Isotropic Word Representation in BERT

by Sara Rajaee and Marzieh Sheikhi

Aguest, 2020

Isotropy

In Wrod Representations

What

It is a property to directly check if the “self-normalization" holds more strongly. i.e., to make the shape of the representation rounding.

Why

- Gradient Descent algorithm may oscillate
- Interpretation of the model is hard

How

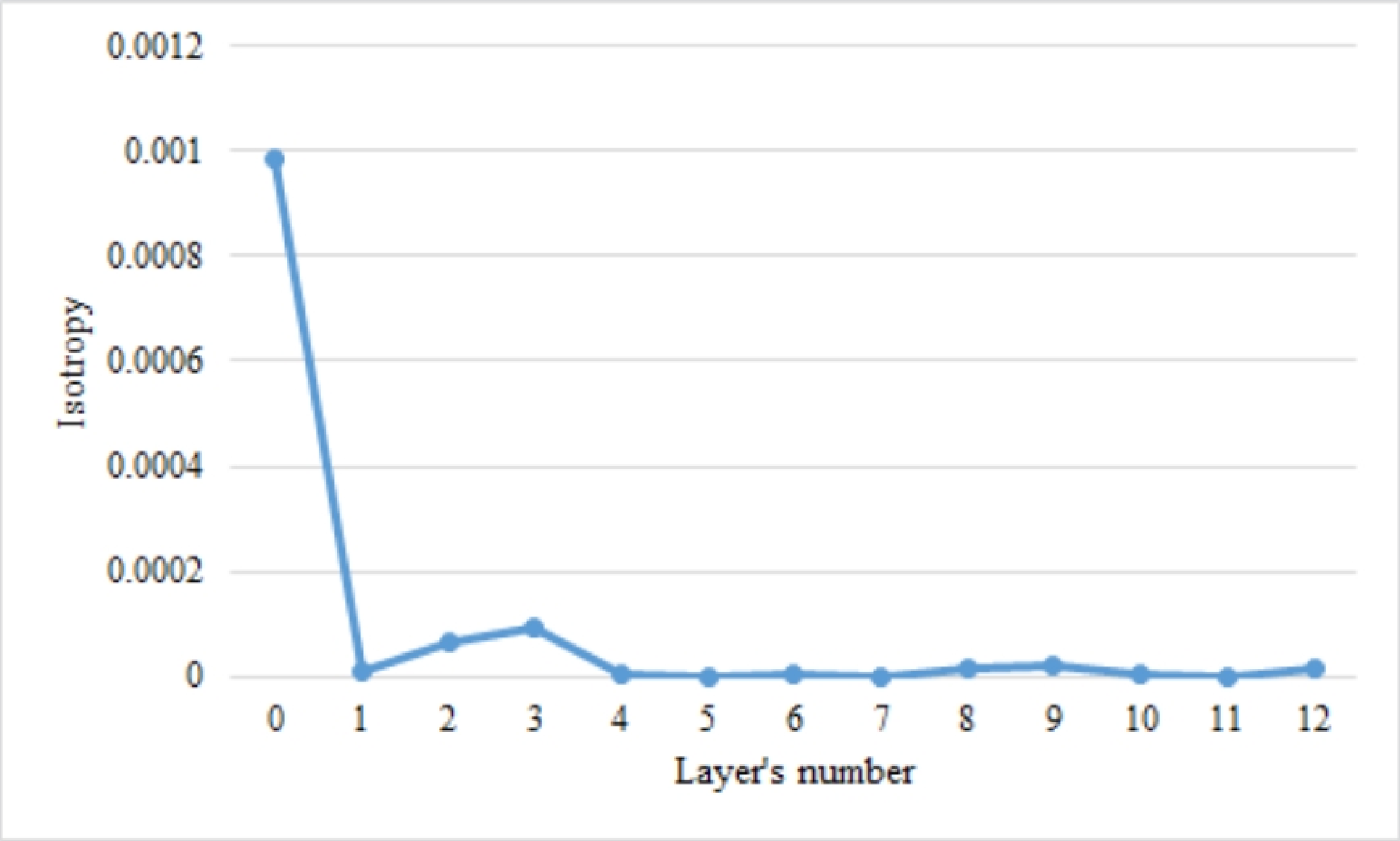
1. Make the zero-mean data
2. Subtract the effect of dominant direction

BERT

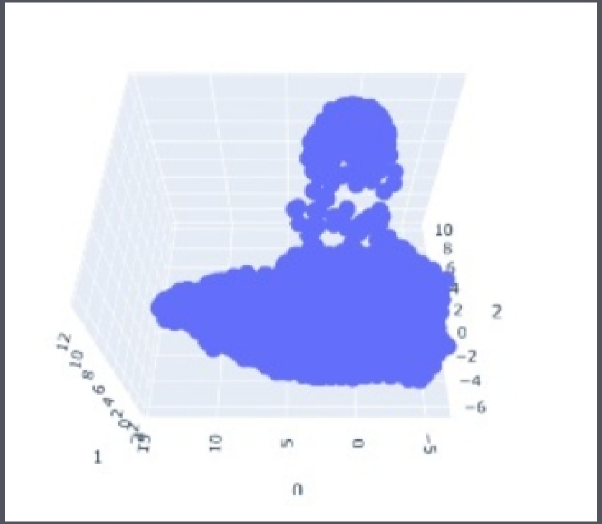
BERT contextual representations are extremely **anisotropic**.

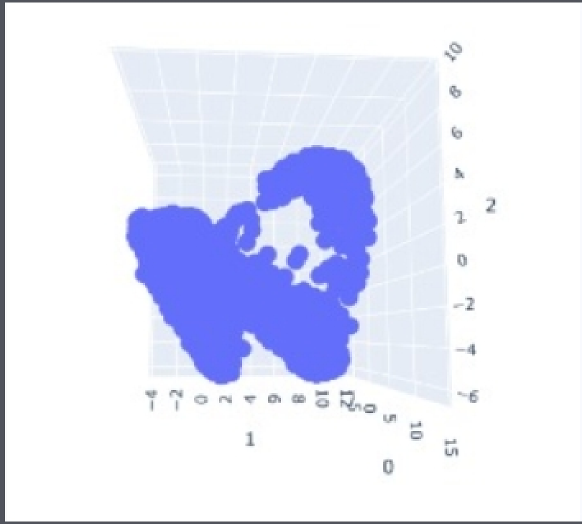
The contextualized hidden layer representations are almost all more anisotropic than the input layer representations, which do not incorporate context. This suggests that high anisotropy is inherent to, or least a by-product of, the process of contextualization

Pre-trained BERT Isotropy



Pre-trained BERT Word Representations





Recent works

1
Make the zero-mean data

2
Apply PCA on the zero-mean data

3
Eliminate the effects of dominant directions

[1] J. Mu, S. Bhat, and P. Viswanath, All-but-the-Top: Simple and Effective Postprocessing for Word Representations, preprint, https://arxiv.org/abs/1702.01417, 2017.

Proposed Method

01

02

03

Step 01

Cluster the word representations and subtract the mean of each cluster from their elements.

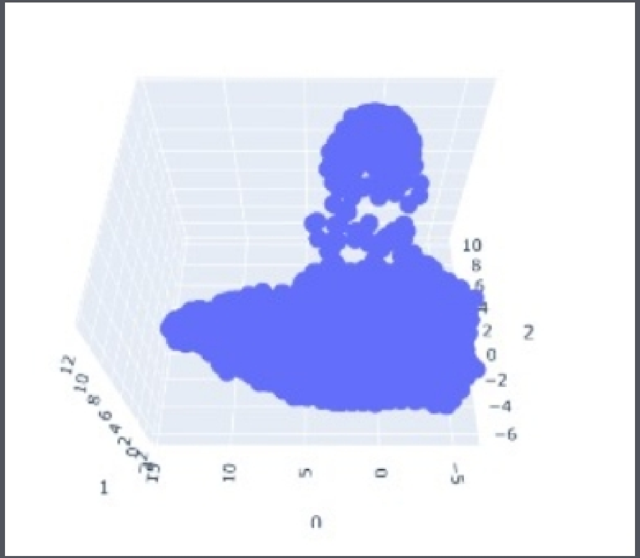
Step 02

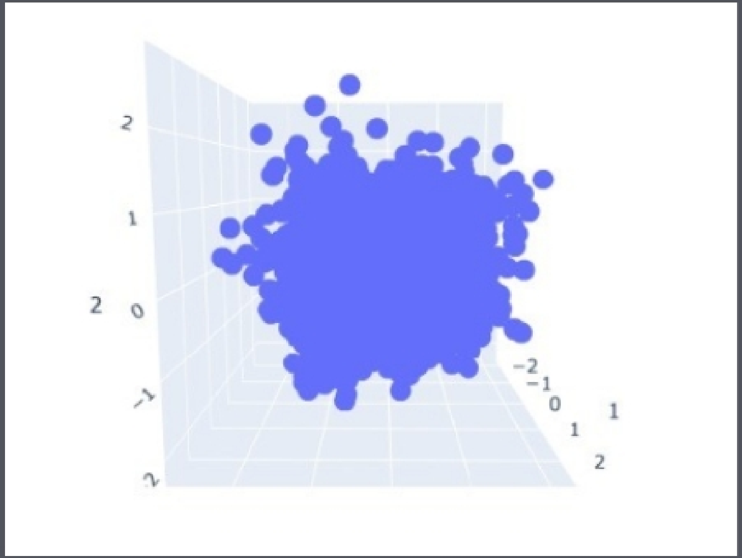
Apply PCA on each cluster

Step 03

Project word embeddings toward the weak directions rather than the dominant directions.

Pre-trained BERT results





Pre-trained Bert Results

Our proposed algorithm in comparison to other studies

	SEMEVAL 2015 –TASK 2			STS-B		
	Pearson	Spearman	Isotropy	Pearson	Spearman	Isotropy
Pre-trained BERT	56.7	53.68	1.35 e-5	56.16	54.09	3.36 e-5
Mean	60.51	56.81	4.60e-6	60.08	57.00	2.16 e-6
Mean + PCA	55.81	53.23	7.30 e-8	47.40	46.67	9.61 e-13
K-means + Mean	66.58	62.38	0.33	68.43	64.25	0.2376
GMM + Mean + PCA	69.18	65.51	0.84	70.53	67.32	**0.6380**
K-means + Mean + PCA	**69.84**	**66.33**	**0.85**	**70.75**	**67.50**	0.6353

Pre-trained BERT results

Our proposed method in comparison to pre-trained BERT

Sematic Similarity

	Pre-trained BERT Base			Proposed Algorithm		
Dataset	Pearson	Spearman	Isotropy	Pearson	Spearman	Isotropy
STS2012	45.46	43.53	2.97 e-5	**72.67**	**64.73**	**0.55**
STS2013	62.39	59.50	2.6 e -4	**69.88**	**68.89**	**0.46**
STS2014	56.71	53.36	3.84 e-6	**66.11**	**62.14**	**0.53**
STS2015	56.70	53.68	1.35 e-5	**69.84**	**66.33**	**0.85**
STS2016	61.33	61.11	1.01 e-4	**67.46**	**66.68**	**0.48**
STS-Benchmark	56.16	54.09	3.36 e-5	**70.75**	**67.50**	**0.63**
SICK	62.32	59.38	2.93 e-4	**66.47**	**63.07**	**0.492**

Test on Classification Task

	Pre-trained BERT Base		Proposed Algorithm	
	Accuracy	Isotropy	Accuracy	Isotropy
SST-2	55.67	7.97 e-6	**56.08**	**84.36**

Fine-tune the BERT

BERT Model

Proposed Algorithm

Classifier/ Regressor

Fine-Tuning BERT

CLS token

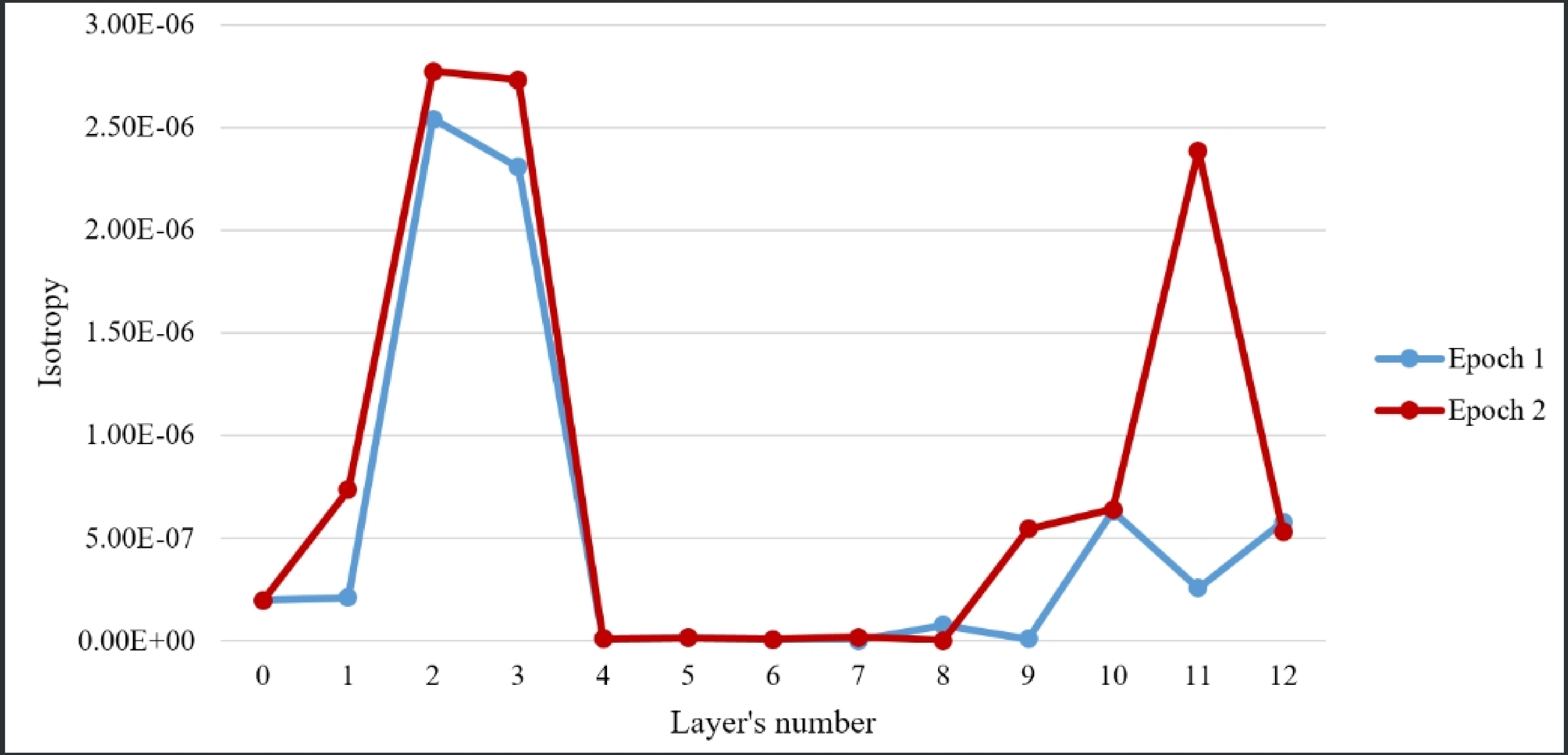
In fine-tuning we consider CLS token instead of all word representations

Batch of Data

We apply clustering algorithm on a batch

Fine-tuning analysis

Isotropy of CLS token in each BERT Layer after fine-tuning in 2 epochs as baseline



CLS Token Analysis

- Our proposed algorithm can not improve the isotropy of CLS tokens during fine-tuning. Even we apply algorithm offline.
- CLS token already are zero-mean. As a result, BERT has learned to make zero-mean CLS tokens.

Fine-tuning results

	BERT-based		Proposed algorithm	
	Accuracy	Isotropy	Accuracy	Isotropy
RTE(Re-Imp)	65.3	4.86 e-5	62.8	**0.29**
WiC	64.04	8.26 e-4	62.1	**0.13**

Results on dev set

Thank You

DO YOU HAVE ANY QUESTIONS?