All we need is a score for every query–document pair
- We’ll sort the results by decreasing score
Query–Document Interaction Models

1. Tokenize the query and the document
2. Embed all the tokens of each
3. Build a query–document interaction matrix
   - Most commonly: store the cos similarity of each pair of words
4. Reduce this dense matrix to a score
   - Learn neural layers (e.g., convolution, linear layers)

Models in this category include KNRM, Conv-KNRM, and Duet.

Chenyan Xiong, et al. End-to-end neural ad-hoc ranking with kernel pooling. SIGIR’17
Bhaskar Mitra, et al. Learning to match using local and distributed representations of text for web search. WWW’17
Query–Document Interaction Models: MS MARCO Results

- Considerable gains in **quality**—at a reasonable increase in computational cost!

These models re-rank the top-1000 passages retrieved by BM25.
All-to-all Interaction with BERT

2. Run this through all the BERT layers
3. Extract the final [CLS] output embedding
   - Reduce to a single score through a linear layer

This is essentially a standard BERT classifier, used for ranking passages.

Of course, we must fine-tune BERT for this task with positives and negatives to be effective.

Zhuyun Dai and Jamie Callan. 2019. Deeper Text Understanding for IR with Contextual Neural Language Modeling. SIGIR’19
BERT Rankers: SOTA 2019 (in quality)

<table>
<thead>
<tr>
<th>Rank</th>
<th>Model</th>
<th>Submission Date</th>
<th>MRR@10 On Eval</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BERT + Small Training Rodrigo Nogueira and Kyunghyun Cho - New York University</td>
<td>January 7th, 2019</td>
<td>35.87</td>
</tr>
<tr>
<td>2</td>
<td>IRNet (Deep CNN/IR Hybrid Network) Dave DeBarr, Navendu Jain, Robert Sim, Justin Wang, Ninupama Chandrasekaran – Microsoft</td>
<td>January 2nd, 2019</td>
<td>28.061</td>
</tr>
</tbody>
</table>

MS MARCO Ranking screenshot as of Jan 2019. From Rodrigo Nogueira’s Brief History of DL applied to IR (UoG talk).
https://blog.google/products/search/search-language-understanding-bert/
BERT Rankers: Efficiency–Effectiveness Tradeoff

- Dramatic gains in **quality**—but also a dramatic increase in **computational cost**!

Can we achieve high MRR **and** low latency?
Toward Faster Ranking: Pre-computation

- BERT rankers are slow because their computations be **redundant**:
  - Represent the query (1000 times for 1000 documents)
  - Represent the document (once for every query!)
  - Conduct matching between the query and the document

- We have the documents in advance.
  - Can we **pre-compute** the document representations?
  - And “cache” these representations for use across queries

Is there a unique value in jointly representing queries and documents?
Neural IR Paradigms: Learning term weights

- BM25 decomposed a document’s score into a summation over term–document weights. **Can we learn term weights with BERT?**

  - Tokenize the query/document
  - Use BERT to produce a score for each token in the document
  - Add the scores of the tokens that also appear in the query
  
  \[ t_1 \quad t_2 \quad t_3 \quad \ldots \quad t_{32} \]

  \[ t_{91} \quad t_2 \quad t_1 \quad \ldots \quad t_{32} \]

  ![Diagram](image)

  - Save term weights to the inverted index
  - Lookup term weights from inverted index
  - Compute sum of scores for the matching terms!

Dai, Zhuyun, and Jamie Callan. "Context-aware term weighting for first stage passage retrieval." SIGIR’20
Learning term weights

- We get to learn the term weights with BERT and to re-use them!
- But our query is back to being a “bag of words”.

DeepCT and doc2query are two major models under this paradigm.

Can we do better?
Next: Can we achieve high MRR \textit{and} low latency?

- Yes! We’ll discuss two rich neural IR paradigms:
  - Representation Similarity
  - Late Interaction
References

Omar Khattab and Matei Zaharia. "ColBERT: Efficient and effective passage search via contextualized late interaction over BERT." SIGIR’20

Chenyang Xiong, et al. End-to-end neural ad-hoc ranking with kernel pooling. SIGIR’17


Bhaskar Mitra, et al. Learning to match using local and distributed representations of text for web search. WWW’17


Zhuyun Dai and Jamie Callan. 2019. Deeper Text Understanding for IR with Contextual Neural Language Modeling. SIGIR’19

Rodrigo Nogueira. “A Brief History of Deep Learning applied to Information Retrieval” (UoG talk). Retrieved from https://docs.google.com/presentation/d/1_mlvmyev0pjdg0CcfbEWMaRRECE0jCdjD3b1tPPvcbk

Zhuyun Dai, and Jamie Callan. "Context-aware term weighting for first stage passage retrieval." SIGIR’20
