◁ BIT ▷

多语种语言模型与处理

答辩人：权双庆、杨麒沄、谢文泽、张易初、李世杰

时间：2022-04-15

德以明理 学以精工

北京理工大學

BEIJING INSTITUTE OF TECHNOLOGY

北京理工大学 BEIJING INSTITUTE OF TECHNOLOGY

目 录

CONTENTS

PART ONE

经典综述

语言模型

语言模型（LM）是很多自然语言处理（NLP）任务的基础。
对于任意的词序列，它能够计算出这个序列是一句话的概率。

假设我们要为中文创建一个语言模型,V 表示词典,
V {今天,你,吃,学习,语言,模型,了,吗...} $w_i \in V$

语言模型就是对：给定词典 V ，能够计算出任意单词序列 $w_1, w_2, \ldots, w_n$ ，是一句话的概率 $p(w_1, w_2, \ldots, w_n)$ 其中， $p \geq 0$ 。

$$P(w_1, w_2, \ldots, w_n) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \cdots P(w_n|w_1, w_2, \cdots w_{n-1})$$

语言模型的应用

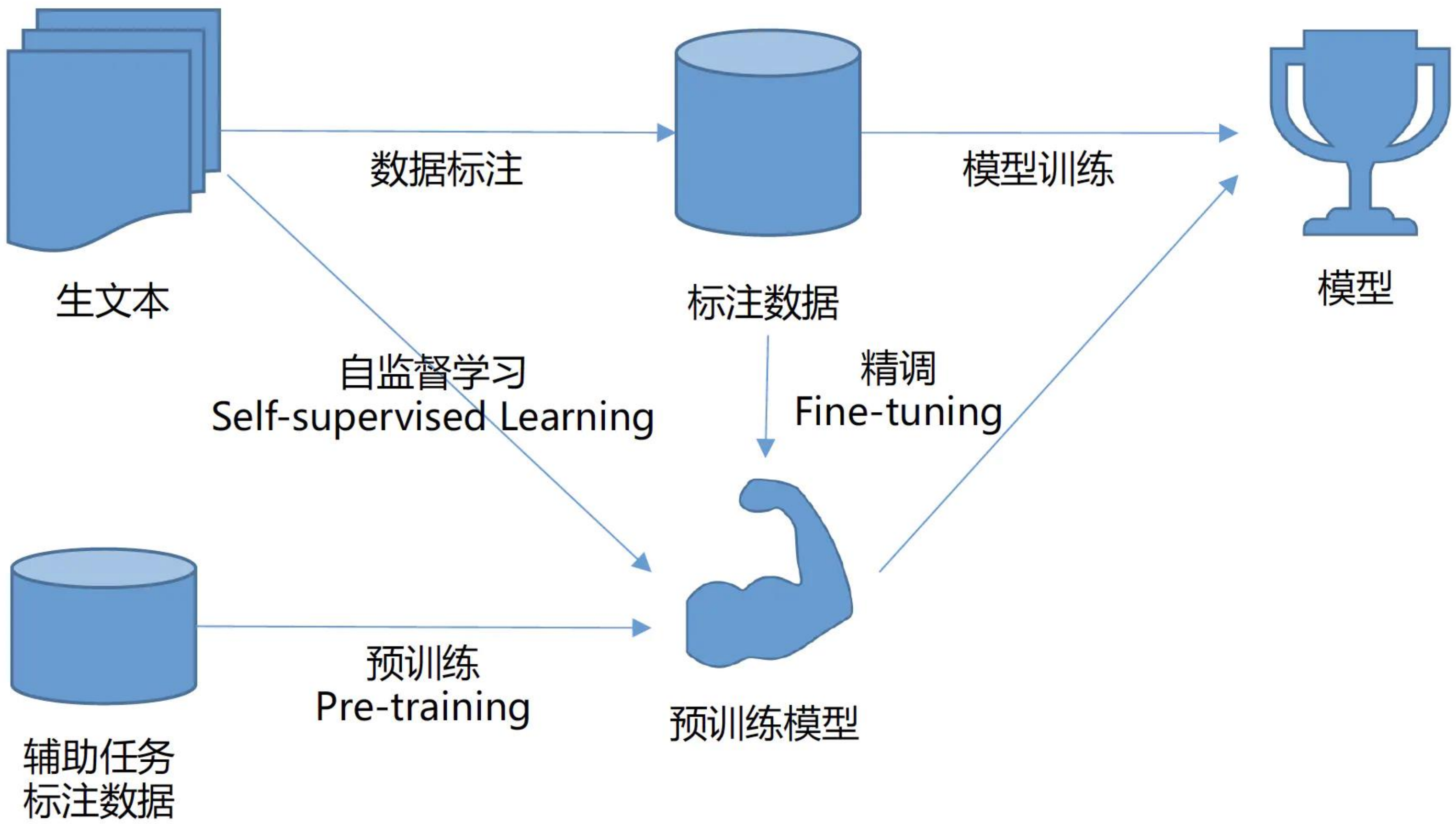
它的应用包括：

（1）做预训练模型

（2）生成文本，给定前面几个词，通过不断地计算生成后续文本

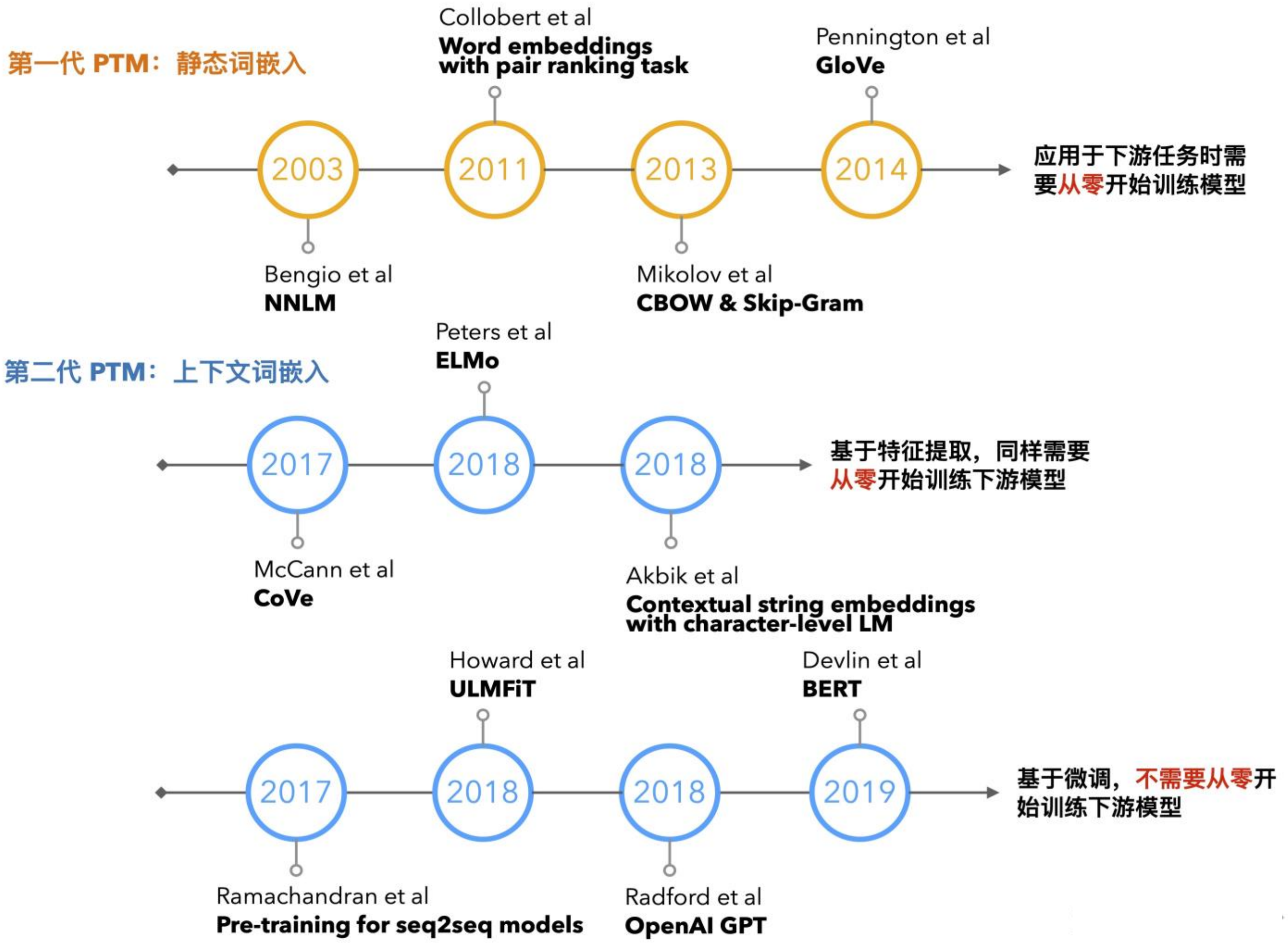
（3）判断多个序列中，哪几个序列更常见

预训练模型


数据标注
模型训练
生文本
标注数据
模型
自监督学习
Self-supervised Learning
精调
Fine-tuning
预训练
Pre-training
预训练模型
辅助任务
标注数据

预训练+精调

静态词向量预训练模型



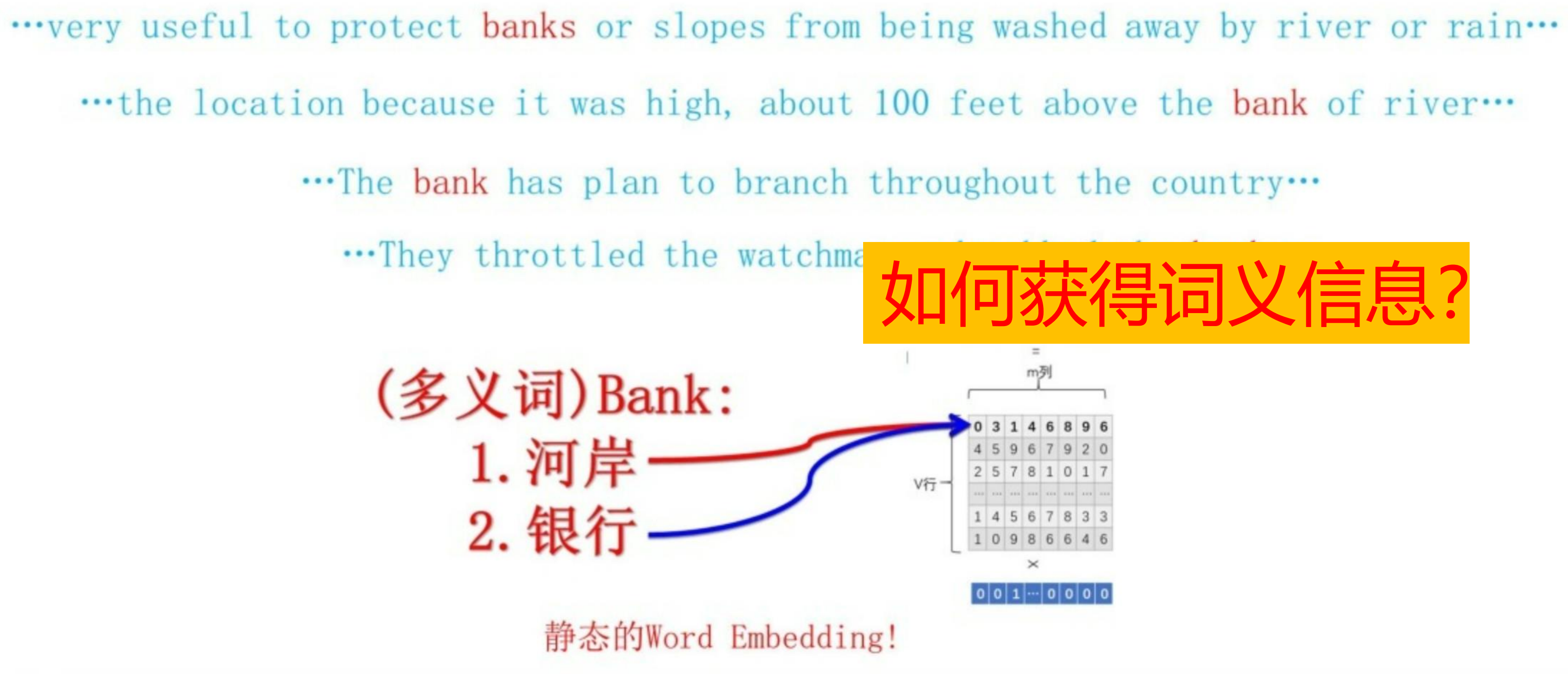
最初，是使用低维、稠密、连续的向量表示词。

（1）词嵌入（word embedding）

（2）通过下游任务直接学习词向量

一词多义问题

当时的静态词向量假设：一个词由唯一的词向量表示
无法处理一词多义的问题



如何获得词义信息?

我 喜欢 吃 土豆
我 刚刚 在 土豆 看视频

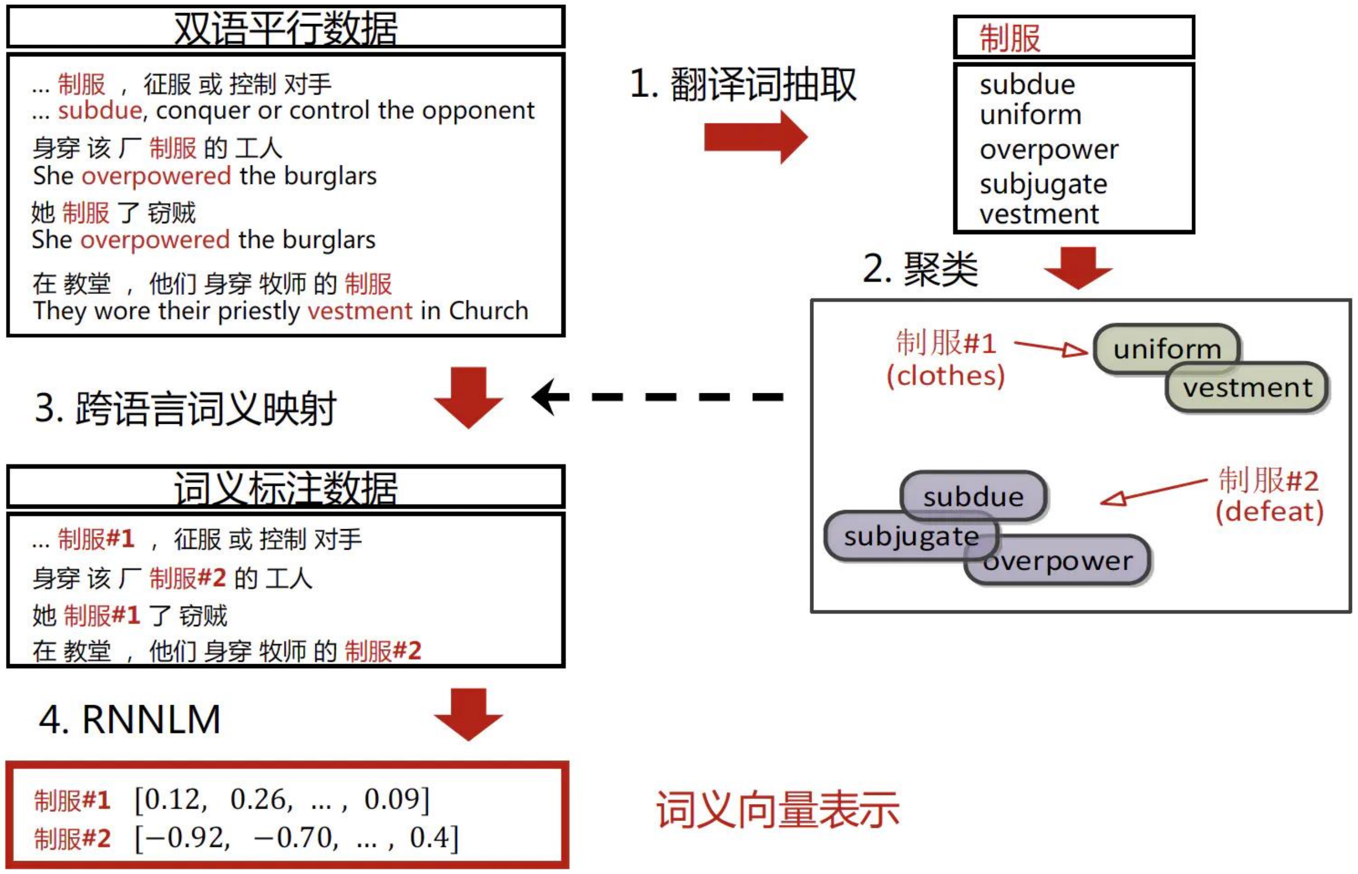
“上下文”信息混乱

我 喜欢 吃 土豆#1
我 刚刚 在 土豆#2 看视频

基于双语的词义向量表示

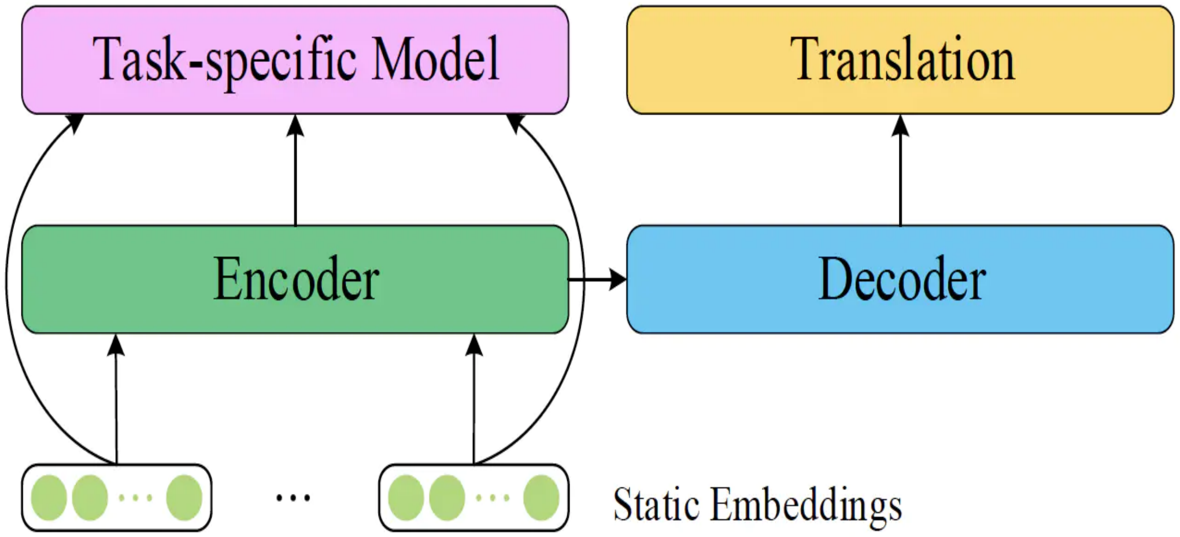
北京理工大学 BEIJING INSTITUTE OF TECHNOLOGY

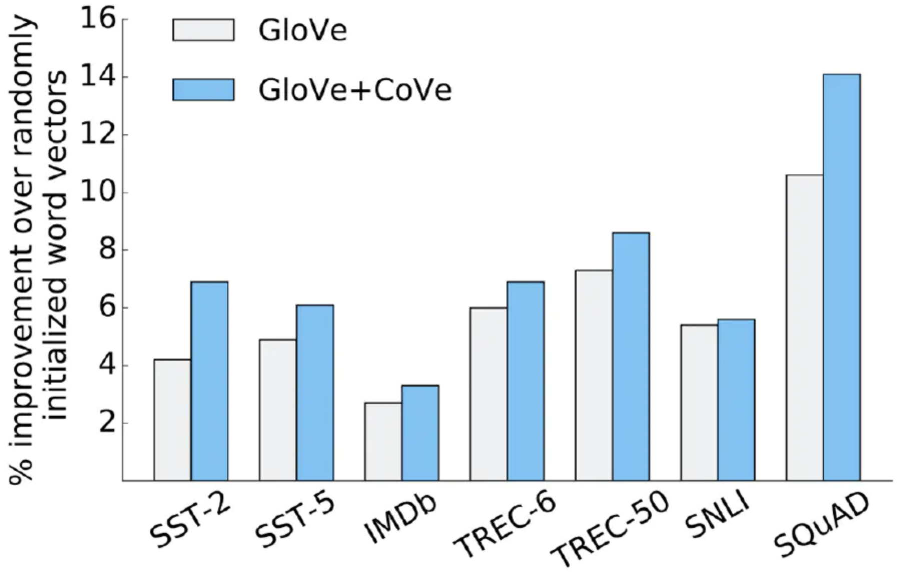
- Learning Sense-specific Word Embeddings By Exploiting Bilingual Resources (Guo et al., Coling 2014)



CoVe

Learned in Translation:Contextualized Word Vectors(McCann et al.,arXiv：1708.00107)
-CoVe：Context Vectors
  预训练NTM模型
  将Encoder作为目标任务的额外特征





ELMo

Deep Contextualized Word Representations(Peters et al.,NAACL 2018)
-ELMo:Embedding from Language Models

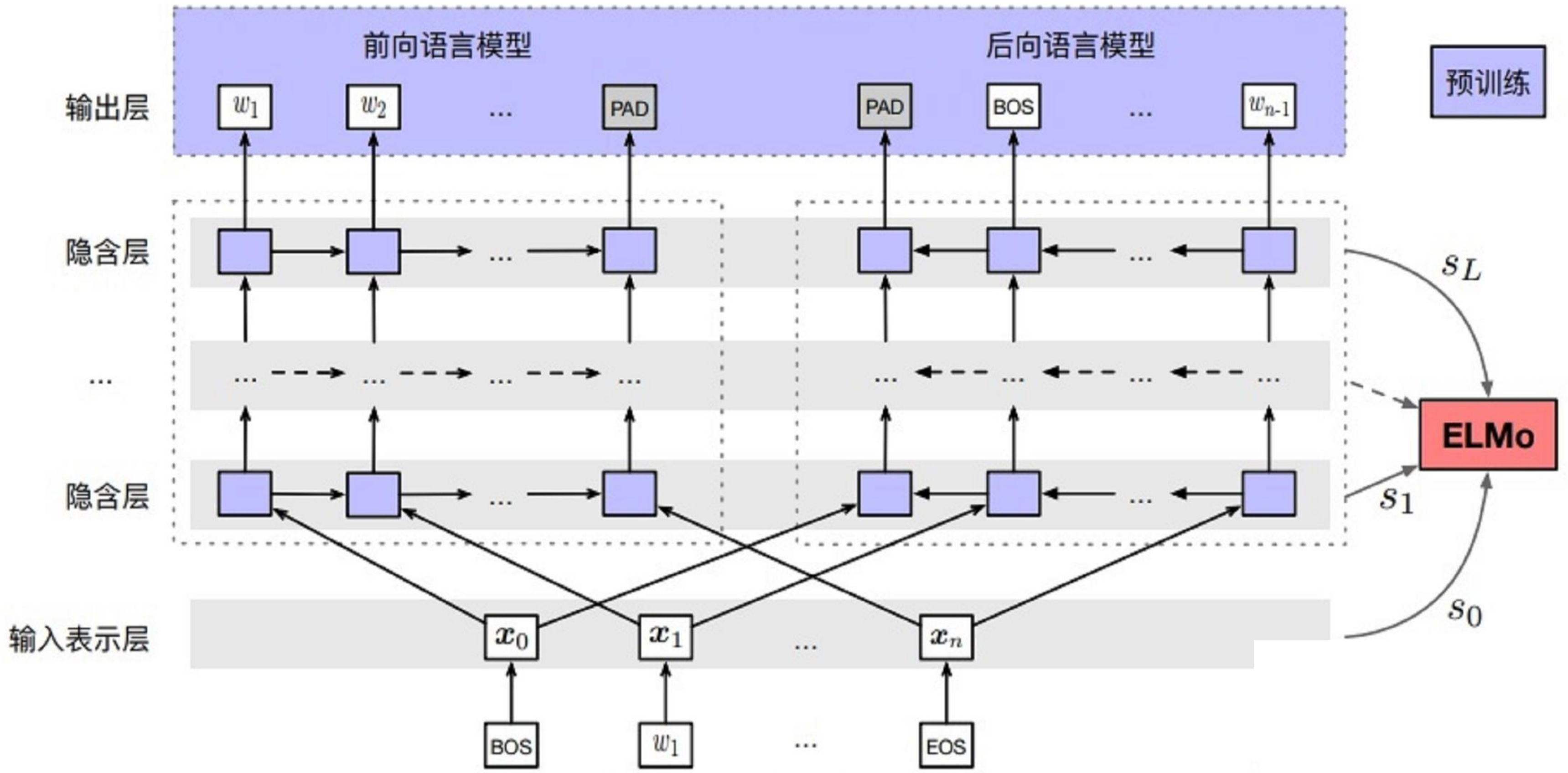


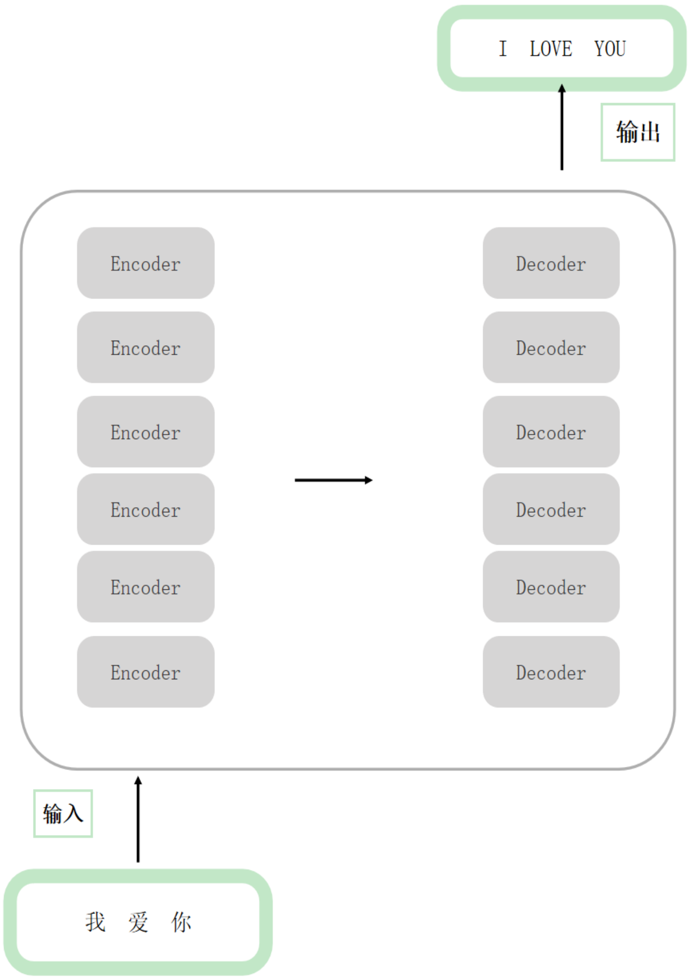
图 6-2 ELMo 模型示意图

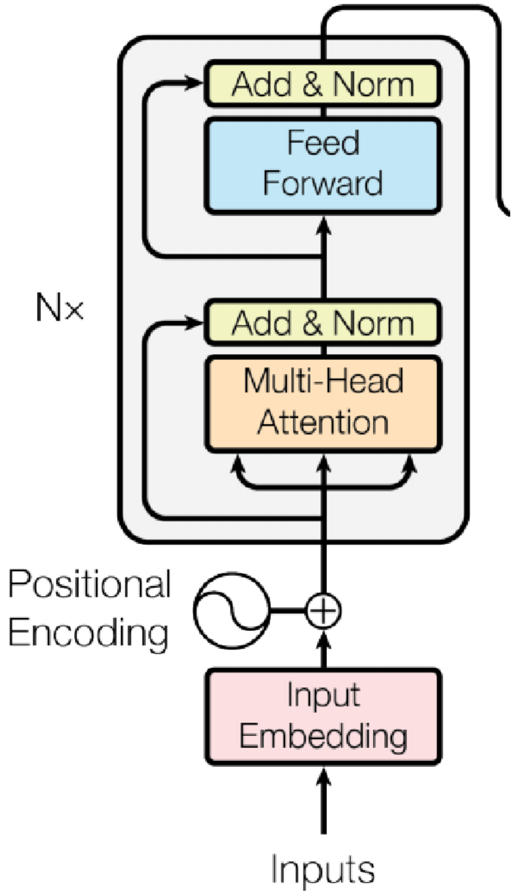
（1）分别训练从左至右、从右至左的语言模型
（2）不再依赖双语义数据
（3）单语训练数据量近乎无限

Bert-编码器

与ELMo相比：
同：根据句中每一个单词返回的词向量，并且词向量要包含自身与周围单词的信息，只不过可能关注在当前词上面。

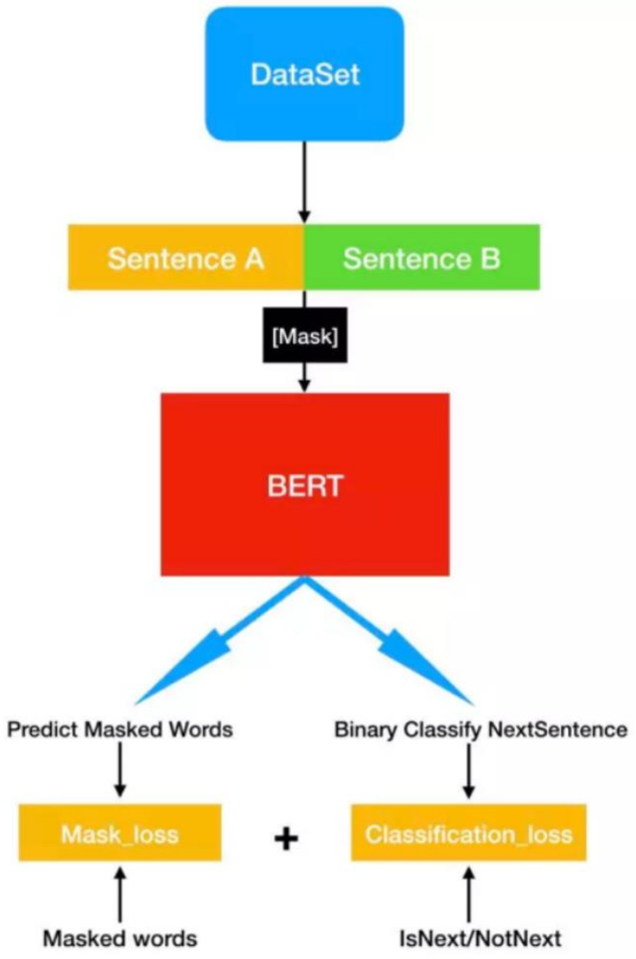
异：创新使用了双向编码器的方式，使用Transfomer模型代替了LSTM模型。



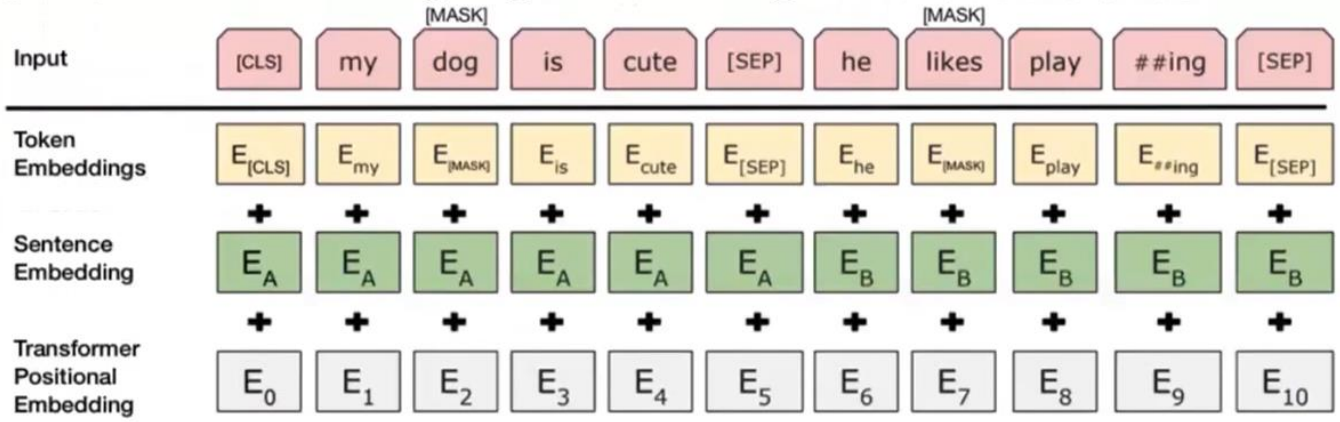


Bert-预训练任务

完形填空 + 下句预测 (NSP)



- 使用 Masked Language Modeling 作为预训练任务

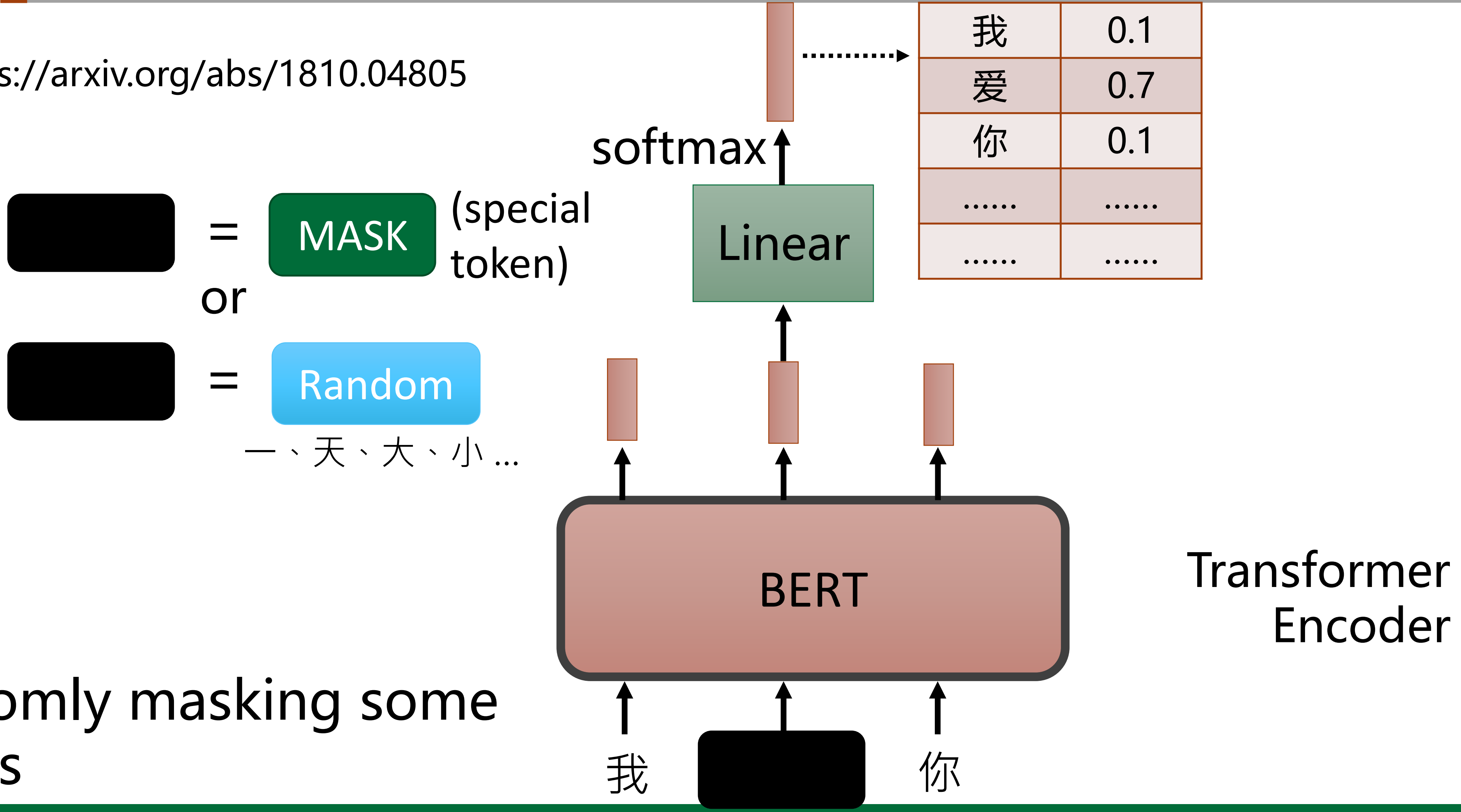


- 随机掩盖一些单词，用周围单词的信息预测这些被掩盖的单词。

模型会随机选择语料中15%的单词，然后其中的80%会用[Mask]掩码代替原始单词，其中的10%会被随机换为另一个单词，剩下10%保持原单词不变，然后要求模型去正确预测被选中的单词。

Bert-预训练任务（完形填空）

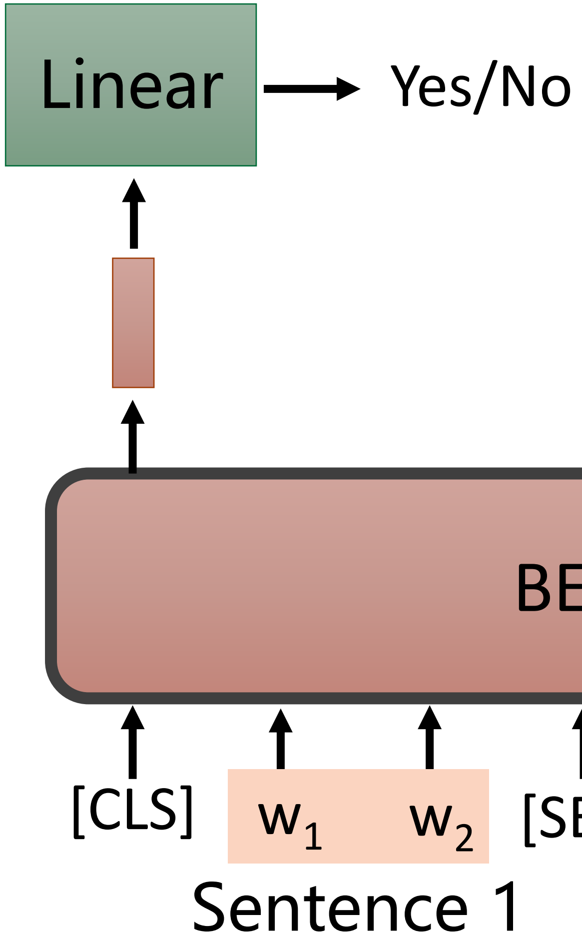
https://arxiv.org/abs/1810.04805



我	0.1
爱	0.7
你	0.1
……	……
……	……

Randomly masking some tokens

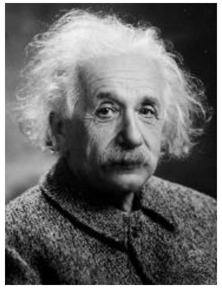
Bert-预训练任务（下一句预测）



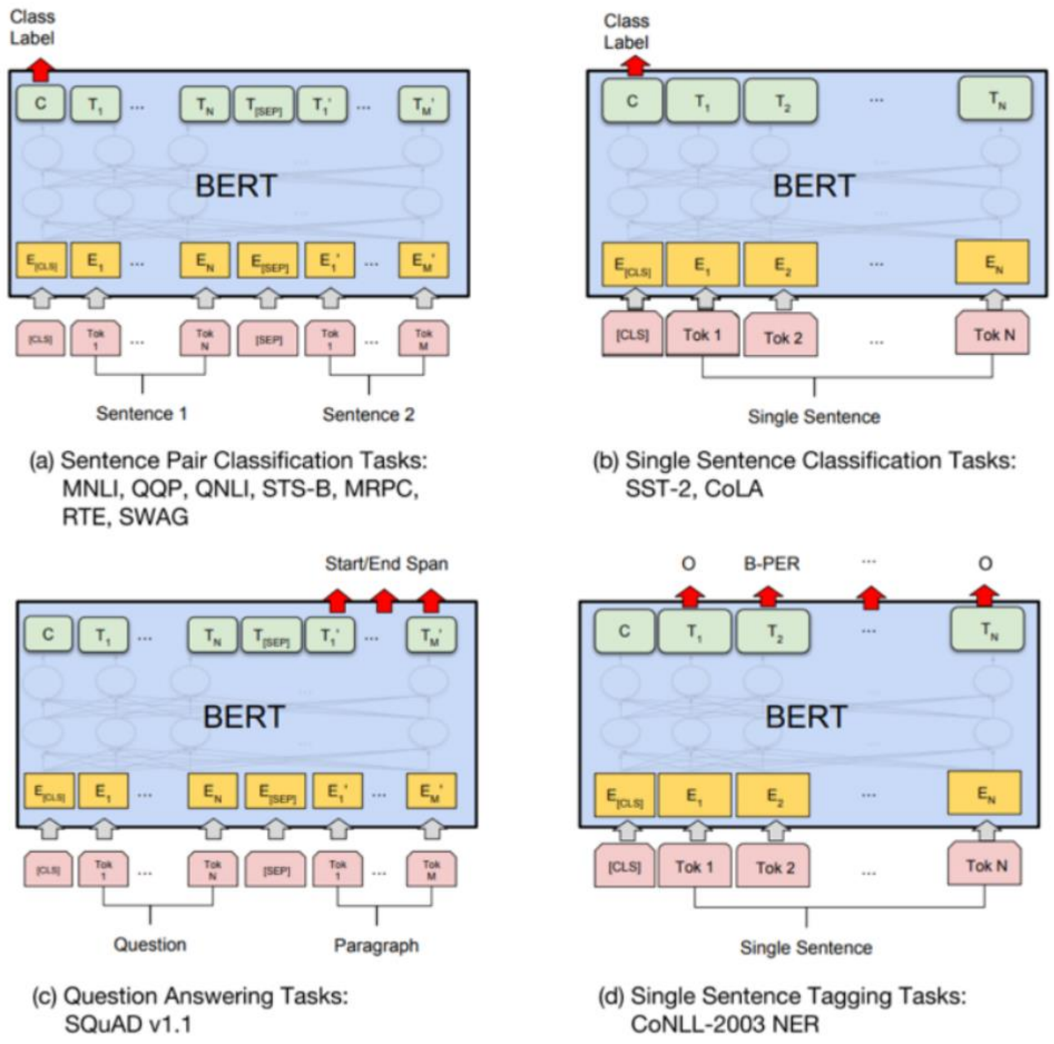
Robustly optimized BERT approach (RoBERTa)

- **SOP**: Sentence order prediction

  Used in ALBERT
  https://arxiv.org/abs/1909.11942



Bert-应用方式



在目标任务上微调
四种任务类型

对于特定下游任务的模型，可以通过将BERT与一个额外的输出层结合而形成，这样需要重新学习的参数量就会比较小，如图所示，只需要按BERT模型要求的格式输入训练数据，就可以完成不同的下游任务。

多语种语言模型

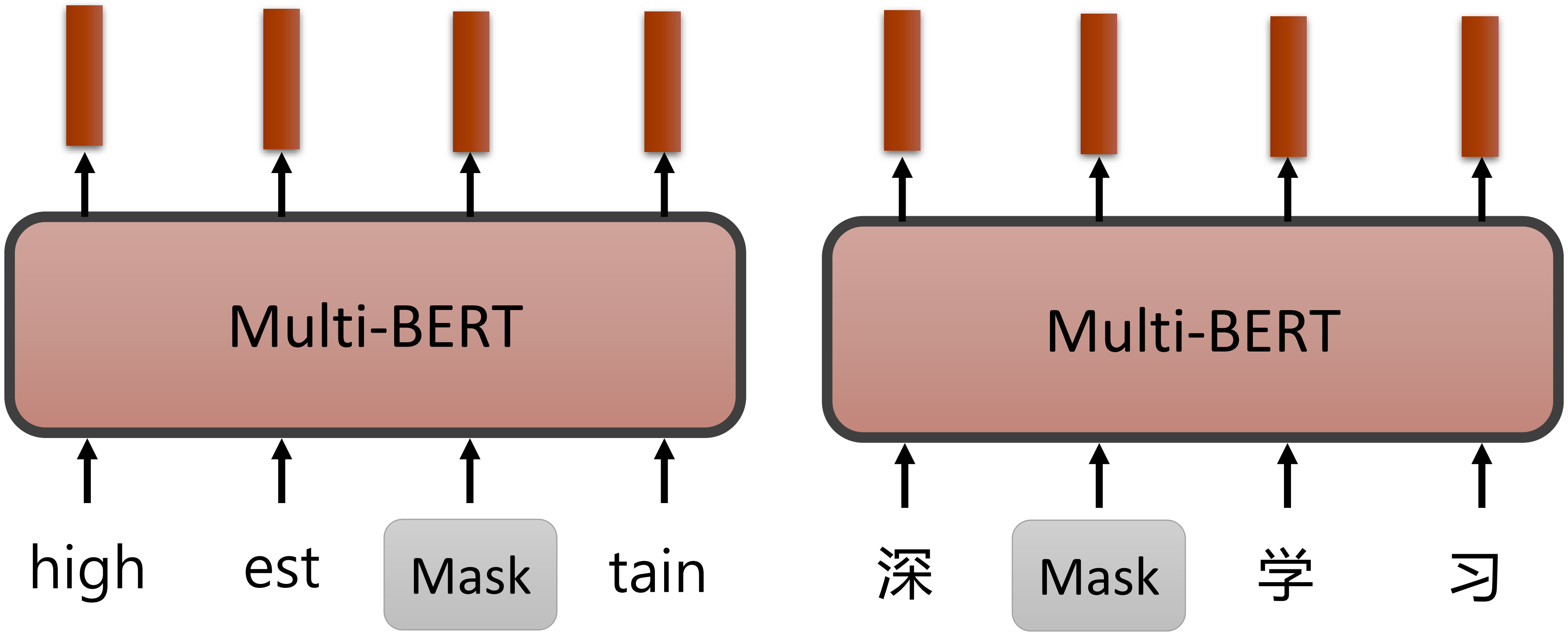
在著名的科幻电影《银河系漫游指南》里有一种叫巴别鱼的神奇生物。将它塞进耳朵里你就能听懂任何语言。多语种语言模型做得事情和巴别鱼很像，人们希望这个模型能用来处理所有的语言。举个例子，大家常用的中文bert有很强的中文处理能力以及一定的英文处理能力，但基本也就只能处理这两种语言；而目前的SOTA多语种模型XLM-RoBERTa能够处理104种语言。

巴别鱼，体型很小，黄色，外形像水蛭，很可能是宇宙中最奇异的事物。它靠接收脑电波的能量为生，并且不是从其携带者身上接收，而是从周围的人身上。它从这些脑电波能量中吸收所有未被人察觉的精神频率，转化成营养。然后它向携带者的思想中排泄一种由被察觉到的精神频率和大脑语言中枢提供的神经信号混合而成的心灵感应矩阵。所有这些过程的实际效果就是，如果你把一条巴别鱼塞进耳朵，你就能立刻理解以任何形式的语言对你说的任何事情。

统一的多语种语言预训练模型

实现“跨语言”的能力

Multi-lingual BERT

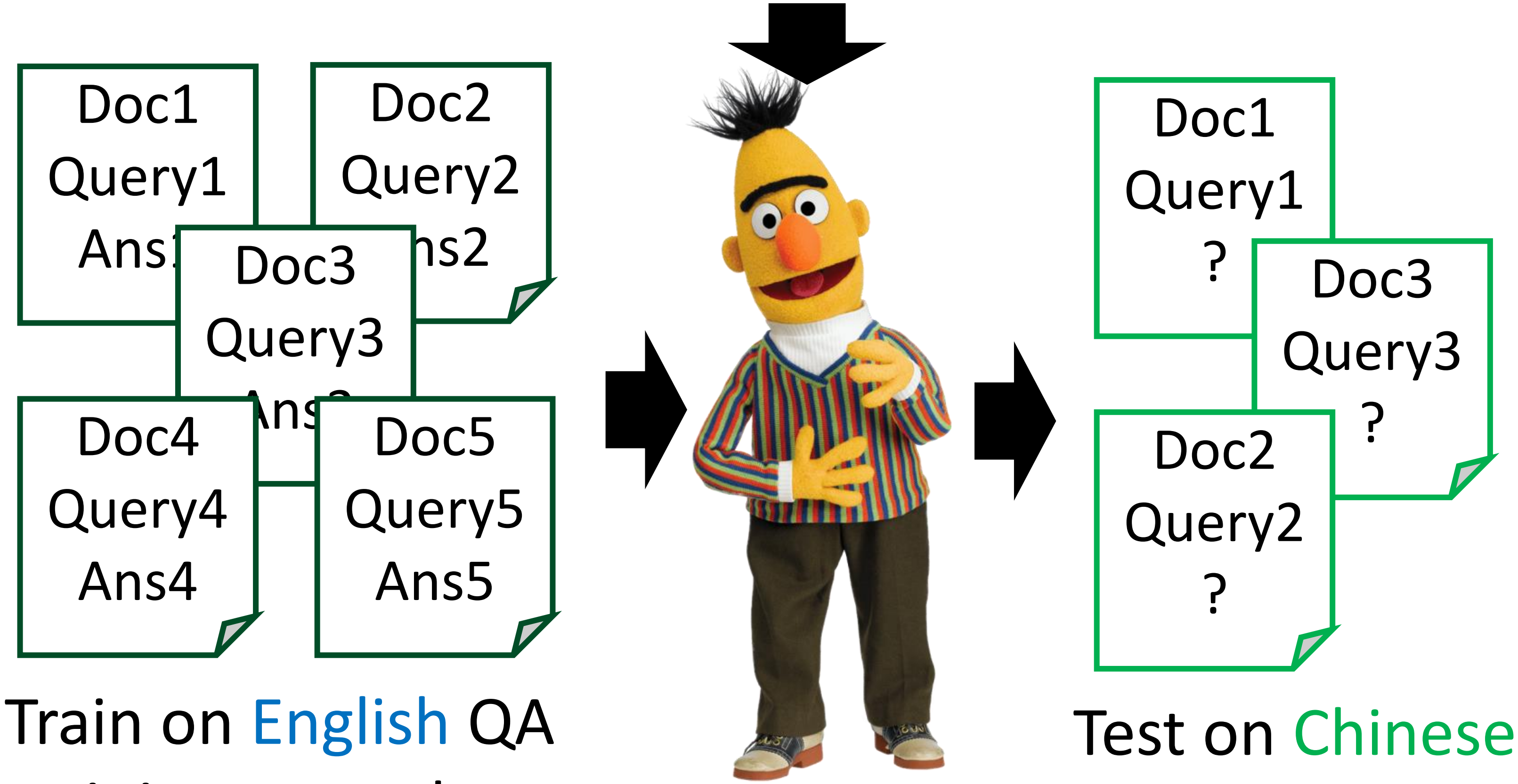


PART TWO

前沿进展

Multi-lingual BERT

Training on the sentences of 104 languages



Train on English QA training examples

Test on Chinese QA test

QA模型对比

English: SQuAD, Chinese: DRCD

Model	Pre-train	Fine-tune	Test	EM	F1
QANet	none	Chinese	Chinese	66.1	78.1
BERT	Chinese	Chinese		82.0	89.1
	104 languages	Chinese		81.2	88.7
		English		63.3	78.8
		Chinese + English		82.6	90.1

F1 score of Human performance is 93.30%

Multi-BERT			
Train / Test	English	Chinese	Korean
English	81.2/88.6	63.3/78.8	49.2/69.3
Chinese	34.1/53.8	81.2/88.7	56.4/78.2
Korean	58.5/68.4	73.4/82.7	69.4/89.3

Multi-BERT			
Train / Test	English	Chinese	Korean
Zh	34.1/53.8	81.2/88.7	56.4/78.2
Zh-En	26.6/44.1	57.7/71.1	40.5/59.5
Zh-Fr	23.4/39.8	44.9/62.0	39.6/59.9
Zh-Jp	25.5/42.6	60.9/72.4	44.9/65.7
Zh-Kr	26.5/42.2	58.2/69.5	47.4/67.7

[Hsu, Liu, et al., EMNLP'19]

为什么可以训练在英文上却可以答在中文上?

原因

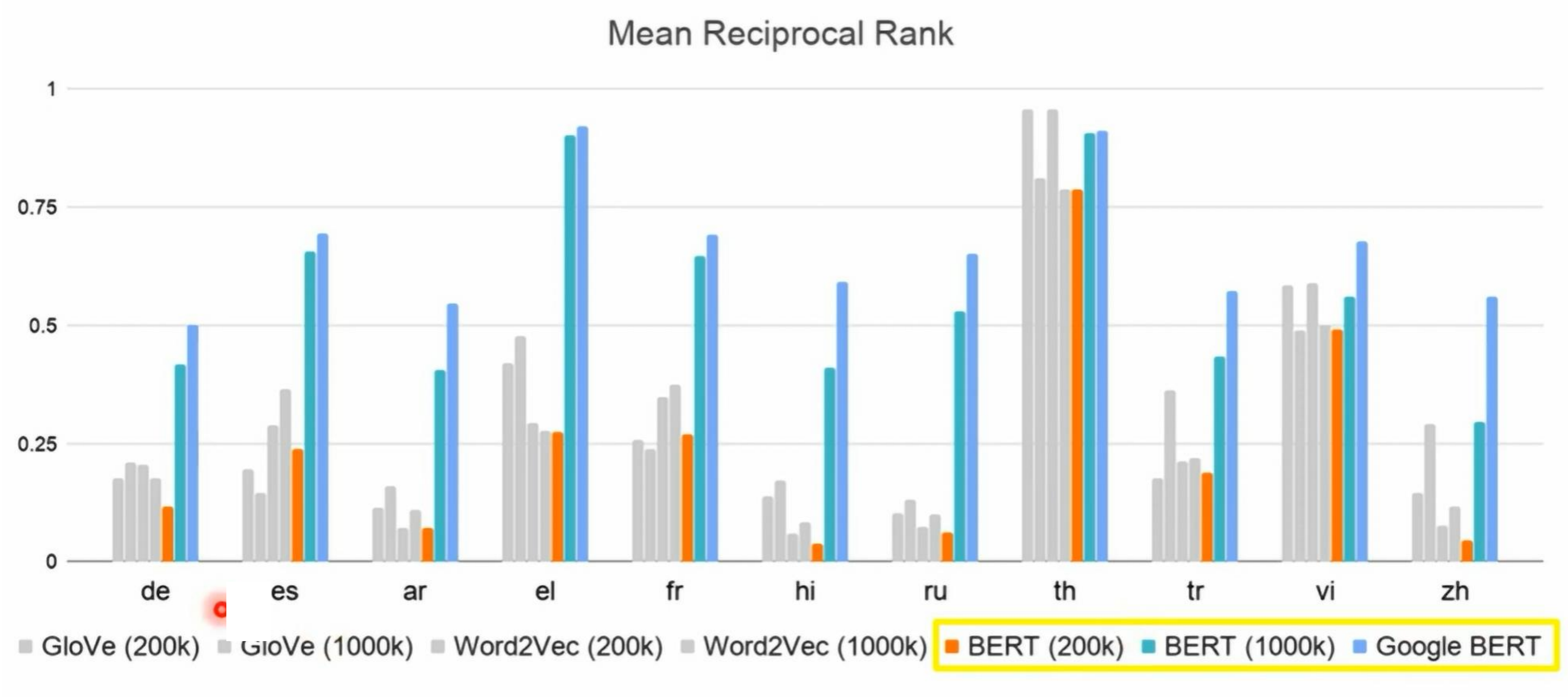
对于mbert来说 不同种语言看起来是一样的
年 years 月 month 的意思是很接近的

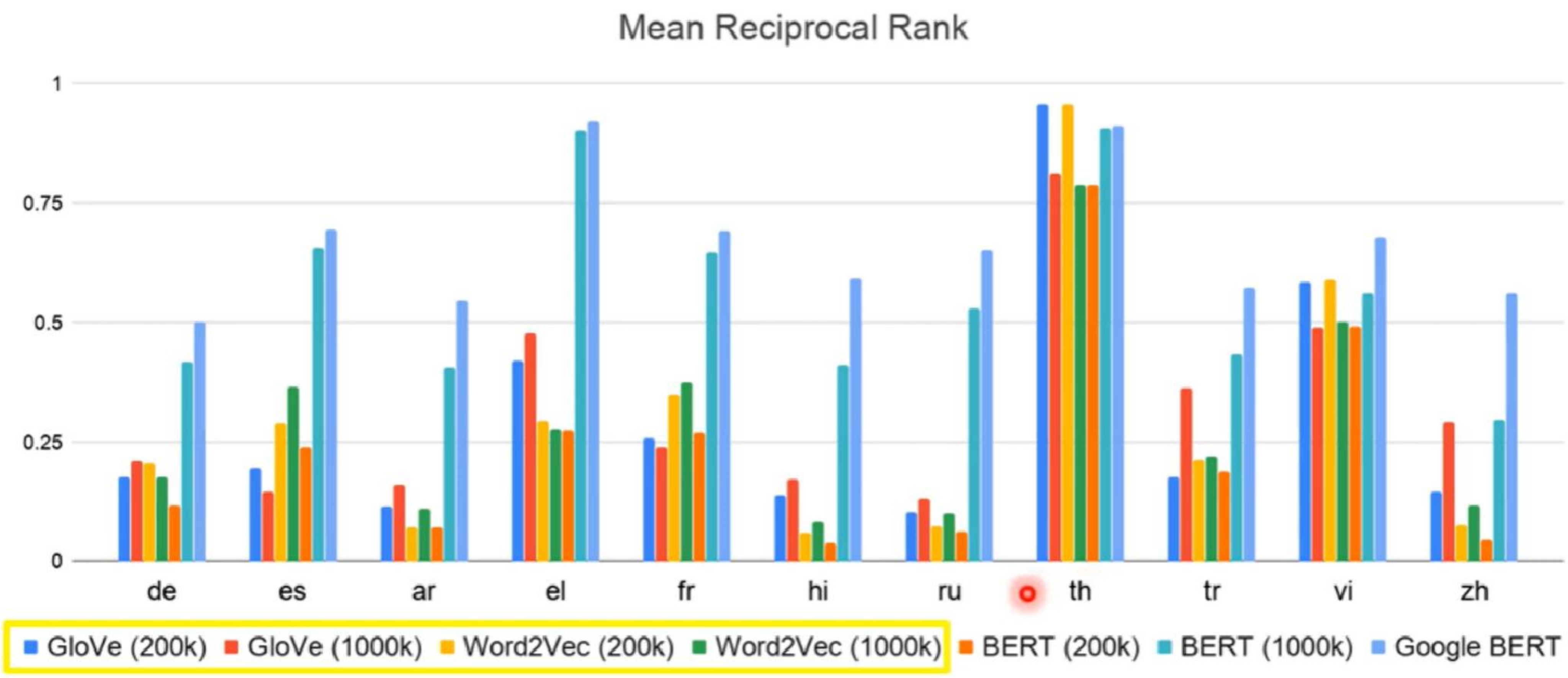
在Multi-Bert里它将不同语言的资讯去掉，只保留语义的咨询
所以只交给mbert某一种语言的任务 就可以完成另一种语言的任务

	年	月	和	村	人	大	他	....	Rank	Score
year	0.7	0.6	0.2	0.1	0.5	0.3	0.4		1	1/1
month	0.5	0.6	0.7	0.8	0.1	0.2	0.3		3	1/3
and	0.1	0.3	0.6	0.5	0.7	0.2	0.4		2	1/2
village	0.5	0.8	0.7	0.6	0.1	0.3	0.1		3	1/3
man	0.1	0.7	0.8	0.6	0.4	0.2	0.3		4	1/4
big	0.3	0.1	0.5	0.8	0.7	0.9	0.2		1	1/1
he	0.5	0.8	0.3	0.6	0.9	0.4	0.7		3	1/3
⋮										

Bi-lingual Dictionary

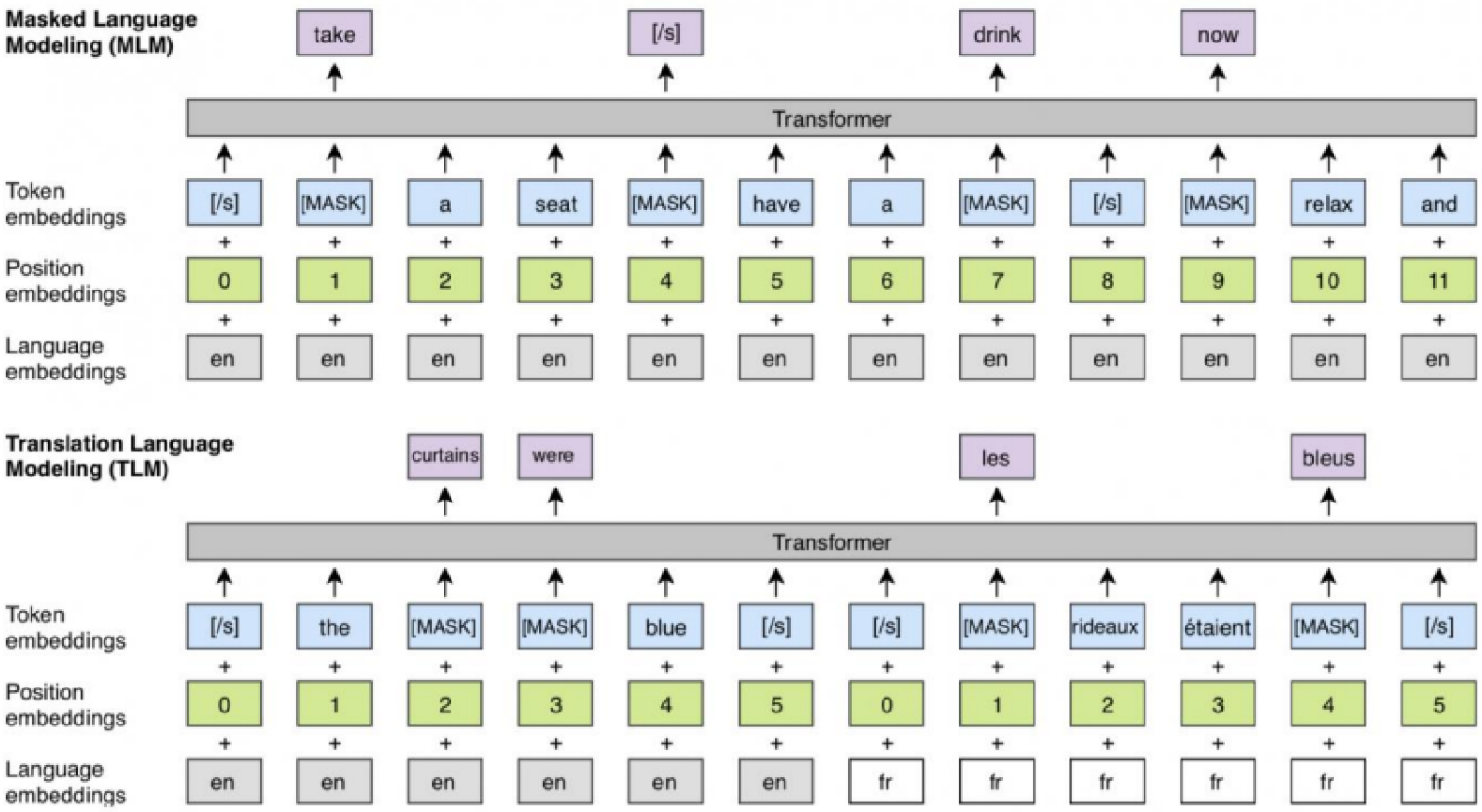
year → 年
month → 月
village → 村
big → 大


Mean Reciprocal Rank
1
0.75
0.5
0.25
0
de
es
ar
el
fr
hi
ru
th
tr
vi
zh
GloVe (200k)
GloVe (1000k)
Word2Vec (200k)
Word2Vec (1000k)
BERT (200k)
BERT (1000k)
Google BERT


Mean Reciprocal Rank
1
0.75
0.5
0.25
0
de
es
ar
el
fr
hi
ru
th
tr
vi
zh
GloVe (200k)
GloVe (1000k)
Word2Vec (200k)
Word2Vec (1000k)
BERT (200k)
BERT (1000k)
Google BERT

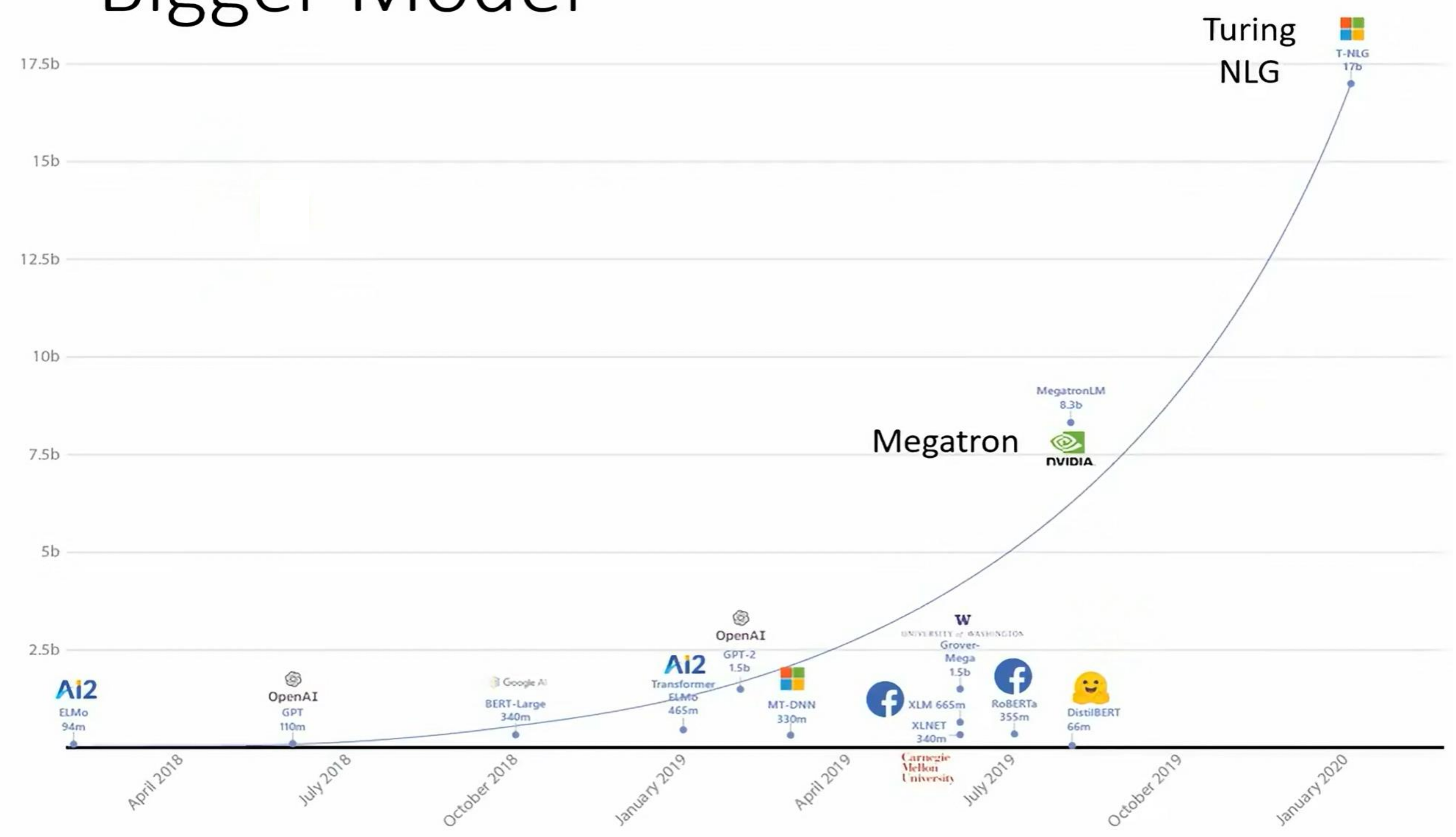
Mbert与其他几种模型的对比

- Cross-Lingual Language Understanding (XLU)



XLM – Enhancing BERT for Cross-lingual Language Model

Bigger Model

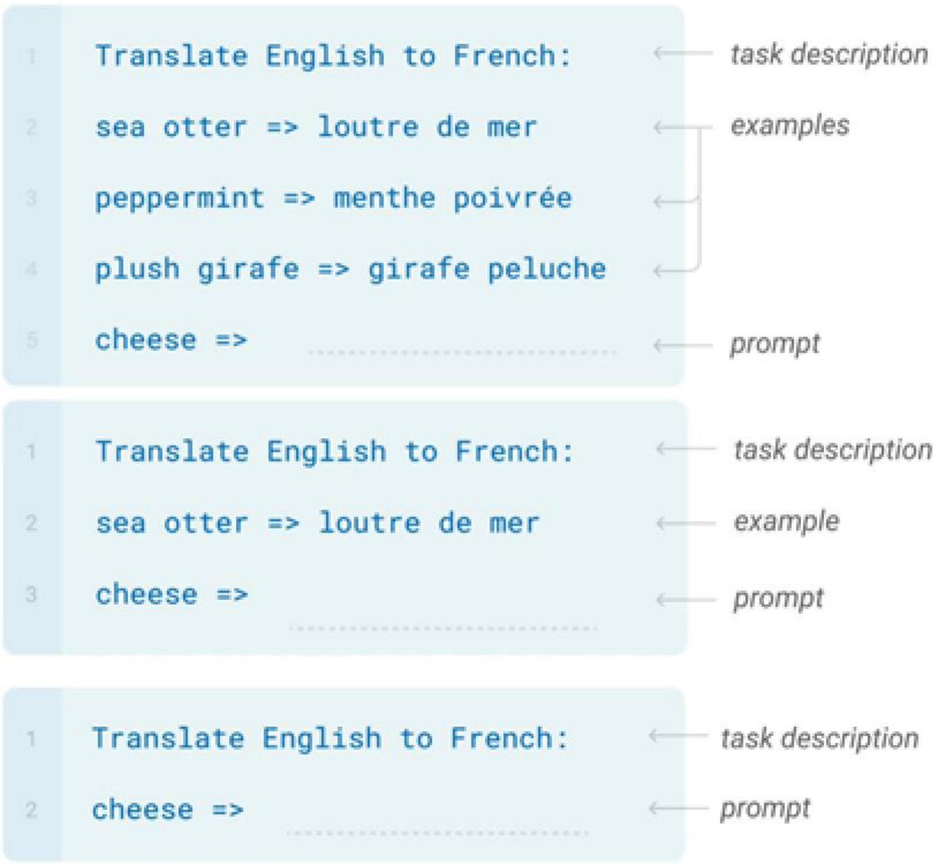


Few-shot Learning

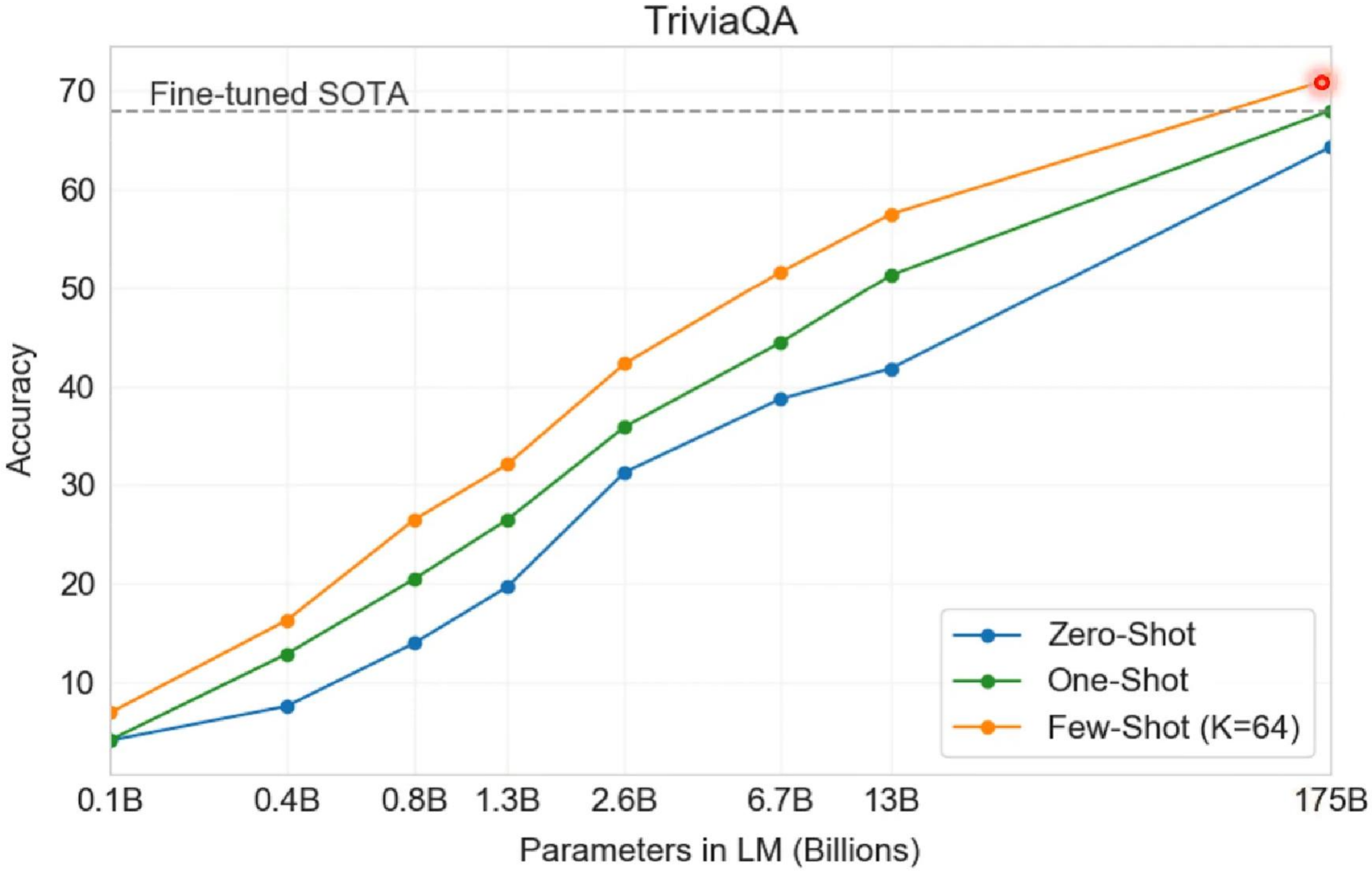
(no gradient descent)

One-shot Learning

Zero-shot Learning



Closed Book QA



GPT-3

GPT-3在两个方面达到了史无前例的高度，一是参数量，达到了1750亿，比刚推出时世界最大NLP模型Tururing大10倍，比同系列的GPT-2高出116倍。二是数据集。英语维基百科的全部内容（涵盖约600万篇文章）仅占其训练数据集的0.6%。除此之外，它还涵盖了其他数字化书籍和各种Web链接。这意味着数据集的文本类型非常丰富，人类所能查询到的知识领域均在其中。

mbert的问题

（1）mbert对于NLU的任务能完成的非常好但generate language却比较困难。所以要根据不同的下游任务去选择合适的预训练模型的结构

（2）不适用距离较短的语言对

mBert未来应用

聊天机器人

Comparison of Transfer-Learning Approaches for Response Selection in Multi-Turn Conversations

语句分析

Assessing BERT’ s Syntactic Abilities

PART THREE

Demo

-mbert判断仇恨言论

Preprocess

```python
"""
Preprocess HateSonar
"""

with open("./HateSonar/data/labeled_data.csv", encoding="utf8") as hate_sonar_data:
    next(hate_sonar_data)

    # tuple (id, count, hate_speech, offensive_language, neither, class, tweet)
    rows_in_hate_sonar = [line for line in csv.reader(hate_sonar_data)]


def preprocess_hate_sonar(text: str):
    text = re.sub("^!+ RT", "", text)  # remove retweet
    text = re.sub("@[A-Za-z0-9_]+:?", "", text)  # remove user id
    text = re.sub("&[^;]+;", "", text)  # remove emojis and special tokens
    text = re.sub("!!+", "", text)
    text = text.replace('"', "")
    text = text.strip()
```

Preprocess

北京理工大学 BEIJING INSTITUTE OF TECHNOLOGY

名称	修改日期	类型	大小
hate_sonar	2022/4/12 22:17	Microsoft Excel ...	1,800 KB
hate_speech_mlma_ar_dataset	2022/4/12 22:17	Microsoft Excel ...	426 KB
hate_speech_mlma_en_dataset	2022/4/12 22:17	Microsoft Excel ...	333 KB
hate_speech_mlma_en_dataset_with_...	2022/4/12 22:17	Microsoft Excel ...	469 KB
hate_speech_mlma_fr_dataset	2022/4/12 22:17	Microsoft Excel ...	356 KB
korean_hate_speech	2022/4/12 22:17	Microsoft Excel ...	1,599 KB
korean-malicious-comments	2022/4/12 21:45	Microsoft Excel ...	1,050 KB
LICENSE	2021/1/12 2:29	文件	12 KB
multilingual_fairness_lrec_English	2022/4/12 21:45	Microsoft Excel ...	8,984 KB
multilingual_fairness_lrec_Italian	2022/4/12 21:45	Microsoft Excel ...	622 KB
multilingual_fairness_lrec_Polish	2022/4/12 21:45	Microsoft Excel ...	1,090 KB
multilingual_fairness_lrec_Portuguese	2022/4/12 21:45	Microsoft Excel ...	201 KB
multilingual_fairness_lrec_Spanish	2022/4/12 21:45	Microsoft Excel ...	522 KB

training

```
datasets = [filename for filename in os.listdir(".") if filename.endswith(".csv")]

dataset_file_to_lang = {
    "multilingual_fairness_lrec_English.csv": "english",
    "hate_speech_mlma_en_dataset_with_stop_words.csv": "english",
    "multilingual_fairness_lrec_Spanish.csv": "spanish",
    "hate_speech_mlma_fr_dataset.csv": "france",
    "hate_speech_mlma_ar_dataset.csv": "arabic",
    "hate_sonar.csv": "english",
    "multilingual_fairness_lrec_Italian.csv": "itailian",
    "multilingual_fairness_lrec_Portuguese.csv": "portuguese",
    "hate_speech_mlma_en_dataset.csv": "english",
    "multilingual_fairness_lrec_Polish.csv": "polish",
}

print(set(dataset_file_to_lang.values()))

dataset_by_lang = {
```

training


```python
dataset_by_lang = {
    "polish": [],
    "portuguese": [],
    "arabic": [],
    "france": [],
    "spanish": [],
    "itailian": [],
    "english": [],
}

for filename, lang in dataset_file_to_lang.items():
    with open(filename, encoding="utf8") as f:

        dataset_by_lang[lang].extend(
            [
                (line[0], int(line[1])) if len(line) == 2 else ()
                 for line in csv.reader(f)
            ]
        )
```

training


```
training
{'portuguese', 'english', 'itailian', 'arabic', 'spanish', 'france', 'polish'}
polish
  train: 19655
  dev: 2183
portuguese
  train: 3417
  dev: 379
arabic
  train: 6036
  dev: 670
france
  train: 7226
  dev: 802
spanish
  train: 8867
  dev: 985
itailian
  train: 10401
  dev: 1155
english
  train: 217115
  dev: 24123
272717 30297
```

analysis


```
dataset_to_lang = {
    "multilingual_fairness_lrec_English.csv": "english",
    "hate_speech_mlma_en_dataset_with_stop_words.csv": "english",
    "multilingual_fairness_lrec_Spanish.csv": "spanish",
    "hate_speech_mlma_fr_dataset.csv": "france",
    "hate_speech_mlma_ar_dataset.csv": "arabic",
    "hate_sonar.csv": "english",
    "multilingual_fairness_lrec_Italian.csv": "itailian",
    "multilingual_fairness_lrec_Portuguese.csv": "portuguese",
    "hate_speech_mlma_en_dataset.csv": "english",
    "multilingual_fairness_lrec_Polish.csv": "polish",
}

lang_count = collections.defaultdict(lambda: {"1": 0, "0": 0})

for dataset, lang in dataset_to_lang.items():
    with open(dataset, encoding="utf8") as f:
        labels = [line[-1] if len(line) == 2 else () for line in csv.reader(f)]
        lang_count[lang]["1"] += len([l for l in labels if l == "1"])
        lang_count[lang]["0"] += len([l for l in labels if l == "0"])

for key, value in lang_count.items():
    print(key)
    print("   1: " + str(value["1"]))
    print("   0: " + str(value["0"]))
    print()
```

analysis

北京理工大学
BEIJING INSTITUTE OF TECHNOLOGY

analysis

```
D:\ANACONDA3\envs\conda3\python.exe G:/school/multilingual-bert-hate-speech-detection-main/analysis.py
english
   1: 63004
   0: 57615

spanish
   1: 1952
   0: 2974

france
   1: 3193
   0: 821

arabic
   1: 2438
   0: 915

itailian
   1: 1129
   0: 4649

portuguese
   1: 385
   0: 1513

polish
   1: 977
   0: 9942
```

北京理工大學
BEIJING INSTITUTE OF TECHNOLOGY

谢谢观看

答辩人：权双庆、杨麒沄、谢
文泽、张易初、李世杰