



文字探勘 (Text Mining)

深度學習和通用句子嵌入模型 (Deep Learning and Universal Sentence-Embedding Models)

1082TM11

MBA, BDABI, TKU (E3611) (8480) (Spring 2020)

Mon, 7, 8, 9 (14:10-17:00) (B206)



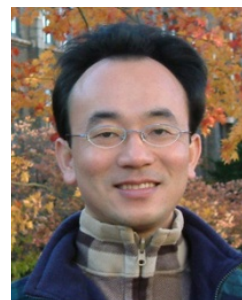
Chichang Jou

周清江

Associate Professor

副教授

cjou@mail.tku.edu.tw



Min-Yuh Day

戴敏育

Associate Professor

副教授

myday@mail.tku.edu.tw

Dept. of Information Management, Tamkang University

淡江大學 資訊管理學系

課程大綱 (Syllabus)

週次 (Week)	日期 (Date)	內容 (Subject/Topics)
1	2020/03/02	文字探勘課程介紹 (Course Orientation on Text Mining)
2	2020/03/09	文字探勘基礎：自然語言處理 (Foundations of Text Mining: Natural Language Processing; NLP)
3	2020/03/16	Python自然語言處理 (Python for Natural Language Processing)
4	2020/03/23	處理和理解文本 (Processing and Understanding Text)
5	2020/03/30	文本表達特徵工程 (Feature Engineering for Text Representation)
6	2020/04/06	人工智慧文本分析個案研究 I (Case Study on Artificial Intelligence for Text Analytics I)

課程大綱 (Syllabus)

週次 (Week)	日期 (Date)	內容 (Subject/Topics)
7	2020/04/13	文本分類 (Text Classification)
8	2020/04/20	文本摘要和主題模型 (Text Summarization and Topic Models)
9	2020/04/27	期中報告 (Midterm Project Report)
10	2020/05/04	文本相似度和分群 (Text Similarity and Clustering)
11	2020/05/11	語意分析和命名實體識別 (Semantic Analysis and Named Entity Recognition; NER)
12	2020/05/18	情感分析 (Sentiment Analysis)

課程大綱 (Syllabus)

週次 (Week)	日期 (Date)	內容 (Subject/Topics)
13	2020/05/25	人工智慧文本分析個案研究 II (Case Study on Artificial Intelligence for Text Analytics II)
14	2020/06/01	深度學習和通用句子嵌入模型 (Deep Learning and Universal Sentence-Embedding Models)
15	2020/06/08	問答系統與對話系統 (Question Answering and Dialogue Systems)
16	2020/06/15	期末報告 I (Final Project Presentation I)
17	2020/06/22	期末報告 II (Final Project Presentation II)
18	2020/06/29	教師彈性補充教學

Deep Learning and Universal Sentence-Embedding Models

Outline

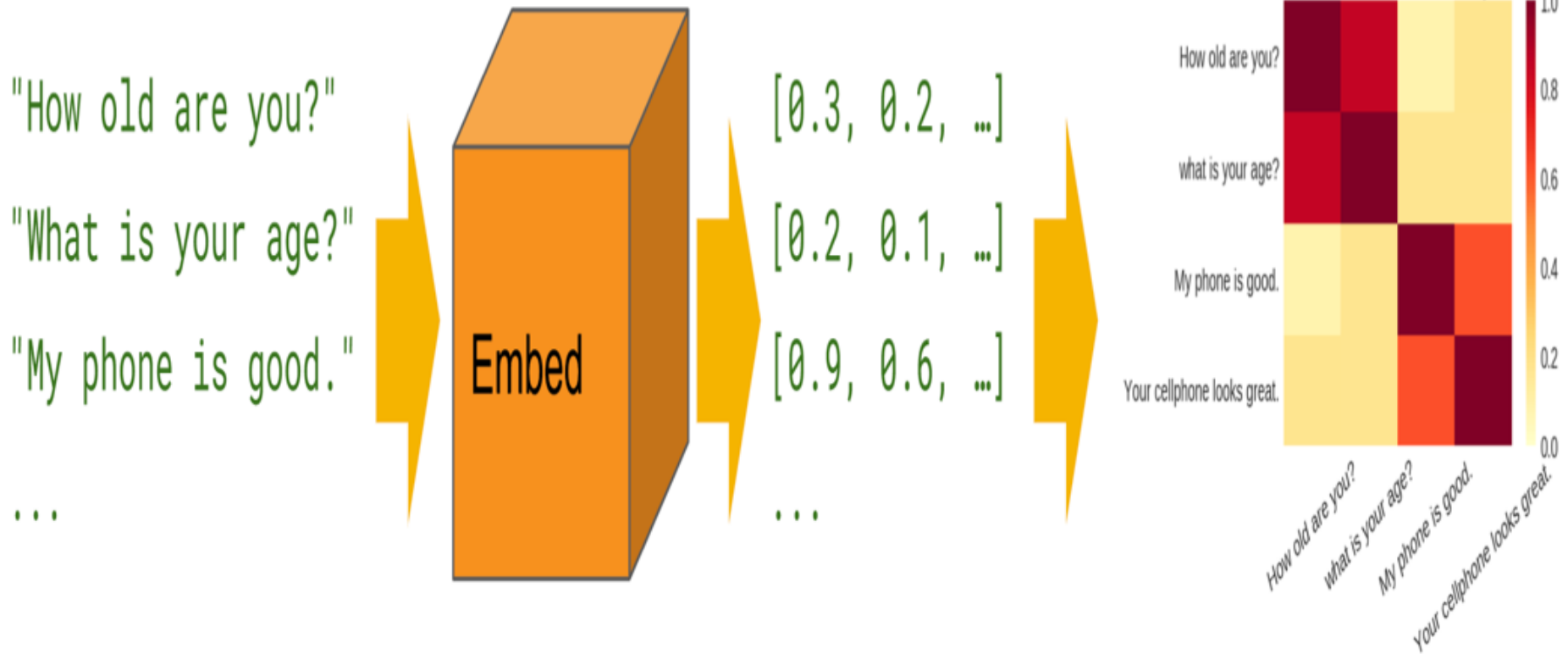
- Universal Sentence Encoder (USE)
- Universal Sentence Encoder Multilingual (USEM)
- Semantic Similarity

Universal Sentence Encoder (USE)

- The Universal Sentence Encoder encodes **text** into high-dimensional **vectors** that can be used for text classification, semantic similarity, clustering and other natural language tasks.
- The universal-sentence-encoder model is trained with a **deep averaging network (DAN)** encoder.

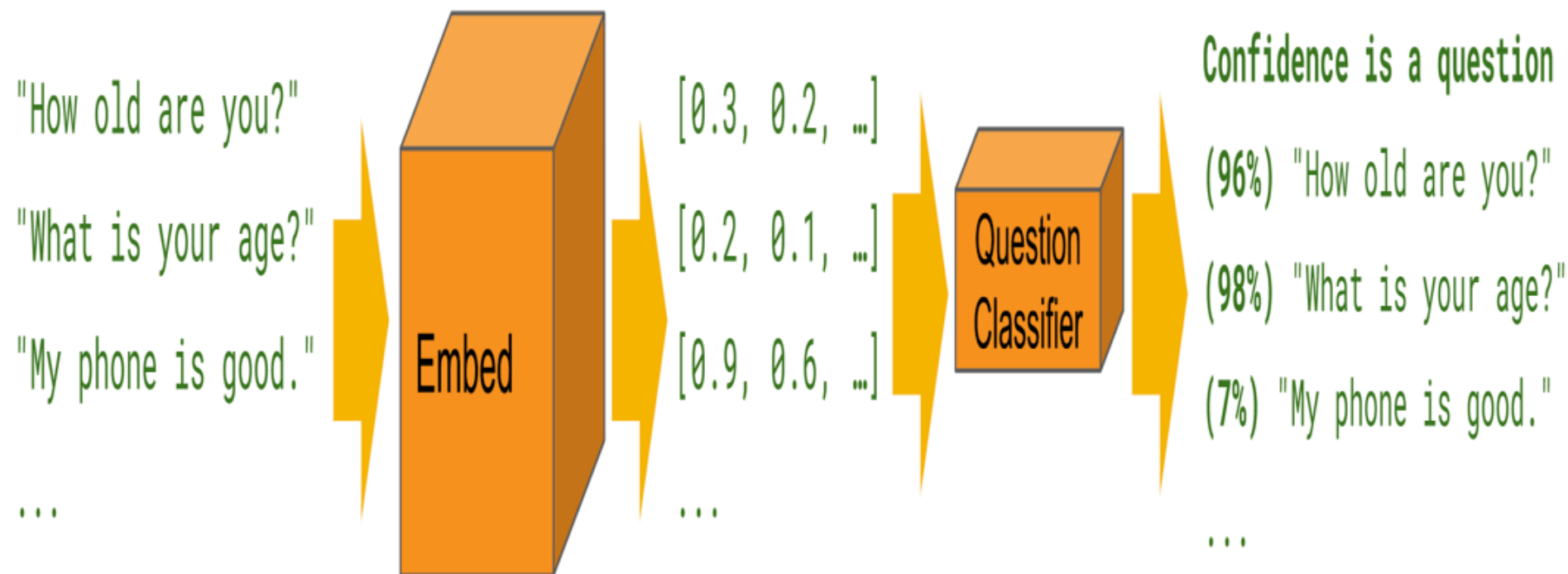
Universal Sentence Encoder (USE)

Semantic Similarity



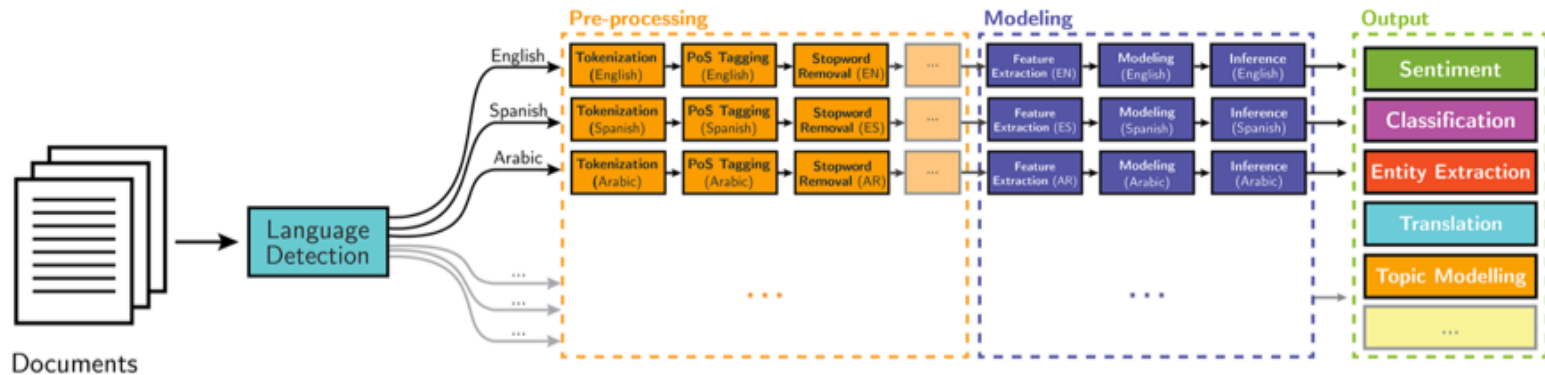
Universal Sentence Encoder (USE)

Classification

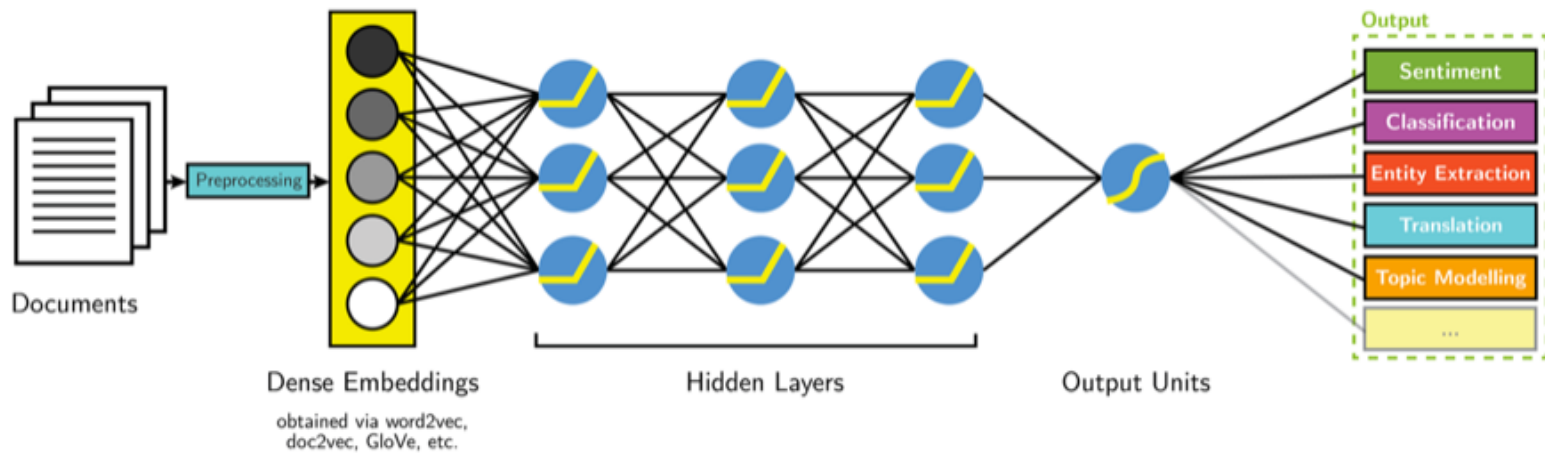


NLP

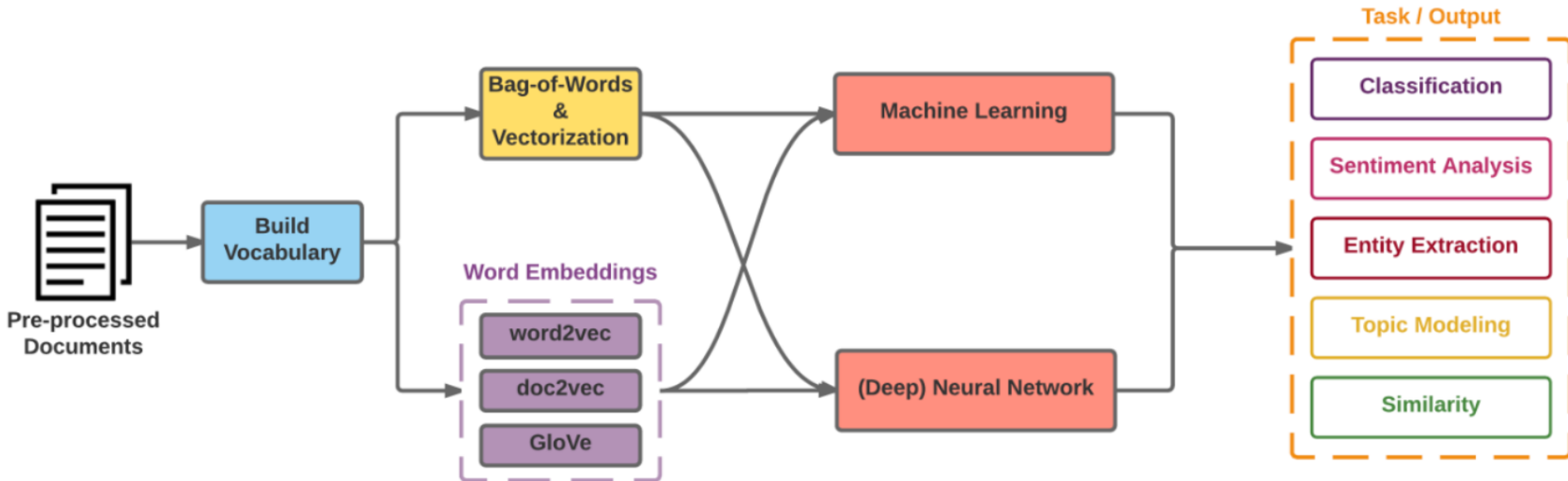
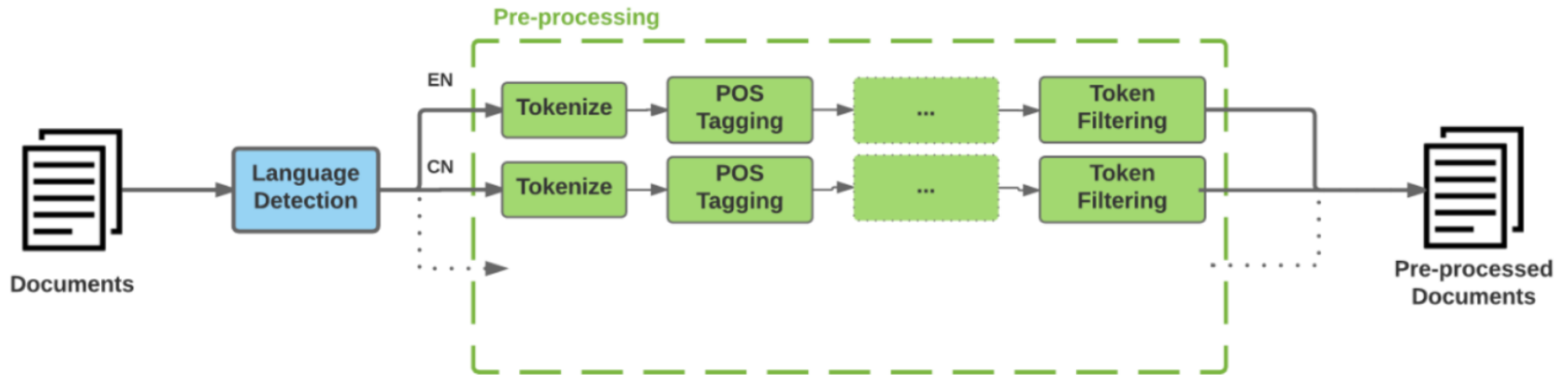
Classical NLP



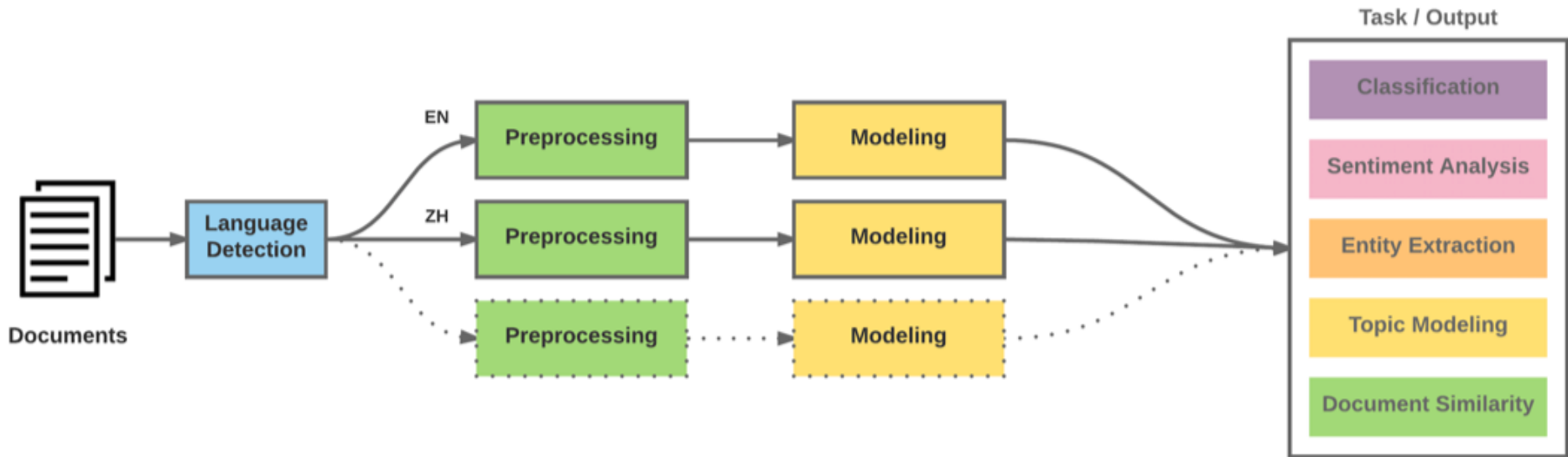
Deep Learning-based NLP



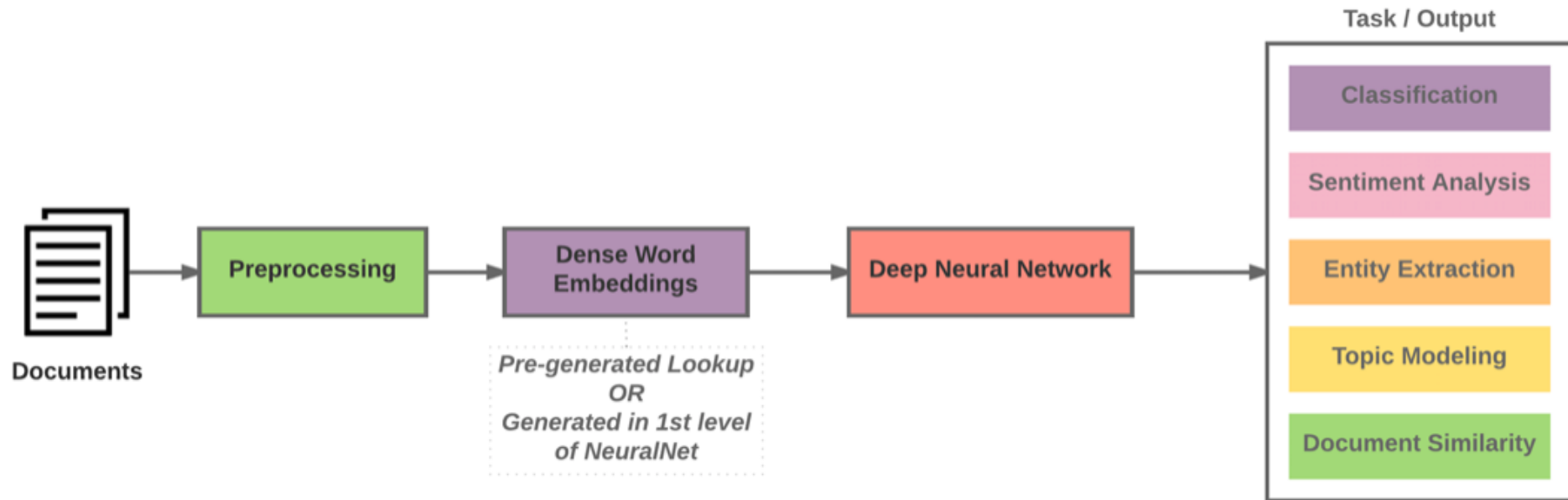
Modern NLP Pipeline



Modern NLP Pipeline



Deep Learning NLP



Natural Language Processing (NLP) and Text Mining

Raw text

Sentence Segmentation

Tokenization

Part-of-Speech (POS)

Stop word removal

Stemming / Lemmatization

Dependency Parser

String Metrics & Matching

word's stem

am → am

having → hav

word's lemma

am → be

having → have

Python in Google Colab (Python101)

<https://colab.research.google.com/drive/1FEG6DnGvwfUbeo4zJ1zTunjMqf2RkCrT>

 python101.ipynb ☆
File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment Share ⚙️ A

RAM  Disk  Editing ^

Table of contents ×

<>

Text Clustering

Semantic Analysis and Named Entity Recognition (NER)

Semantic Analysis

Named Entity Recognition (NER)

Sentiment Analysis

Sentiment Analysis - Unsupervised Lexical

Sentiment Analysis - Supervised Machine Learning

Sentiment Analysis - Supervised Deep Learning Models

Sentiment Analysis - Advanced Deep Learning

Deep Learning and Universal Sentence-Embedding Models

Universal Sentence Encoder (USE)

Universal Sentence Encoder Multilingual (USEM)

Data Visualization

+ Section

+ Code + Text

▼ Deep Learning and Universal Sentence-Embedding Models

▼ Universal Sentence Encoder (USE)

- Source: Universal Sentence Encoder: <https://tfhub.dev/google/universal-sentence-encoder/4>

```
[ ] 1 import tensorflow as tf
    2 import tensorflow_hub as hub
    3 import numpy as np
    4 import pandas as pd
    5 import os
    6 import re
    7 import matplotlib.pyplot as plt
    8 import seaborn as sns
    9
   10 module_url = "https://tfhub.dev/google/universal-sentence-encoder/4"
   11 #"https://tfhub.dev/google/universal-sentence-encoder-large/5"
   12 model = hub.load(module_url)
   13 print ("module %s loaded" % module_url)
   14 def embed(input):
   15     return model(input)
```

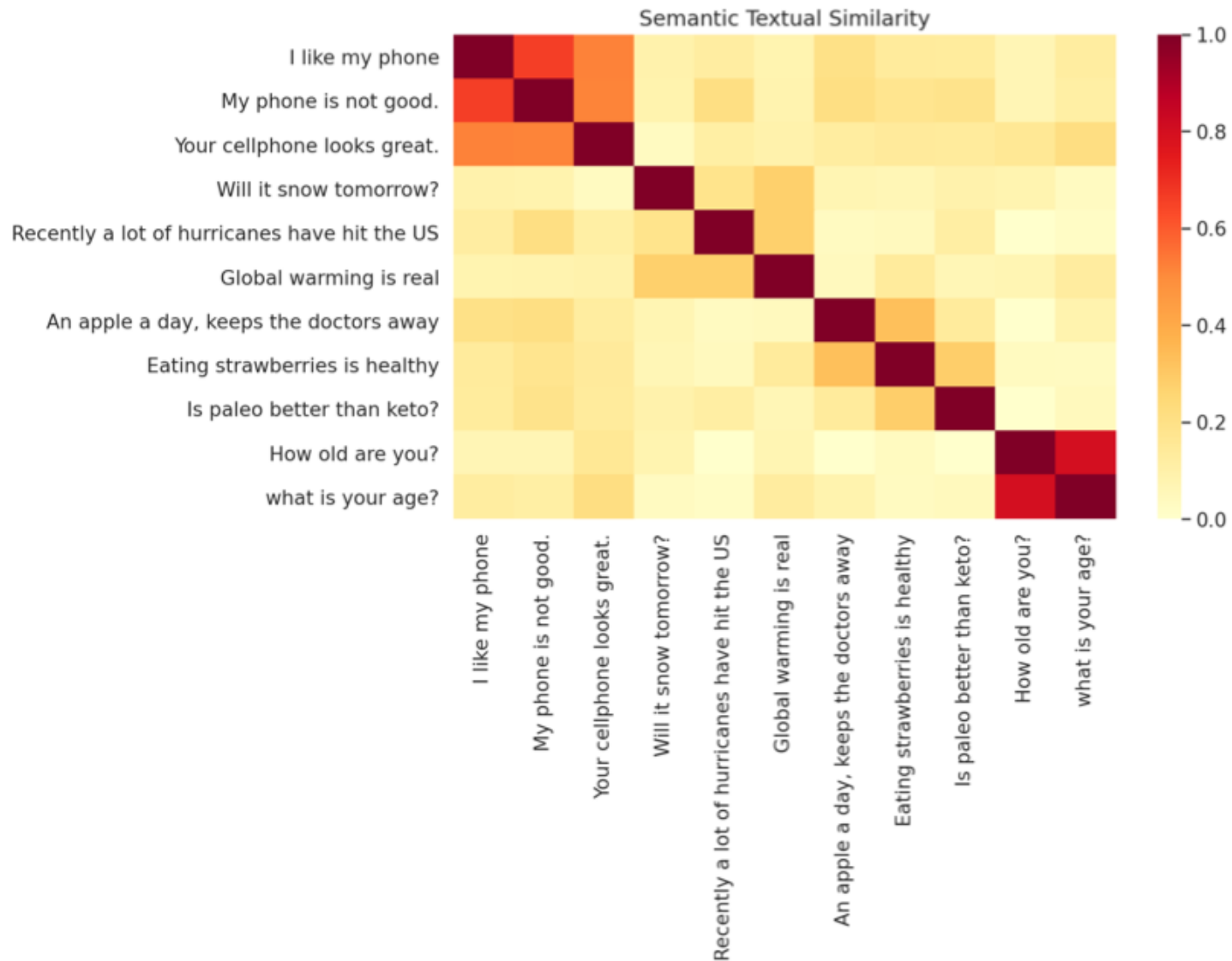
📄 module <https://tfhub.dev/google/universal-sentence-encoder/4> loaded

```
[ ] 1 word = "Elephant"
    2 sentence = "I am a sentence for which I would like to get its embedding."
```

<https://tinyurl.com/imtkupython101>

Python in Google Colab (Python101)

<https://colab.research.google.com/drive/1FEG6DnGvwfUbeo4zJ1zTunjMqf2RkCrT>



<https://tinyurl.com/imtkupython101>

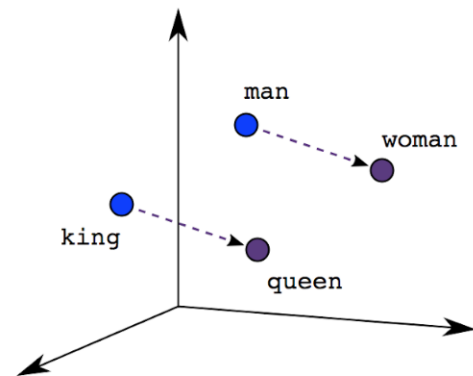
One-hot encoding

'The mouse ran up the clock' =

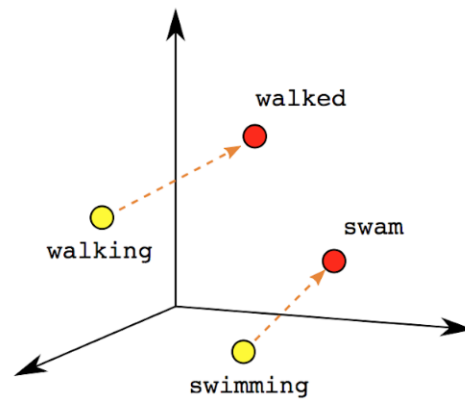
The	1	[	[0, 1, 0, 0, 0, 0, 0],
mouse	2		[0, 0, 1, 0, 0, 0, 0],
ran	3		[0, 0, 0, 1, 0, 0, 0],
up	4		[0, 0, 0, 0, 1, 0, 0],
the	1		[0, 1, 0, 0, 0, 0, 0],
clock	5		[0, 0, 0, 0, 0, 1, 0]]

[0, 1, 2, 3, 4, 5, 6]

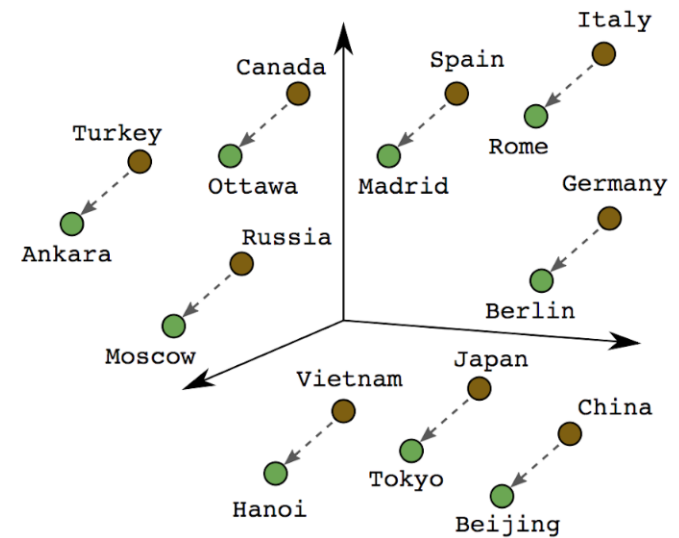
Word embeddings



Male-Female

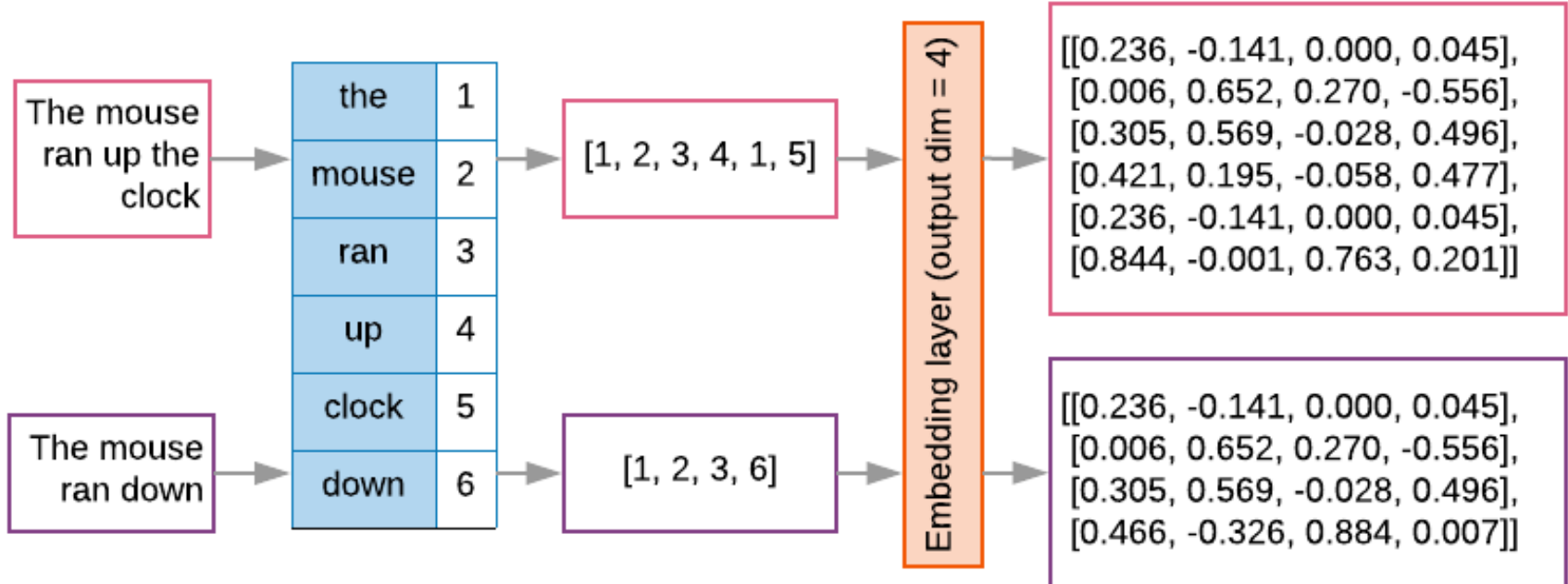


Verb Tense

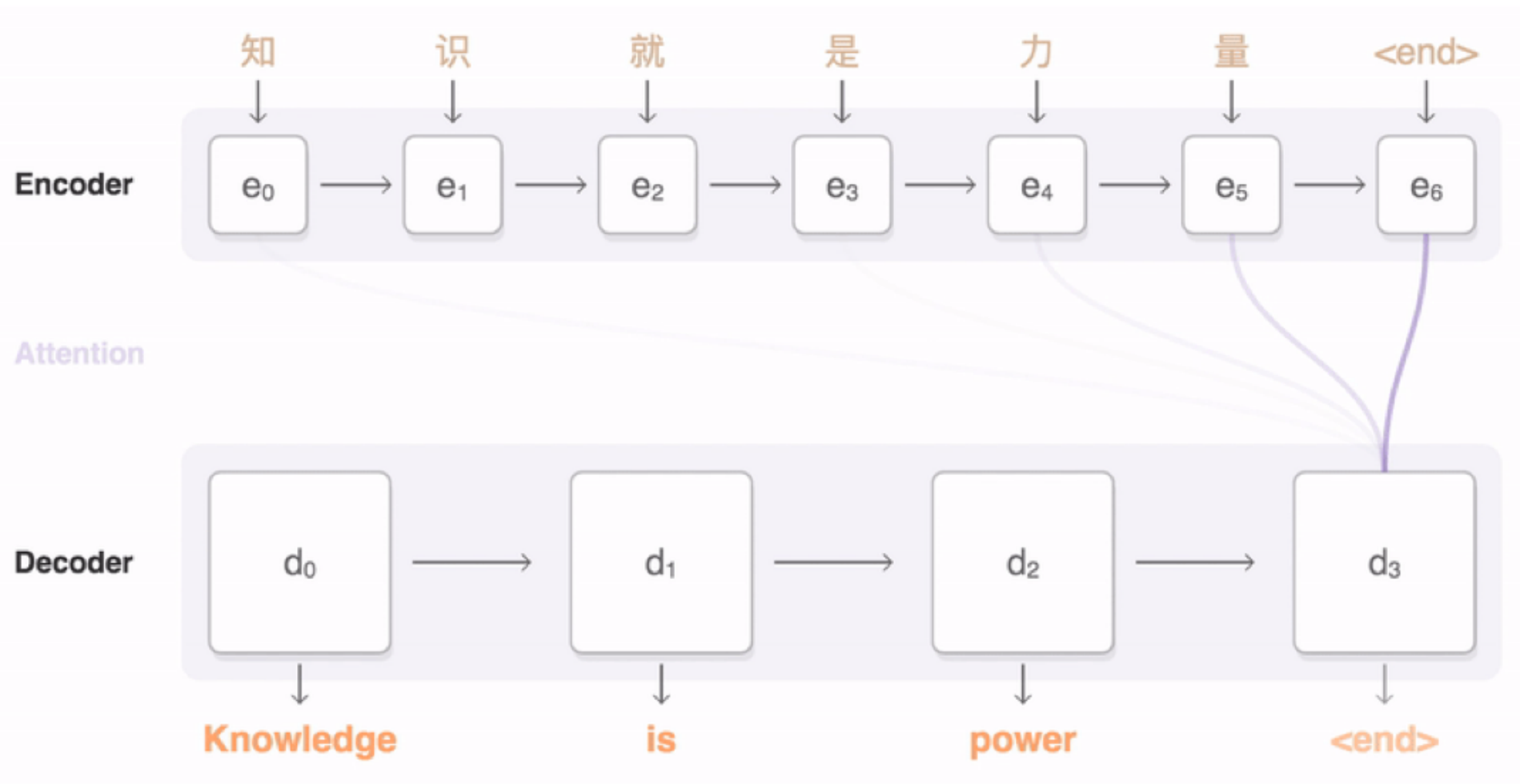


Country-Capital

Word embeddings

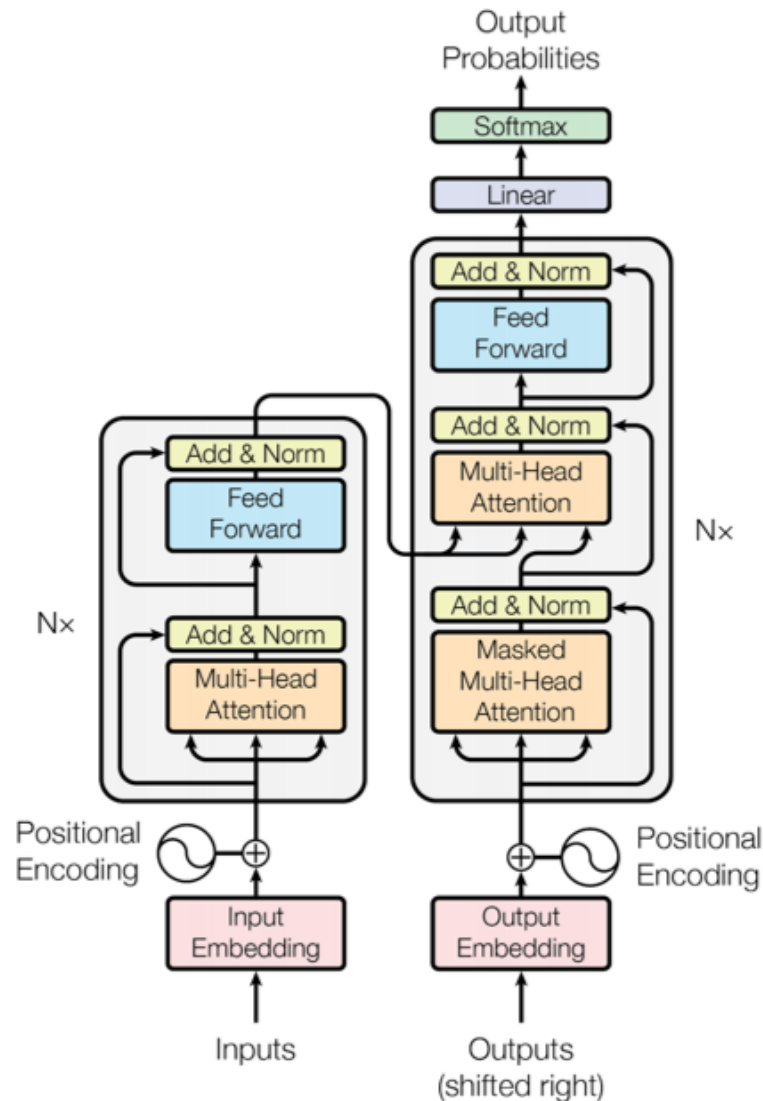


Sequence to Sequence (Seq2Seq)



Transformer (Attention is All You Need)

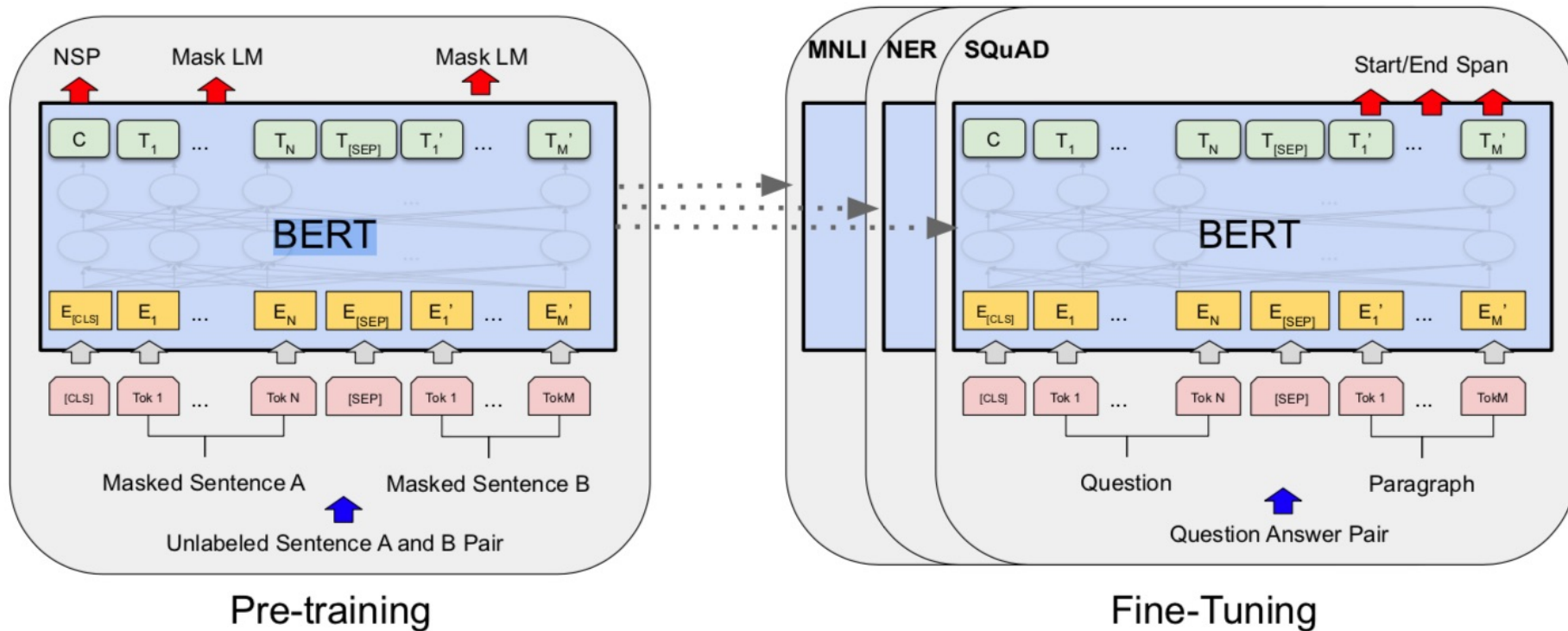
(Vaswani et al., 2017)



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

BERT (Bidirectional Encoder Representations from Transformers)

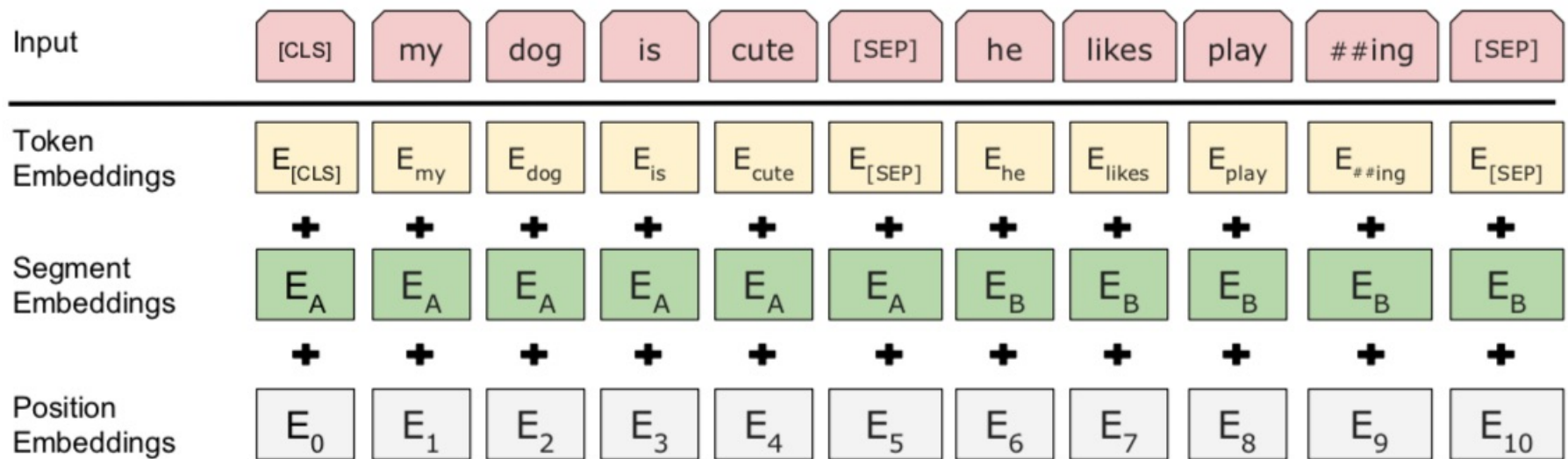
Overall pre-training and fine-tuning procedures for BERT



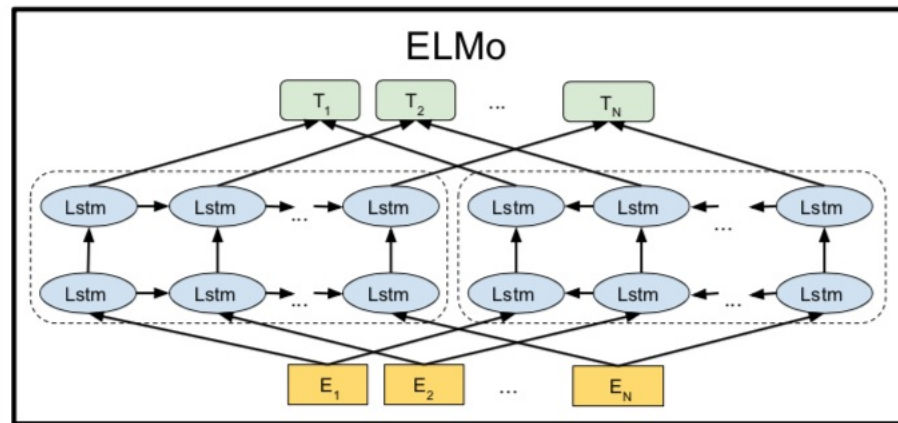
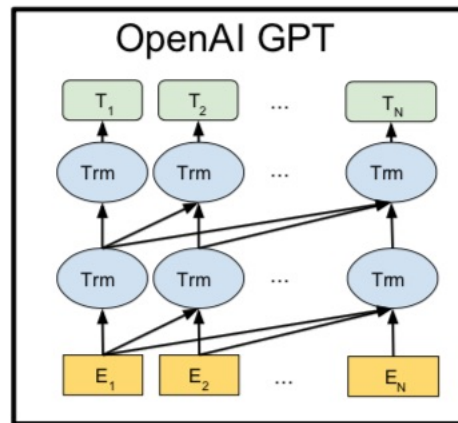
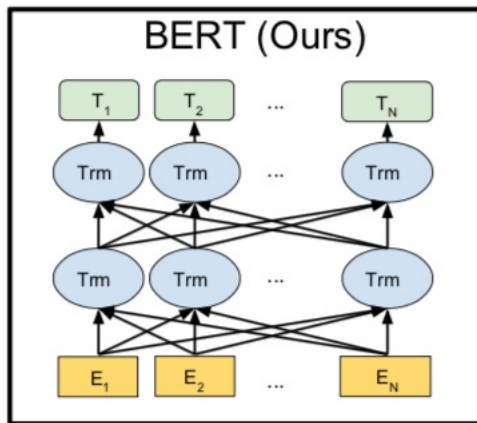
BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

BERT (Bidirectional Encoder Representations from Transformers)

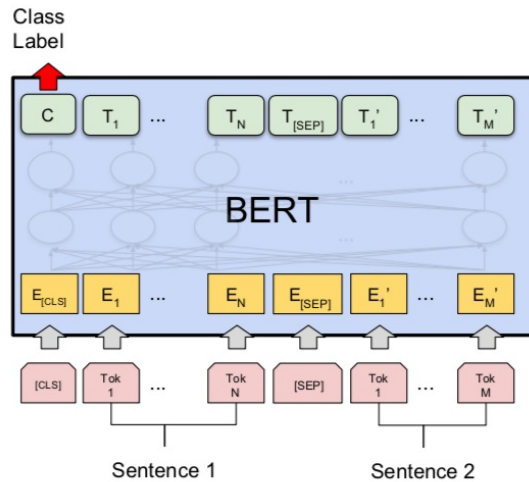
BERT input representation



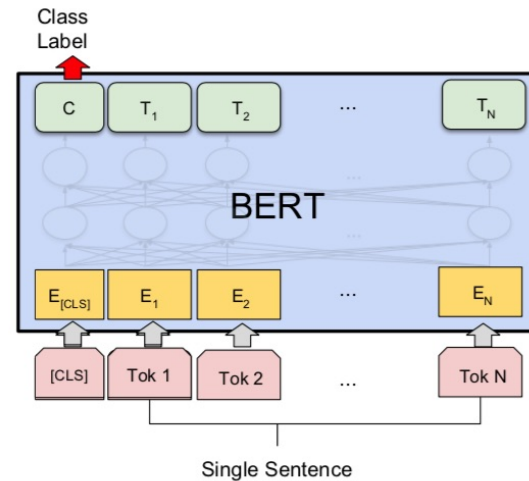
BERT, OpenAI GPT, ELMo



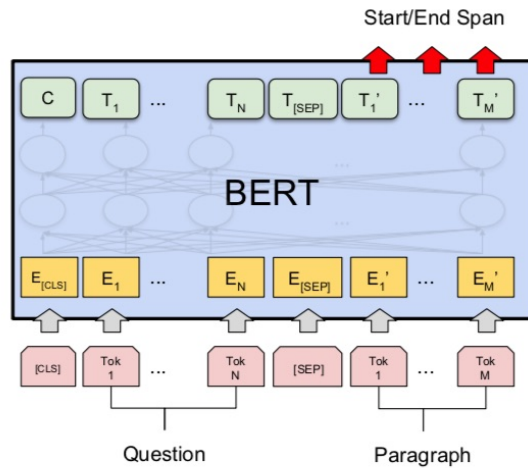
Fine-tuning BERT on Different Tasks



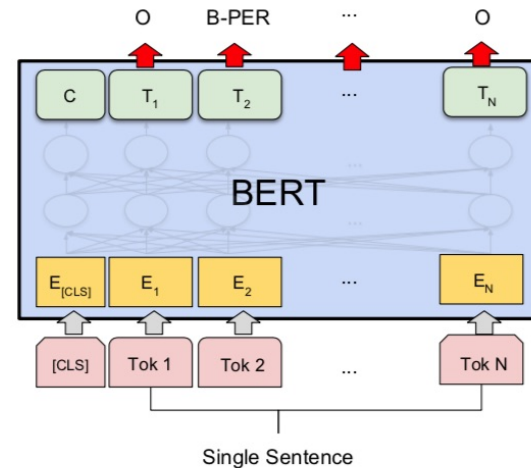
(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG



(b) Single Sentence Classification Tasks:
SST-2, CoLA



(c) Question Answering Tasks:
SQuAD v1.1

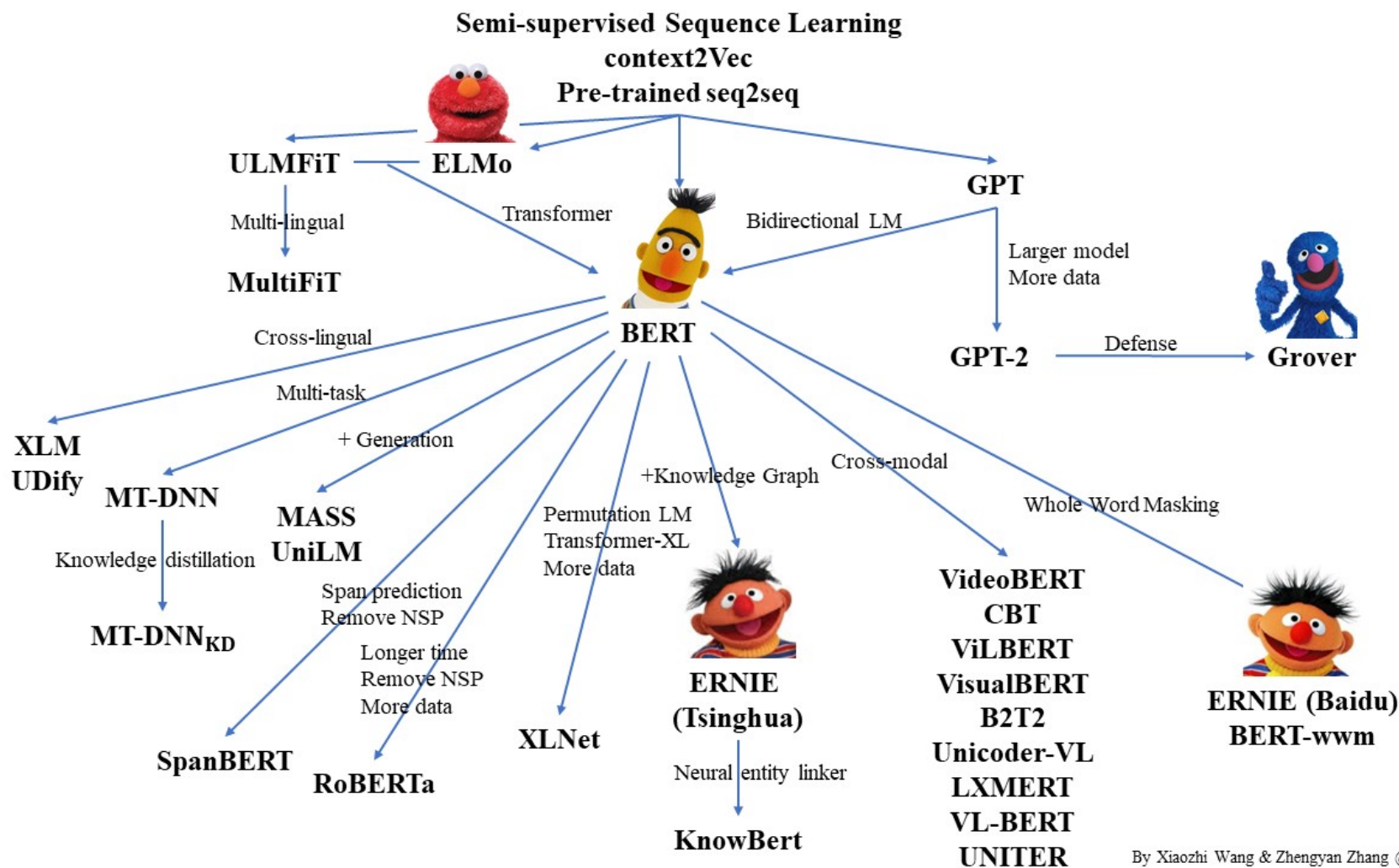


(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

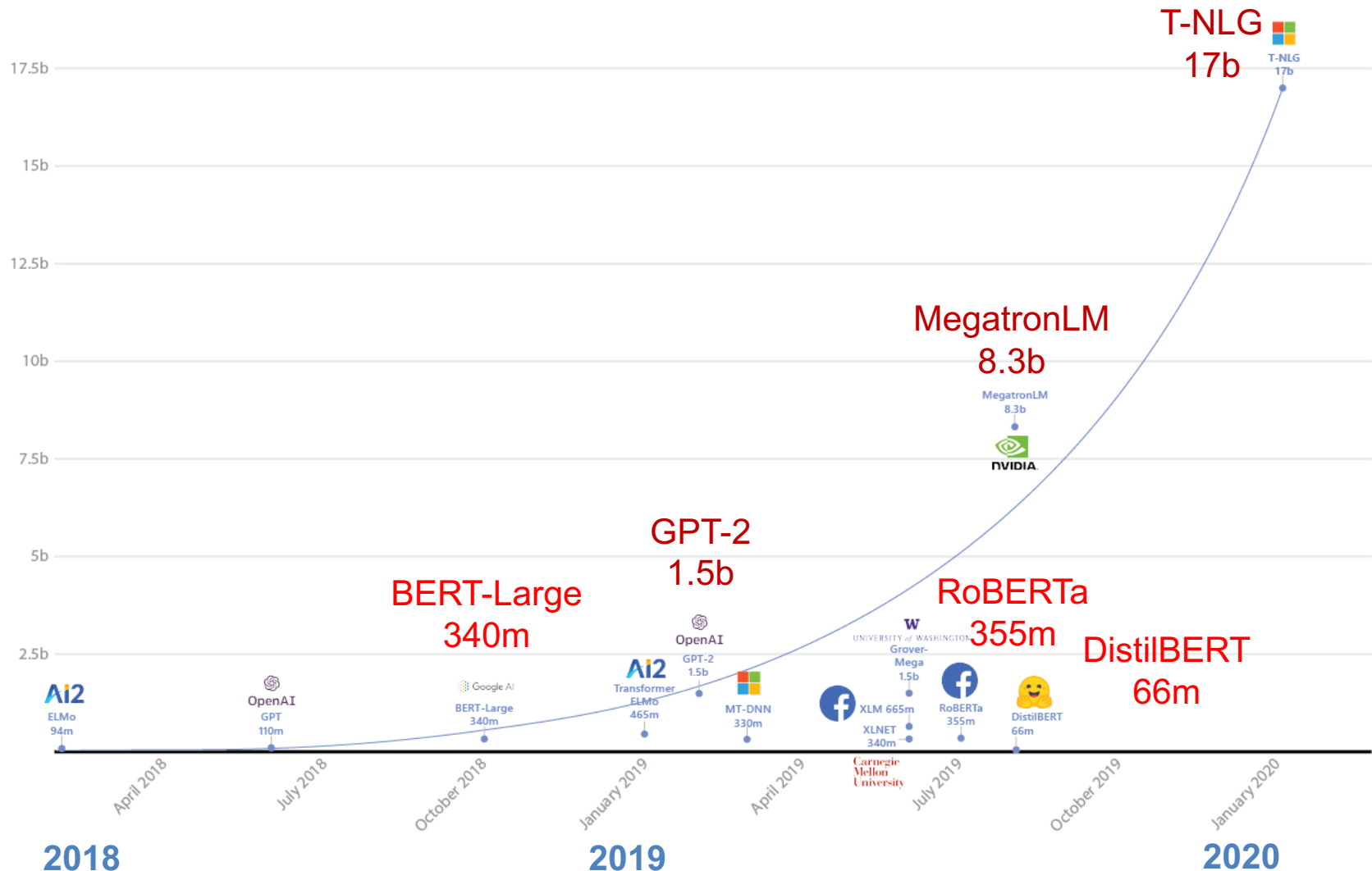
Source: Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2018).

"Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805.

Pre-trained Language Model (PLM)

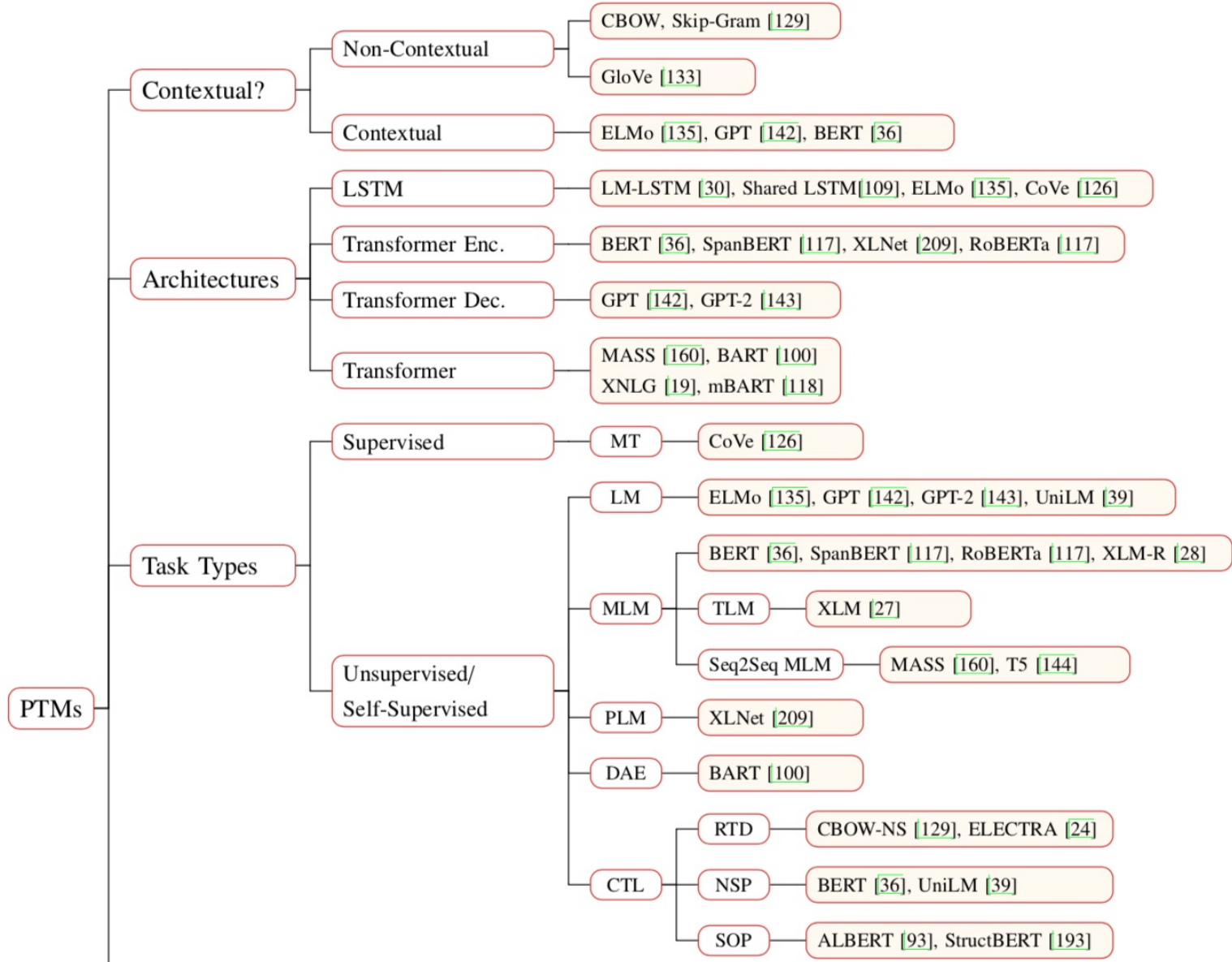


Turing Natural Language Generation (T-NLG)

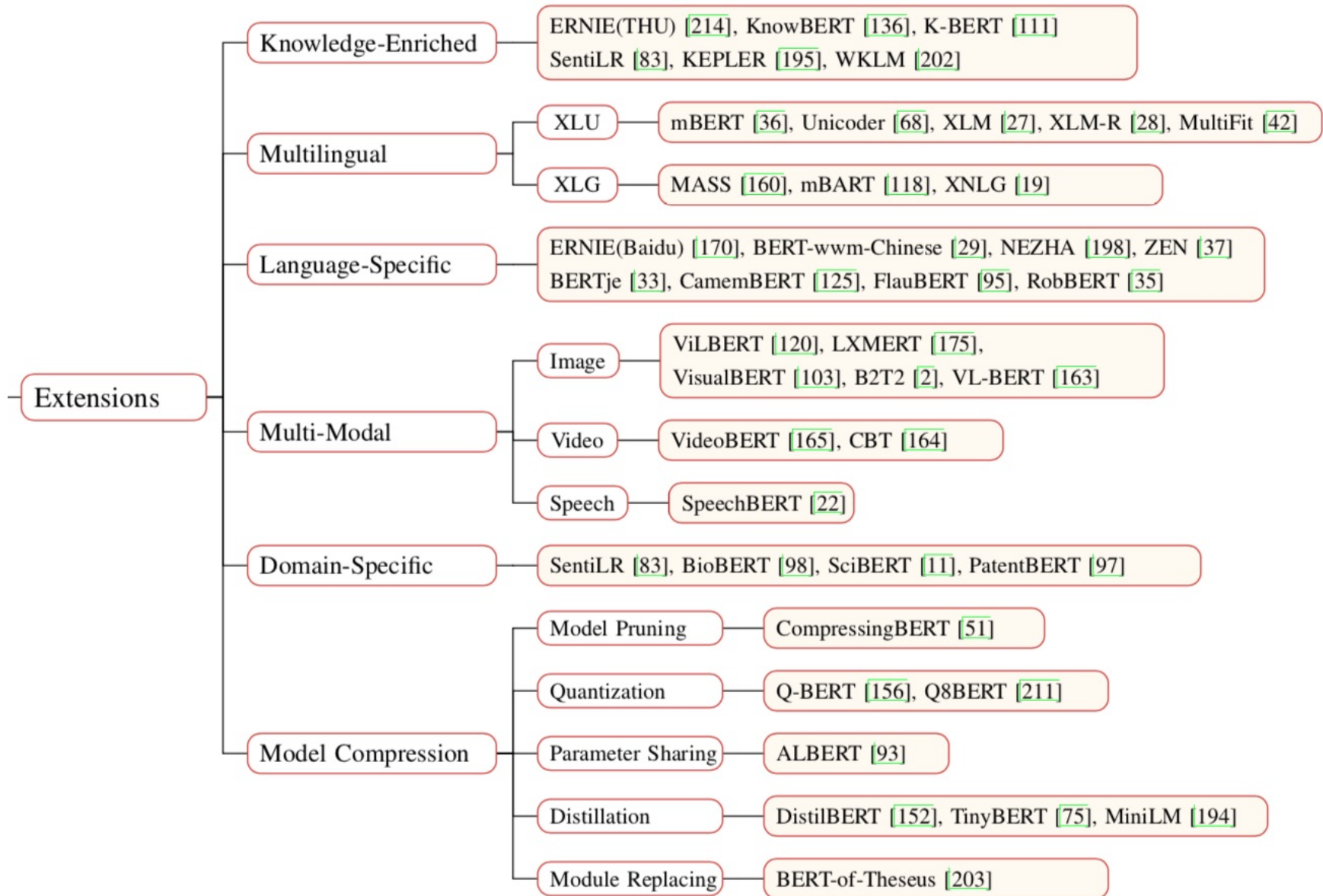


Source: <https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/>

Pre-trained Models (PTM)



Pre-trained Models (PTM)



Transformers Transformers

State-of-the-art Natural Language Processing for TensorFlow 2.0 and PyTorch

- Transformers
 - pytorch-transformers
 - pytorch-pretrained-bert
- provides state-of-the-art general-purpose architectures
 - (BERT, GPT-2, RoBERTa, XLM, DistilBert, XLNet, CTRL...)
 - for Natural Language Understanding (NLU) and Natural Language Generation (NLG)
with over 32+ pretrained models
in 100+ languages
and deep interoperability between TensorFlow 2.0 and PyTorch.

NLP Benchmark Datasets

Task	Dataset	Link
Machine Translation	WMT 2014 EN-DE WMT 2014 EN-FR	http://www-lium.univ-lemans.fr/~schwenk/csml_joint_paper/
Text Summarization	CNN/DM Newsroom DUC Gigaword	https://cs.nyu.edu/~kcho/DMQA/ https://summari.es/ https://www-nlpir.nist.gov/projects/duc/data.html https://catalog.ldc.upenn.edu/LDC2012T21
Reading Comprehension Question Answering Question Generation	ARC CliCR CNN/DM NewsQA RACE SQuAD Story Cloze Test NarrativeQA Quasar SearchQA	http://data.allenai.org/arc/ http://aclweb.org/anthology/N18-1140 https://cs.nyu.edu/~kcho/DMQA/ https://datasets.maluuba.com/NewsQA http://www.qizhexie.com/data/RACE_leaderboard https://rajpurkar.github.io/SQuAD-explorer/ http://aclweb.org/anthology/W17-0906.pdf https://github.com/deepmind/narrativeqa https://github.com/bdhingra/quasar https://github.com/nyu-dl/SearchQA
Semantic Parsing	AMR parsing ATIS (SQL Parsing) WikiSQL (SQL Parsing)	https://amr.isi.edu/index.html https://github.com/jkkummerfeld/text2sql-data/tree/master/data https://github.com/salesforce/WikiSQL
Sentiment Analysis	IMDB Reviews SST Yelp Reviews Subjectivity Dataset	http://ai.stanford.edu/~amaas/data/sentiment/ https://nlp.stanford.edu/sentiment/index.html https://www.yelp.com/dataset/challenge http://www.cs.cornell.edu/people/pabo/movie-review-data/
Text Classification	AG News DBpedia TREC 20 NewsGroup	http://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html https://wiki.dbpedia.org/Datasets https://trec.nist.gov/data.html http://qwone.com/~jason/20Newsgroups/
Natural Language Inference	SNLI Corpus MultiNLI SciTail	https://nlp.stanford.edu/projects/snli/ https://www.nyu.edu/projects/bowman/multinli/ http://data.allenai.org/scitail/
Semantic Role Labeling	Proposition Bank OneNotes	http://propbank.github.io/ https://catalog.ldc.upenn.edu/LDC2013T19

Summary

- Universal Sentence Encoder (USE)
- Universal Sentence Encoder Multilingual (USEM)
- Semantic Similarity

References

- Dipanjan Sarkar (2019), Text Analytics with Python: A Practitioner's Guide to Natural Language Processing, Second Edition. APress. <https://github.com/Apress/text-analytics-w-python-2e>
- Benjamin Bengfort, Rebecca Bilbro, and Tony Ojeda (2018), Applied Text Analysis with Python, O'Reilly Media. <https://www.oreilly.com/library/view/applied-text-analysis/9781491963036/>
- Kumar Ravi and Vadlamani Ravi (2015), "A survey on opinion mining and sentiment analysis: tasks, approaches and applications." Knowledge-Based Systems, 89, pp.14-46.
- Qiu, Xipeng, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. "Pre-trained Models for Natural Language Processing: A Survey." arXiv preprint arXiv:2003.08271 (2020).
- HuggingFace (2020), Transformers Notebook, <https://huggingface.co/transformers/notebooks.html>
- The Super Duper NLP Repo, <https://notebooks.quantumstat.com/>
- Min-Yuh Day (2020), Python 101, <https://tinyurl.com/imtkupython101>