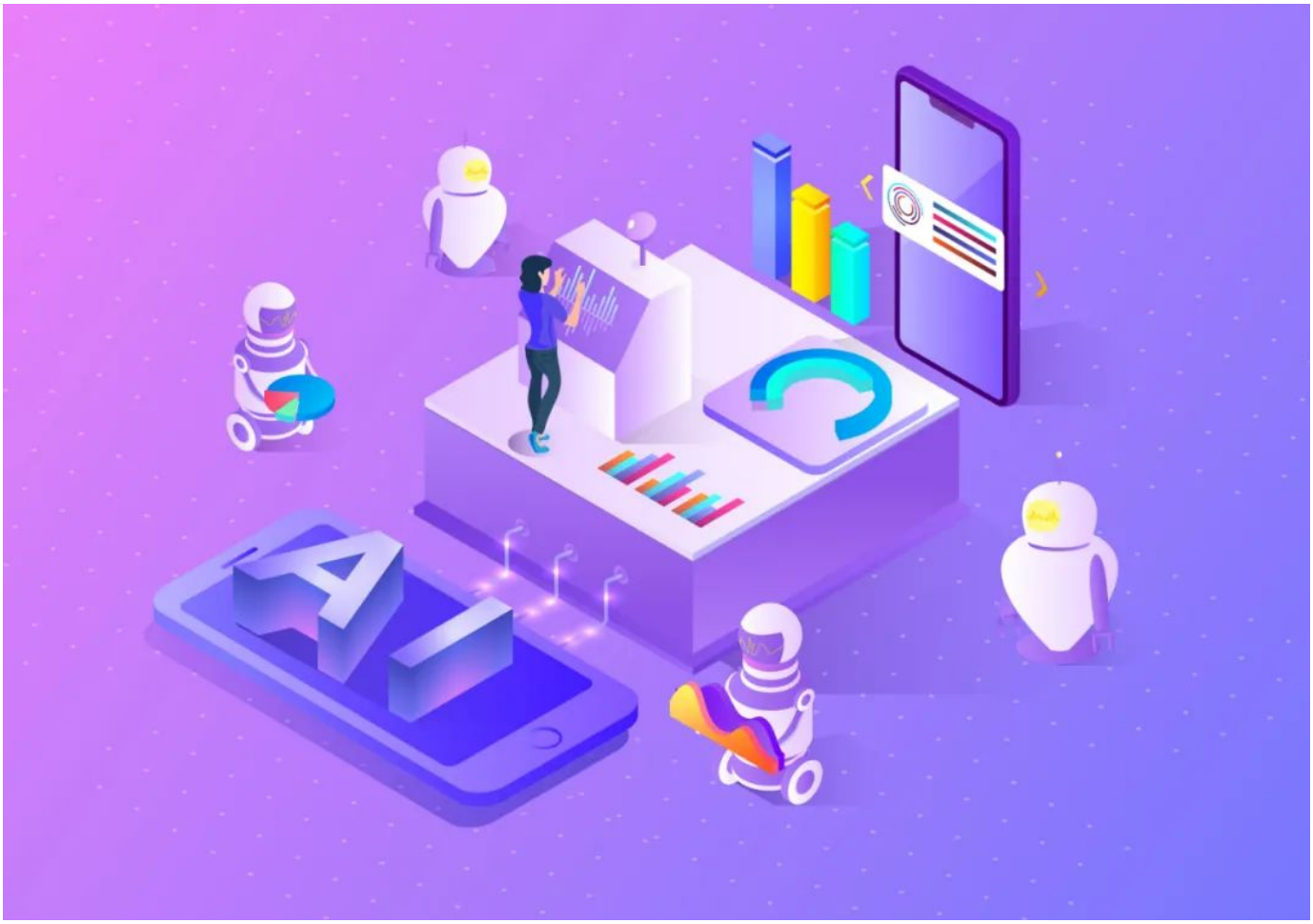
文本增强技术的研究进展及应用实践

李渔 博士 DataFunTalk



分享嘉宾：李渔 博士 熵简科技

文章整理：Hoh

内容来源：作者授权

出品平台：DataFun

导读：本文摘自熵简科技NLP团队的内部技术沙龙，文章系统性地回顾了自然语言处理领域中的文本增强技术在近几年的发展情况，重点列举和讨论了18年、19年中人们常用的五类文本增强技术路径以及对应的代表性技术，希望对于大家的日常研究工作有所启发。

01. 为什么要了解文本增强技术

在开始介绍具体方法之前，先简单列举一下目前自然语言处理任务中运用文本增强技术的常见场景：

① 少样本场景

在少样本场景下，能够搜集到的样本数目不满足模型训练的需求，导致模型处于欠拟合的状态。自然而然，在现有数据基础上，运用文本增强技术来扩充样本集，是一件又快又省，性价比很高的事。很多研究也已经表明，这种方法可以明显提高模型的性能[1-3]；

② 分类任务中样本分布不均衡的场景

除了一些基准 benchmark，真实场景中大部分文本分类任务中的各类别样本数目都是不均衡的，很多时候样本数最多类别的数目可能比最少的类别高两个数量级。这会带来很多问题，比如模型对于小样本类别往往处于欠拟合状态，在实际预测时，几乎不会对这一类别给予太高的概率。

自然，面对这样的问题，一种常见的处理方式是针对小样本类别，运用数据增强技术进行样本扩充，从而降低样本间的不均衡性，提高模型的泛化能力。这种方法也在实际中被多次证明了其有效性[2,4]。

当然，对于样本不均衡问题，已经有很多解决方法，大家可以参考这篇 google 引用数快1万的论文[5]及其引文，以及知乎问题下的高赞回答：

欠采样（undersampling）和过采样（oversampling）会对模型带来怎样的影响?

https://www.zhihu.com/question/269698662/answer/352279936

③ 半监督训练场景

至少从19年 NLP方向 google出品的半监督学习算法 UDA 可以看出[6]，文本数据增强技术可以用在无标签样本上，以构造出半监督训练所需的样本对，以此让模型从无标签的数据中获取到优化所需的梯度。关于半监督学习的具体进展，后面如果有时间，可以单开一篇文章介绍。

④ 提高模型的鲁棒性

数据增强技术在不严谨的情况下可以分为两类，一类是在保持语义不变的情况下，变换文本的表达形式，例如接下来提到的回译、文本复述等；另一类是，按照某种策略对原文进行局部调整，例如后面提到同义词替换，随机删除等等。不论是哪种方法，都可以认为是提高了模型的鲁棒性，使得模型更关注文本的语义信息，并对文本的局部噪声不再敏感。举个例子，"文本数据强增技术帮助可以模型对于噪声局部不再感敏"，如果你依然能够看明白这句话的意思，说明你对于文本局部噪声也是不敏感的。

基于这种考虑，无论是少样本场景还是大语料场景，文本增强技术都有助于提高模型的鲁棒性，提高其泛化能力。关于这一点，深度学习领域著名的花书[7] 的 7.4 节表达了类似的观点。

从上面的介绍可以看出来，文本数据增强技术在自然语言处理中属于基础性技术，具有广阔的应用场景，因此有必要对其进行系统性的讨论。

02. 典型技术方案

① 回译（Back translation）

得益于近几年文本翻译领域的显著进展、各种先进翻译模型的开源（包括百度、google 等翻译工具的接口开放），基于回译（back translation）方法的文本数据增强成为了质量高又几乎无技术门槛的通用文本增强技术。回译方法的基本流程很简单，利用翻译模型将语种1的原始文本翻译为语种2的文本表达，基于语种2的表达再翻译为语种3的文本表达，最后再直接从语种3的形式翻译回语种1的文本表达，此文本即是原始文本增强后的文本。当然，很多时候只采用一种中间语种也可以实现很好的增强效果。

我们利用 google 翻译举个例子：

原始文本为：文本数据增强技术在自然语言处理中属于基础性技术；

翻译为日语：テキストデータ拡張技術は、自然言語処理の基本的な技術です；

日语再翻译为英语：Text data extension technology is a basic technology of natural language processing；

英语再翻译回中文：文本数据扩展技术是自然语言处理的基本技术。

可以看出来，由于 google 翻译足够优秀，增强前后的文本在语义上基本保持一致。因此，对于回译这一增强技术，翻译模型的好坏决定了数据增强的最终效果。

其中还有一些细节值得说一下：

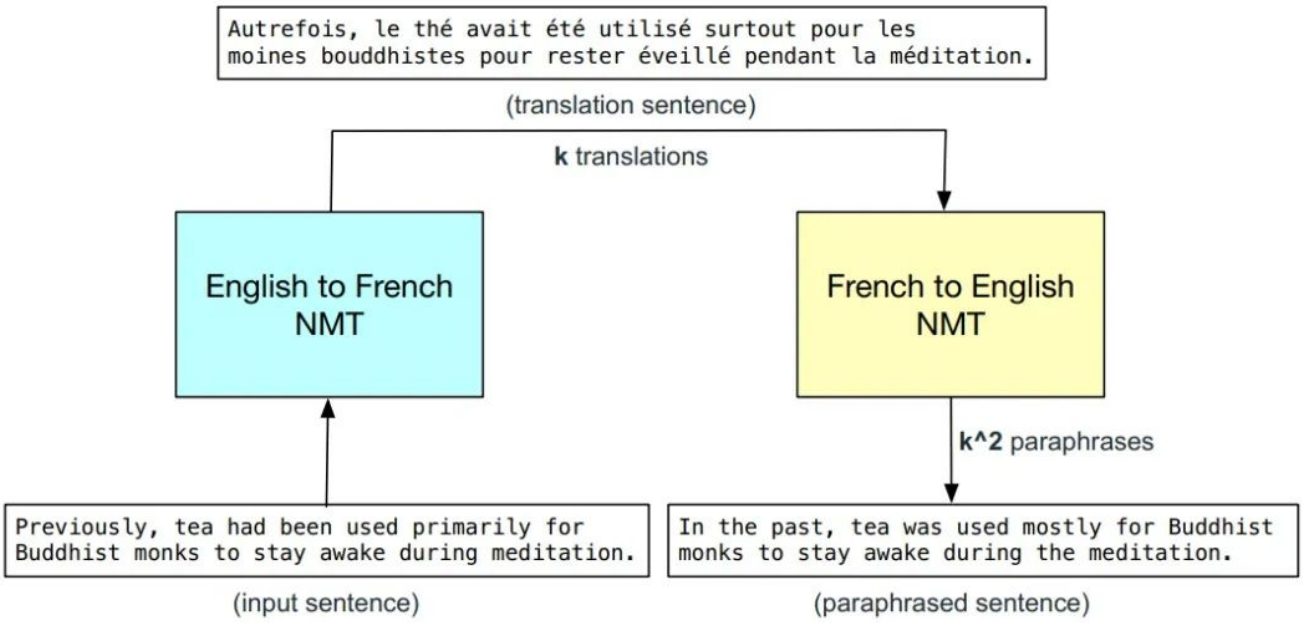
- 第一，如果采用翻译模型，可以采用 random sample 或 beam search 等策略实现成倍数的数据扩充。如果采用google等翻译工具，通过更换中间语种，也可以实现N倍的数据扩充。

- 第二，目前翻译模型对长文本输入的支持较弱，因此在实际中，一般会将文本按照"。"等标点符号拆分为一条条句子，然后分别进行回译操作，最后再组装为新的文本。

说了这么多，我们看一下 回译 技术在近几年研究报道中的实际应用情况。

据我们所知，早期人们主要将回译技术用于神经网络翻译模型（NMT）的性能提升[8,9]，通过回译可以将单语语料（monolingual data）构造成双语语料，从而帮助模型提升性能。实验证明，回译可以帮助 NMT 模型带来平均 1.7 BLEU 的性能提升，帮助 facebook 的团队在WMT'14 English-German 测试集上实现了当时的 SOTA 性能，更多的细节大家可以移步文献[9]，里面有详细讨论。

到了 2018年，CMU 和 google brain 的团队将回译技术独立出来作为一个专门的数据增强技术用来优化问答模型的性能[10]。他们同时训练了两个NMT模型，分别是 English to French 和 French to English，用来实现回译，如下图所示：



最终的实验证明，回译技术帮助他们的模型带了至少一个百分点的性能提升，如下图红框所示。我们知道，对于问答系统而言，能够有一个百分点的提升，也是很不错的工作。

Single Model	Published[12] EM / F1	LeaderBoard[13] EM / F1
LR Baseline (Rajpurkar et al., 2016)	40.4 / 51.0	40.4 / 51.0
Dynamic Chunk Reader (Yu et al., 2016)	62.5 / 71.0	62.5 / 71.0
Match-LSTM with Ans-Ptr (Wang & Jiang, 2016)	64.7 / 73.7	64.7 / 73.7
Multi-Perspective Matching (Wang et al., 2016)	65.5 / 75.1	70.4 / 78.8
Dynamic Coattention Networks (Xiong et al., 2016)	66.2 / 75.9	66.2 / 75.9
FastQA (Weissenborn et al., 2017)	68.4 / 77.1	68.4 / 77.1
BiDAF (Seo et al., 2016)	68.0 / 77.3	68.0 / 77.3
SEDT (Liu et al., 2017a)	68.1 / 77.5	68.5 / 78.0
RaSoR (Lee et al., 2016)	70.8 / 78.7	69.6 / 77.7
FastQAExt (Weissenborn et al., 2017)	70.8 / 78.9	70.8 / 78.9
ReasoNet (Shen et al., 2017b)	69.1 / 78.9	70.6 / 79.4
Document Reader (Chen et al., 2017)	70.0 / 79.0	70.7 / 79.4
Ruminating Reader (Gong & Bowman, 2017)	70.6 / 79.5	70.6 / 79.5
jNet (Zhang et al., 2017)	70.6 / 79.8	70.6 / 79.8
Conductor-net	N/A	72.6 / 81.4
Interactive AoA Reader (Cui et al., 2017)	N/A	73.6 / 81.9
Reg-RaSoR	N/A	75.8 / 83.3
DCN+	N/A	74.9 / 82.8
AIR-FusionNet	N/A	76.0 / 83.9
R-Net (Wang et al., 2017)	72.3 / 80.7	76.5 /84.3
BiDAF + Self Attention + ELMo	N/A	**77.9/ 85.3**
Reinforced Mnemonic Reader (Hu et al., 2017)	73.2 / 81.8	73.2 / 81.8
Dev set: QANet	**73.6 / 82.7**	N/A
Dev set: QANet + data augmentation ×2	**74.5 / 83.2**	N/A
Dev set: QANet + data augmentation ×3	**75.1 / 83.8**	N/A
Test set: QANet + data augmentation ×3	**76.2 / 84.6**	76.2 / 84.6

Table 2: The performances of different models on SQuAD dataset.

同时，他们详细研究了不同增强倍数以及不同采样比例下，回译对于模型提升的大小，如下图所示：

	EM / F1	Difference to Base Model EM / F1
Base QANet	73.6 / 82.7	
- convolution in encoders	70.8 / 80.0	-2.8 / -2.7
- self-attention in encoders	72.2 / 81.4	-1.4 / -1.3
replace sep convolution with normal convolution	72.9 / 82.0	-0.7 / -0.7
+ data augmentation ×2 (1:1:0)	74.5 / 83.2	+0.9 / +0.5
+ data augmentation ×3 (1:1:1)	74.8 / 83.4	+1.2 / +0.7
+ data augmentation ×3 (1:2:1)	74.3 / 83.1	+0.7 / +0.4
+ data augmentation ×3 (2:2:1)	74.9 / 83.6	+1.3 / +0.9
+ data augmentation ×3 (2:1:1)	75.0 / 83.6	+1.4 / +0.9
+ data augmentation ×3 (3:1:1)	**75.1 / 83.8**	**+1.5 / +1.1**
+ data augmentation ×3 (4:1:1)	75.0 / 83.6	+1.4 / +0.9
+ data augmentation ×3 (5:1:1)	74.9 / 83.5	+1.3 / +0.8

对比图中的两个红框部分，研究人员发现，在最佳情况下，回译带来的性能提升与在模型中加入 self-attention 组件带来的提升几乎相当。这似乎表明，挖掘数据多维度的信息和优化模型架构具有同等的重要性。

时间到了2019年下半年，google 团队提出了一种可用于 NLP 任务的半监督学习算法（UDA）[6]，前面已经提到过了。这篇文章本身并不复杂，主要是实验证明了 回译 等文本增强技术可以用于半监督学习，而且结果看起来很惊人，他们仅用了 20 条样本作为标签数据，就在 IMDb 数据集上实现了接近 SOTA 的性能。当然，我们觉得这里面至少有一半的原因是算法采用的 BERT 模型原本就已经在大规模预料上学习过。关于 UDA 更具体的分析，感兴趣的同学可以移步文献[6]，这里就不再详细展开。

最后，再从机器学习本身来讨论一下回译技术：

- 第一，回译技术的有效性本质上来源于迁移学习。通过文本增强的过程，回译技术将翻译模型学到的关于词义、语法、句法等知识转移到了新生成的样本上，从而为当前的自然语言处理任务引入了新的信息和知识来源；

- 第二，回译技术产生的新样本如果有益，隐含着这样一个先验，即模型对于具有不同语言表达形式但同样语义的输入文本，应该具有不变性，或者应该具有相近的输出。那么，是否所有的NLP任务都具备这样的先验假设呢?

② 随机词替换

此处所谓的基于随机词替换的数据增强方法是对一类文本数据增强方法的统称，其基本方法类似于图像增强技术中的随机裁剪、图像缩放，通常是随机地选择文本中一定比例的词，并对这些词进行同义词替换、删除等简单操作，不像回译等模型，需要外部预训练好的模型的辅助。

19年有研究团队提出了一种称为 EDA（Easy data augmentation）的文本增强方法[11]，该方法可以认为是这一类方法的集大成者。EDA 主要包含四种操作：同义词替换、随机插入、随机交换和随机删除。详细说明如下：

- 同义词替换(SR)：从句子中随机选择非停止词。用随机选择的同义词替换这些单词；

- 随机插入(RI)：随机的找出句中某个不属于停用词集的词，并求出其随机的同义词，将该同义词插入句子的一个随机位置。重复n次；
- 随机交换(Random Swap, RS)：随机的选择句中两个单词并交换它们的位置。重复n次；
- 随机删除(RD)：以概率p随机删除句子中每个单词。

下面列举了以上四类操作的例子：

原始文本：今天天气很好。

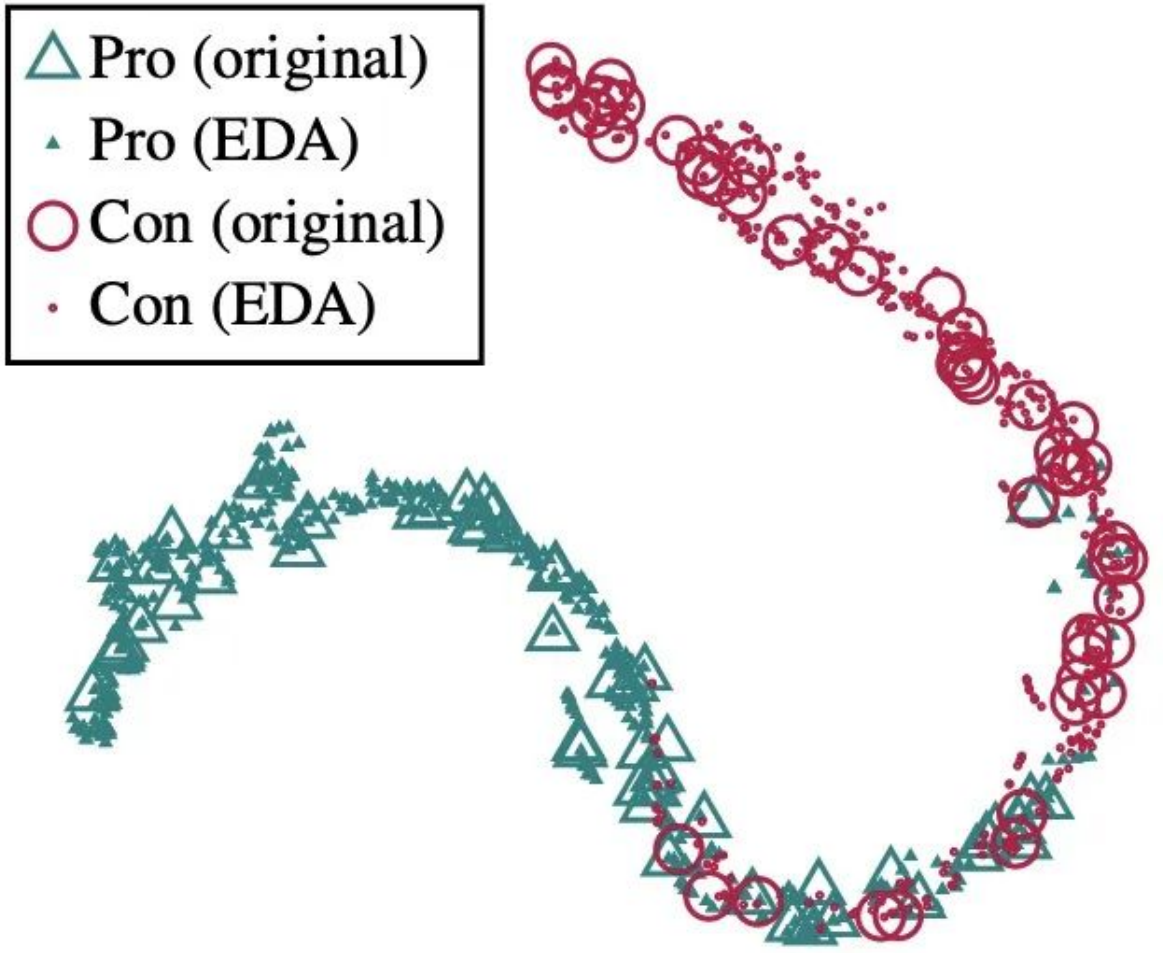
同义词替换(SR)：今天天气不错。（好 替换为 不错）

随机插入(RI)：今天不错天气很好。（插入 不错）

随机交换(RS)：今天很好天气。（很好 和 天气交换位置）

随机删除(RD)：今天天气好。（删除 很）

对于这种方法，最大的一个疑问是，经过EDA操作之后，文本的类别标签（label）是否还能保持不变，毕竟这是对文本进行随机操作。研究人员对于这个问题专门进行了实验分析。首先，他们仅用原始训练集（未经过数据增强）训练了一个分类模型，这里姑且称之为"模型A"。接下来，利用EDA 方法对测试集进行了数据拓展。最后，将原有的测试集和拓展出的语料输入到模型A中，并对模型在最后线性层的输出进行了比较。他们发现原有测试集和拓展出的语料，在高维空间中，距离很小。二者经过 t-SNE 算法降维之后的结果对比，如下图所示：



从上面的分析可以看出来，经过 EDA 变换之后，原始数据集一方面在原有基础上扩展吸收了很多噪声，扩大了数据量，同时还保持了原有的标签，因而有效的扩大了原始样本集的信息容量。

接下来，我们看一下 EDA 技术应用到实际问题中的效果怎么样。

研究人员在五项公开的文本分类数据集中进行实验测试，为了更充分地对比，实验中分别采用了卷积神经网（CNN）和循环神经网（RNN）作为分类模型，最终在五项任务中的平均表现如下表所示[2.2.1]：

Model	Training Set Size			
	500	2,000	5,000	full set
RNN	75.3	83.7	86.1	87.4
+EDA	79.1	84.4	87.3	88.3
CNN	78.6	85.6	87.7	88.3
+EDA	80.7	86.4	88.3	88.8
Average	76.9	84.6	86.9	87.8
+EDA	79.9	85.4	87.8	**88.6**

从上表的结果中，我们至少可以得出两个结论：

- 第一：EDA 技术可以有效提到模型的泛化能力，降低泛化误差，即使在完整数据集下，EDA 技术也可以带来平均 0.8 个百分点的提升；

- 第二：数据集越小，EDA 技术对模型带来的提升越明显。当样本数量只有 500 时，EDA 技术可以带来平均 三个 百分点的提升。因此，很适合用在少样本的场景。值得注意的是，在 EDA 技术的帮助，数据量规模仅有原数据集的 50% 时，模型性能已经超过了不使用 EDA 时在 100% 数据上的表现。

此外，在19年11月由 IBM 研究团队发表的一项新的文本增强技术的研究中[2]，也对 EDA 技术进行了对照实验：

Dataset		BERT	SVM	LSTM
ATIS	Baseline	53.3	35.6	29.0
	EDA	62.8	35.7	27.3
	CVAE	60.6	27.6	14.9
	CBERT	51.4	34.8	23.2
	LAMBADA	**75.7***	**56.5***	**33.7***
TREC	Baseline	60.3	42.7	17.7
	EDA	62.6	**44.8***	23.1
	CVAE	61.1	40.9	25.4*
	CBERT	61.4	43.8	24.2
	LAMBADA	**64.3***	43.9*	**25.8 ***
WVA	Baseline	67.2	60.2	26.0
	EDA	67.0	60.7	28.2
	CVAE	65.4	54.8	22.9
	CBERT	67.4	60.7	28.4
	LAMBADA	**68.6***	**62.9***	**32.0***

其中，LAMBADA 技术为IBM研究团队所提出的文本增强方法，将本文的后面部分做详细介绍。从表中可以看出，EDA方法在多数训练集中的表现仅次于IBM最新研究成果LAMBADA，这再次验证了EDA方法的简单有效。

关于 EDA 技术，另一个需要重点关注的问题是，在运用 EDA 技术，如何设置替换比例 和 增强的文本倍数，比如2000条语句应对多少数据进行随机删除，增加等。原文给出的建议比例如下[11]:

N_{train}	α	n_{aug}
500	0.05	16
2,000	0.05	8
5,000	0.1	4
More	0.1	4

其中，α是替换删除等的比例，比如同义词替换中，替换的单词数n=α*L，L是句子长度，随机插入、随机替换类似；Naug 是使用EDA方法从每一个句子拓展出的句子数量。

综上，我们可以知道，采用EDA文本增强技术进行模型性能的提升，简单而有效，尤其是在小样本场景下。

③ 非核心词替换

在上文的 EDA 技术中，对于要替换的词是随机选择的，因此一种直观感受是，如果一些重要词被替换了，那么增强后文本的质量会大打折扣。这一部分介绍的方法，则是为了尽量避免这一问题，所实现的词替换技术，姑且称之为"基于非核心词替换的数据增强技术"。

我们最早是在 Google 提出 UDA 算法的那篇论文中发现的这一技术[6]，是否在更早的文献中出现过，我们没有再深究了，有了解的同学请留言告知。

整个技术的核心点也比较简单，用词典中不重要的词去替换文本中一定比例的不重要词，从而产生新的文本。

我们知道在信息检索中，一般会用 TF-IDF值来衡量一个词对于一段文本的重要性，下面简单介绍一下 TF-IDF 的定义：

TF（词频）即一个词在文中出现的次数，统计出来就是词频TF，显而易见，一个词在文章中出现很多次，那么这个词可能有着很大的作用，但如果这个词又经常出现在其他文档中，如"的"、"我"，那么其重要性就要大打折扣，后者就是用 IDF 来表征。

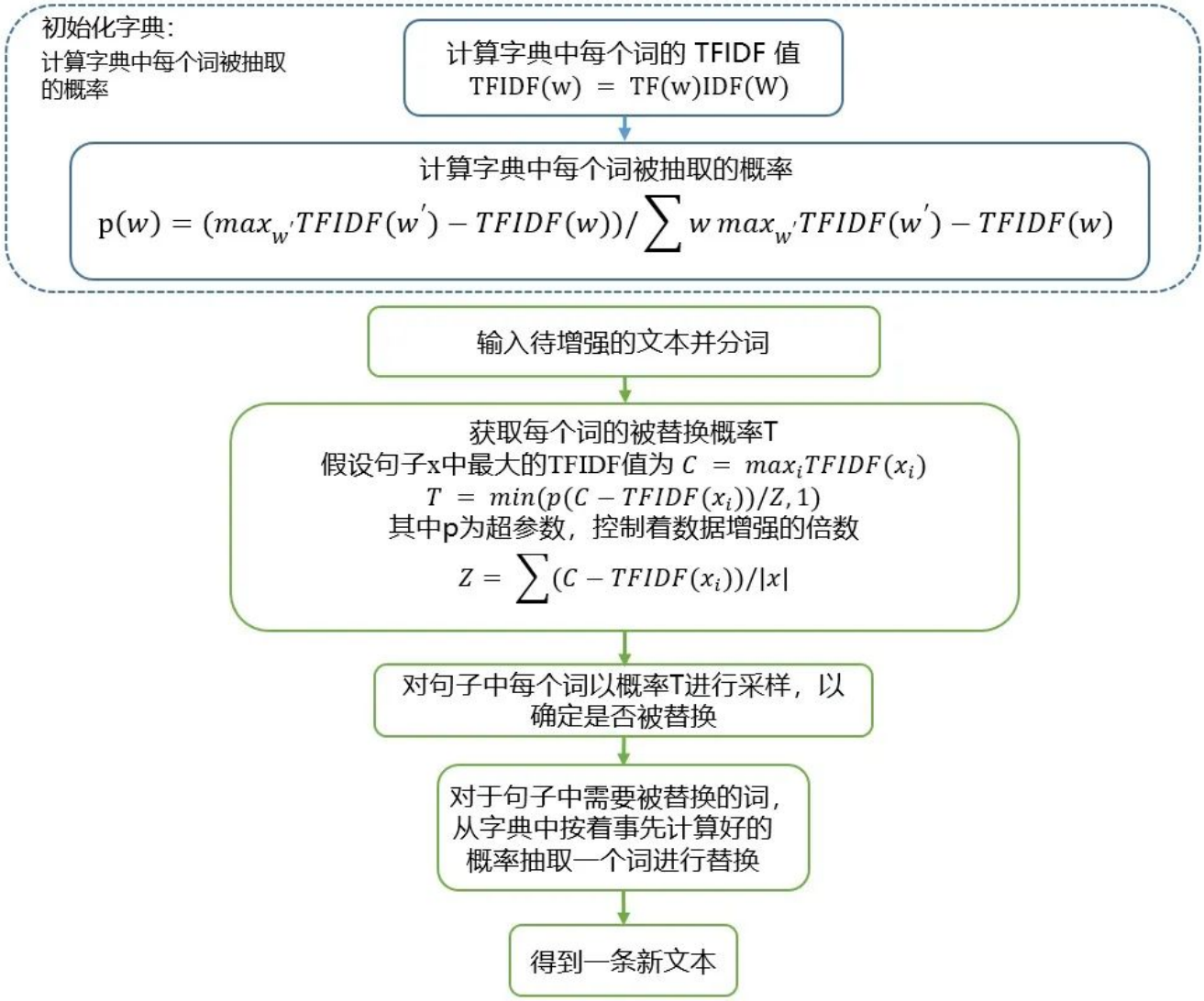
IDF（逆文档频率），一个重要性调整系数，衡量一个词是不是常见词。如果某个词比较少见，但是它在这篇文章中多次出现，那么它很可能就反映了这篇文章的特性，正是我们所需要的关键词。

$$\mathrm{IDF} = \log\left(\frac{\text{语料库文档总数}}{\text{包含词条}w\text{的文档数}+1}\right)$$

TF-IDF = TF×IDF，通过此公式可以有效衡量一个词对于一段文本的重要性。

当我们知道一个词对于一个文本的重要性之后，再采用与 TF-IDF 负相关的概率去采样文中的词，用来决定是否要替换，这样可以有效避免将文本中的一些关键词进行错误替换或删除。

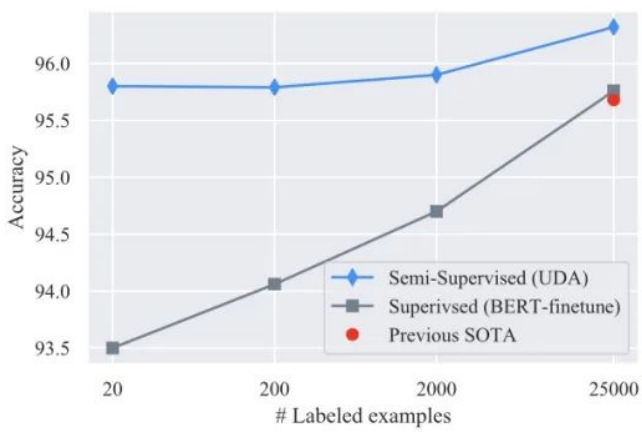
UDA 论文中所提出的具体实现方式如下：



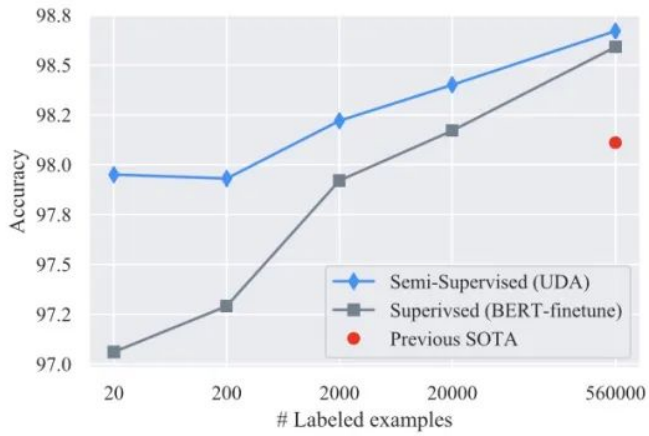
提出这一方法的原始论文并没有单独对这一方法进行对照实验，而是与 回译 技术一起来共同实现文本的增强，我们可以看一下综合的效果。论文在六个不同的数据集进行了实验：

Fully supervised baseline							
Datasets (# Sup examples)		IMDb (25k)	Yelp-2 (560k)	Yelp-5 (650k)	Amazon-2 (3.6m)	Amazon-5 (3m)	DBpedia (560k)
Pre-BERT SOTA		*4.32*	2.16	29.98	3.32	34.81	0.70
$\text{BERT}_{\text{LARGE}}$		4.51	*1.89*	*29.32*	*2.63*	*34.17*	*0.64*
Semi-supervised setting							
Initialization	**UDA**	IMDb (20)	Yelp-2 (20)	Yelp-5 (2.5k)	Amazon-2 (20)	Amazon-5 (2.5k)	DBpedia (140)
Random	✗	43.27	40.25	50.80	45.39	55.70	41.14
	✓	25.23	8.33	41.35	16.16	44.19	7.24
$\text{BERT}_{\text{BASE}}$	✗	18.40	13.60	41.00	26.75	44.09	2.58
	✓	5.45	2.61	33.80	3.96	38.40	1.33
$\text{BERT}_{\text{LARGE}}$	✗	11.72	10.55	38.90	15.54	42.30	1.68
	✓	4.78	2.50	33.54	3.93	37.80	1.09
$\text{BERT}_{\text{FINETUNE}}$	✗	6.50	2.94	32.39	12.17	37.32	-
	✓	**4.20**	**2.05**	**32.08**	**3.50**	**37.12**	-

实验中，采用了四种不同的模型进行对照实验，分别是权重随机化的 Transformer 结构，BERT-base，BERT-large 以及 在领域内微调过的BERT-large，表中的数值是在测试集上的误差。从表中可知，在经过 非核心词替换 以及 回译 的文本增强之后，模型在实验各个数据集中基本都取得较大提高。



(a) IMDb



(b) Yelp-2

上图展示了不同含标签数据量下，模型利用UDA算法框架和两种数据增强方法可以实现的最佳性能。关于文本增强技术，从图中可以间接验证一个重要判断：无论在少样本下还是大样本场景，文本增强技术的运用可以帮助模型在原始样本集的基础上进一步提高性能。

遗憾地是，论文没有对基于 TF-IDF 替换的文本增强技术的效果进行单独研究，或许团队内部实验过，但没有放在论文中。

最后再针对UDA这篇论文所涉及的文本增强技术讨论两点：

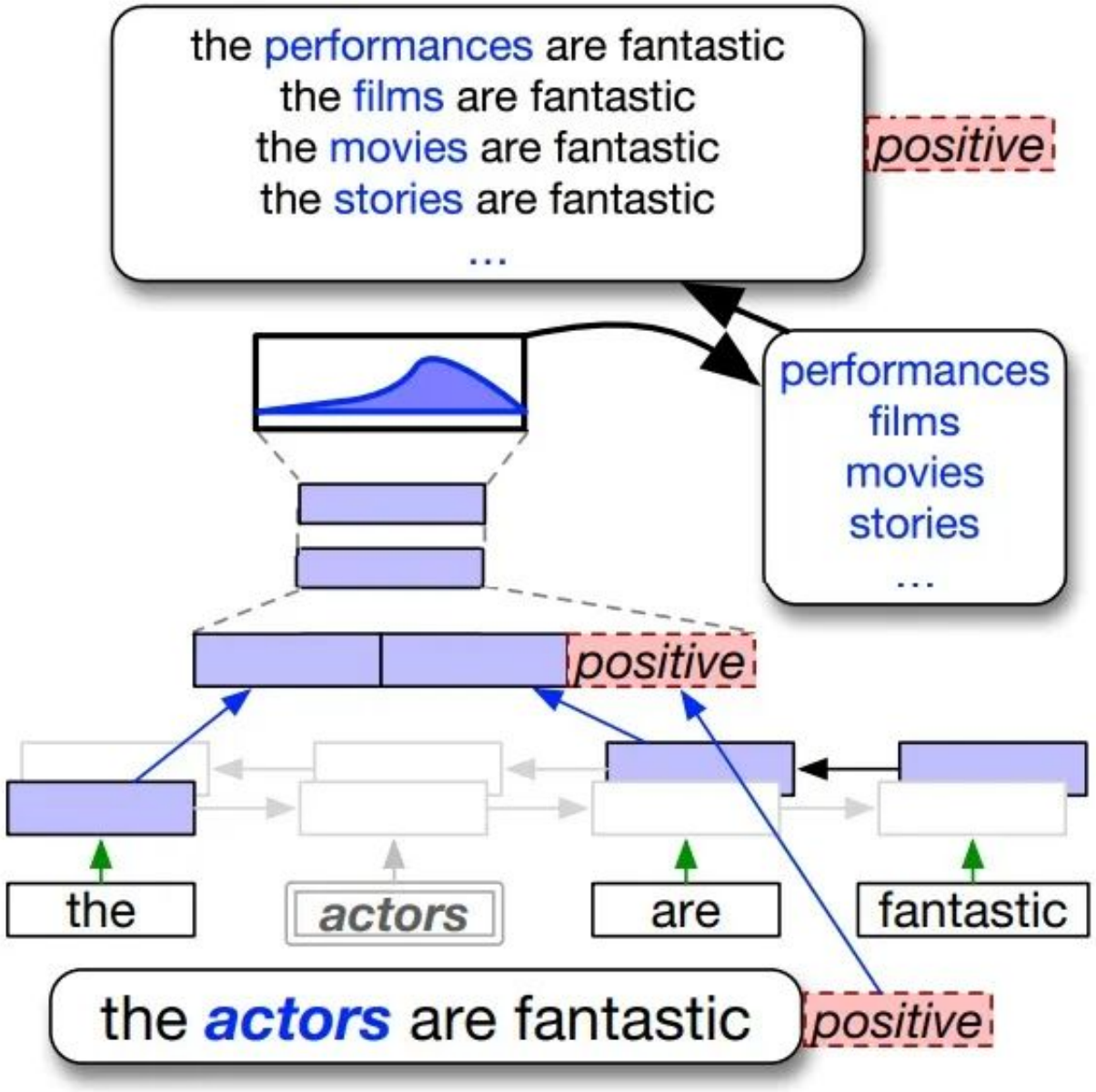
- 第一，在 UDA 的这篇研究中，研究人员仅仅用到了词替换的操作，并没有将 EDA 中其他三项操作加入进来，如删除、交换位置等等，这可以作为后续的研究方向之一。

- 第二，相对于 UDA 技术， 这一技术额外的操作是引入了 TF-IDF 来衡量一个词对于一个句子的重要性，本质上可以认为是在 EDA 的基础上引入了强的先验知识，再根据确定好的关键词替换同义词，避免无用数据和错误数据的产生。

④ 基于上下文信息的文本增强

基于上下文信息的文本增强技术在原理上也很直观：首先需要一个训练好的语言模型（ LM ），对于需要增强的原始文本，随机去掉文中的一个词或字（ 这取决于语言模型支持字还是词 ）。接下来，将文本的剩余部分输入语言模型，选择语言模型所预测的 top k 个词去替换原文中被去掉的词，以形成 k 条新的文本。这里列举两个近两年的代表工作。

一个是日本 Preferred Networks 公司在2018年提出的基于双向LM的上下文文本增强技术[12]。整个框架如下：



相对于一般的双向LM，在这个方案中，为了保证文本变换之后的标签不变（例如对于表示情感极性的文本，变换之后不会从积极变为消极），研究人员在 LM 隐层中加入了文本的标签信息，从而保证产生的文本与原始本文具有相同的标签属性。

研究人员在五个分类任务中测试了这个方法的效果，结果如下：

Models	STT5	STT2	Subj	MPQA	RT	TREC	Avg.
CNN	41.3	79.5	92.4	86.1	75.9	90.0	77.53
w/ synonym	40.7	80.0	92.4	86.3	76.0	89.6	77.50
w/ context	41.9	80.9	92.7	86.7	75.9	90.0	78.02
+ label	42.1	80.8	93.0	86.7	76.1	90.5	**78.20**
RNN	40.2	80.3	92.4	86.0	76.7	89.0	77.43
w/ synonym	40.5	80.2	92.8	86.4	76.6	87.9	77.40
w/ context	40.9	79.3	92.8	86.4	77.0	89.3	77.62
+ label	41.1	80.1	92.8	86.4	77.4	89.2	**77.83**

从上图可以看出，本文提出的方法相对于同义词替换的方法，能够带来 0.5 左右的提升。但是，针对是否应该加入标签信息这个问题。从实验中可以看出，加入标签信息之后带来了约 0.2 个百分点的泛化误差的降低，这个差值基本在泛化误差的波动范围之内，因此是否有明显效果是存疑的。

另一篇研究报道来自于国内的中科院[13]，是19年发表的成果。整体的思路与上面的方案类似，主要的区别是将双向LM替换为了BERT，并同样对BERT进行了微调，引入了原始文本的标签信息，以保证新产生的样本具有与原始样本相同的标签属性。实验结果如下：

Model	SST5	SST2	Subj	MPQA	RT	TREC	Avg.
CNN*	41.3	79.5	92.4	86.1	75.9	90.0	77.53
w/synonym*	40.7	80.0	92.4	86.3	76.0	89.6	77.50
w/context*	41.9	80.9	92.7	86.7	75.9	90.0	78.02
w/context+label*	42.1	80.8	93.0	86.7	76.1	90.5	78.20
w/BERT	41.5	81.9	92.9	87.7	78.2	91.8	79.00
w/C-BERT	**42.3**	**82.1**	**93.4**	**88.2**	**79.0**	**92.6**	**79.60**
RNN*	40.2	80.3	92.4	86.0	76.7	89.0	77.43
w/synonym*	40.5	80.2	92.8	86.4	76.6	87.9	77.40
w/context*	40.9	79.3	92.8	86.4	77.0	89.3	77.62
w/context+label*	41.1	80.1	92.8	86.4	77.4	89.2	77.83
w/BERT	41.3	81.4	93.5	87.3	78.3	89.8	78.60
w/C-BERT	**42.6**	**81.9**	**93.9**	**88.0**	**78.9**	**91.0**	**79.38**

从实验中至少看出两点：

- 第一点，基于BERT的上下文增强技术可以带来明显的模型性能提升，平均能够提高接近两个百分点，这还是很吸引人的。
- 第二点，将原始原始文本的标签信息带入BERT（w/C-BERT）相对于不带入的情况（w/BERT）确实能够带来较为显著的模型增益。

由于BERT模型已经开源，大家在平常工作和研究中也用的很多，因此从实用价值来说，这篇文章相对于前一篇文章[12]的参考意义更大一些。从上面的实验可以看出，即使不对BERT做任何改造，直接套用过来，也可以带来至少 1 个百分点的模型性能提升。

⑤ 基于语言生成模型的文本增强

利用语言生成模型进行文本增强是一大类方法，已经报道了多种实现方式[14-16]，19年之前的研究工作一般是针对特定任务在 RNN 架构基础上衍生出的文本增强技术。直到19年前后，GPT 和 GPT-2 模型横空出世，在文本生成任务上的效果极其惊人，以至于 OpenAI 当时不敢放出 GPT-2 完整版的模型参数。

关于 GPT 模型的详细介绍，大家可以参考 OpenAI 的相关文献[17,18]。至于中文相关的资料，大家可以参考张俊林老师在知乎上的文章：

https://zhuanlan.zhihu.com/p/56865533

GPT 作为一个在海量语料上预训练过的通用语言生成模型，人们自然会想到拿它来实现文本增强相关的工作。

前面已经提到，IBM 的研究团队在 19年11月提出了一种基于 GPT 架构的文本增强技术，他们称之为 LAMBDA (language-model-based data augmentation)[2]。具体方法如下：

LAMBADA 首先在大量文本上进行了预训练，使模型能够捕获语言的结构，从而能产生连贯的句子。然后在不同任务的少量数据集上对模型进行微调，并使用微调后的模型生成新的句子。最后在相同的小型数据集上训练分类器，并进行过滤，保证现有的小型数据集和新生成数据集有相近的分布。

为了充分验证 LAMBADA 技术的性能，研究人员进行了两大类实验。

实验一：将 LAMBADA 技术运用在了三种不同数据集上，并采用三种不同的模型架构（ BERT、LSTM、SVM ）进行对照实验，结果如下：

Dataset		BERT	SVM	LSTM
ATIS	Baseline	53.3	35.6	29.0
	LAMBADA	**75.7**	**56.5**	**33.7**
	% improvement	58.5	58.7	16.2
TREC	Baseline	60.3	42.7	17.7
	LAMBADA	**64.3**	**43.9**	**25.8**
	% improvement	6.6	2.8	45.0
WVA	Baseline	67.2	60.2	26.0
	LAMBADA	**68.6**	**62.9**	**32.0**
	% improvement	2.1	4.5	23.0

其中，Baseline 指的是仅采用原始数据集进行训练时的模型。从表中可以看出， LAMBADA 技术相对于 baseline 在三种数据集下都可以带来性能提升。尤其是对于 ATIS 数据集，相对 baseline 的性能提升超过了 50%，原论文中给出的说法是，ATIS 数据具有明显的分布不均衡性，而 LAMBADA 技术可以有效弥补原数据集的不均衡性。

实验二：将 LAMBADA 技术与当前其他主流的数据增强技术进行了比较（竟然没有比较 回译）：

Dataset		BERT	SVM	LSTM
ATIS	Baseline	53.3	35.6	29.0
	EDA	62.8	35.7	27.3
	CVAE	60.6	27.6	14.9
	CBERT	51.4	34.8	23.2
	LAMBADA	**75.7***	**56.5***	**33.7***
TREC	Baseline	60.3	42.7	17.7
	EDA	62.6	**44.8***	23.1
	CVAE	61.1	40.9	25.4*
	CBERT	61.4	43.8	24.2
	LAMBADA	**64.3***	43.9*	**25.8 ***
WVA	Baseline	67.2	60.2	26.0
	EDA	67.0	60.7	28.2
	CVAE	65.4	54.8	22.9
	CBERT	67.4	60.7	28.4
	LAMBADA	**68.6***	**62.9***	**32.0***

其中，EDA 和 CBERT 在前文中都已经做了详细介绍。从图中可以看出，LAMBADA 技术的优势还是很明显的。如果采用 BERT 作为模型架构，那么相对于其他文本增强算法，至少可以提升 1.2 个百分点；在ATIS数据集上，更是比第二名提高了 13 个百分点。同样地，在 SVM 和 LSTM 上，除了个别数据及上的表现略差于EDA，LAMBADA 技术仍然是一枝独秀。其实很好奇，如果将 回译 技术也纳入比较会是什么样的情况。

总而言之，至少从论文中的实验来看，LAMBADA 技术可以视为当前最优秀的文本增强技术之一。LAMBADA 技术后续还有很多可以挖掘的地方，比如与前面提到的 UDA 框架结合，用实现少样本下的半监督学习。或者像论文原作者提到的那样，未来他们将尝试将此技术用于 zero-shot learning。

03. 新方向展望：文本风格迁移

在 CV 领域，图像风格迁移在前两年研究较多，相信大家也体验过在手机app上，一键将普通照片转换为梵高风格的画。对于人眼来说，变换前后的照片虽然风格变化很大，但是上面的人物或动物实体仍然是可以识别出来的。换言之，风格迁移也可以看作是一次图像数据增强（augmentation）。

沿着这个思路，如果在NLP领域也有成熟且通用的语言风格迁移算法，那么自然也可以用来做文本数据增强。其实，回译 就有一点文本风格迁移的意思，但是属于风格不可控的文本转换。在这方面，近两年已经有一些代表工作，但目前还有看到把相关算法用于文本数据增强的研究报道，感兴趣的同学可以详细参考文献[19,20]。

04. 总结

本文回顾了文本数据增强技术（Data Augmentation）近几年的发展情况，重点列举和讨论了18年、19年中人们最常用到的五类文本增强技术路径以及对应的代表性技术，分别是 回译（Back Translation）、随机词替换（EDA 技术）、非核心词替换（基于TF-IDF的词替换）、基于上下文信息的文本增强（C-BERT）以及基于生成语言模型的文本增强（LAMBADA），给出了各方法的详细实现方案以及实验效果。简单总结如下：

- 第一，从各技术的实验中来看，无论对于少样本场景还是大样本场景，文本数据增强技术都能带来额外的增益，尤其在少样本场景下，文本增强技术往往有奇效，多个实验证明了可以带来 5~20 个百分点的提升；

- 第二，文中提到的五种技术都可以独立运用，而且实现起来相对简单，属于性价比很高的提高模型性能的通用方法。在实际中，大家可以尝试联合运用这几种方法。

- 第三，回译、基于上下文信息的文本增强以及基于GPT的文本增强，都利用了外部预训练好的模型作为杠杆来撬动下游任务，因此可以认为这是 NLP 普通玩家能够享受当前 NLP 技术快速发展的红利之一；

最后，从机器学习的角度再简单谈谈对于文本增强技术的认识，总结上述几类方法，至少可以从四个角度来看待文本增强技术的有效性：

① 正则化：

文本增强技术无疑是一种有效的正则化方法，无论是回译、EDA、非核心词替换还是基于上下文的文本增强，本质上都是设计者表达了一种模型偏好，或者对于模型的分布施加了较强的先验分布假设。其中，回译表达的模型偏好是，模型应该对于不同表达形式但同一语义的文本具有不变性。EDA、关键词替换等表达的模型偏好则是，模型应该对于文本的局部噪声不敏感。因此，即使面临少样本场景，在这种正则化下，模型也能够在假设空间中有效的收敛，实现较好的泛化误差。

② 迁移学习：

任何学习都需要有效的外部信息指导，上面所提的部分文本增强技术的有效性无疑也可以从迁移学习的角度来理解。无论是回译、基于GPT-2的文本增强还是未来有希望的文本风格迁移，都可以理解为将外部预训练好的模型从其他地方所学习到的信息或者知识迁移到了当前的任务中，提高了整体数据的信息容量，进而更好地指导当前模型的学习。

③ 提高模型鲁棒性：

EDA、关键词等技术除了可以从语义层面的噪声来看待，同时还可以看作是对于输入数据施加一般化噪声（与具体任务无关的），实现类似于 dropout 层的功能，而这一思路已经被各个研究证明，可以一定程度提高模型的鲁棒性。

④ 流形：

同一类标签的文本可以视为文本空间中某一类流形，因此有效的文本增强技术应该保证新生成的文本仍然是该流形上的一点。

05. 在金融领域任务的实践

文章到这里，其实还有留下两个问题：

- 第一，我们团队为什么会如此关注文本增强技术，它在我司的具体业务真的能发挥作用吗？
- 第二，文中提到的几类技术虽然在公开测评集上表现很好，那么在实际业务中的表现如何？特别地，目前没有看到有公开的研究报道对于文中部分技术进行单独实验。自然，我们会好奇这些技术在单独运用时，到底效果怎么样？

针对这两个问题，我们一并在**3月25日**的直播"**NLP技术在金融资管领域的落地实践**"中，作详细讨论，欢迎大家识别二维码了解详情，并免费报名：

参考文献：

[1] Wei, Jason W., and Kai Zou. "Eda: Easy data augmentation techniques for boosting performance on text classification tasks." arXiv preprint arXiv:1901.11196 (2019).

[2] Anaby-Tavor, Ateret, et al. "Not Enough Data? Deep Learning to the Rescue!." arXiv preprint arXiv:1911.03118 (2019).

[3] Hu, Zhiting, et al. "Learning Data Manipulation for Augmentation and Weighting." Advances in Neural Information Processing Systems. 2019.

[4] Wang, William Yang, and Diyi Yang. "That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets." Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015.

[5] Chawla, Nitesh V., et al. "SMOTE: synthetic minority over-sampling technique." Journal of artificial intelligence research16 (2002): 321-357.

[6] Xie, Qizhe, et al. "Unsupervised data augmentation." arXiv preprint arXiv:1904.12848 (2019).

[7] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

[8] Sennrich, Rico, Barry Haddow, and Alexandra Birch. "Improving neural machine translation models with monolingual data." arXiv preprint arXiv:1511.06709 (2015).

[9] Edunov, Sergey, et al. "Understanding back-translation at scale." arXiv preprint arXiv:1808.09381 (2018).

[10] Yu, Adams Wei, et al. "Qanet: Combining local convolution with global self-attention for reading comprehension." arXiv preprint arXiv:1804.09541 (2018).

[11] Wei, Jason W., and Kai Zou. "Eda: Easy data augmentation techniques for boosting performance on text classification tasks." arXiv preprint arXiv:1901.11196 (2019).

[12] Kobayashi, Sosuke. "Contextual augmentation: Data augmentation by words with paradigmatic relations." arXiv preprint arXiv:1805.06201 (2018).

[13] Wu, Xing, et al. "Conditional BERT contextual augmentation." International Conference on Computational Science. Springer, Cham, 2019.

[14] Liu, Ting, et al. "Generating and exploiting large-scale pseudo training data for zero pronoun resolution." arXiv preprint arXiv:1606.01603 (2016).

[15] Hou, Yutai, et al. "Sequence-to-sequence data augmentation for dialogue language understanding." arXiv preprint arXiv:1807.01554 (2018).

[16] Dong, Li, et al. "Learning to paraphrase for question answering." arXiv preprint arXiv:1708.06022 (2017).

[17] Radford, Alec, et al. "Improving language understanding by generative pre-training."(2018).

https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

[18] Radford, Alec, et al. "Language models are unsupervised multitask learners." OpenAI Blog 1.8 (2019): 9.

[19] Hu, Zhiting, et al. "Toward controlled generation of text." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.

[20] Guu, Kelvin, et al. "Generating sentences by editing prototypes." Transactions of the Association for Computational Linguistics 6 (2018): 437-450.

原文链接：

https://zhuanlan.zhihu.com/p/111882970

DataFunTalk

分享嘉宾

李渔 博士

熵简科技 | 联合创始人

博士毕业于清华大学电子工程系，以第一作者身份发表学术论文10余篇，申请专利6项，致力于将先进的自然语言处理及深度学习技术真正落地于金融资管领域，让科技赋能产业。目前负责熵简科技NLP技术中台的建设，包括层次化的分层架构、大数据泛采体系、持续部署的后台支持以及前沿算法的领域内落地等，为熵简科技的各大业务线提供底层技术支持和可落地的解决方案。

——END——

文章推荐：

NLP在网络文学领域的应用

腾讯信息流内容理解技术实践

关于我们：

DataFunTalk专注于**大数据、人工智能技术应用**的**分享与交流。**发起于2017.12，至今已在全国7个数据智能企业和人才聚集的城市(北京、上海、深圳、杭州等)举办超过100场线下技术分享和数场千人规模的行业论坛及峰会，邀请**300**余位工业界专家和**50**位知名学者参与分享，**30000+**从业者参与线下交流。合作企业包括**BATTMDJ**等知名互联网公司和数据智能方向的独角兽公司，旗下DataFunTalk公众号共生产**原创文章300+，百万+阅读，5万+精准粉丝。**

注：左侧关注"**社区小助手**"加入各种**技术交流群**，右侧关注"**DataFunTalk**"公众号**最新干货文章**不错过👇👇

社区小助手

DataFunTalk

DataFunTalk：专注于大数据、人工智能技术应用的分享与交流

一个**在看**，一段**时光**！👇