Model Knowledge Stimulation with Prompts for Pre-trained Language Models

Zhiyuan Liu
Tsinghua University
Background

- NLP is the key to pass Turing Test and Realize AI

Turing Test

Alan Turing
(1912 - 1954)
Key founder of CS and AI, proposed Turing test based on language understanding

Dartmouth Conference
(1956)
Proposed AI for the first time and listed NLP as the key research problem
Background

• Deep language understanding requires complicated knowledge

Language understanding requires the ability of **knowledge acquisition, representation and application**
Research Spectrum of NLP

1960

Noam Chomsky

Modern grammar (Linguistics) theory proposed in 1950s has been introduced in NLP but cannot well cover complex language usage.
An expert system represents facts and rules with the knowledge base, and conducts inference based on the knowledge base.

Modern grammar (Linguistics) theory proposed in 1950s has been introduced in NLP but cannot well cover complex language usage.
Research Spectrum of NLP

**Noam Chomsky**
Modern grammar (Linguistics) theory proposed in 1950s has been introduced in NLP but cannot well cover complex language usage.

**Edward Feigenbaum**
An expert system represents facts and rules with the knowledge base, and conducts inference based on the knowledge base.

**Symboledge (Symbolic Knowledge)**
- linguistic rules
- knowledge bases
  - human-friendly, discrete, sparse
Edward Feigenbaum
An **expert system** represents facts and rules with the knowledge base, and **conducts inference based on the knowledge base**

Noam Chomsky
Modern grammar (**Linguistics**) theory proposed in 1950s has been introduced in NLP but **cannot well cover complex language usage**.

Robert Mercer
The data-driven **statistical models** proposed in 1990s only **take advantages of shallow lexical information**.
Research Spectrum of NLP

The data-driven statistical models proposed in 1990s only take advantages of shallow lexical information.

Modeledge (Model Knowledge)

- SVM
- Decision Tree
- CRF, LDA

machine-friendly, discrete/continuous, shallow
Edward Feigenbaum
An expert system represents facts and rules with the knowledge base, and conducts inference based on the knowledge base.

Noam Chomsky
Modern grammar (Linguistics) theory proposed in 1950s has been introduced in NLP but cannot well cover complex language usage.

Yoshua Bengio
Neural models are introduced in NLP in 2010s but challenged by deep understanding with structured knowledge.

Robert Mercer
The data-driven statistical models proposed in 1990s only take advantages of shallow lexical information.
Yoshua Bengio

**Neural models** are introduced in NLP in 2010s but challenged by **deep understanding with structured knowledge**.

**Embeledge (Embedding Knowledge)**

- word embedding
- knowledge graph embedding
- ......

**machine-friendly, continuous, shallow**
Research Spectrum of NLP

Yoshua Bengio

Neural models are introduced in NLP in 2010s but challenged by deep understanding with structured knowledge.

Embeledge (Embedding Knowledge)

- word embedding
- knowledge graph embedding
- ......

Modeledge (Model Knowledge)

- CNN、RNN、GNN
- BERT、GPT、T5、BART
- ......

machine-friendly、continuous、shallow

machine-friendly、continuous、deep
Edward Feigenbaum
An expert system represents facts and rules with the knowledge base, and conducts inference based on the knowledge base.

Noam Chomsky
Modern grammar (Linguistics) theory proposed in 1950s has been introduced in NLP but cannot well cover complex language usage.

Yoshua Bengio
Neural models are introduced in NLP in 2010s but challenged by deep understanding with structured knowledge.

Robert Mercer
The data-driven statistical models proposed in 1990s only take advantages of shallow lexical information.

Acquisition, Representation and Application of knowledge for language understanding.
Closed-Loop of Knowledge in NLP

1. **Knowledge Acquisition**
   - How to extract **accurate** knowledge from **noisy** text

2. **Knowledge Representation**
   - How to represent **complex** knowledge in machine

3. **Knowledge Guide**
   - How to improve NLP models by incorporating knowledge

- Human Knowledge
- Machine Knowledge
- Text Data
Knowledge Extraction from Open Text

• Challenge: From noisy text to accurate knowledge

  Filtering - Instance-level attention to remove noise

  Context - Use rich context to improve accuracy & coverage

Instance-Level Attention

Transfer

Relation Extraction from Open Text
Representation Learning for Complex Knowledge

• Challenge: Efficient knowledge representation for machine
  
  **Fusion** – Consider internal and external information of KGs
  
  **Unified** - Build unified knowledge embeddings

![Diagram showing unified representation of human and machine knowledge](image-url)
Knowledge-Guided NLP Models

- Challenge: Incorporate knowledge in heterogeneous models

  **Arch** – Design learning architecture with knowledge
  **In/Out** – Design inputs and objectives with knowledge

Dihydroartemisinin, a derivative of artemisinin, has a powerful killing effect on the red stage of malaria parasites.
Pre-trained Language Models as Advanced Model Knowledge
Pretrained Language Model as a Breakthrough in 2018

- Impressive progress of deep learning on unsupervised text corpora

- 2001: Neural language models
- 2008: Multi-task learning
- 2013: Word embeddings
- 2013: Neural networks for NLP
- 2014: Sequence-to-sequence models
- 2015: Attention
- 2015: Memory-based networks
- 2018: Pretrained language models
Challenge of Deep Learning in NLP

• Deep Learning has achieved the best performance in most NLP tasks
• Challenges: require large-scale supervised training
Pretrained Language Models

- Pre-trained Language Models (PLMs) can learn language patterns from large-scale unlabeled data, and improve the performance on downstream tasks by fine-tuning parameters.
Superior Performance on Language Understanding

GLUE Benchmark

- CBOW: 58.6
- BiLSTM: 64.2
- BiLSTM+ELMO: 67.7
- Human: 87.1
- Roberta: 88.4
- ELECTRA: 89.4
- MT-DNN: 89.9
- T5: 90.3
- ALBERT: 90.6
Superior Performance on Language Generation

The diagram shows the comparison of different language models based on their Interactive SSA (%) against Perplexity. The models include:

- Human (86%)
- Meena (79%)
- Meena (base) (72%)
- Cleverbot (56%)
- Mitsuku (56%)
- DiagoGPT (48%)
- Xiaolce (31%)
Contests of Pretrained Language Models

[Diagram showing various models and their relationships, with labels for each model.

https://github.com/thunlp/PLMpapers]
Knowledgeable PLM

- Incorporate external symbolic knowledge with model knowledge of PLMs

**Bob Dylan** wrote *Blowin’ in the Wind* in 1962, and wrote *Chronicles: Volume One* in 2004.
How to Make PLMs Knowledgeable

• **Knowledgeable Input**: input augmentation as extra features

• **Knowledgeable Tasks**: knowledge-guided pre-training tasks

• **Knowledgeable Framework**: knowledge-guided neural architecture
Knowledgeable Input

- ERNIE: Enhanced Language Representation with Informative Entities
  - Lower layers for text, and higher layers for knowledge integration
  - Link Prediction Objective with MLM

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, Qun Liu. ERNIE: Enhanced Language Representation with Informative Entities. ACL 2019.
Knowledgeable Tasks

• KEPLER: Joint learning of knowledge and language modeling
• Unify knowledge embedding and language representation into the same semantic space

Wang et al. KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation. To appear at TACL.
Knowledgeable Tasks

• Coreference: Two or more expressions in a text refer to the same entity

Antoine published *The Little Prince* in 1943. *The book* follows a young prince who visits various planets in space.

• CorefBERT: Learn coreferential reasoning ability from large-scale unlabeled corpus
  • Mask one or several mentions and requires model to predict the masked mention’s corresponding referents

Knowledgeable Framework

- LM with mechanisms for selecting and copying facts from KG

Super Mario Land is a 1989 side-scrolling platform video game developed and published by [Nintendo] as a [launch title] for their [Game Boy] [handheld game console].

Framework of Knowledgeable Learning

- More methods to incorporate multiple knowledge into deep learning

Dihydroartemisinin, a derivative of artemisinin, has a powerful killing effect on the red stage of malaria parasites.
Model Knowledge Stimulation with Prompts
GPT-3 and Prompts

- GPT-3 has 175 billion parameters, almost impossible to fine-tune.
- GPT-3 introduces prompts to stimulate knowledge in PLMs.
- Prompts are typically task descriptions and language triggers to give models hints to generate words.
- By adding prompts, downstream tasks are formalized as language modeling problems.

GPT-3 and Prompts

• Prompts stay **untuned**
• Prompts have great performance on **few-shot** and **zero-shot** tasks
• Big models contain more knowledge from large unlabeled corpora, and have better performance

Prompt-Oriented Fine-Tuning

• Prompts can be tuned together with PLMs for downstream tasks

Masked Language Model Fine-tuning

Task-oriented Fine-tuning

Prompt-oriented Fine-tuning

Map the predicted words to a class
Use the pre-training task to predict words
Prompt-Oriented Fine-Tuning

- Auto generated prompts
- Use encoder-decoder model to generate templates

A fun ride. \textbf{great} \textit{Y}
A pleasure to watch. \textbf{great} \textit{Y}

No reason to watch. \textbf{terrible} \textit{Y}
This junk. \textbf{terrible} \textit{Y}

\textit{positive: great, negative: terrible}

\textbf{Label mapping } \mathcal{M}(Y)
Prompt Tuning

- Keep PLM fixed and tune soft prompts
- Achieve comparable performance with tuning all model parameters

Pre-trained Prompt Tuning

• Tuning soft prompts under few-shot setting is not easy
• Pre-train general soft prompts, keep PLMs fixed and tune pre-trained soft prompts for downstream tasks

Pre-trained Prompt Tuning

- As compared with prompt tuning, pre-trained prompt tuning works better under few-shot settings

<table>
<thead>
<tr>
<th>Model</th>
<th>Method</th>
<th>SST-2 Acc.</th>
<th>SST-5 Acc.</th>
<th>RACE-m Acc.</th>
<th>RACE-h Acc.</th>
<th>BoolQ Acc.</th>
<th>RTE Acc.</th>
<th>CB F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>FT (11B)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T5-Small</td>
<td>-</td>
<td>72.83.1</td>
<td>31.10.4</td>
<td>26.40.6</td>
<td>26.30.5</td>
<td>59.20.6</td>
<td>54.01.7</td>
<td>70.14.6</td>
</tr>
<tr>
<td>T5-Base</td>
<td>-</td>
<td>74.62.7</td>
<td>28.81.8</td>
<td>27.20.5</td>
<td>26.70.2</td>
<td>61.92.1</td>
<td>56.12.3</td>
<td>70.42.6</td>
</tr>
<tr>
<td>T5-Large</td>
<td>-</td>
<td>89.12.2</td>
<td>42.41.2</td>
<td>48.21.6</td>
<td>43.21.7</td>
<td>74.60.9</td>
<td>64.43.4</td>
<td>82.32.2</td>
</tr>
<tr>
<td>T5-XL</td>
<td>-</td>
<td>89.63.2</td>
<td>38.45.1</td>
<td>55.02.8</td>
<td>50.92.6</td>
<td>77.21.1</td>
<td>62.36.8</td>
<td>81.99.0</td>
</tr>
<tr>
<td>T5-XXL</td>
<td>-</td>
<td>91.40.8</td>
<td>40.62.0</td>
<td>62.98.9</td>
<td>54.83.0</td>
<td>80.82.4</td>
<td>64.12.0</td>
<td>86.55.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Method</th>
<th>SST-2 Acc.</th>
<th>SST-5 Acc.</th>
<th>RACE-m Acc.</th>
<th>RACE-h Acc.</th>
<th>BoolQ Acc.</th>
<th>RTE Acc.</th>
<th>CB F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>PT (410K)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T5-XXL</td>
<td>Vanilla PT</td>
<td>70.515.5</td>
<td>32.38.3</td>
<td>34.78.2</td>
<td>31.63.5</td>
<td>61.05.3</td>
<td>53.53.5</td>
<td>50.74.1</td>
</tr>
<tr>
<td>Hybrid PT</td>
<td>87.66.6</td>
<td>40.92.7</td>
<td>53.58.2</td>
<td>44.26.4</td>
<td>79.81.5</td>
<td>56.82.6</td>
<td>66.57.2</td>
<td></td>
</tr>
<tr>
<td>LM Adaptation</td>
<td>77.67.5</td>
<td>36.23.6</td>
<td>27.30.2</td>
<td>26.50.4</td>
<td>62.00.3</td>
<td>55.31.0</td>
<td>61.21.7</td>
<td></td>
</tr>
<tr>
<td>PPT</td>
<td>93.50.3</td>
<td>50.20.7</td>
<td>60.01.2</td>
<td>53.00.4</td>
<td>66.45.7</td>
<td>58.91.6</td>
<td>71.26.2</td>
<td></td>
</tr>
<tr>
<td>Hybrid PPT</td>
<td>93.80.1</td>
<td>50.10.5</td>
<td>62.50.9</td>
<td>52.20.7</td>
<td>82.01.0</td>
<td>59.83.2</td>
<td>73.27.0</td>
<td></td>
</tr>
<tr>
<td>Unified PPT</td>
<td>94.40.3</td>
<td>46.01.3</td>
<td>58.00.9</td>
<td>49.91.3</td>
<td>76.02.7</td>
<td>65.82.1</td>
<td>82.25.4</td>
<td></td>
</tr>
</tbody>
</table>

A Brief Comparison

Performance comparisons among different strategies

<table>
<thead>
<tr>
<th>Model Parameters</th>
<th>Tuning</th>
<th>Full data</th>
<th>Few-shot Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tune</td>
<td>Classifier</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Tune</td>
<td>Prompts</td>
<td>≈</td>
<td>↑</td>
</tr>
<tr>
<td>Fix</td>
<td>Classifier</td>
<td>↓</td>
<td>↓</td>
</tr>
<tr>
<td>Fix</td>
<td>Prompts</td>
<td>≈ (Big Model)</td>
<td>≈ (Big Model with PPT)</td>
</tr>
</tbody>
</table>
Knowledgeable Prompt Tuning

• Combine prompts (model knowledge) with human prior knowledge
• Use logic rules to enhance prompt tuning to downstream classification tasks

Han et al. PTR: Prompt Tuning with Rules for Text Classification. 2021.
Knowledgeable Prompt Tuning

- Incorporate knowledge base into verbalizer design in prompt tuning

Cross-modal Prompt Tuning

- Cross-model prompts: use prompt-learning in computer vision

CPT: Colorful Prompt Tuning for Pre-trained Vision-Language Models. 2021
Application: Information Extraction

- 60~80-way classification for fine-grained entity typing

[CLS] London is one of the biggest cities in the world. [London is a [MASK]. [SEP]
Application: Information Extraction

- 60~80-way classification for fine-grained entity typing

<table>
<thead>
<tr>
<th>Shot</th>
<th>Metric</th>
<th>Few-NERD</th>
<th>OntoNotes</th>
<th>BBN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Fine-tuning</td>
<td>PLET</td>
<td>Fine-tuning</td>
</tr>
<tr>
<td>1</td>
<td>Acc</td>
<td>8.94</td>
<td>43.87 (+34.93)</td>
<td>3.70</td>
</tr>
<tr>
<td></td>
<td>MiF</td>
<td>19.85</td>
<td>60.60 (+45.75)</td>
<td>18.98</td>
</tr>
<tr>
<td></td>
<td>MaF</td>
<td>19.85</td>
<td>60.60 (+40.75)</td>
<td>19.43</td>
</tr>
<tr>
<td>2</td>
<td>Acc</td>
<td>20.83</td>
<td>47.78 (+26.95)</td>
<td>7.27</td>
</tr>
<tr>
<td></td>
<td>MiF</td>
<td>32.67</td>
<td>62.09 (+29.42)</td>
<td>24.89</td>
</tr>
<tr>
<td></td>
<td>MaF</td>
<td>32.67</td>
<td>62.09 (+29.42)</td>
<td>25.64</td>
</tr>
<tr>
<td>4</td>
<td>Acc</td>
<td>33.09</td>
<td>57.00 (+23.91)</td>
<td>11.15</td>
</tr>
<tr>
<td></td>
<td>MiF</td>
<td>44.14</td>
<td>68.61 (+24.47)</td>
<td>27.69</td>
</tr>
<tr>
<td></td>
<td>MaF</td>
<td>44.14</td>
<td>68.61 (+24.47)</td>
<td>28.26</td>
</tr>
<tr>
<td>8</td>
<td>Acc</td>
<td>46.44</td>
<td>55.75 (+9.31)</td>
<td>18.37</td>
</tr>
<tr>
<td></td>
<td>MiF</td>
<td>57.76</td>
<td>68.74 (+10.98)</td>
<td>38.16</td>
</tr>
<tr>
<td></td>
<td>MaF</td>
<td>57.76</td>
<td>68.74 (+10.98)</td>
<td>37.77</td>
</tr>
<tr>
<td>16</td>
<td>Acc</td>
<td>60.98</td>
<td>61.58 (+0.60)</td>
<td>32.26</td>
</tr>
<tr>
<td></td>
<td>MiF</td>
<td>71.59</td>
<td>72.39 (+0.80)</td>
<td>51.40</td>
</tr>
<tr>
<td></td>
<td>MaF</td>
<td>71.59</td>
<td>72.39 (+0.80)</td>
<td>51.45</td>
</tr>
</tbody>
</table>
Analysis: Effectiveness in few-shot learning

• How many data points is a prompt worth?

• Using 50 examples with prompts is comparable with 200 data points
Analysis: Stability

- Templates have huge impact, and different templates mean different context for [MASK]
- Human-defined, automatically generated, randomly initialized...

Prompt-learning could be unstable for different templates

Zero-shot entity typing. With appropriate templates, the performance is promising.
Implementation Issues for Prompt-learning

• Prompt-learning is a synthesis of pre-trained tasks, deep models, human prior knowledge and current tasks

• The implementation may face problems
  • What? What model? What template? Hard or soft? What verbalizer?
  • When? When to insert the template?
  • Where? Where to insert the template?
  • How? How to generate templates and verbalizers?
OpenPrompt: A Prompt-learning Programming Framework

https://github.com/thunlp/OpenPrompt
Remaining Challenges

• Converge speed of prompt-tuning for super large models
• The convergence speed is still very slow
• Fast estimation for prompt-tuning
Remaining Challenges

• Only tune prompts, adapters, or biases. Are they all the same?
• Additional parameters in different pattern: contexts, MLPs, matrices...
• Assumption: They are just switches for knowledge distributed in PLMs
Remaining Challenges

• Still vanilla pre-training?
• Pursue the grand unity for pre-training and model tuning
• A central model with toolkit can do all the things
Summary

• Knowledge is the key to deep understanding of human languages
• Knowledge can be represented in appropriate ways: symbol vs. model
• Big PLMs are the most advanced approach to model knowledge
• Big PLMs do capture knowledge from plain text including commonsense
• The challenge is to stimulate and stabilize model knowledge in PLMs
• **Prompt Tuning** seems a promising approach to stimulate model knowledge for NLP
• Prompt Tuning is friendly to deploy big PLMs in applications, one PLM vs. thousands of prompts and applications
Future Work

Knowledgeable Learning for NLP

- Knowledge Acquisition
- Knowledge Representation
- Knowledge Guided NLP

Existing Works

- Knowledge Too Coarse
- Organization Too Loose
- Fusion Too Heuristic

Background

- Fine-Grained Knowledge
- Hierarchical Organization
- Unified Knowledge Modeling

Approach

Knowledgeable and Interpretable NLP
Open Source

- Packages for representation and acquisition of linguistic and world knowledge
- The projects obtain 40000+ stars on GitHub

https://github.com/thunlp
BMInf - https://github.com/OpenBMB

- Low-cost Inference Package for Big Pretrained Language Models (PLMs)

<table>
<thead>
<tr>
<th>Implementation</th>
<th>GPU</th>
<th>Encoder Speed (tokens/s)</th>
<th>Decoder Speed (tokens/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMInf</td>
<td>NVIDIA GeForce GTX 1060</td>
<td>533</td>
<td>1.6</td>
</tr>
<tr>
<td>BMInf</td>
<td>NVIDIA GeForce GTX 1080Ti</td>
<td>1200</td>
<td>12</td>
</tr>
<tr>
<td>BMInf</td>
<td>NVIDIA GeForce GTX 2080Ti</td>
<td>2275</td>
<td>19</td>
</tr>
<tr>
<td>BMInf</td>
<td>NVIDIA Tesla V100</td>
<td>2966</td>
<td>20</td>
</tr>
<tr>
<td>BMInf</td>
<td>NVIDIA Tesla A100</td>
<td>4365</td>
<td>26</td>
</tr>
<tr>
<td>PyTorch</td>
<td>NVIDIA Tesla V100</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>PyTorch</td>
<td>NVIDIA Tesla A100</td>
<td>-</td>
<td>7</td>
</tr>
</tbody>
</table>
## Resource: **Chinese Pre-Trained Models (CPM)**

<table>
<thead>
<tr>
<th>训练数据</th>
<th>模型大小</th>
<th>任务</th>
</tr>
</thead>
<tbody>
<tr>
<td>新闻</td>
<td>参数量</td>
<td>文本分类</td>
</tr>
<tr>
<td>参数量</td>
<td>109M</td>
<td>12</td>
</tr>
<tr>
<td>层数</td>
<td>334M</td>
<td>24</td>
</tr>
<tr>
<td>2.6B</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>百科</td>
<td>层数</td>
<td>自然语言推理</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>1024</td>
</tr>
<tr>
<td>对话</td>
<td>隐向量维度</td>
<td>阅读理解</td>
</tr>
<tr>
<td></td>
<td>768</td>
<td>64</td>
</tr>
<tr>
<td>网页</td>
<td>每层注意力数</td>
<td>完形填空</td>
</tr>
<tr>
<td>每层注意力数</td>
<td>1,024</td>
<td>64</td>
</tr>
<tr>
<td>2,560</td>
<td>64</td>
<td></td>
</tr>
<tr>
<td>故事</td>
<td>注意力向量维度</td>
<td>对话生成</td>
</tr>
<tr>
<td>注意力向量维度</td>
<td>12</td>
<td>80</td>
</tr>
<tr>
<td>16</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>实体生成</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


**CPM: A Large-scale Generative Chinese Pre-trained Language Model**

**Authors:** Zhengyan Zhang, Xu Han, Hao Zhou, Pei Ke, Yuxian Gu, Deming Ye, Yujia Qin, Yusheng Su, Haozhe Ji, Jian Guan, Fanchao Qi, Xiaozhi Wang, Yanan Zheng, Guoyang Zeng, Huanqi Cao, Shengqi Chen, Daixuan Li, Zhenbo Sun, Zhiyuan Liu, Minlie Huang, Wentao Han, Jie Tang, Juanzi Li, Xiaoyan Zhu, Mao Song Sun

**Abstract:** As the training corpus of GPT-3 is primarily English, and the parameters are not publicly available. In this technical report, we release the Chinese Pre-trained Language Model (CPM) with generative pre-training on large-scale Chinese training data. To the best of our knowledge,... More

Submitted 1 December, 2020; originally announced December 2020.
CPM-2: Large-scale Cost-effective Pre-trained Language Models

<table>
<thead>
<tr>
<th>训练数据</th>
<th>模型架构</th>
<th>能力评测</th>
</tr>
</thead>
<tbody>
<tr>
<td>电子书</td>
<td></td>
<td>识记</td>
</tr>
<tr>
<td>百科</td>
<td>层数</td>
<td>阅读</td>
</tr>
<tr>
<td>问答</td>
<td>隐向量维度</td>
<td>分类</td>
</tr>
<tr>
<td>科学文献</td>
<td>每层注意力数</td>
<td>计算</td>
</tr>
<tr>
<td>小说</td>
<td>注意力向量维度</td>
<td>跨语</td>
</tr>
<tr>
<td>50TB文本数据</td>
<td>110亿模型参数</td>
<td>生成</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>概括</td>
</tr>
<tr>
<td>MoE</td>
<td>4,096</td>
<td></td>
</tr>
<tr>
<td>1980亿模型参数</td>
<td>64</td>
<td></td>
</tr>
<tr>
<td>Expert 数目</td>
<td>32</td>
<td></td>
</tr>
</tbody>
</table>

CPM-3: A Large-Scale Continual Pre-Trained Language Model
CPM-2: Large-Scale Cost-Effective Pre-Trained Language Models
CPM-1: A Large-Scale Chinese Pre-Trained Language Model
Thanks!

liuzy@tsinghua.edu.cn

http://nlp.csai.tsinghua.edu.cn/~lzy