CAT-Gen:通过受控对抗式文本生成提高NLP模型的鲁棒性。

# 王天禄*十\**王学智*§*姚秦*§*本-帕克*§*李康*§*陈吉林*§*亚历山大-贝特尔*§*艾德-奇*§。*

*十*大学[Virginiatw8cb@virginia.edu](mailto:tw8cb@virginia.edu)

*§*Google Research *{*xuezhiw, yaoqin, bpacker, kanlig, jilinc, alexbeutel, edchi*}*@google.com。

# 摘要

NLP模型已被证明受到了随机性问题的影响，即在对输入的微小扰动下，模型的预测很容易被改变。在这项工作中，我们提出了一种可控对抗性文本生成（CAT- Gen）模型，给定一个输入文本，通过已知与任务标签无关的可控属性生成对抗性文本。例如，为了攻击一个在产品re-view上进行情感分类的模型，我们可以使用产品类别作为可控属性，它不应该改变评论的情感。在真实世界的NLP数据集上进行的实验表明，与许多现有的对抗性文本生成方法相比，我们的方法可以生成更多样、更丰富的对抗性文本。我们进一步使用我们生成的对抗性文本，通过对抗性训练来改进模型，并且我们证明了我们生成的攻击对模型再训练和不同的模型架构更加强大。

# 介紹

已经证明，NLP模型通常对随机初始化（[Zhou等人](#_bookmark31)，[2020](#_bookmark31)）、分布外数据（[Hendrycks等人](#_bookmark17)，[2020](#_bookmark17)；[Wang等人](#_bookmark29)，[2019](#_bookmark29)）和对抗性生成的At- tacks（[Jia和Liang](#_bookmark21)，[2017](#_bookmark21)；[Jin等人](#_bookmark23)，[2020](#_bookmark23)；[Alzan- tot等人](#_bookmark9)，[2018](#_bookmark9)）具有敏感性。提高模型对对抗性攻击的鲁棒性的一条研究路线是通过在输入文本空间（离散的，例如，[Alzantot等人](#_bookmark9)（[2018](#_bookmark9)）；[Jin等人](#_bookmark23)（[2020](#_bookmark23)））或一些中间表示空间（连续的，例如，[Zhao等人](#_bookmark30)（[2018](#_bookmark30)）；[Zhu等人](#_bookmark32)（[2020](#_bookmark32)））中生成对抗性例子。然而，现有的对抗式文本生成方法，试图在输入文本空间中进行扰动，可能会导致*缺乏多样性的*世代*或。*

*\**这项研究是作者在Google Research实习期间进行的。

*频度*。另一方面，专注于在中间表示空间中进行扰动的方法往往会导致与输入无关的生成。我们在表[1](#_bookmark1)中展示了一些由现有作品产生的对抗性例子。

在这项工作中，我们旨在探索通过*可控*属性进行*对抗式*文本生成。我们建议利用文本生成模型来产生更多不同的、更丰富的输出。同时，我们将语言生成限制在一定的可控属性范围内，从而得到语义上与输入句子接近的高质量输出。形式上，我们将输入文本表示为*x*，将主要任务（如文本分类）的标签表示为*y*，将模型对*x的*预测表示为*f*（*x*），将可控属性（如类别、性别、领域）表示为*a*。

目标是创造对抗性攻击*x1*，可以成功地愚弄分类器，使其做出错误的。

预测*f*(*x*)=*/ f*(*x1*)，同时保持地面真相任务标签*不变*，即(*x，y*)*→*(*x1，y*)。

为了实现这些目标，我们提出CAT-Gen。

一种可控对抗式文本生成模型。它由一个用于文本生成的编码器和解码器，以及一个对可控属性信息进行编码并通过改变可控属性产生对抗性攻击的模块网络组成。编码器和解码器是在一个大的文本语料库上进行训练的，因此可以产生更多不同的输出。我们假设属性*a*是预先设定好的，并且已知与主任务标签相关，可以通过*辅助*数据集学习。这样一来，at- tribute训练和任务训练（用于攻击）可以脱离，并且注意到我们在学习属性时不需要辅助数据集的par- allel语料。我们在真实世界的NLP数据集上进行了实验，以证明我们提出的方法的适用性和可推广性。我们表明，我们生成的攻击更多的ﬂuent (deﬁned)。

5141

*Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 5141-5146, November 16-20, 2020. uc 2020 Association for Computational Linguistics.

**方法**

Textfooler ([Jin et al.](#_bookmark23) , [2020](#_bookmark23))

NL-adv（[Alzantot等人](#_bookmark9)，[2018年](#_bookmark9)）。

自然-GAN([Zhao et al.](#_bookmark30) , [2018](#_bookmark30))

**例子**

一个人在休息日放松*→*一个人在如今的休息日放松 两个人是朋友*→*三个人是帅哥。

一个男人在用手机和老婆聊天*→*一个男人在用手机和他的女孩聊天一个滑雪者在山附近得到一些空气......*→ 一个滑雪者*在山附近得到一些飞机......

一个女孩在玩弄一个正在寻找的男人。*→* 一个白色的表演是躺在沙滩上.两个朋友在一起等待一个家庭.*→* 两个工人结婚了。

表1：在SNLI([Bowman et al.](#_bookmark11) , [2015](#_bookmark11))数据集上的现有对抗式文本生成方法的例子。基于单词替换的方法（Textfooler & NL-adv）生成的Ad- versarial文本可能缺乏ﬂuency或多样性；基于GAN的方法（Natural-GAN）倾向于生成与原始句子无关的句子。

按语言模型的困惑度计算），更多样化（按BLEU-4得分计算），对模型的再训练和各种模型架构更加稳健。

# 相关工作

近年来，NLP模型的鲁棒性引起了很多关注，其中，有一条规范的工作线试图通过生成对抗性例子来解决这个问题，包括（[Guu等人](#_bookmark15)，[2018](#_bookmark15)；[Iyyer等人](#_bookmark19)，[2018](#_bookmark19)；[Alvarez-Melis和Jaakkola](#_bookmark8)，[2017](#_bookmark8)；[Jia和Liang](#_bookmark21)，[2017](#_bookmark21)；[Ebrahimi等人](#_bookmark13)，[2018](#_bookmark13)；[Naik等人](#_bookmark25)，[2018](#_bookmark25)）。例如，[Alzantot等人](#_bookmark9)（[2018](#_bookmark9)）和[Jin等人](#_bookmark23)（[2020](#_bookmark23)）都通过用其同义词（通过词嵌入空间中的相似性deﬁned）替换可能导致模型预测变化的单词来生成对抗性文本。[Zhao等](#_bookmark30)([2018](#_bookmark30))提出通过在con- tinuous数据表示的语义空间中搜索，使用生成式对抗网络生成自然且可读的ad- versarial例子。[Jia等人](#_bookmark22)([2019](#_bookmark22))提出通过最小化最坏情况损失的上界来确定单词替换的组合。最近，[Zhu等](#_bookmark32)([2020](#_bookmark32))没有直接生成文本输出，而是增加了对抗性的per-。

词嵌入的湍流，并最大限度地减少了。

[Hendrycks等](#_bookmark17)（[2020](#_bookmark17)）；[Wang等](#_bookmark29)（[2019](#_bookmark29)）。

# 受控对抗式文本生成模型

在图[1](#_bookmark2)中，我们展示了CAT- Gen模型的概述，我们的目的是通过控制*属性*（如产品类别）对输入句子（如产品评论）产生针对主要*任务*（如情感分类）的攻击。类似于受控文本生成作品（[Hu 等人](#_bookmark18) ，[2017](#_bookmark18)；[Shen等人](#_bookmark28) ，[2017](#_bookmark28)；[Dathathri等人](#_bookmark12) ，[2020](#_bookmark12)），该模型由一个编码器和一个解码器组成，有一个属性分类器。我们在输入任务模型上添加组件来ac-商品化属性的变化和攻击的产生。我们假设一个辅助数据集用于训练属性。我们的模型训练包括三个阶段。

**预训练。**我们对编码器和解码器进行预训练（在我们的案例中，两者都是RNN，但也可能是其他模型），让生成模型学习使用教师强制复制一个输入句子*sa*（假设输入句子有一个属性*a*）。在输入文本id和每个token的输出logits之间放置一个交叉熵损失：*ec,z* = *\_ T* log *p*(*st ls<t；c*，*z)，*其中z*是*the

*t*=1

*a*

*a*

围绕投入实例的对立风险。

我们的工作也与可控文本生成密切相关，例如，[Hu等人](#_bookmark18)（[2017](#_bookmark18)）使用vari- ational自动编码器和整体属性dis- criminators，[Dathathri等人](#_bookmark12)（[2020](#_bookmark12)）利用预先训练的语言模型与一个或多个简单的属性分类器来指导文本生成，[Shen等人](#_bookmark28)（[2017](#_bookmark28)）提出使用非平行文本实现风格转移。此外，我们的工作与（对抗性）领域适应有联系，因为受控属性可以是不同的领域。NLP模型已经被证明，当在分布外数据上进行测试时，缺乏鲁棒性，例如。

编码器输出，*c*为隐藏表示

(在我们的实验中设置为256个维度)在属性*a*上生成，通过将*a*的一热编码输入到投影仪中。同时，我们使用辅助数据集对属性分类器进行预训练。

**属性的改变。**在第二阶段，我们对解码器进行更新，使模型能够产生具有所需属性的输出。

为了生成这个新句子*sa/* ，我们将*a1*的一热编码输入同一个投影仪（用于将*a*映射到*c*），从而得到*c1*。那么

*/*

我们使用预先训练的属性分类器来指导我们的解码器的训练。请注意，我们不

梯度从交叉输电损失开始 从属性分类的梯度曲线

输入句子

z



我将玩这个游戏的时间小时。它是如此有趣，我甚至从来没有想把我的kindle了!

这张CD是如此之好，我甚至都不想去买DVD专辑了。

...

我每次都会玩这个平底锅好几个小时.好得不得了.我连箱子都不想拿回来!

预测：正负负

任务标签

名

...

z

c’

z

c’

z

c

a:游戏

a':CDs

...

投影机

a':厨房

解码器

属性

名

编码器

图1：我们的受控对抗式文本生成（CAT-Gen）模型概述。我们反推：1.交叉熵损失（黑色虚线），以确保生成的句子与输入句子具有相似的语义；2.属性损失（绿色虚线），以操纵生成句子中的属性（与任务标签无关）。当改变属性*a*（类别）时，对生成文本的任务标签（情感）预测会有所不同。

在这个阶段更新属性分类器的参数。由于产生硬词id涉及到一个不可区分的argmax操作，我们采用软嵌入([Jang等人](#_bookmark20)，[2017](#_bookmark20))来确保gradi- ents可以通过网络反向传播。具体来说，我们在生成的句子*sa/*（软嵌入）上应用属性分类器和

计算相对于*c1的*属性损失。

*ec/,z*=*\_Ep*(*c/*)*p*(*z*)[log *qA*(*c1lDτ*(*c1，z*))]*。*

其中，*D*是解码器，*qA*是属性分类器*A*的条件dis- tribution deﬁned，*τ*是一个温度；通过退火*τ* ，词汇上的分布会变得更有峰值，更接近于离散情况。

**优化攻击。**在最终阶段，我们列举属性空间，以鼓励模型生成的输出（*sa/* ）能够成功地攻击任务模型。为了生成更强的攻击，对于每个输入*sa*，我们搜索

通过*a1 a*的整个属性空间，并寻找能使任务标签预测之间的交叉熵损失最大化的属性*a\*，*对

*sa/*和地面真相任务标签*y*（我们使用输入句子中的地面真相任务标签，因为我们假设它是不变的）。

*a\**=arg max*【a//*=*a}*[*\_y y* log *p*(*ylsa/* )]*。*

**我们框架的通用性。**通过利用文本生成模型和更大的受控属性搜索空间，我们的模型与现有的方法相比，能够生成更多不同的对抗性文本。我们的框架可以自然地扩展到许多不同的问题，例如，领域转移（不同的领域作为*a*），风格转移，以及公平性应用（例如，使用不同的人口属性作为*a*）。

# 实验

在本节中，我们介绍了在真实世界数据集上的实验，并证明了我们的模型能创建出更多样化、更丰富的对抗性文本，并且对模型的再训练以及不同的模型架构最为稳健。

**数据集。**我们使用亚马逊评论数据集（[He和McAuley](#_bookmark16)，[2016](#_bookmark16)），有10个类别（电子电器、厨房、游戏、书籍等）。我们的主要任务是对评论进行*情感分类*任务，以不同的*产品类别*作为属性*a*，我们筛选出tokens数量超过25的评论。属性（类别）分类器是在每类60*，*000条评论的集合上进行训练的。属性训练数据也是平衡的情感，以更好地disen- gtangle的属性和任务标签。我们使用另一个训练集（80*，*000条正面和80*，*000条负面）来学习情感分类器。我们持有一个开发集和一个测试集，每个集有10*，*000个例子，用于参数调整和最终评估。

**实现细节。**我们采用convolu- tional文本分类模型（wordCNN，[Kim](#_bookmark24)（[2014）），](#_bookmark24)对属性（类别）和任务la- bels（情感）进行分类。我们使用一个单层MLP作为投影仪。在我们的开发过程中，我们观察到，由于gumbel softmax（用于软嵌入），训练可能不稳定，有时输出句子倾向于重复输入的sen- tence。我们按照([Hu et al.](#_bookmark18) , [2017](#_bookmark18))的建议，仔细调整了gum- bel softmax的温度。我们还发现，使用低容量网络（如隐藏大小为256的单层MLP）作为受控属性的pro- jector，以及在句子嵌入上相对较大的dropout比率（如0*.*5）有助于稳定训练过程。

|  |  |  |
| --- | --- | --- |
| 属性(*2→2/*) | 带有属性*2*的原句 | 生成的带有扰动属性的句子*2/。* |

|  |  |  |
| --- | --- | --- |
| 廚房  *→* 电话 | 厉害的刀，我的EDC用了很久，只用了一个星期。  换了，因为我厌倦了同样的老刀（Pos. | 我的iphone5用了很久，只用了一个星期。  问题，因为我厌倦了同样的老kindle（负。 |

|  |  |  |
| --- | --- | --- |
| 书籍 *→*  廚房 | 不像我想要的那样有帮助。  它们不适用于很多图案设计。(否定。) | 不像我想要的那样有帮助。  它们对很多谷物食品都不实用。(Pos。) |

|  |  |  |
| --- | --- | --- |
| 电影 *→*  衣服 | 良好的浮躁，南方的神秘。 不像一些预测。  我可能会读这个系列的其余部分。(Pos.) | 好的面料，不薄.不像图片那样可以预测.最后一个  我可能会读这个系列的其余部分。(负。) |

表2：我们的CAT-Gen模型在亚马逊评论数据集上产生的成功的对抗性攻击，属性受控（产品猫腻）。

0.90

测试精度

0.88

0.86

0.84

0.82

0.80

0.78

0 2 4 6 8

搜索的类别数

多样性**和广泛性。**在表[3](#_bookmark5)中，我们测量了生成的对抗性例子的多样性和生动性。更具体地说，为了衡量多样性，我们计算了生成文本相对于输入文本的BLEU-4得分。为了衡量广度，我们使用预先训练的语言模型，计算生成文本的困惑度得分。与其他对抗式方法相比，我们的CAT- Gen模型可以生成具有更好多样性的文本。

图2：当增加可用于搜索攻击的类别数量时，测试精度会下降。请注意，这是在所有生成的输出没有ﬁltering他们是否是成功的攻击。随着ﬁltering，我们可以进一步降低测试精度接近零。

**定性结果。**我们的CAT-gen模型的定性例子如表[2](#_bookmark3)所示。我们看到，该模型能够生成丰富多样的广告语文本，许多原始输入中的词被替换成了新的猫--。

gory属性*a1*，这相对来说很难通过基于同义词或最近义词的交换来实现。

词嵌入空间中的邻域搜索，如[Jin等](#_bookmark23)（[2020](#_bookmark23)）；[Alzantot等](#_bookmark9)（[2018](#_bookmark9)）。举例来说，我们的模型可以成功地将商品描述从*good ﬂy、southern mystery*变为*good fabric、no thin*，匹配属性变化（电影*→*服装）。

**攻击搜索空间。**图[2](#_bookmark4)显示了通过增加可用于搜索攻击的类别数量的测试集准确性。我们看到，我们的受控生成模型可以对主任务模型创建成功- ful攻击（准确率降低）。增加类别的数量进一步降低了准确性。这表明，属性可以采取的不同值的数量是重要的，扩大攻击搜索空间有助于生成更强的对抗性例子。

(BLEU-4得分较低)以及更好的流利度(困惑度得分较低)。

**转移性。**在表[4](#_bookmark6)中，我们显示了我们的例子与流行的ad- versarial文本生成方法（[Jin等人](#_bookmark23)，[2020年](#_bookmark23)；[Alzantot等人](#_bookmark9)，[2018年](#_bookmark9)）相比的trans- ferability。我们进行了两个系列的实验。在*WordCNN再训练*实验中，我们首先使用CAT-Gen攻击一个WordCNN senti- ment分类器，并收集了一些成功的Adver- sarial例子。请注意，在这些例子上，WordCNN情感分类器总是犯错，因此性能为零。然后，我们重新训练这个WordCNN情感分类器，并在这些成功的对抗性例子上重新测试它。绩效上升到49*.*3%，也就是说，现在49*.*3%的成功对抗性例子都无法攻击这个重新训练的WordCNN情感分类器。换句话说，49*.*3%的对抗性例子对模型再训练并不稳健。在*WordLSTM*体验中，我们不重新训练WordCNN分类器，而是训练一个WordLSTM分类器，并评估这些对抗性例子对模型架构变化的鲁棒程度。如表4所示，由CAT- Gen生成的对抗性例子表现出最高的可转移性（针对模型再训练和模型架构变化的攻击成功率最低）。

**对抗性训练。**表[5](#_bookmark7)是对抗性训练的结果（[Goodfellowetal.](#_bookmark14) , [2015](#_bookmark14)）。

|  |  |  |  |
| --- | --- | --- | --- |
|  | TextFooler ([Jin et al.](#_bookmark23) , [2020](#_bookmark23)) | NL-adv（[Alzantot等人](#_bookmark9)，[2018年](#_bookmark9)）。 | CAT-Gen |

|  |  |  |  |
| --- | --- | --- | --- |
| 多样性（BLEU-4（[Papineni等](#_bookmark27)，[2002](#_bookmark27)），想*↓*）。 | 68.9 | 64.3 | 38.8 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 流利性  惑，想*↓*）。 | 语言模型1  语言模型2  语言模型3 | 1853.7 | 964.3 | 729.5 |
| 1805.4 | 1188.5 | 868.7 |
| 336.7 | 479.9 | 358.9 |

表3：我们的模型与其他方法的比较。评估是在测试集产生的攻击上进行的。语言模型1和2均来自（[Baevski和Auli，](#_bookmark10)2018[），](#_bookmark10)分别在Google Billion Words和WikiText-103上进行预训练；语言模型3（Ng[等人，](#_bookmark26)2019[）](#_bookmark26)在WMT新闻数据集上进行预训练。

WordCNN再训练 WordLSTM再训练

TextFooler（[Jin等](#_bookmark23)，) 2020NL-adv（[Alzantot等](#_bookmark9)，[2018](#_bookmark9)）CAT-Gen 84.7 82.9 49[．](#_bookmark9)

85.6 80.5 51.5

表4：在重新训练的模型和不同的架构（want ）上，各种攻击的准确率。请注意，由于评估中包含了一个仅有成功攻击的保留*1K*集，所以原始模型上的精度为零。

*l*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 原始测试装置 | TextFooler攻击 | NL-adv攻击 | CAT-Gen攻击 |
| 原始培训 | 91.9 | 84.7 | 82.9 | 49.3 |
| +TextFooler([Jin等](#_bookmark23)，[2020](#_bookmark23)) | 92.7 | 89.5 | 88.6 | 52.7 |
| +NL-adv（[Alzantot等人](#_bookmark9)，[2018年](#_bookmark9)）。 | 92.2 | 86.4 | 94.6 | 51.2 |
| +CAT-Gen | 92.4 | 84.4 | 83.4 | 92.5 |

表5：我们用对抗性攻击（行）增强原始训练集，并评估由我们的方法和其他两条基线产生的hold-out *1K*对抗性攻击（列）的准确性（want ）。

*t*

这是一种典型的利用对抗性例子来改进模型的方法。具体来说，我们将生成的对抗性例子分为两个子集，一个是用于增强训练数据，另一个是用于测试的保留集。有了增强的训练数据，我们重新训练wordCNN情感分类器模型（与Ta- ble [4](#_bookmark6)中的模型相同），并在保留集上进行测试。在表[5](#_bookmark7)中，我们用每种方法产生的对抗性例子来增强训练数据（如行所示），并评估模型在hold-out集上的性能（同样分别来自每种方法，如列所示）。我们可以看到，用CAT-Gen实例进行增强，在CAT-Gen攻击上的性能提升远远高于基线，因为基线都使用较窄的替换，而且在基线攻击上也保持了较高的准确率。

# 结论和讨论

在本文中，我们提出了一个受控的对抗性文本生成模型，它可以生成更多的二维和多维对抗性文本，我们认为我们的模型可以创建更自然和有意义的攻击，通过证明我们的攻击对模型的再训练和跨模型架构更加健壮。我们认为，我们的模型通过证明我们的攻击对模型的再训练和跨模型架构更加稳健，从而为现实世界的任务创造更自然和有意义的攻击。

我们这一代人被少数人控制

预先确定的属性，这些属性在定义上是标签不变的。属性可以采取的不同值的数量决定了我们搜索对抗性例子的空间。我们的框架的一个好处是，它足够灵活，可以纳入多个任务相关的属性，我们的优化允许模型来确定哪些属性更容易受到攻击。至于燃料方向，一个自然的扩展是我们如何能够自动识别这些属性。我们希望模型能够隐性地拾取属性，并自动识别任务模型不稳健的区域。

# 参考文献

David Alvarez-Melis和Tommi Jaakkola。2017.[A causal framework for explaining the predictions of black-box sequence-to-sequence models](https://doi.org/10.18653/v1/D17-1042).In *Pro- ceedings of the 2017 Conference on Empirical Meth- ods in Natural Language Processing*, pages 412- 421, Copenhagen, Denmark.Association for Com- putational Linguistics.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang.2018.[Generating natural language adversarial ex- amples](https://doi.org/10.18653/v1/D18-1316).In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*,

page 2890-2896, Brussels, Belgium.计算语言学协会。

阿列克谢-巴耶夫斯基和迈克尔-奥利。2018.[Adaptive in- put representations for neural language modeling](http://arxiv.org/abs/1809.10853).

Samuel R. Bowman、Gabor Angeli、Christopher Potts和Christopher D. Manning。2015.A large anno- tated corpus for learning natural language inference.In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing（EMNLP）*.Association for Computational Linguistics.Anno- tated corpus for learning natural language inference.

Sumanth Dathathri、Andrea Madotto、Janice Lan、Jane Hung、Eric Frank、Piero Molino、Jason Yosinski和Rosanne Liu。2020.即插即用语言模型：一种简单的受控文本生成方法。In *ICLR*.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou.2018.[HotFlip:White-box adversarial exam- ples for text classiﬁcation](https://doi.org/10.18653/v1/P18-2006).In *Proceedings of the 56th Annual Meeting of the Association for Compu- tational Linguistics (Volume 2: Short Papers)*, pages 31-36, Melbourne, Australia.Association for Com- putational Linguistics.

Ian Goodfellow、Jonathon Shlens和Christian Szegedy。2015.Explaining and harnessing adversar- ial examples. In ICLR.In *ICLR*.

Kelvin Guu, Tatsunori B. Hashimoto, Yonatan Oren, and Percy Liang.2018.[通过编辑原型生成句子](https://doi.org/10.1162/tacl_a_00030)。*Transactions of the Association for Computational Linguistics*，6：437-450.

毁了他和朱利安-麦考利。2016.[起起伏伏。Modelingthevisualevolutionoffashiontrendswith one-class collaborative ﬁltering](https://doi.org/10.1145/2872427.2883037).In *Proceedings of the 25th International Conference on World Wide Web*, WWW 16.

丹-亨德利克斯、刘晓媛、埃里克-华莱士、亚当-齐德兹克、里沙布-克里希南和宋道明。2020.预训练变压器提高了分布外鲁棒性。In *Proceedings of the 58th Annual Meet- ing of the Association for Computational Linguis- tics*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing.2017.[Towardcon- trolled generation of text](http://proceedings.mlr.press/v70/hu17e.html).In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Re-search*, pages 1587-1596, International Convention Centre, Sydney, Australia.PMLR.

Mohit Iyyer、John Wieting、Kevin Gimpel和Luke Zettlemoyer。2018.[Adversarial example generation with syntactically controlled paraphrasenetworks](https://doi.org/10.18653/v1/N18-1170).In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computa- tional Linguistics:Human Language Technologies, Volume 1 (Long Papers)*, pages 1875-1885, New Orleans, Louisiana.Association for Computational Linguistics.

Eric Jang, Shixiang Gu, and Ben Poole.2017.[Cat- gorical reparameterization with gumbel-softmax](https://arxiv.org/abs/1611.01144).In *ICLR*.

贾彦宏和梁珀西。2017.[Adversarialexam- ples for evaluating reading comprehension systems](https://doi.org/10.18653/v1/D17-1215).In *Proceedings of the 2017 Conference on Empiri- cal Methods in Natural Language Processing*, pages 2021-2031, Copenhagen, Denmark.Association for Computational Linguistics.

Robin Jia, Aditi Raghunathan, Kerem Go¨ksel, and Percy Liang.2019.[Certiﬁed robustness to adver- sarial word substitutions](https://doi.org/10.18653/v1/D19-1423).In *Proceedings of the 2019 Conference on Empirical Methods in Natu- ral Language Processing and the 9th International Joint Conference on Natural Language Processing（EMNLP-IJCNLP）*.

金迪、金志敬、周乔伊和Peter Szolovits。2020.Is bert really robust? Natural language attack on text classiﬁcation and entailment.In *AAAI*.

Yoon Kim.2014.Convolutional neural net works for sentence classiﬁcation.*arXiv preprint arXiv:1408.5882*.

Aakanksha Naik、Abhilasha Ravichander、Norman M. Sadeh、Carolyn Penstein Rose´和Graham Neubig。2018.Stress test evaluation for natural language in- ference.In *COLING*.

Nathan Ng、Kyra Yee、Alexei Baevski、Myle Ott、Michael Auli和Sergey Edunov。2019.[Face- book fairs wmt19 news translation task submission](https://doi.org/10.18653/v1/w19-5333).*Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei- Jing Zhu.2002.BLEU：一种自动评估机器翻译的方法。In *Proc.of ACL*.

沈天啸、陶磊、Regina Barzilay和Tommi Jaakkola。2017.[Style transfer from non-parallel text by cross-alignment](http://papers.nips.cc/paper/7259-style-transfer-from-non-parallel-text-by-cross-alignment.pdf).In *Advances in Neural Informa- tion Processing Systems 30*, pages 6830-6841.

王华正，甘哲，刘晓东，刘晶晶，高剑锋，王红宁。2019.[Adversar- ial domain adaptation for machine reading compre- hension](https://doi.org/10.18653/v1/D19-1254).In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natu- ral Language Processing（EMNLP-IJCNLP）*.

Zhengli Zhao，Dheeru Dua，and Sameer Singh.2018.Generating natural adversarial examples.In *ICLR*.

周翔，聂一心，谭浩，和Mohit Bansal.2020.The curse of performance instability in analy- sis datasets:Consequences, source, and suggestions.*arXiv preprint arXiv:2004.13606*.

Chen Zhu，Yu Cheng，Zhe Gan，Siqi Sun，Thomas Goldstein，and Jingjing Liu.2020.Freelb。En- hanced adversarial training for language understand- ing.In *ICLR*.