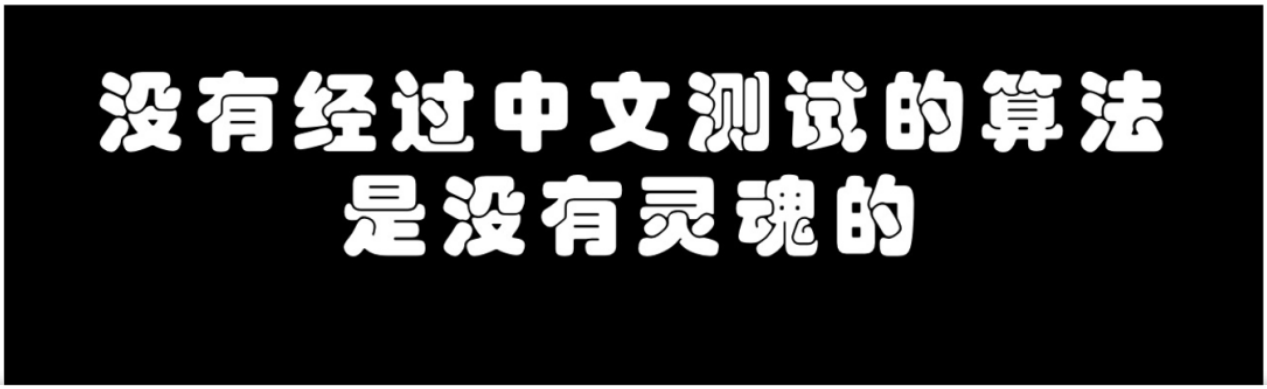
丹琦女神的对比学习新SOTA，在中文表现如何？我们补充实验后，惊了！

原创 苏剑林 夕小瑶的卖萌屋 2021-04-28 20:43



文 | 苏剑林（追一科技）
编 | 小戏

> 小编注：他来了他来了，苏神带着他的文章走来了！在小屋这篇**《丹琦女神新作：对比学习，简单到只需要Dropout两下》**推出后，苏神发来了他在 **SimCSE** 上的中文实验，实验结果表明在不少任务上 **SimCSE** 确实相当优秀，能明显优于 **BERT-whitening**。那么话不多说，让我们接着前篇的讨论，来看看苏神的文章吧~

今年年初，笔者受到 **BERT-flow** 的启发，构思了 **BERT-whitening** 方法，一度成为了语义相似度的新 **SOTA**——参考《你可能不需要 **BERT-flow** ：一个线性变换媲美 **BERT-flow** 》[1]，对应论文为《**Whitening Sentence Representations for Better Semantics and Faster Retrieval**》[2]。

。然而“好景不长”，在 **BERT-whitening** 提交到 **Arxiv** 的不久之后，笔者刷到了至少有两篇新论文里边的结果明显优于 **BERT-whitening** 了。

第一篇是《 **Generating Datasets with Pretrained Language Models** 》，这篇借助模板从 **GPT2_XL** 中无监督地构造了数据对来训练相似度模型，个人认为虽然有一定的启发而且效果还可以，但是复现的成本和变数都太大。另一篇则是本文的主角《 **SimCSE: Simple Contrastive Learning of Sentence Embeddings** 》，它提出的 **SimCSE** 在英文数据上显著超过了 **BERT-flow** 和 **BERT-whitening** ，并且方法特别简单～

那么，**SimCSE 在中文上同样有效吗？能大幅提高中文语义相似度的效果吗**？本文就来做些补充实验。

SimCSE 简介

首先，简单对 **SimCSE** 做个介绍。事实上，**SimCSE** 可以看成是 **SimBERT** 的简化版（关于 **SimBERT** 请阅读《鱼与熊掌兼得：融合检索和生成的 SimBERT 模型》[3]），它简化的部分如下：

> 1、SimCSE 去掉了 SimBERT 的生成部分，仅保留检索模型；
> 2、由于 SimCSE 没有标签数据，所以把每个句子自身视为相似句传入。

说白了，本质上来说就是(自己,自己)作为正例、(自己,别人)作为负例来训练对比学习模型。当然，事实上还没那么简单，如果仅仅是完全相同的两个样本作为正例，那么泛化能力会大打折扣。**一般来说，我们会使用一些数据扩增手段**，让正例的两个样本有所差异，但是在 NLP 中如何做数据扩增本身又是一个难搞的问题，**SimCSE** 则提出了一个极为简单的方案：**直接把 Dropout 当作数据扩增！**

具体来说，N个句子经过带 **Dropout** 的 **Encoder** 得到向量$\boldsymbol{h}_1^{(0)},\boldsymbol{h}_2^{(0)},\cdots,\boldsymbol{h}_N^{(0)}$，然后再过一遍 **Encoder**（这时候是另一个随机 **Dropout**）得到向量$\boldsymbol{h}_1^{(1)},\boldsymbol{h}_2^{(1)},\cdots,\boldsymbol{h}_N^{(1)}$，我们可以将$(\boldsymbol{h}_i^{(0)},\boldsymbol{h}_i^{(1)})$视为一对（略有不同的）正例了，那么训练目标为：

$$-\sum_{i=1}^{N}\sum_{\alpha=0,1}\log\frac{e^{\cos(\boldsymbol{h}_i^{(\alpha)},\boldsymbol{h}_i^{(1-\alpha)})/\tau}}{\sum\limits_{j=1,j\neq i}^{N}e^{\cos(\boldsymbol{h}_i^{(\alpha)},\boldsymbol{h}_j^{(\alpha)})/\tau}+\sum\limits_{j}^{N}e^{\cos(\boldsymbol{h}_i^{(\alpha)},\boldsymbol{h}_j^{(1-\alpha)})/\tau}}$$

英文效果

原论文的（英文）实验还是颇为丰富的，读者可以仔细阅读原文。但是要注意的是，原论文正文表格的评测指标跟 **BERT-flow** 、 **BERT-whitening** 的不一致，指标一致的表格在附录：

Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
$\text{BERT}_{\text{base}}$ (first-last avg.)♠	57.86	61.97	62.49	70.96	69.76	59.04	63.75	63.69
+ flow (NLI)♠	59.54	64.69	64.66	72.92	71.84	58.56	65.44	65.38
+ flow (target)♠	63.48	72.14	68.42	73.77	75.37	70.72	63.11	69.57
+ whitening (NLI)♠	61.69	65.70	66.02	75.11	73.11	68.19	63.60	67.63
+ whitening (target)♠	63.62	73.02	69.23	74.52	72.15	71.34	60.60	69.21
∗ Unsup. $\text{SimCSE-BERT}_{\text{base}}$	**68.92**	**78.70**	**73.35**	**79.72**	**79.42**	**75.49**	**69.92**	**75.07**
$\text{SBERT}_{\text{base}}$ (first-last avg.)♠	68.70	74.37	74.73	79.65	75.21	77.63	74.84	75.02
+ flow (NLI)♠	67.75	76.73	75.53	80.63	77.58	79.10	78.03	76.48
+ flow (target)♠	68.95	78.48	77.62	81.95	78.94	81.03	74.97	77.42
+ whitening (NLI)♠	69.11	75.79	75.76	82.31	79.61	78.66	76.33	76.80
+ whitening (target)♠	69.01	78.10	77.04	80.83	77.93	80.50	72.54	76.56
∗ Sup. $\text{SimCSE-BERT}_{\text{base}}$	**70.90**	**81.49**	**80.19**	**83.79**	**81.89**	**84.25**	**80.39**	**80.41**

Table B.3: STS results with "wmean" setting (Spearman). ♠: from Li et al. (2020); Su et al. (2021).

▲SimCSE与BERT-flow、BERT-whitening的效果对比

不管怎样比，**SimCSE** 还是明显优于 **BERT-flow** 和 **BERT-whitening** 的。那么 **SimCSE** 的这个优势是不是普遍的呢？在中文上有没有这个优势呢？我们马上就来做实验。

实验配置

我们的中文实验基本与《无监督语义相似度哪家强？我们做了个比较全面的评测》[4]对齐，包括之前测试的 5 个任务、4 种 **Pooling** 以及所有 **base**、**small**、**tiny** 版的模型，**large** 没有跑是因为相同配置下 **large** 模型 **OOM** 了。

经过调参，笔者发现中文任务上 **SimCSE** 的最优参数跟原论文中的不完全一致，具体区别如下：

> 1、原论文batch_size=512，这里是batch_size=64（实在跑不起这么壕的batch_size）；
> 2、原论文的学习率是5e-5，这里是1e-5；
> 3、原论文的最优dropout比例是0.1，这里是0.3；
> 4、原论文的无监督 SimCSE 是在额外数据上训练的，这里直接随机选了1万条任务数据训练。

最后一点再说明一下，原论文的无监督 **SimCSE** 是从维基百科上挑了 100 万个句子进行训练的，**至于中文实验，为了实验上的方便以及对比上的公平，直接用任务数据训练（只用了句子，没有用标签，还是无监督的）**。

不过除了 **PAWSX** 之外，其他 4 个任务都不需要全部数据都拿来训练，**经过测试，只需要随机选 1 万个训练样本训练一个 epoch 即可训练到最有效果（更多样本更少样本效果都变差）**。

> **开源地址**：
> *https://github.com/bojone/SimCSE*

中文效果

SimCSE 的所有中文实验结果如下：

	ATEC	BQ	LCQMC	PAWSX	STS-B
BERT-P1	16.59/20.61/33.14	29.35/25.76/50.67	41.71/48.92/69.99	15.15/17.03/12.95	34.65/61.19/69.
BERT-P2	9.46/22.16/25.18	16.97/18.97/41.19	28.42/49.61/56.45	13.93/16.08/12.46	21.66/60.75/57.
BERT-P3	20.79/18.27/32.89	33.08/22.58/49.58	59.22/60.12/71.83	16.68/18.37/14.47	57.48/63.97/70.
BERT-P4	24.51/27.00/31.96	38.81/32.29/48.40	64.75/64.75/71.49	15.12/17.80/16.01	61.66/69.45/70.
RoBERTa-P1	24.61/29.59/32.23	40.54/28.95/50.61	70.55/70.82/74.22	16.23/17.99/12.25	66.91/69.19/71.
RoBERTa-P2	20.61/28.91/20.07	31.14/27.48/39.92	65.43/70.62/62.65	15.71/17.30/12.00	59.50/70.77/61.
RoBERTa-P3	26.94/29.94/32.66	40.71/30.95/51.03	66.80/68.00/73.15	16.08/19.01/16.47	61.67/66.19/70.
RoBERTa-P4	27.94/28.33/32.40	43.09/33.49/49.78	68.43/67.86/72.74	15.02/17.91/16.39	64.09/69.74/70.
NEZHA-P1	17.39/18.83/32.14	29.63/21.94/46.08	40.60/50.52/60.38	14.90/18.15/16.60	35.84/60.84/68.
NEZHA-P2	10.96/23.08/15.70	17.38/28.81/32.20	22.66/49.12/21.07	13.45/18.05/12.68	21.16/60.11/43.
NEZHA-P3	23.70/21.93/31.47	35.44/22.44/46.69	60.94/62.10/69.65	18.35/21.72/18.17	60.35/68.57/70.
NEZHA-P4	27.72/25.31/30.26	44.18/31.47/46.57	65.16/66.68/67.21	13.98/16.66/14.41	61.94/69.55/68.
WoBERT-P1	23.88/22.45/32.66	43.08/32.52/49.13	68.56/67.89/72.99	18.15/19.92/12.36	64.12/66.53/70.
WoBERT-P2	-	-	-	-	-
WoBERT-P3	24.62/22.74/34.03	40.64/28.12/49.77	64.89/65.22/72.44	16.83/20.56/14.55	59.43/66.57/70.
WoBERT-P4	25.97/27.24/33.67	42.37/32.34/49.09	66.53/65.62/71.74	15.54/18.85/14.00	61.37/68.11/70.
RoFormer-P1	24.29/26.04/32.33	41.91/28.13/49.13	64.87/60.92/71.61	20.15/23.08/15.25	59.91/66.96/69.
RoFormer-P2	-	-	-	-	-
RoFormer-P3	24.09/28.51/34.23	39.09/34.92/50.01	63.55/63.85/72.01	16.53/18.43/15.25	58.98/55.30/71.
RoFormer-P4	25.92/27.38/34.10	41.75/32.36/49.58	66.18/65.45/71.84	15.30/18.36/15.17	61.40/68.02/71.
SimBERT-P1	38.50/23.64/36.98	48.54/31.78/51.47	76.23/75.05/74.87	15.10/18.49/12.66	74.14/73.37/75.
SimBERT-P2	38.93/27.06/37.00	49.93/35.38/50.33	75.56/73.45/72.61	14.52/18.51/19.72	73.18/73.43/75.
SimBERT-P3	36.50/31.32/37.81	45.78/29.17/51.24	74.42/73.79/73.85	15.33/18.39/12.48	67.31/70.70/73.
SimBERT-P4	33.53/29.04/36.93	45.28/34.70/50.09	73.20/71.22/73.42	14.16/17.32/16.59	66.98/70.55/72.

SimBERTsmall-P1	30.68/27.56/31.16	43.41/30.89/44.80	74.73/73.21/74.32	15.89/17.96/14.69	70.54/71.39/69.
SimBERTsmall-P2	31.00/29.14/30.76	43.76/36.86/45.50	74.21/73.14/74.55	16.17/18.12/15.18	70.10/71.40/69.
SimBERTsmall-P3	30.03/21.24/30.07	43.72/31.69/44.27	72.12/70.27/71.21	16.93/21.68/12.10	66.55/66.11/64.
SimBERTsmall-P4	29.52/28.41/28.56	43.52/36.56/43.38	70.33/68.75/68.35	15.39/21.57/14.47	64.73/68.12/63.
SimBERTtiny-P1	30.51/24.67/30.04	44.25/31.75/43.89	74.27/72.25/73.47	16.01/18.07/12.51	70.11/66.39/70.
SimBERTtiny-P2	30.01/27.66/29.37	44.47/37.33/44.04	73.98/72.31/72.93	16.55/18.15/13.73	70.35/70.88/69.
SimBERTtiny-P3	28.47/19.68/28.08	42.04/29.49/41.21	69.16/66.99/69.85	16.18/20.11/12.21	64.41/66.72/64.
SimBERTtiny-P4	27.77/27.67/26.25	41.76/37.02/41.62	67.55/65.66/67.34	15.06/20.49/13.87	62.92/66.77/60.

其中每个单元的数据是**“a/b/c”**的形式，a 是不加任何处理的原始结果，b 是 **BERT-whitening** 的结果（没有降维），c 则是 **SimCSE** 的结果，如果 c ＞ b，那么 c 显示为绿色，否则为红色，也就是说绿色越多，说明 **SimCSE** 比 **BERT-whitening** 好得越多。

关于其他实验细节，可以看原代码以及《无监督语义相似度哪家强？我们做了个比较全面的评测》[4]。注意由于又有 **Dropout** ，训练时又是只采样 1 万个样本，**因此结果具有随机性，重跑代码结果肯定会有波动，请读者知悉**。

一些结论

从实验结果可以看出，除了 **PAWSX** 这个“异类”外， **SimCSE** 相比 **BERT-whitening** 确实有压倒性优势，**有些任务下还能好 10 个点以上**，而且像 **SimBERT** 这种已经经过监督训练的模型还能获得进一步的提升，确实强大。（至于 **PAWSX** 为什么“异”，文章《无监督语义相似度哪家强？我们做了个比较全面的评测》[4]已经做过简单分析。）

同时，我们还可以看出在 **SimCSE** 之下，在 **BERT-flow** 和 **BERT-whitening** 中表现较好的 **first-last-avg** 这种 **Pooling** 方式已经没有任何优势了，反而较好的是直接取[**CLS**]向量，但让人意外的是，**Pooler** （取[**CLS**]的基础上再加个 **Dense** ）的表现又比较差，真让人迷惘～

由于 **BERT-whiteing** 只是一个线性变换，所以笔者还实验了 **SimCSE** 是否能浮现这个线性变换的效果。具体来说，就是固定 **Encoder** 的权重，然后接一个不加激活函数的 **Dense** 层，然后以 **SimCSE** 为目标，只训练最后接的 **Dense** 层。结果发现这种情况下的 **SimCSE** 并不如 **BERT-whitening** 。那就意味着， **SimCSE 要有效必须要把 Encoder 微调才行，同时也说明 BERT-whitening 可能包含了 SimCSE 所没有东西的**，也许两者以某种方式进行结合会取得更好的效果（构思中...）。

相关工作

简单调研了一下，发现**“自己与自己做正样本”**这个思想的工作，最近都出现好几篇论文了，除了 SimCSE 之外，同期出现的还有《**Augmented SBERT: Data Augmentation Method for Improving Bi-Encoders for Pairwise Sentence Scoring Tasks**》[5]、《**Semantic Re-tuning with Contrastive Tension**》[6]都是极度相似的。其实类似的idea笔者也想过，只不过没想到真的能 work（就没去做实验了），也没想到关键点是 **Dropout** ，看来还是得多多实验啊～

本文小结

本文分享了笔者在 **SimCSE** 上的中文实验，结果表明不少任务上 **SimCSE** 确实相当优秀，能明显优于 **BERT-whiteining** 。

参考文献

[1] https://kexue.fm/archives/8069.

[2] Jianlin Su et al. Whitening Sentence Representations for Better Semantics and Faster Retrieval. https://arxiv.org/abs/2103.15316.

[3] https://kexue.fm/archives/7427.

[4] https://kexue.fm/archives/8321.

[5] https://arxiv.org/abs/2010.08240.

[6] https://openreview.net/forum?id=Ov_sMNau-PF.

阅读原文

喜欢此内容的人还喜欢

若被制裁，中国AI会雪崩吗?

夕小瑶的卖萌屋