Automatic Detection of Generated Text is Easiest when Humans are Fooled

Daphne Ippolito†‡*
daphnei@seas.upenn.edu

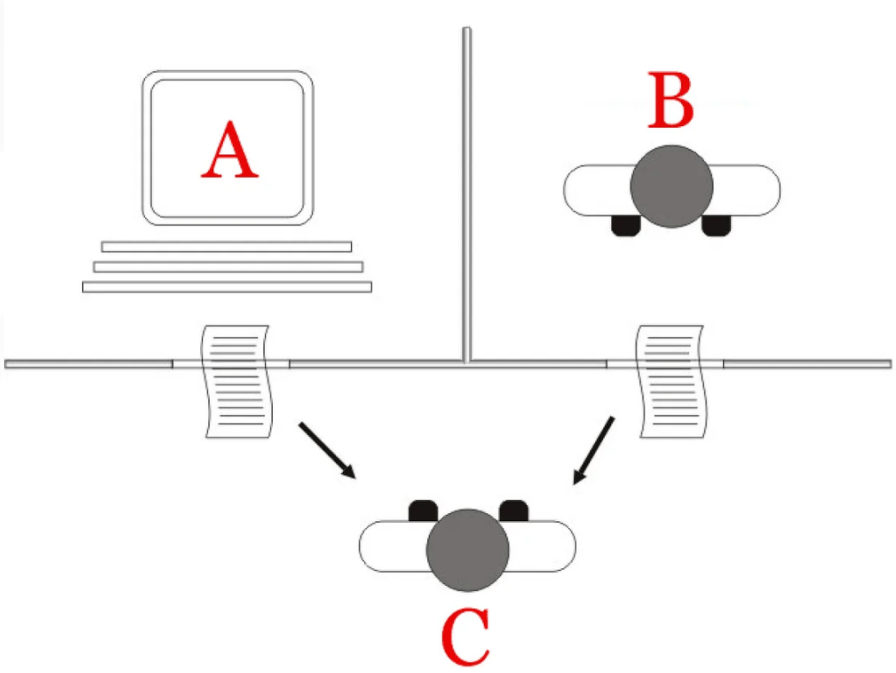
Daniel Duckworth‡*
duckworthd@google.com

Chris Callison-Burch†‡
ccb@seas.upenn.edu

Douglas Eck‡
deck@google.com

目录 | CONTENT

生成文本检测（Detection of Generated Text)



起源于图灵测试，及给定一段文本，需要判断出这段文本是由机器生成的，还是由人类书写的。

本文贡献

- 研究模型结构、解码策略、以及文本长度对自动文本检测系统的影响。
- 分析人类对这类问题判断的能力，并且人类判断和使用模型来判断有什么不同。

模型结构

当一个句子出现反常、错误或者话题漂移的表达时往往则可以判定这句话是有机器生成的，而这类的错误如果判别模型仅仅依靠语言的概率分布是不够的，同时还需要明确的因果事实，对句子的理解，类似于人类的这方面的能力。

文中认为BERT在需要语言理解方面的任务往往有不错的效果，因此认为BERT在这方面的任务中，也会有着不错的效果，因此本文中也使用了BERT和一些仅仅使用统计知识的方法进行对比

解码策略

文中认为文本生成过程中的解码策略会对检测模型和人类判断有着不同方向的影响。

由于人类对一个句子的判断，主要依赖于这个句子的中是否存在一些不正常的词语错误或者语法错误。而检测模型主要依赖于词语的概率分布信息来检测。

因此本文提出，类似于top-k这样的解码策略，让句子更加符合事实，因此导致人类更难判断，但同样也导致生成的句子中更加倾向于一些高频词，所以更容易被检测模型找到

目录 | CONTENT

对比内容

- 人类判断与机器判断
- 不同的解码策略
- 不同的句子长度
- 不同的评测模型

数据集设计

使用训练GPT2模型的训练的语料集以及GPT2生成的语句来进行数据集的构造。

使用训练数据集当作人类书写的数据集，然后使用不同的解码策略结合GPT2生成模型，生成出不同的语料集，然后用这些语料集构成训练数据以及测试数据，来训练检测模型。

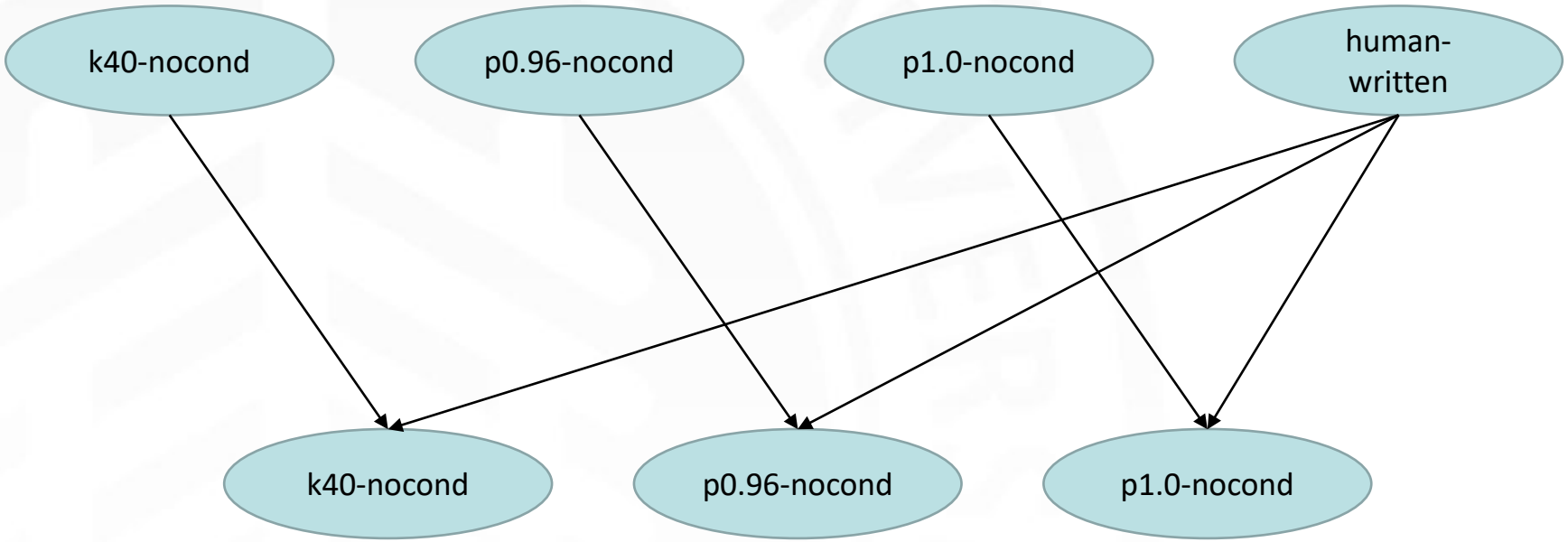
数据集设计

Top-k：每次得到词语的概率分布，只从概率最高的k个词中选取。

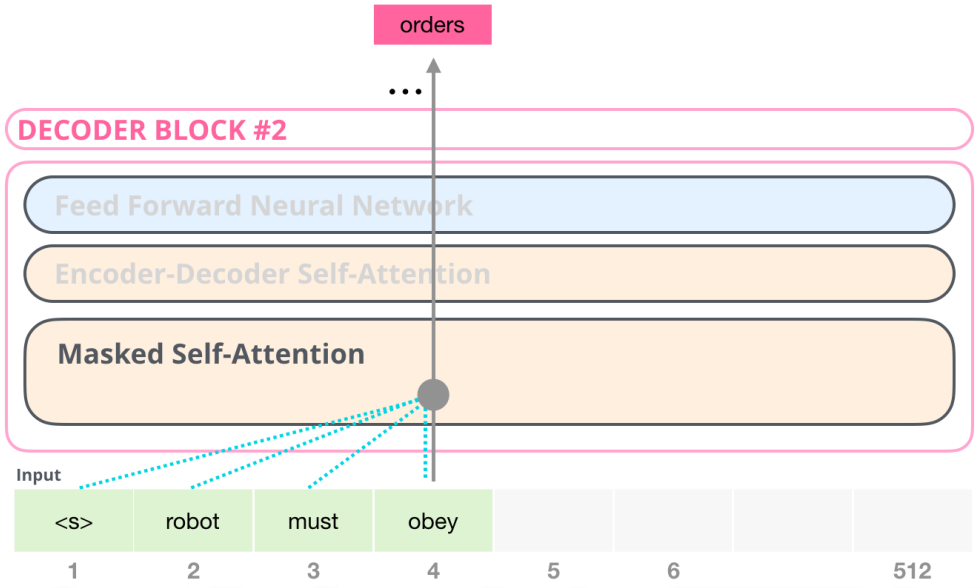
Nucleus sampling：得到词语的概率分布，从前t个词中抽取，要求前t个词概率总和达到预先设定的p值。

随机抽取：得到词语的概率分布，使用随机抽取的方法进行选择。

数据集设计

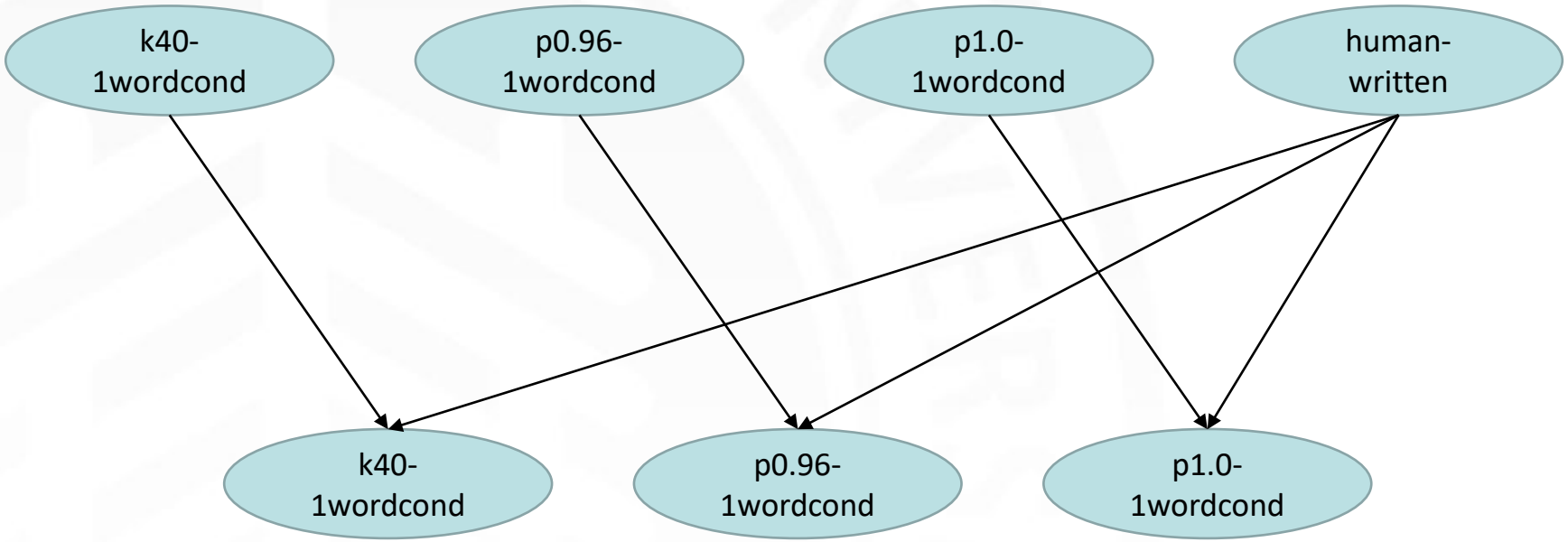

k40-nocond
p0.96-nocond
p1.0-nocond
human-written
k40-nocond
p0.96-nocond
p1.0-nocond

数据集设计



生成的时候是否给模型一个启动序列（priming sequence)因为如果不给定，并且搭配上解码策略，例如top-k，那所有句子的第一个词选择范围就更小了，会导致更加易于辨认

数据集设计


k40-1wordcond
p0.96-1wordcond
p1.0-1wordcond
human-written
k40-1wordcond
p0.96-1wordcond
p1.0-1wordcond

数据集设计

6个数据集按照从长度为2到192直接去10个值进行划分，又分为10个数据集，每个数据集中句子长度不超过给定的长度。最后则共60个数据集进行实验

检测模型设计

- 使用BERT进行微调进行判断
- 词袋模型，然后使用回归模型进行判断。
- Histogram-of-Likelihood Ranks 统计排名在某个区间的次数，按照次数制成向量，然后用逻辑回归二分类
- 计算一句话的概率，接近所有生成语句的平均概率则是生成的，接近人写的概率，则是人写的

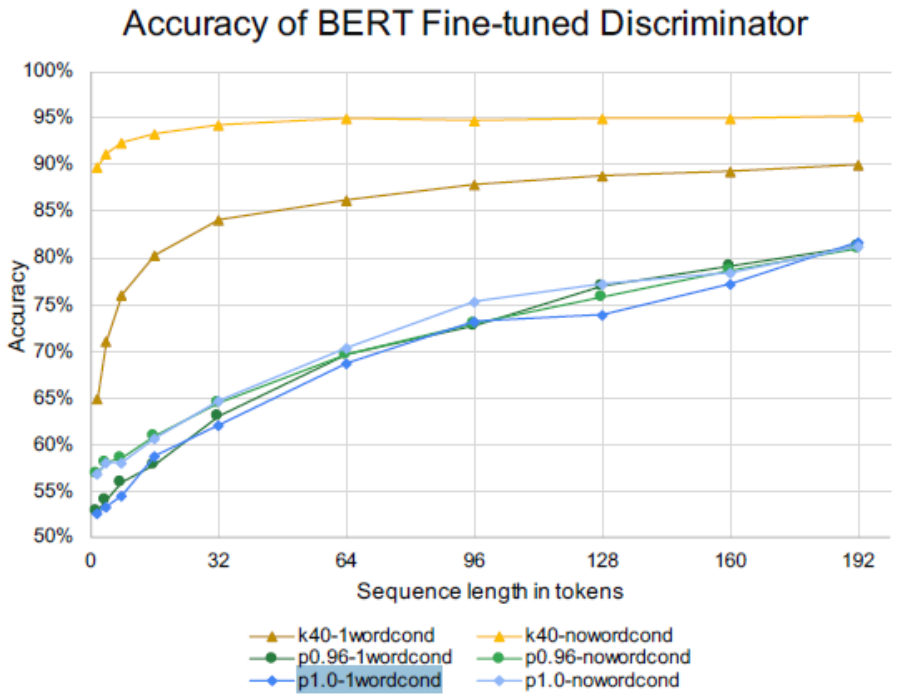
目录 | CONTENT

检测模型结果

Method	BERT		BagOfWords		HistGLTRBuckets		Hist50Buckets		TotalProb	Human
	acc	AUC	acc	AUC	acc	AUC	acc	AUC	acc	acc
k40-1wordcond	0.88	0.99	0.79	0.87	0.52	0.52	0.69	0.76	0.61	0.64
p0.96-1wordcond	0.81	0.89	0.60	0.65	0.53	0.56	0.54	0.56	0.63	0.77
p1.0-1wordcond	0.79	0.92	0.59	0.62	0.53	0.55	0.54	0.55	0.65	0.71

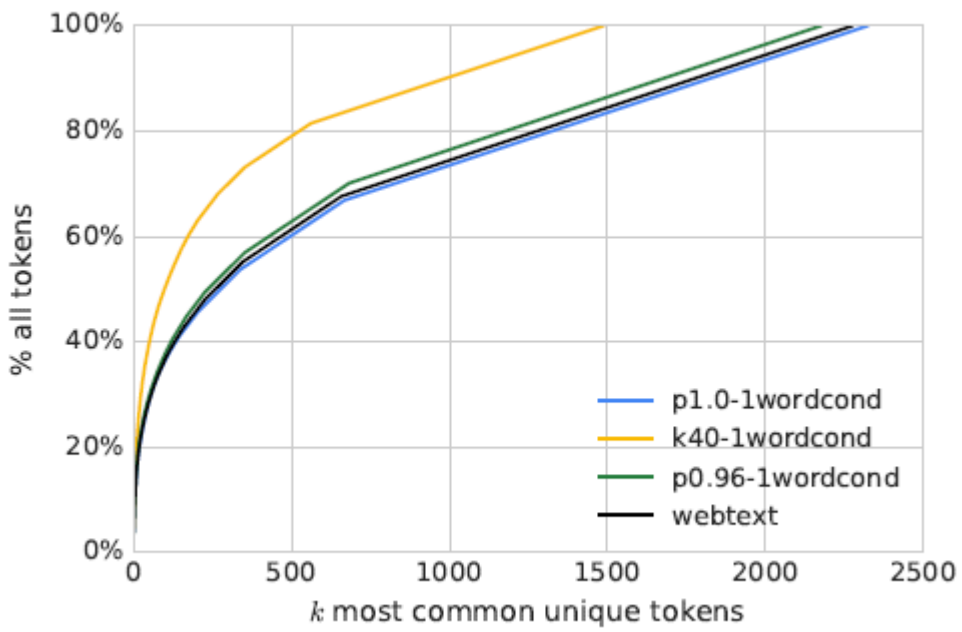
对192长度的数据集进行测试，Bert的效果最好，本文认为是BERT可以更好的理解文本。同时对其他的baseline进行分析，总体感觉是对文本统计信息越精细，可以得到的结果最好，例如Histogram-of-Likelihood Ranks的方法，对词语的统计比较粗粒度，因此效果有点差，而50个bucket的效果比4个的bucket的效果又好一些

检测模型结果



Bert对不同长度，不同解码方式生成出的文本进行判断，得到不同的结果

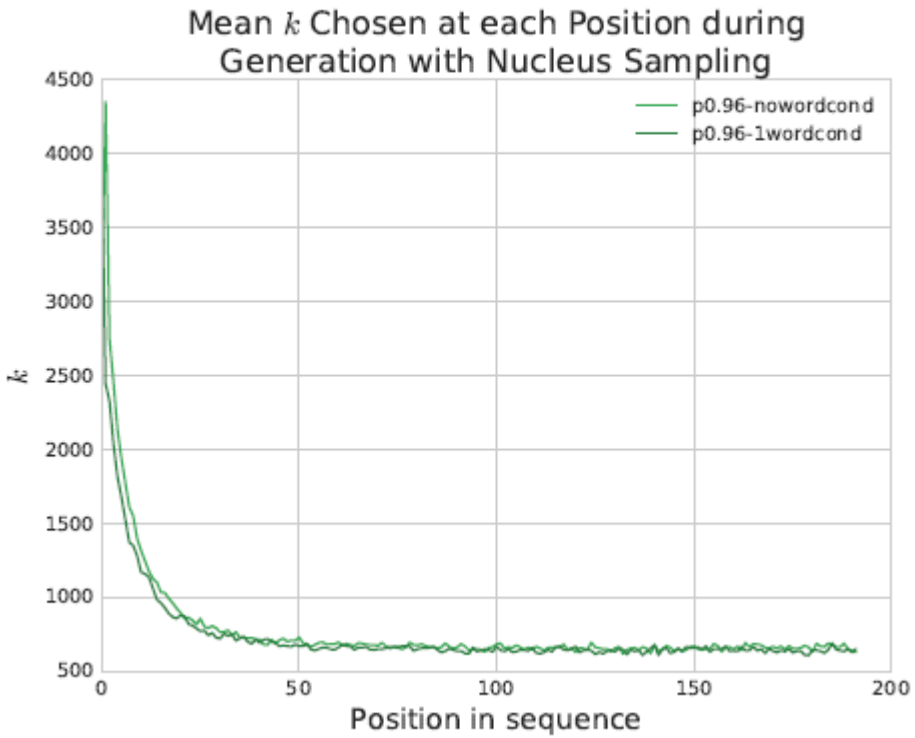
检测模型结果



按照频次统计最常见的k个单词占总体文本的比例。可以明显的看出top-k生成的文本用词明显较少，且和其他集中相差较远。

论文中也提到一篇其他的论文中论述的需要k达到1000才和人类真实分布比较相似

检测模型结果



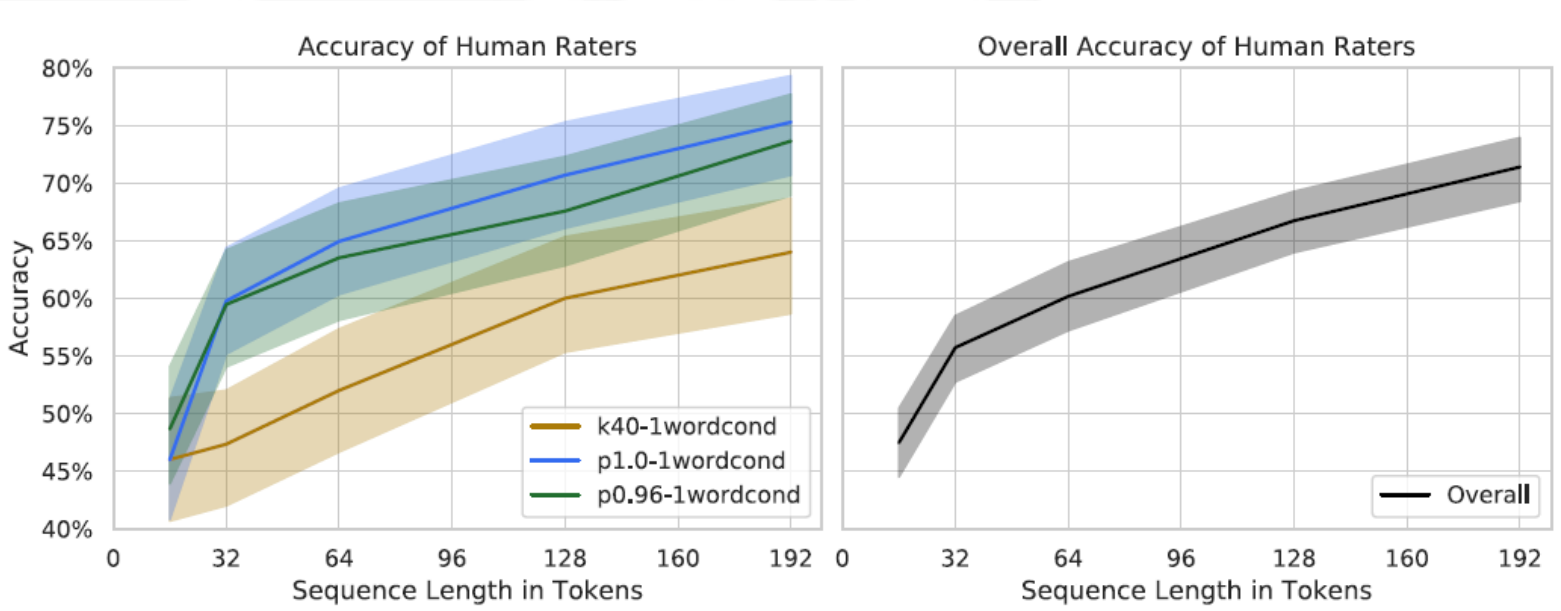
其他两种方法对比，nucleus sampling 第一个词有3075的选择，后面的则有500个，随机选择则是整个此表，而top-k只有k个

迁移性能

		Eval		
		top-k	nucleus	random
Train	top-k	**90.1**	57.1	43.8
	nucleus	79.1	**81.3**	78.4
	random	47.8	63.7	**81.7**
	mixed	88.7	74.2	72.2

不同数据集上训练的检测模型在不同数据上进行测试发现效果有很大的差别

人类判断


Accuracy of Human Raters
80%
75%
70%
65%
60%
55%
50%
45%
40%
Accuracy
0
32
64
96
128
160
192
Sequence Length in Tokens
k40-1wordcond
p1.0-1wordcond
p0.96-1wordcond
Overall Accuracy of Human Raters
0
32
64
96
128
160
192
Sequence Length in Tokens
Overall

目录 | CONTENT

人类判断

Identifying ways to improve the language models and decoding strategies we use in order to generate text that is both exciting (ie. unlikely) and semantically plausible.

Building better world understanding into automatic discriminators so that they are more capable of detecting the types of errors that humans notice.