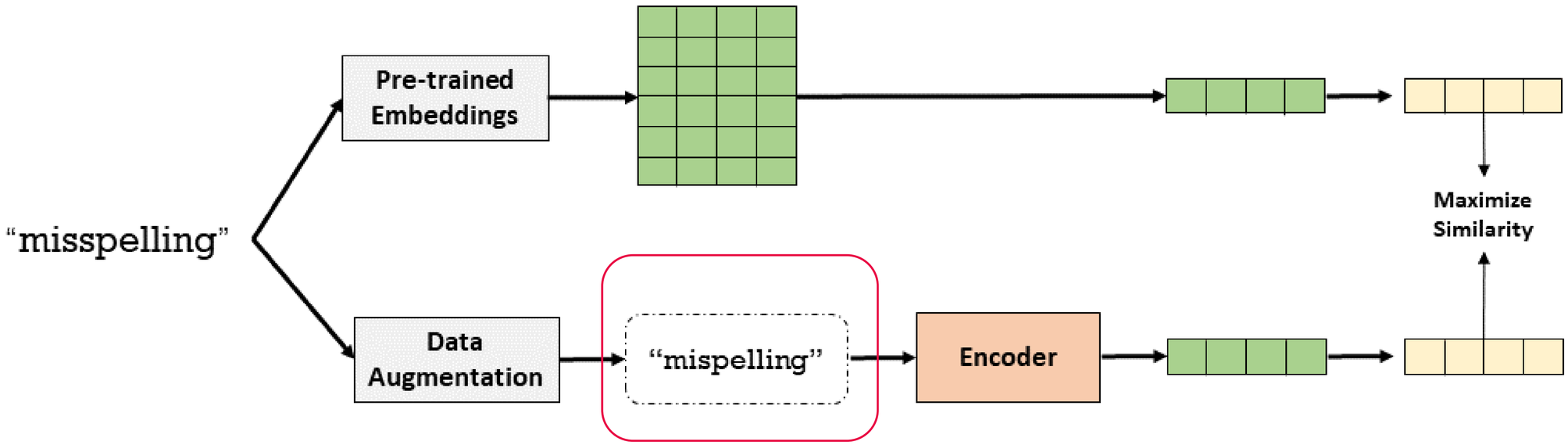
Framework

LOVE (Learning Out-of-Vocabulary Embeddings) draws on the principles of contrastive learning to maximize the similarity between target and generated vectors, and to push apart negative pairs.

Five key components: **(1) Mixed Input; (2) PAM encoder; (3) Contrastive Loss; (4) Data Augmentation; (5) Hard Negatives**



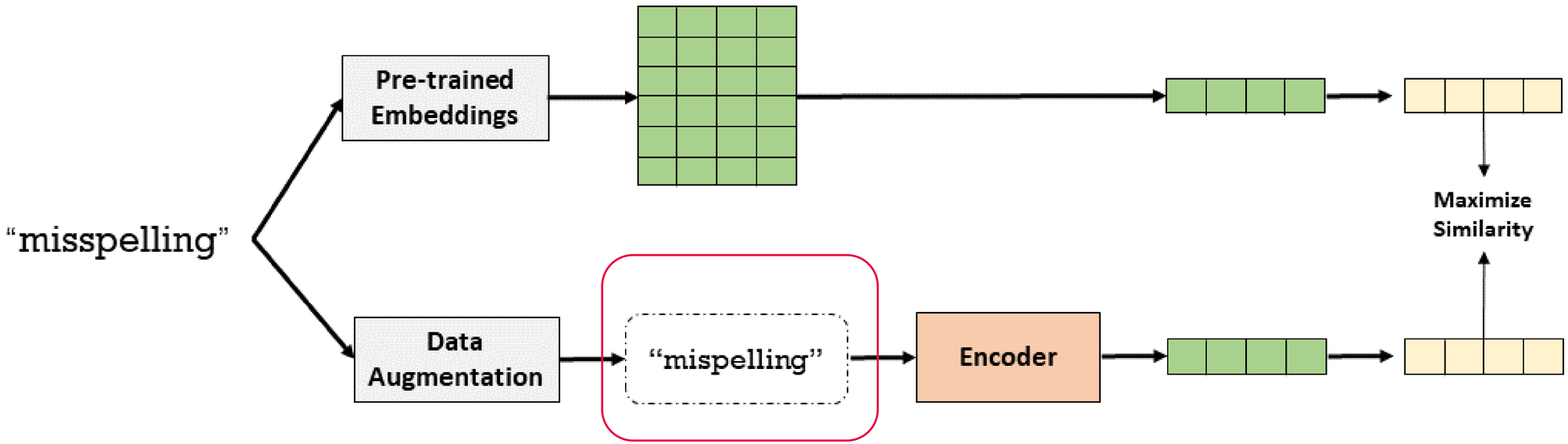
Input Method



"misspelling" $\Longrightarrow$ *{m, i, s, s, p, e, l, l, i, n, g}*

Characters cannot yield good word representations

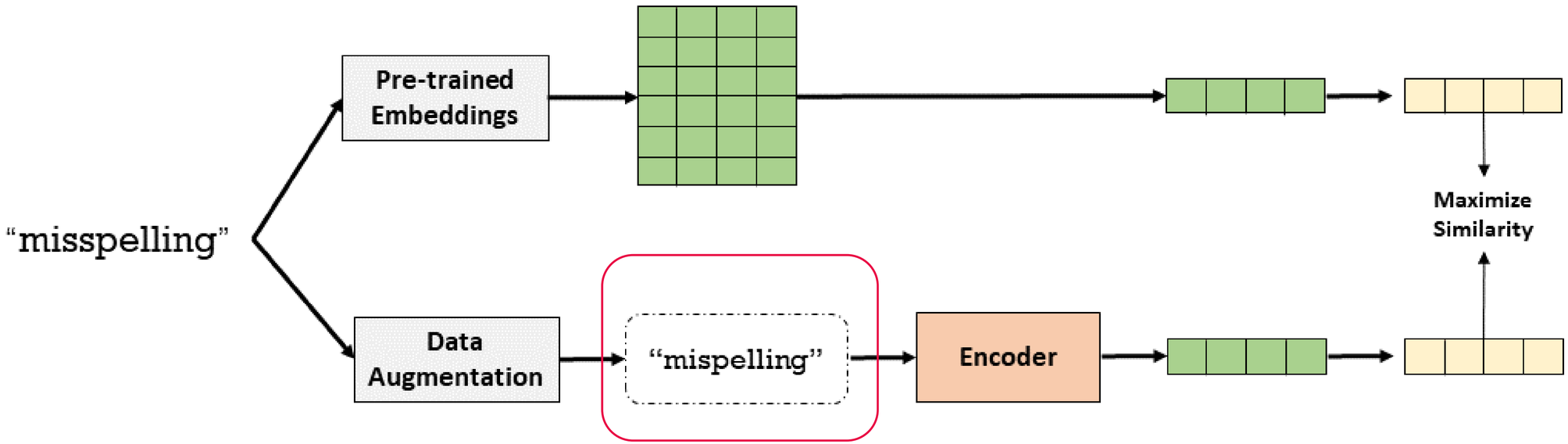
Input Method



"misspelling" ⟹ {*mis*, *iss*, *ssp*, *spe*, *pel*, *ell*, *lli*, *lin*, *ing*, *miss*, *issp*, *sspe*, *spel*, *pell*, *elli*, *llin*, *ling*, *missp*, *isspe*, *sspel*, *spell*, *pelli*, *ellin*, *lling*}

N-Gram Characters are effective while highly redundant

Input Method

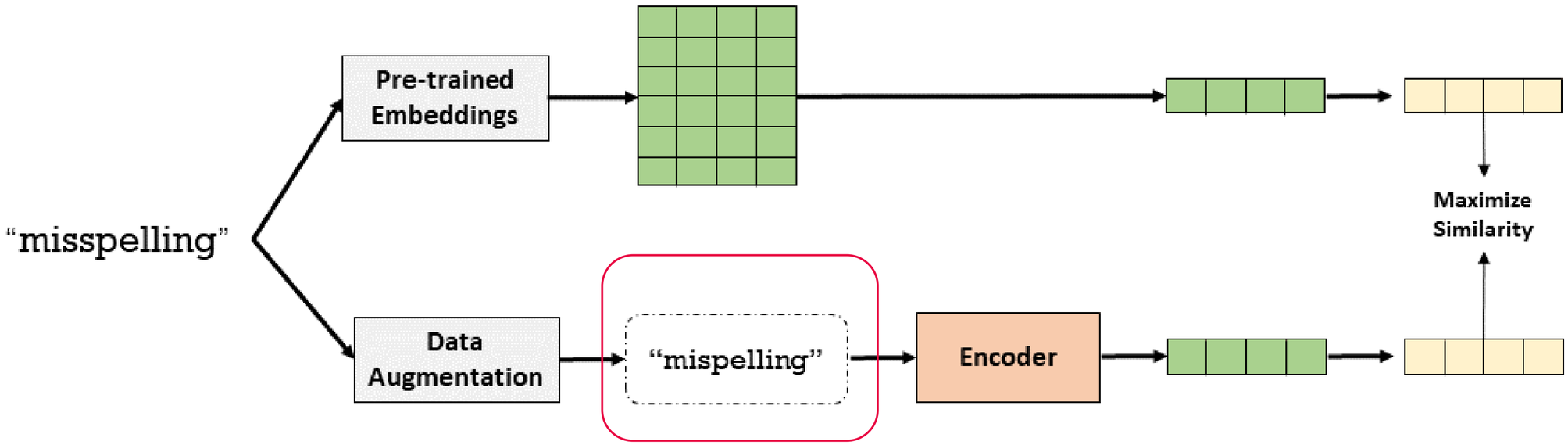


"misspelling" ⟹ *{miss, ##pel, ##ling}*

"mispselling" ⟹ {mi, ##sp, ##sell, ##ing *}*

Subwords are sensitive to typos

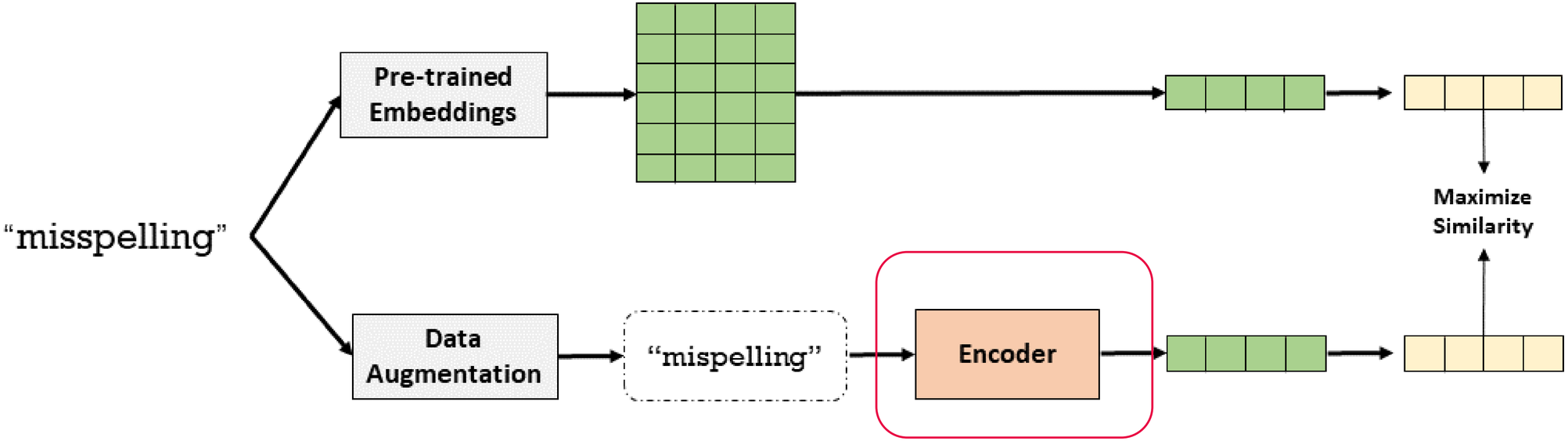
Input Method



"misspelling" ⟹ {m, i, s, s, p, e, l, l, i, n, g, miss, ##pel, ##ling}

LOVE uses both the character sequence and subwords

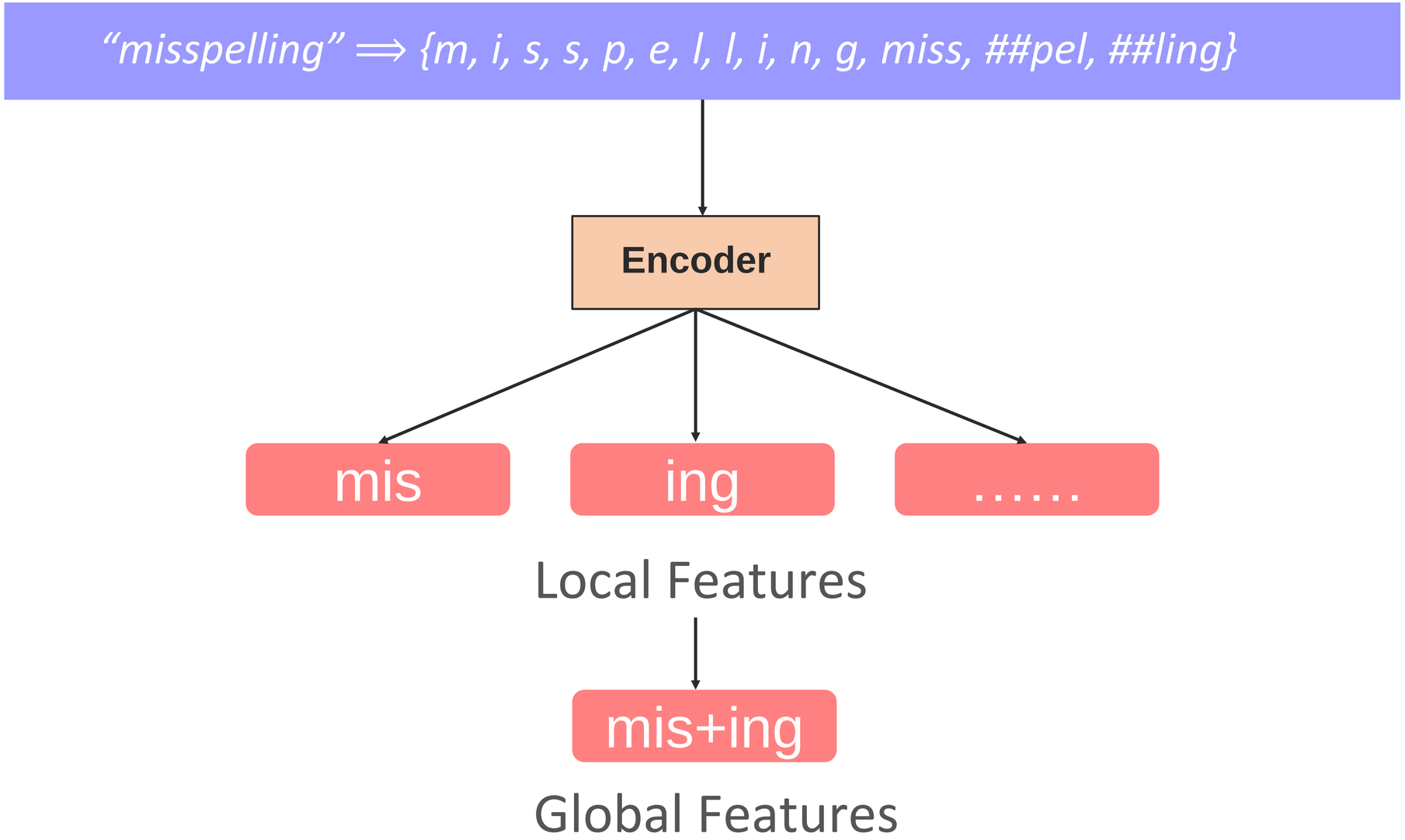
Encoder



	Input	Encoder	Loss
MIMICK (2017)	character sequence {s,p,e,l,l}	RNNs	$\mathcal{L}_{\text{dis}}$
BoS (2018)	n-gram subword {spe,pel,ell}	SUM	$\mathcal{L}_{\text{dis}}$
KVQ-FH (2019)	adapted n-gram subword {spe,pel,ell}	Attention	$\mathcal{L}_{\text{dis}}$

Table 1: Details of different mimick-like models, with the word `spell` as an example.

Encoder


“misspelling” ⟹ {m, i, s, s, p, e, l, l, i, n, g, miss, ##pel, ##ling}
Encoder
mis
ing
……
Local Features
mis+ing
Global Features

Encoder

Positional Attention Module (PAM)

"misspelling" $\Longrightarrow$ *{m, i, s, s, p, e, l, l, i, n, g, miss, ##pel, ##ling}*

1. $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \in \mathbb{R}^{n \times d}, \mathbf{x}_i \in \mathbf{V} \in \mathbb{R}^{|\mathcal{V}| \times d}$

2. $\mathrm{PA}(\mathbf{X}) = \mathrm{Softmax}\left(\frac{\mathbf{P}\mathbf{P}^\top}{\sqrt{d}}\right)(\mathbf{X}\,\mathbf{W}^V), \mathbf{P} \in \mathbb{R}^{n \times d}$

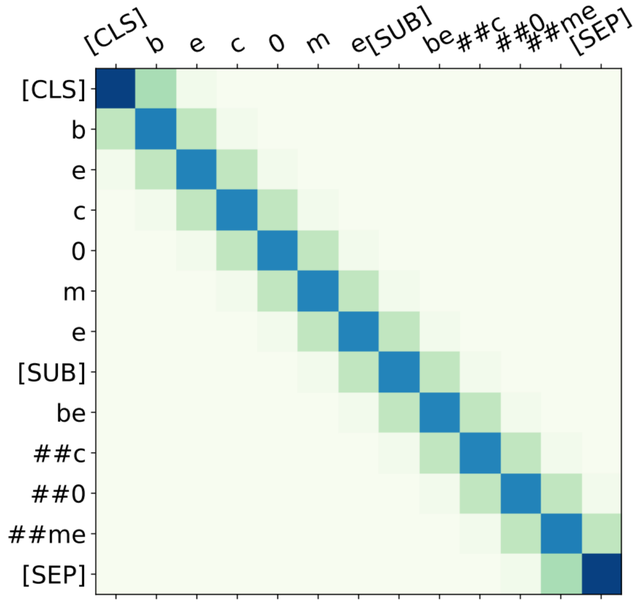
3. $\bar{\mathbf{X}} = \mathrm{SA}(\mathrm{PA}(\mathbf{X}))\,\mathbf{W}^O$

Encoder

"bec0me" ⟹ {[cls], b, e, c, 0, m, e, [sub], be, ##c, ##0, ##me, [sep]}

2 $$\mathrm{PA}(\mathbf{X}) = \mathrm{Softmax}\left(\frac{\mathbf{P}\mathbf{P}^{\mathsf{T}}}{\sqrt{d}}\right)(\mathbf{X}\,\mathbf{W}^{V}), \mathbf{P} \in \mathbb{R}^{n \times d}$$

Position Embeddings **P** are from the original Transformer [10]

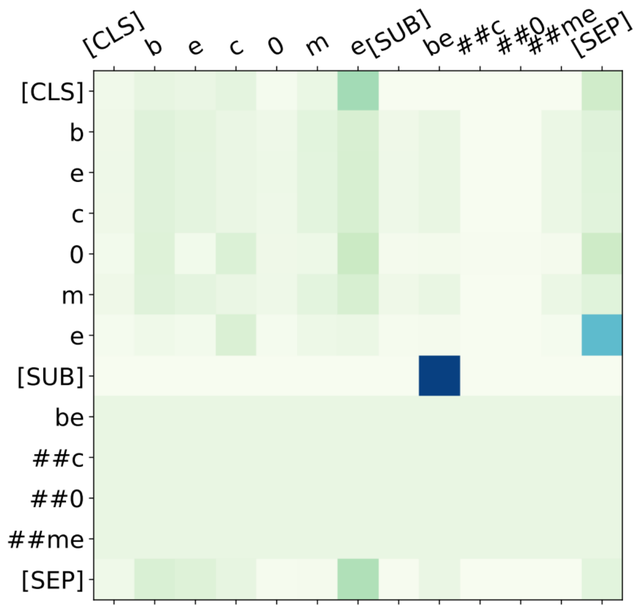


Encoder

"bec0me" ⟹ {[cls], b, e, c, 0, m, e, [sub], be, ##c, ##0, ##me, [sep]}

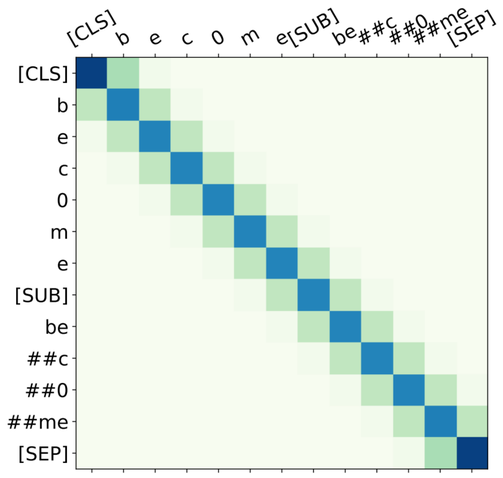
3 $$\bar{\mathbf{X}} = \mathrm{SA}(\mathrm{PA}(\mathbf{X}))\,\mathbf{W}^{O}$$

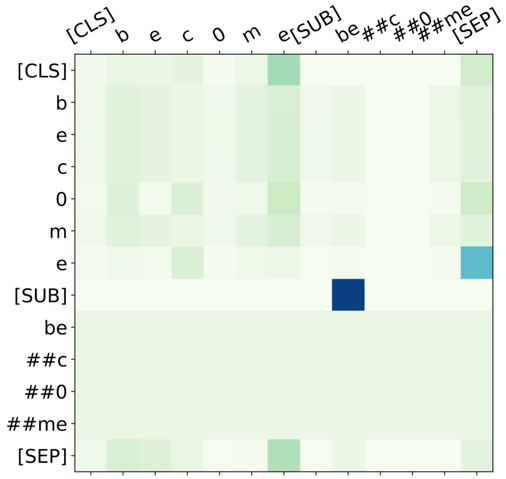
We use the self-attention mechanism in the original Transformer [10]



Encoder

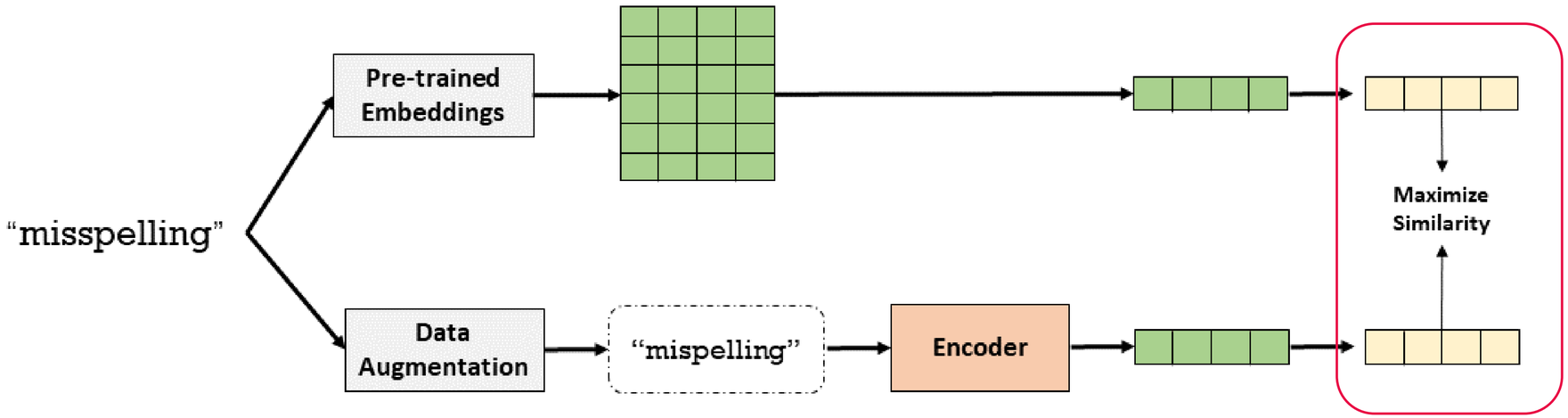
“bec0me” ⟹ {[cls], b, e, c, 0, m, e, [sub], be, ##c, ##0, ##me, [sep]}





Our Positional Attention Module (PAM) extracts both local and global information

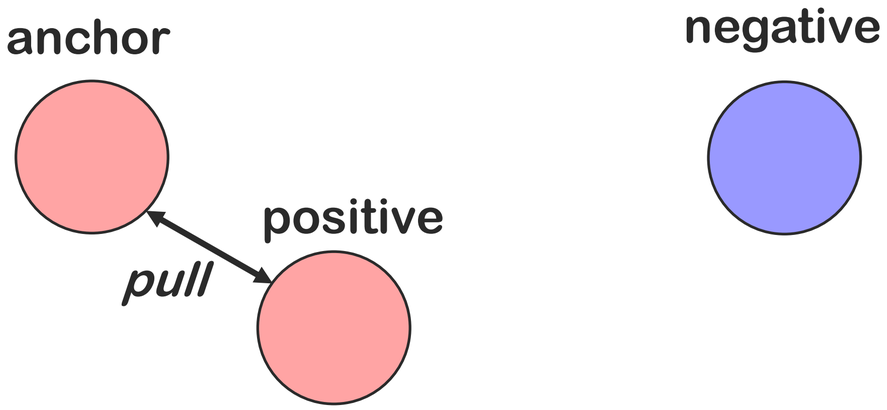
Loss Function



$$\mathcal{L}_{\text{dis}} = \frac{1}{|\mathcal{V}|} \sum_{w \in \mathcal{V}} \psi(\mathbf{u}_w, \mathbf{v}_w), \mathbf{u} \in \mathbb{R}^m, \mathbf{v} \in \mathbb{R}^m$$

MSE only pulls positive word pairs closer while ignoring negative pairs.

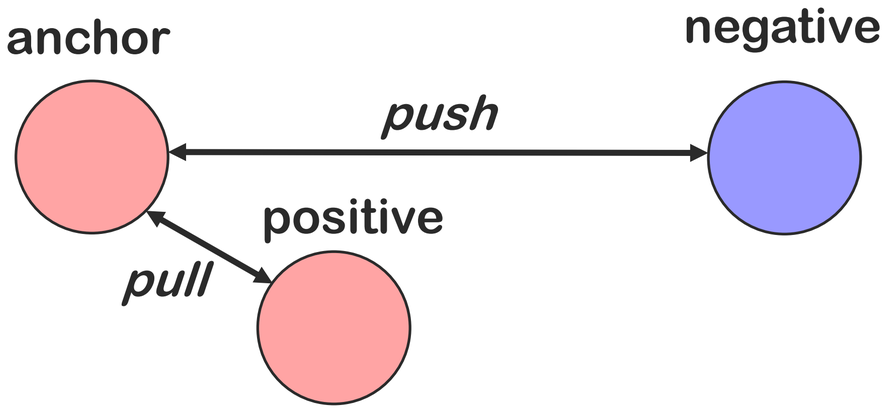
Loss Function



$$\mathcal{L}_{\text{dis}} = \frac{1}{|\mathcal{V}|} \sum_{w \in \mathcal{V}} \psi(\mathbf{u}_w, \mathbf{v}_w), \mathbf{u} \in \mathbb{R}^m, \mathbf{v} \in \mathbb{R}^m$$

MSE only pulls positive word pairs closer while ignoring negative pairs.

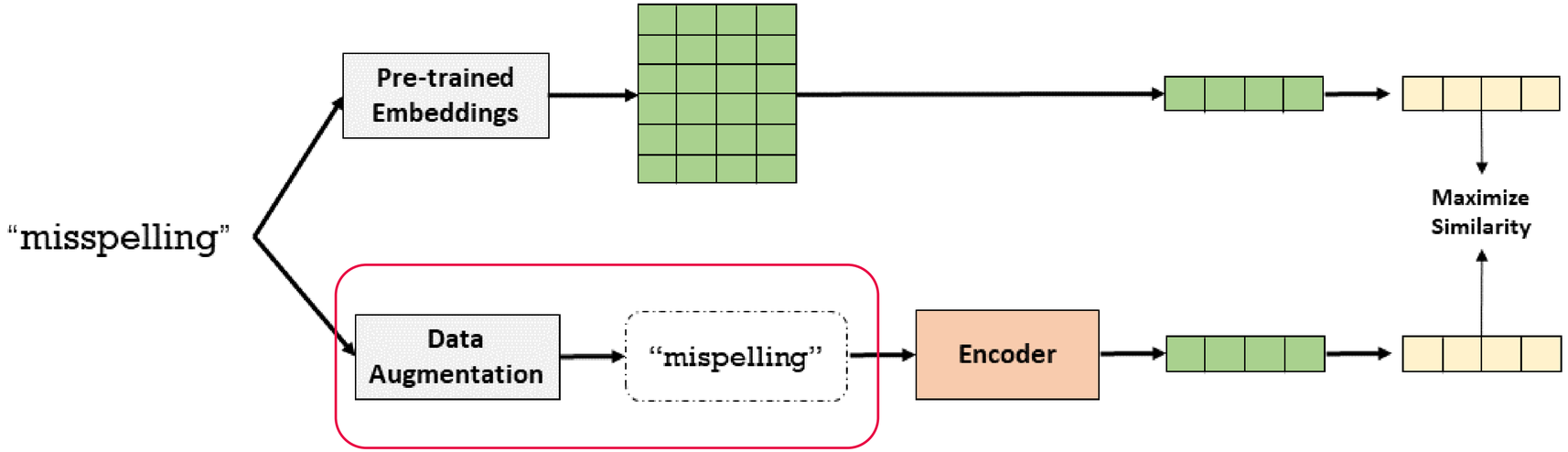
Loss Function



$$\ell_{cl} = -\log \frac{e^{\mathrm{sim}(\mathbf{u}_i^\top \mathbf{u}^+)/\tau}}{e^{\mathrm{sim}(\mathbf{u}_i^\top \mathbf{u}^+)/\tau} + \sum e^{\mathrm{sim}(\mathbf{u}_i^\top \mathbf{u}^-)/\tau}}$$

LOVE adopts the contrastive loss instead of MSE

Data Augmentation



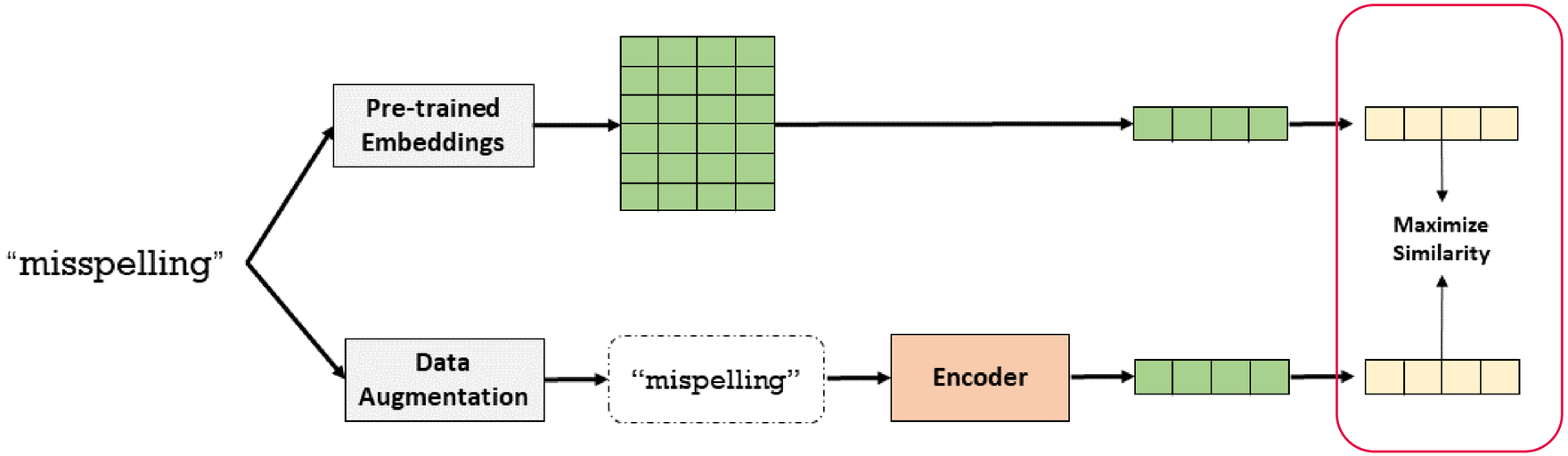
LOVE uses data augmentation to increase the diversity of training samples

Data Augmentation

Swap	misspelling -> misspleling
Drop	misspelling -> mispelling
Insert	misspelling -> misspellling
Keyboard	misspelling -> mosspelling
Synonym	misspelling -> heterography

LOVE uses data augmentation to increase the diversity of training samples

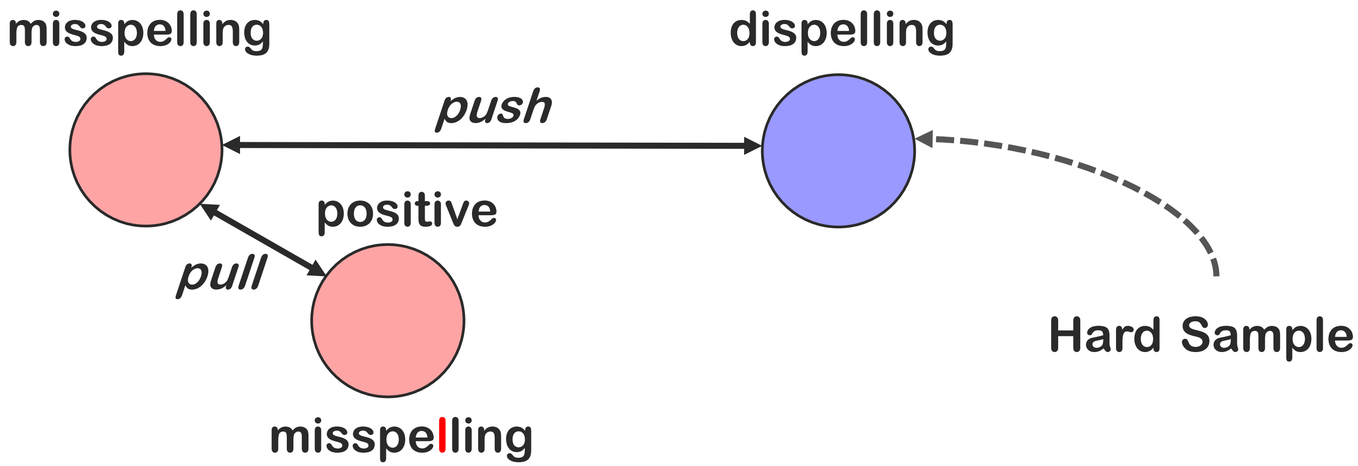
Hard Negative



$$\ell_{cl} = -\log \frac{e^{\mathrm{sim}(\mathbf{u}_i^\top \mathbf{u}^+)/\tau}}{e^{\mathrm{sim}(\mathbf{u}_i^\top \mathbf{u}^+)/\tau} + \sum e^{\mathrm{sim}(\mathbf{u}_i^\top \mathbf{u}^-)/\tau}}$$

Hard Negative

$$\ell_{cl} = -\log \frac{e^{\mathrm{sim}(\mathbf{u}_i^\mathsf{T}\mathbf{u}^+)/\tau}}{e^{\mathrm{sim}(\mathbf{u}_i^\mathsf{T}\mathbf{u}^+)/\tau} + \sum e^{\mathrm{sim}(\mathbf{u}_i^\mathsf{T}\boxed{\mathbf{u}^-})/\tau}}$$



Experimental Results

Performance on the intrinsic tasks

	parameters		Word Similarity						Word Cluster		Avg
	embedding	others	RareWord	SimLex	MTurk	MEN	WordSim	SimVerb	AP	BLESS	
FastText (2017)	969M	-	48.1	30.4	66.9	78.1	68.2	25.7	58.0	71.5	55.9
MIMICK (2017)	9M	517K	27.1	15.9	32.5	36.5	15.0	7.5	**59.3**	**72.0**	33.2
BoS (2018)	500M	-	**44.2**	27.4	55.8	65.5	53.8	22.1	41.8	39.0	43.7
KVQ-FH (2019)	12M	-	42.4	20.4	55.2	63.4	53.1	16.4	39.1	42.5	41.6
LOVE	6.3M	200K	42.2	**35.0**	**62.0**	**68.8**	**55.1**	**29.4**	53.2	51.5	**49.7**

Performance on the extrinsic tasks

	parameters		**SST2**		**MR**		**CoNLL-03**		**BC2GM**		Avg
	embedding	others	original	+typo	original	+typo	original	+typo	original	+typo	
FastText (2017)	969M	-	82.3	60.5	73.3	62.2	86.4	66.3	71.8	53.4	69.5
Edit Distance	969M	-	-	67.4	-	68.3	-	76.2	-	66.6	-
MIMICK (2018)	9M	517K	69.7	62.3	73.6	61.4	68.0	65.2	56.6	56.7	64.2
BoS (2018)	500M	-	79.7	72.6	73.6	**69.5**	**79.5**	68.6	**66.4**	61.5	71.5
KVQ-FH (2019)	12M	-	77.8	71.4	72.9	66.5	73.1	**70.4**	46.2	53.5	66.5
LOVE	6.3M	200K	**81.4**	**73.2**	**74.4**	66.7	78.6	69.7	64.7	**63.8**	**71.6**

LOVE achieves similar or even better performances than prior competitors while using fewer parameters

Experimental Results

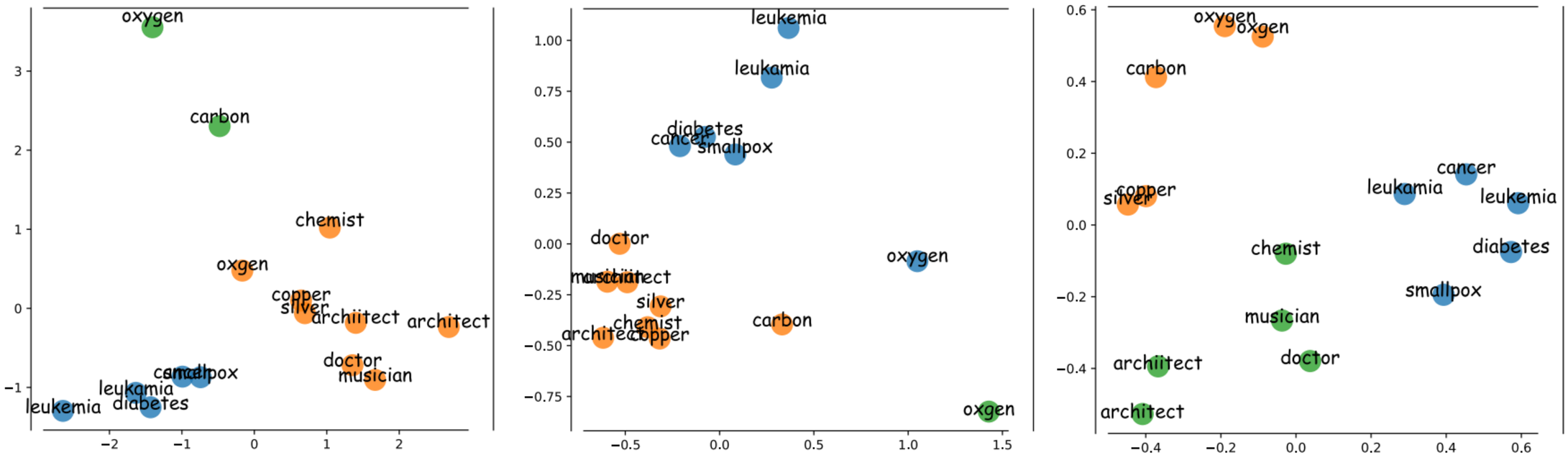
Robust evaluation

	SST2						CoNLL-03						
Typo Probability	original	10%	30%	50%	70%	90%	original	10%	30%	50%	70%	90%	Avg
Static Embeddings													
FastText	**82.3**	68.2	59.8	56.7	57.8	60.3	**86.4**	81.6	78.9	73.9	70.2	63.4	70.0
FastText + LOVE	82.1	**79.8**	**74.9**	**74.2**	**68.8**	**67.2**	86.3	**84.7**	**81.8**	**77.5**	**73.1**	**71.3**	**76.8**
Dynamical Embeddings													
BERT	**91.5**	88.2	78.9	74.7	69.0	60.1	**91.2**	**89.8**	**86.2**	83.4	79.9	76.5	80.7
BERT + LOVE	**91.5**	**88.3**	**83.7**	**77.4**	**72.7**	**63.3**	89.9	88.3	86.1	**84.3**	**80.8**	**78.3**	**82.1**

LOVE can be used in a plug-and-play fashion to robustify existing language models

Experimental Results

Visualizations of word clusters



BoS Model (500M) **KVQ-FH Model** (12M) **LOVE** (6.5M)

LOVE can produce better word vectors while consuming fewer parameters

Conclusion

We present a simple contrastive learning framework, LOVE, which can make language models robust with little cost. There are several advantages of LOVE:

- No need of pre-training
- Small model size
- Plug and Play

Lihu Chen

Gaël Varoquaux

Fabian Suchanek

paper

code

ACL 2022

22ND – 27TH MAY | 60TH MEETING | DUBLIN

Chen, Lihu, Gael Varoquaux, and Fabian Suchanek. "Imputing Out-of-Vocabulary Embeddings with LOVE Makes LanguageModels Robust with Little Cost." Proceedings of ACL 2022.

References I

[1] Schick, Timo, and Hinrich Schütze. Schick, Timo, and Hinrich Schütze. "Rare words: A major problem for contextualized embeddings and how to fix it by attentive mimicking." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 05. 2020.

[2] Sun, Lichao, et al. "Adv-BERT: BERT is not robust on misspellings! Generating nature adversarial samples on BERT." arXiv. 2020.

[3] El Boukkouri, Hicham, et al. "CharacterBERT: Reconciling ELMo and BERT for Word-Level Open-Vocabulary Representations From Characters." International Conference on Computational Linguistics. 2020.

[4] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5:135–146.

[5] Yuval Pinter, Robert Guthrie, and Jacob Eisenstein. 2017. Mimicking word embeddings using subword rnns. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 102–112.

[6] Jinman Zhao, Sidharth Mudgal, and Yingyu Liang. 2018. Generalizing word embeddings using bag of subwords. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 601–606.

References II

[7] Shota Sasaki, Jun Suzuki, and Kentaro Inui. 2019. Subword-based compact reconstruction of word embeddings. 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 3498–3508.

[8] Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, Hiroshi Noji, Pierre Zweigenbaum, and Jun'ichi Tsujii. 2020. Characterbert: Reconciling elmo and bert for word-level open-vocabulary representations from characters. In Proceedings of the 28th International Conference on Computational Linguistics, pages 6903–6915.

[9] Wentao Ma, Yiming Cui, Chenglei Si, Ting Liu, Shijin Wang, and Guoping Hu. 2020. Charbert: Characteraware pre-trained language model. In Proceedings of the 28th International Conference on Computational Linguistics, pages 39–50.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. Advances in neural information processing systems, 30.

[11] Ariel S Schwartz and Marti A Hearst. 2002. A simple algorithm for identifying abbreviation definitions in biomedical text. In Biocomputing 2003, pages 451– 462. World Scientific.