PROP: Pre-training with Representative Words

Prediction for Ad-hoc Retrieval

Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Xiang Ji and Xueqi Cheng

1. CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences
2. University of Chinese Academy of Sciences
New Paradigm of NLP

- Pre-training and then fine-tuning paradigm
- Significant benefit for tasks with limited training data

NLP Tasks
- Machine Translation
- Sentiment Analysis
- Question Answering
- Dialogue & Chatbot
- Textual Entailment
- Paraphrasing
- Semantic Parsing
- ...
• BERT: Bidirectional Encoder Representations from Transformers

- Pre-trained with mask language model and next sentence prediction on Wikipedia and BookCorpus.

- A comparison of BERT with previous SOTA on GLUE, SQUAD 1.1, SQUAD 2.0, from Devlin et.al.

• BERT outperform previous SOTA on many natural language understanding tasks.
Directly applying BERT to IR

- Usage of BERT for IR. Concatenate query and document, take [CLS] for relevance computation

- Pre-trained models also benefit the search tasks, but not very significant

A comparison of BERT with BM25 and previous SOTA on downstream IR tasks.
Observation

• Pegasus for Abstractive Summarization
• SSPT for Question Answering
• SentiLARE for sentiment analysis
• ERNIE (THU) for entity-related tasks
• ...

The pre-training objective that more closely resembles the downstream tasks leads to better and faster fine-tuning performance.
Gap sentence generation (GSG): selected by ROUGE scores

PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, ICML, 2020
**SSPT for Question Answering**

- **Span Selection PreTraining (SSPT):** predict masked span (noun phrase or entity, pseudo answer), jointly pre-training with MLM

- Span selection training instance generation. Masked span will be predicted by the passage containing it.

- BERT+SSPT is significantly better than original BERT

- A comparison of BERT+SSPT with BERT on SQUAD 1.1, SQUAD 2.0, HotpotQA and Natural Questions.
However, pre-training objectives tailored for ad-hoc retrieval have not been well explored.
Revisit the Pre-training Objectives

Sequence-based tasks:
- Masked Language Modeling
- Permutated Language Modeling

Learn contextual representations

Sequence pair-based tasks:
- Next Sentence Prediction
- Sentence Order Prediction

Learn inter-sequence coherence

IR requirements

Good representations for the query and the document

Relevance matching between short queries and long documents
- sentence-pair vs. query-document
- coherence vs. relevance
Pre-training for Passage Retrieval in openQA

• Design three pre-training tasks that resemble the relevance relationship between natural language questions and answer passages

- Inverse Cloze Task (ICT)
- Body First Selection (BFS)
- Wiki Link Prediction (WLP)

- Natural language questions-answer passages vs. short queries-long documents
- Depend on document structure, e.g., WLP
- Marginal benefit for ad-hoc retrieval
Design a novel pre-training objective tailored for IR, which more closely resembles the relevance relationship between query and document.
The user has a reasonable idea of the terms that are likely to appear in the “ideal” document that can satisfy his/her information need.

The query is generated as the piece of text representative of the “ideal” document.
Query likelihood scoring function derived by the Bayesian theorem

$$P(D|Q) \propto P(Q|\theta_D)P(D) \propto P(Q|\theta_D)$$

- Query generation probability
- Uniform distribution

Smoothing methods for zero probability problem
- E.g., Jelinek-Mercer, Dirichlet prior, Absolute discounting
- Query likelihood with Dirichlet smoothing is one of the most effective method (Zhai et.al. 2001)

$$P(q_i|\theta_D) = \frac{c(w,D)}{|D|} \Rightarrow \frac{c(w,D)+\mu P(w|C)}{|D|+\mu}$$, \(\mu\) is smoothing parameter, \(P(w|C)\) is collection language model
Pre-training Task for Ad-hoc Retrieval: ROP

- Representative words prediction (ROP) task
  - Given a document, sample word sets according to the document language model
  - The word set with higher likelihood is deemed as more “representative” of the document
  - Pre-train the Transformer model to predict the representativeness

From https://en.wikipedia.org/wiki/Information_retrieval
Representative Word Sets Sampling

1. Given document $d$, initialize document language model with Dirichlet smoothing $\theta_d$

2. Choose length $l \sim \text{Poisson}(\lambda)$

3. **Paired Sampling**: Sample N pairs of word sets for each document where $w_i \sim P(w_i|\theta_d)$
   - Why? Likelihood comparable

4. Higher likelihood deemed as more representative
Pre-training with the ROP task

- Pre-training Loss function

\[ \mathcal{L}_{ROP} = \max(0, 1 - P(S_1|D) + P(S_2|D)) \]

\[ \mathcal{L}_{MLM} = - \sum_{\tilde{x} \in X} \log p(\tilde{x}|X_{\tilde{x}}) \]
The ROP objective belongs to the category of model-based pre-training objective where the labels are produced by some automatic model rather than simple MASKs.

- **Electra** leverages a generative model to replace masked tokens
- **PEGASUS** leverages the ROUGE1-F1 score to select top-m sentences
- ......

**Pre-training**
- Only documents
- MLM + ROP
- A variety of retrieval tasks

**Weak supervision**
- Query and document, label is missing
- Same as final ranking objective
- Designed for each retrieval task

**VS.**
- What data is available?
- Learning objective
- Scope of application
Experiment Setting

• Pretraining datasets:
  • Wikipedia, over 10 million documents
  • MS MARCO, about 3.4 million documents

• 5 downstream ad-hoc retrieval tasks:
  • Robust04, ClueWeb09-B, Gov2, MQ2007, MQ2008

• Baseline models:
  • Traditional retrieval models: BM25, QL
  • Previous state-of-the-art neural ranking models on each dataset: BERT-MaxP, HiNT et.al.
  • Other pretraining method: BERT, Transformer_{ICT}
Experiments – Main Results

Table 2: Comparisons between PROP and the baselines. *, † and ‡ indicate statistically significance with $p-value \leq 0.05$ over BM25, BERT and Transformer$_{ICT}$, respectively.

<table>
<thead>
<tr>
<th>Model</th>
<th>Robust04</th>
<th>ClueWeb09-B</th>
<th>Gov2</th>
<th>MQ2007</th>
<th>MQ2008</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>nDCG@20</td>
<td>P@20</td>
<td>nDCG@20</td>
<td>P@20</td>
<td>nDCG@20</td>
</tr>
<tr>
<td>QL</td>
<td>0.413</td>
<td>0.367</td>
<td>0.225</td>
<td>0.326</td>
<td>0.409</td>
</tr>
<tr>
<td>BM25</td>
<td>0.412</td>
<td>0.363</td>
<td>0.230</td>
<td>0.334</td>
<td>0.421</td>
</tr>
<tr>
<td>Previous SOTA</td>
<td><strong>0.538</strong></td>
<td><strong>0.467</strong></td>
<td>0.296</td>
<td>-</td>
<td>0.422</td>
</tr>
<tr>
<td>BERT</td>
<td>0.459*</td>
<td>0.389*</td>
<td>0.295*</td>
<td>0.367*</td>
<td>0.495*</td>
</tr>
<tr>
<td>Transformer$_{ICT}$</td>
<td>0.460*</td>
<td>0.388*</td>
<td>0.298*</td>
<td>0.369*</td>
<td>0.499†</td>
</tr>
<tr>
<td>PROP$_{Wikipedia}$</td>
<td><strong>0.502††</strong></td>
<td><strong>0.421††</strong></td>
<td><strong>0.316††</strong></td>
<td><strong>0.384††</strong></td>
<td><strong>0.519††</strong></td>
</tr>
<tr>
<td>PROP$_{MSMARCO}$</td>
<td><strong>0.484††</strong></td>
<td><strong>0.408††</strong></td>
<td><strong>0.329††</strong></td>
<td><strong>0.391††</strong></td>
<td><strong>0.525††</strong></td>
</tr>
</tbody>
</table>

1. PROP significantly outperforms previous SOTA in 4 of 5 tasks (8.9%, 24.4%, 6.7% and 9% in terms of NDCG@20), except for the Robust04 (BERT + Neu-IR ensemble).
2. PROP is significantly better than BERT and Transformer$_{ICT}$.
3. Pre-training in related domain corpus is more effective.
Experiments – Impact of Pre-training Objectives

Table 3: Impact of pre-training objectives. † indicates statistically significance with $p-value < 0.05$.

<table>
<thead>
<tr>
<th></th>
<th>nDCG@20</th>
<th>nDCG@10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Robust04 ClueWeb09-B Gov2</td>
<td>MQ2007 MQ2008</td>
</tr>
<tr>
<td>w/ MLM</td>
<td>0.467</td>
<td>0.306</td>
</tr>
<tr>
<td>w/ ROP</td>
<td>0.481†</td>
<td>0.321†</td>
</tr>
<tr>
<td>w/ ROP+MLM</td>
<td><strong>0.484†</strong></td>
<td><strong>0.329†</strong></td>
</tr>
</tbody>
</table>

1. Pretraining with ROP achieves significant improvements over MLM.
2. MLM and ROP are both helpful for downstream tasks.
Experiments – Impact of Sampling Strategies

• docLM–based vs. Random sampling

Table 5: Impact of Different Sampling Strategies. Two-tailed t-tests demonstrate the improvements of document language model-based sampling to the random sampling strategy are statistically significant († indicates p-value < 0.05).

<table>
<thead>
<tr>
<th></th>
<th>nDCG@20</th>
<th></th>
<th>nDCG@10</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Robust04</td>
<td>ClueWeb09-B</td>
<td>Gov2</td>
<td>MQ2007</td>
</tr>
<tr>
<td>Random</td>
<td>0.471</td>
<td>0.304</td>
<td>0.505</td>
<td>0.513</td>
</tr>
<tr>
<td>docLM-based</td>
<td>0.493†</td>
<td>0.317†</td>
<td>0.517†</td>
<td>0.516†</td>
</tr>
</tbody>
</table>

1. docLM-based sampling converges faster and leads to better performance.
2. docLM-based sampling strategy is a more suitable way than the random sampling strategy to generate representative word sets for a document.
1. PROP fine-tuned on limited supervised data can achieve comparable performance with BERT fine-tuned on the full supervised datasets, e.g., 30 queries on Robust04.

2. Under the zero–shot setting, PROP also achieves exciting performance
   • On Gov2, PROP beats BM25 in terms of nDCG@20, and achieves about 90% performance of BERT fine-tuned on the full dataset
Conclusion & Future Work

• Conclusion
  • We proposed PROP, a new pre-training method tailored for ad-hoc retrieval
  • PROP achieved significant improvements over the baselines without pre-training or with other pre-training methods
  • PROP can achieve strong performance under both the zero- and low-resource IR settings

• Future work
  • Go beyond the ad-hoc retrieval, and test the ability of PROP over other downstream IR tasks, such as passage retrieval in QA or response retrieval in dialog systems
  • Investigate new ways to further enhance the pre-training objective tailored for IR
Code and the pre-training models are available at:
https://github.com/Albert-Ma/PROP

Thanks!

Xinyu Ma
maxinyu17g@ict.ac.cn