Data-efficient and Explainable Ranking with BERT models

Leo (Leonid) Boytsov

Bosch Center for Artificial Intelligence
Fall 2021

Meta-motivation: Ranking Transformers have issues!

- Robustness to distribution shift
- Lack of interpretability (attention is not an explanation)
- Data efficiency (few-shot is underexplored)
- No you cannot use MS MARCO in production (🤗😢)!
- Restricted document length

Talk outline

- Recap of ranking with Transformers
- Data-efficient ranking with BERT-models
- Interpretable ranking layer via neural IBM Model 1

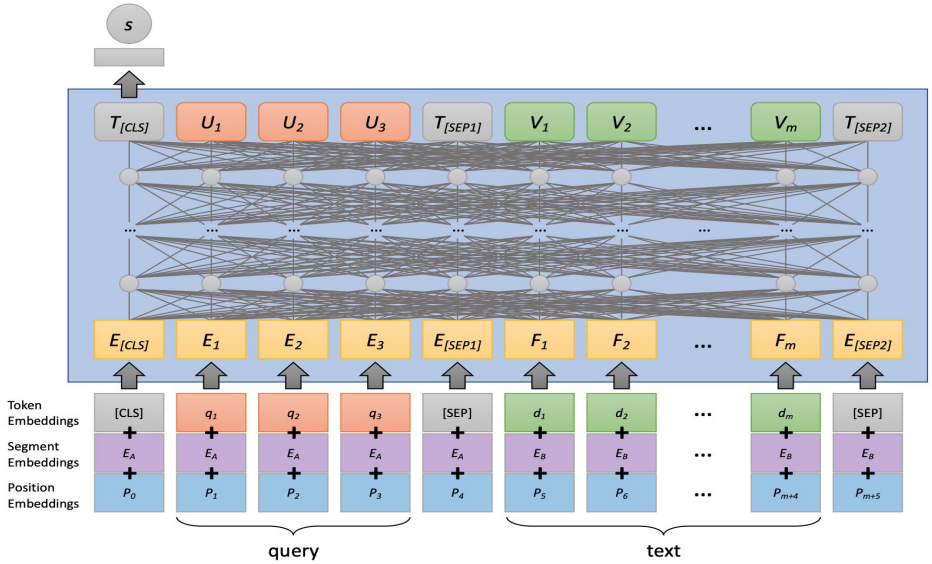
PART 1
Ranking with Transformers (warm-up)

BERT rankers and beyond

Ranking with Transformers (h/t R. Nogueira)

- BERT (decoder-only) and the family: masked LM
- T5 (encoder-decoder) multi-task sequence-to-sequence

BERT-style ranking model



Pretrained Transformers for Text Ranking: BERT and Beyond. J. Lin, R. Nogueira, A. Yates.

BERT-style ranking model: Useful Simplification

- The BERT encoder consume a sequence of sub-word tokens and produces contextualized token vectors
- Last layer provides contextualized token embeddings
- [CLS] is always present
- Prediction head uses [CLS] vector to produce a score

Three types of training objectives

- Self-supervised pre-training (e.g., masked LM)
- Fine-tuning
  - Full supervision
  - Weak supervision

PART 2:
A Systematic Evaluation of Transfer Learning and Pseudo-labeling with BERT-based Ranking Models

Iurii Mokrii*, Leonid Boytsov*, Pavel Braslavski

*) equal contribution

Motivation

- Training a competitive IR model from scratch may require a lot of costly human annotations
- Few-shot learning is a reasonable option in such cases, but most work focuses on transfer to datasets with small query sets.
- Transfer learning is often impossible due to licensing issues.
- Pseudo-labeling combined with a subsequent training on a small number of human annotated queries may provide a solution.

Key features of our study

- BERT-based neural re-ranking
- Diverse data-sets (lots of queries!)
- Zero-shot, few-shot, and full-shot evaluation
- Comparison against BM25 pseudo-labeling

Datasets

Dataset	#queries train	#docs	#relev. /query	#tok. /query	#tok. /doc
Yahoo! Answers	100K	819.6K	5.7	11.9	63
MS MARCO doc	357K	3.2M	1	3.2	1197
MS MARCO pass	788.7K	8.8M	0.7	3.5	75
DPR NQ	53.9K	21M	7.9	4.5	141
DPR SQuAD	73.7K	21M	4.8	5	141

Notes: Development sets have 5K queries, test sets have 1.5K queries. Text length is the # of **BERT** word pieces.

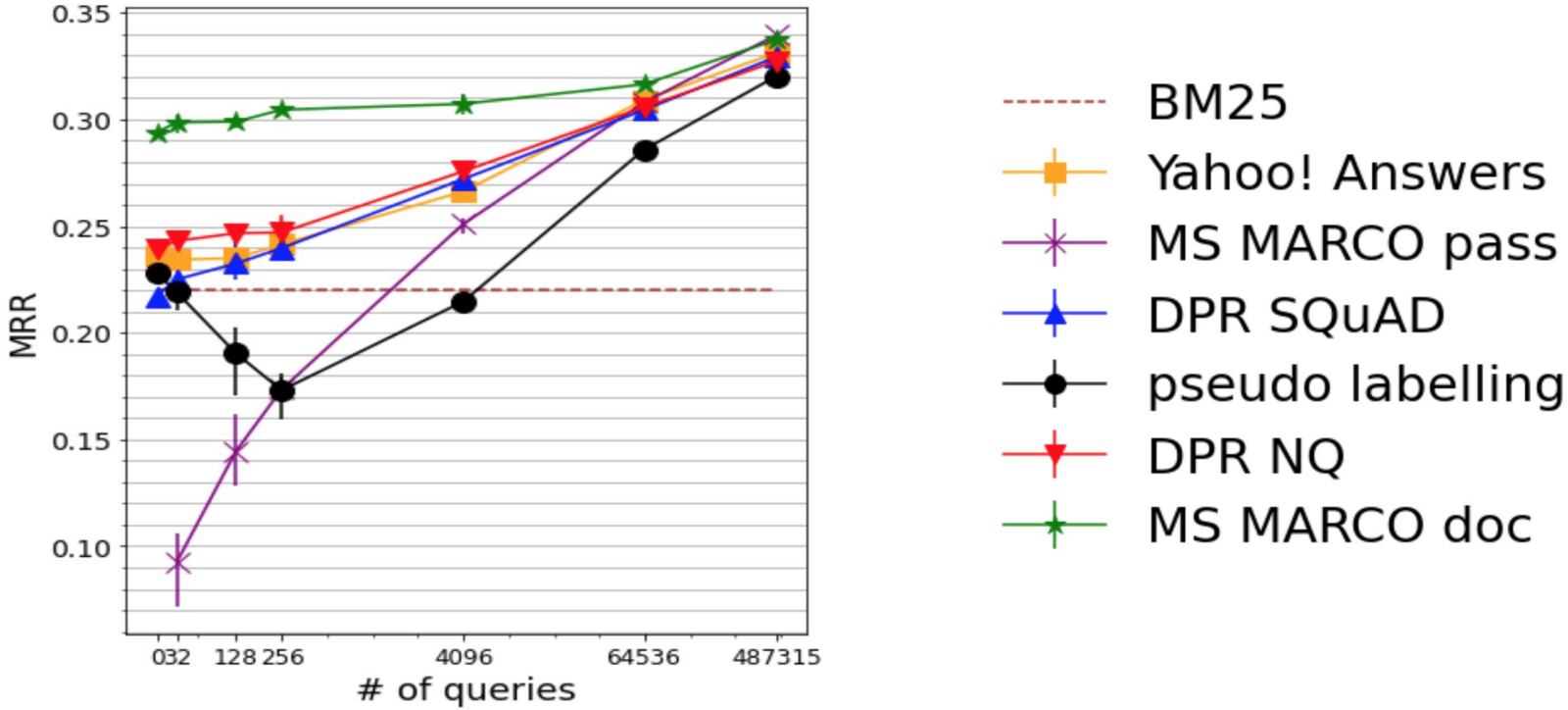
Training and Evaluation Modes Taxonomy

1. Pre-training mode
    a. no pre-training
    b. **different**-collection training data
    c. **same**-collection pseudo-labels
2. Fine-tuning mode
    a. zero-shot
    b. few-shot
    c. full-shot

Research Questions

- When training from scratch, how much data does a BERT-based ranker need to outperform BM25?
- Does a model trained on pseudo-labels outperform BM25 (and by how much)?
- Is transfer learning always more effective than BM25?
- Is transfer learning more effective than training on pseudo-labels?
- Can we improve upon transfer learning and/or pseudo-labeling with a few training examples?

Target Collection: MS MARCO Passage



Summary (part 1)

- Training on pseudo-labels with few-shot fine-tuning is a promising method
- Future work:
  - Improving few-shot fine-tuning
  - More diverse synthetic data

PART 3:
Exploring Classic and Neural Lexical Translation Models for Information Retrieval: Interpretability, Effectiveness, and Efficiency Benefits

Leonid Boytsov, Zico Kolter

Motivation

- Retrieval systems suffer from query-document term mismatch
- Translation models offer a simple solution to reduce mismatch
- Translation models have been around for 20 years
- However, they are underused and underexplored

Lexical Translation Model (IBM Model 1)

$$P(Q|D) = \prod_{q \in Q} \sum_{d \in D} T(q|d) P(d|D)$$

Berger & Lafferty, 1999

Reasons to care about lexical translation models

- Simplicity
- Interpretability
- Sparsity (efficiency on CPU)
- It can be learned by neural networks

Problems with a classic IBM Model 1

- Non-parametric model (rare words, poor generalizability)
- Paired queries and documents need to have similar lengths
- Trained via EM in a translation (not ranking) task

Solutions for classic IBM Model 1

- Use short document sections (e.g., title, url)
- Split long documents into chunks
- Subword tokens (BERT word pieces)
- Neural parametrization of translation probabilities *T(q|d)*
- Training in a ranking task

Research Questions

- Can we train **classic** Model 1 when documents are much longer than queries?

Answers: Not really, but we can train on metadata.

Training a traditional Model 1 on meta-data

Thread

Jimmy Lin
@lintool

Use of IBM Model 1 translation probs learned from MS MARCO to improve ranking by @srchvrs is brilliant and insightful! searchivarius.org/blog/tradition...

10:47 AM · Dec 16, 2020 · Twitter Web App

6 Retweets **4** Quote Tweets **63** Likes

Last-Layer Interpretability (CEDR inspired)

$$P(Q|D) = \prod_{q \in Q} \sum_{d \in D} T(q|d) P(d|D)$$

Berger & Lafferty, 1999

Research Question 2

Question: Is the neural Model 1 layer effective at aggregating BERT embeddings

Answers:

1. There is no performance degradation
2. A small boost for long documents (compared to BERT FirstP).

Last-Layer Interpretability

- Saliency maps do not tell *how* model processes salient input
- Extraneous explanations cannot be trusted
- Models need to be transparent *by design*
- Last-layer transparency is better than none (Rudin 2019) especially if it is "free".
- In 2020, our partially-interpretable BERT-Model1 topped the MS MARCO leaderboard.

Patent warning

Neural Model 1 (trained in ranking mode) can be used in research, but commercial use may be restricted (patent submitted)

Software: FlexNeuART (h/t Sean MacAvaney)

1. Dense, sparse, or dense-sparse retrieval using Lucene & NMSLIB
2. SOTA traditional and neural models
3. AI2 IR datasets
4. Multi-field multi-level forward indices (+parent-child field relations)
5. Python API for retrievers and rankers as well as to access indexed data
6. Basic experimentation framework (+LETOR)

Thank you for attention! Questions?

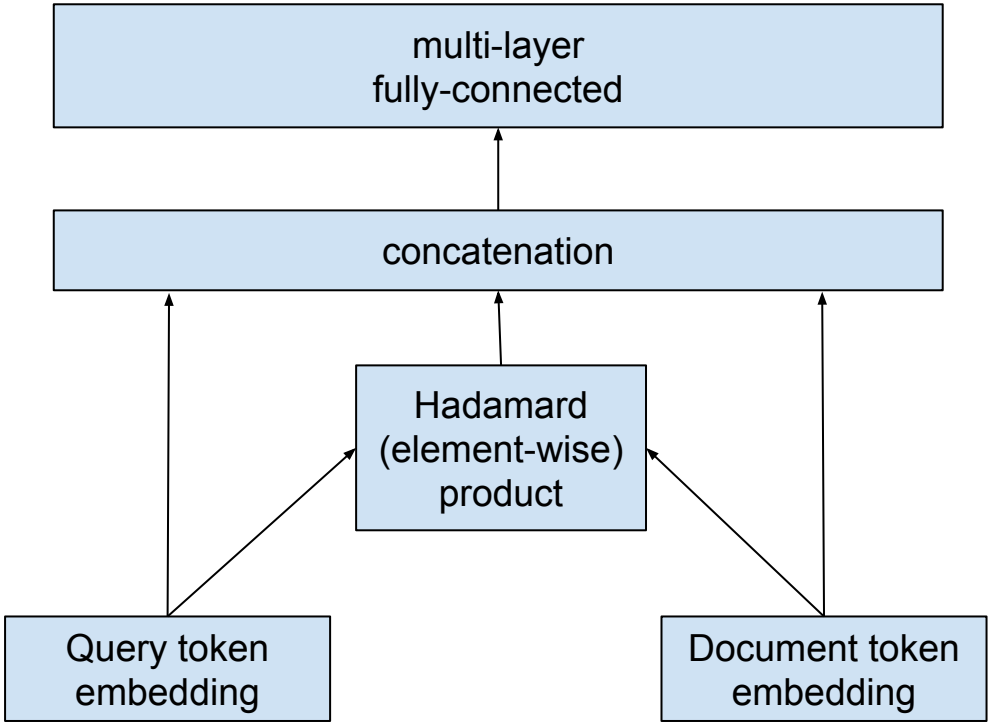
Papers covered in this talk:

- Mokrii, Boytsov, Braslavski. A Systematic Evaluation of Transfer Learning and Pseudo-labeling with BERT-based Ranking Models. SIGIR 2021.
- Boytsov & Kolter, Exploring Classic and Neural Lexical Translation Models for Information Retrieval: Interpretability, Effectiveness, and Efficiency Benefits (blog post). ECIR 2021
- Boytsov. Traditional IR rivals neural models on the MS MARCO Document Ranking Leaderboard, arxiv (blog post)

Appendix

some backup slides

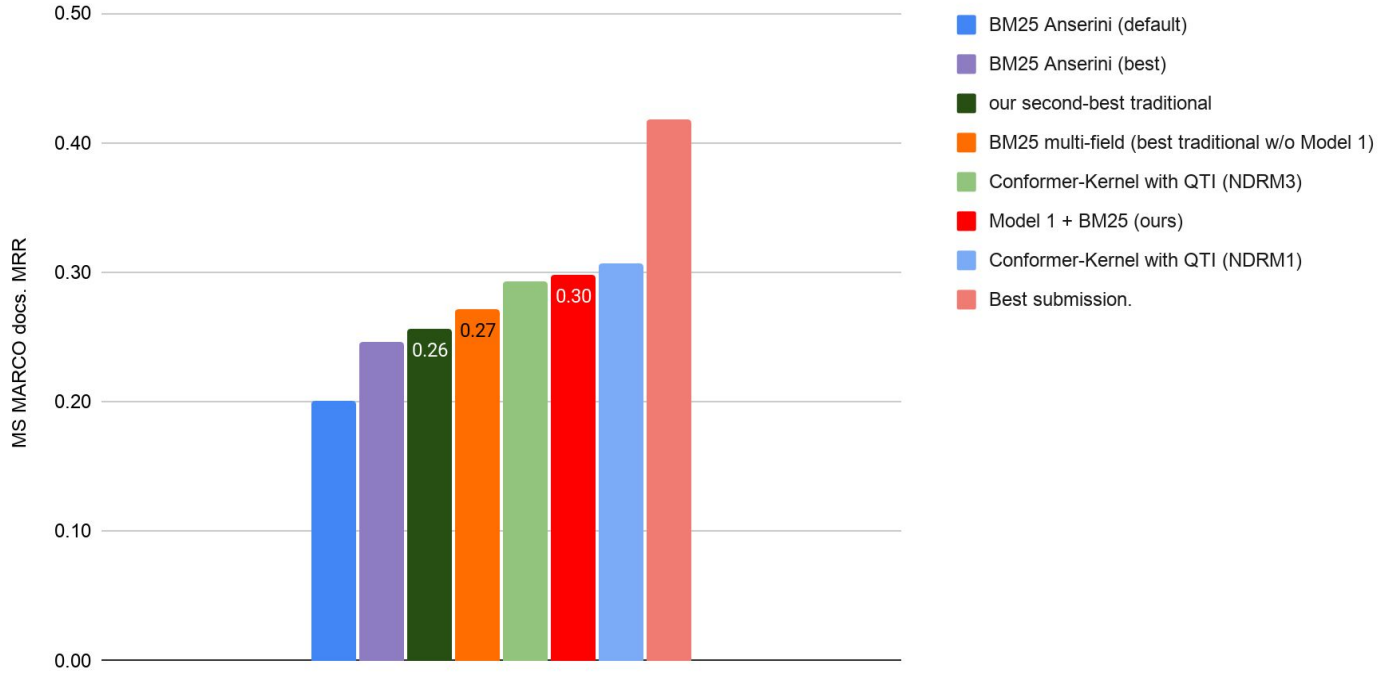
Neural parametrization of probabilities *P(q|d)*



Findings

- *Interpretable* neural Model 1 layer entails virtually no loss in accuracy and efficiency compared to vanilla BERT or CEDR
- In fact, it can be slightly more effective compared to short-document models
- If documents are long, classic Model 1 is marginally useful, but neural Model 1 (trained end-to-end) performs much better
- In contrast, when short documents are available, the classic Model 1 is still effective (follow-up paper)
- Context-free neural Model 1 can be sparsified and run efficiently on CPU (without expensive index- or query-time processing)

Classic Model 1 on MS MARCO Leaderboard



MS MARCO documents (dev subset)