Retrieving Comparative Arguments using Deep Pre-trained Language Models and NLU

Viktoriia Chekalina, Alexander Panchenko

Skolkovo Institute of Science and Technology, Philips Innovations Lab RUS

23 sent 2020

Task Methodology

- Given a set of topics with comparative query, i.e. question like "What is better X or Y ?"
- For each topic:
  - retrieve documents from ClueWeb12[1] corpus by ChatNoir[2] search engine.
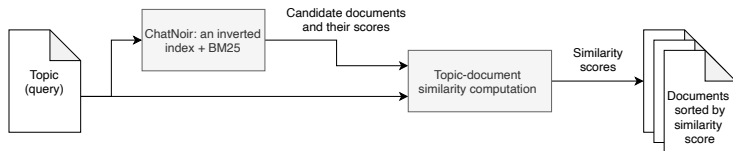 - find documents answering the proposed comparative question most fully and reasonably.

Methodology

- for each topic t_i for every retrieved document d_j count similarity measures $s_{ij} = sim(t_i, d_j)$
- rank documents according to these similarities

Methodology & Approaches

Methodology

Scheme of our methods: candidate documents are obtained using a traditional inverted index-based IR system and then re-ranked on the basis of topic-document similarity

Approaches

▶ Inverted index relevance estimation from ChatNoir system
▶ Approaches based on SOTA NLP models: LSTM, BERT, Transformers
▶ Comparative feature-based approaches: number of comparative structures in document and density of comparative sentences.

Approaches

ChatNoir build-in ranking

▶ ChatNoir has custom BM25 ranking function[3] based on inverted index.

▶ similarity $s_{ij} = cn_{ij}$, where cn_{ij} - scores provided by ChatNoir

LSTM-ULMFit

▶ we use pre-trained bi-directional LSTM language model LSTM-ULMFit

▶ similarity $s_{ij} = cos(h_i, h_j)$ where h_i, h_j are hidden states of model which processes topic title and document body, respectively.

---

[3]https://www.elastic.co/guide/en/elasticsearch/reference/current/index-modules-similarity.html

Approaches

## Bidirectional encoder representations from Transformer (BERT)

- BERT model provides word embeddings
- similarity $s_{ij} = cos(e_{topic\_title}, e_{doc\_title})$, where $e_{topic\_title}, e_{doc\_title}$ are averages between word embeddings in the topic's title and retrieved document's title, respectively.

Approaches

Attention of a Transformer-based language model

- ▶ We concatenate topic title and document body in one sequence
- ▶ We use the output of the self-attention layer in pre-trained Transformer model
- ▶ Base on obtained token-to-token responses we build the attention maps.
- ▶ In these maps, we select the area which describes responses of tokens from one sentence to another. For 3, 4, 10, 11 attention's head, we count the sum around this area, and the obtained sum is a similarity.

Approaches
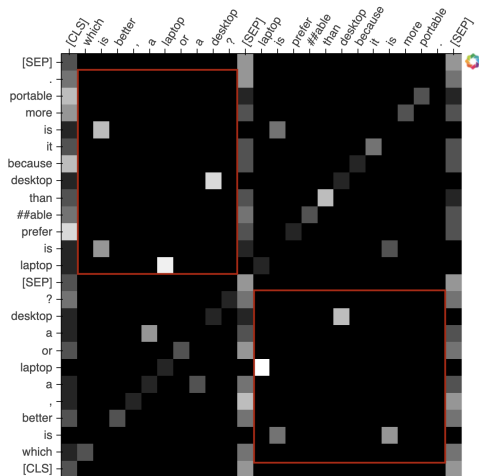
The attention map for input "[CLS]" + Which is better, a laptop or a desktop?+ "[SEP]" + Laptop is preferable than desktop because it is more portable. + "[SEP]". Areas highlighted in red participate in the similarity calculation.

Approaches

Comparative features

For every pair topic-document we count additive multiplicative term r_{ij} based on comparative features. This term describes the competitiveness of the document. Initialy $r_{ij} = 1$

- ▶ From topic we extract objects for comparison and find the same objects in document.
- ▶ Term updates as:

$$r_{ij} = \begin{cases} r_{ij} * 1.2 & \text{if we find one object} \\ r_{ij} * 1.5 & \text{if we find two objects} \end{cases}$$

- ▶ If we find in document l aspects(features of comparison) and predicates, we increase r_{ij} to l.

Approaches

Comparative features

▶ Using classifier of comparative sentences based on InferSent, count number of comparative sentences in the document

▶
$$r_{i,j} = r_{i,j} + n$$

, where n - number of comparative sentences.

Proposed term we add to ChatNoir's score and ULMFit's score, so similarity is $s_{ij} = cn_{ij} * r_{ij}$

Results

The evaluation is carried out manually and based on the existence of relevant information in the document.

Method	NDCG@5
ChatNoir's score based on an inverted index	**0.564**
Combination of ChatNoir's score and comparative features	0.553
Combination of ULMFiT and comparative features	0.464
Bidirectional Encoder Representations from Transformer (BERT)	0.404
Attention of a Transformer-based language model	0.223
Language model LSTM ULMFiT	0.200

The table describes 5@DCG(discount cumulative gain) scores for approaches mentioned above

Discussion

- ▶ We apply several approaches to the Argument retrieval shared task. We use baseline from a search engine, information from pre-trained NLP models, the existence of comparative structures in documents.
- ▶ The best scores gives a model based on inverted indices.
- ▶ Model with consideration of comparative structures gives the better result than one's base on only SOTA NLP models.