



Exploring BERT Synonyms and Quality Prediction for Argument Retrieval

Tommaso Green
Luca Moroldo
Alberto Valente

Search Engines - Team Yeagerists
1st June 2021





Task: Argument Retrieval

Respond to user queries with a set of relevant and high quality arguments.

Argument: “A conclusion (claim) supported by premises (reasons)”

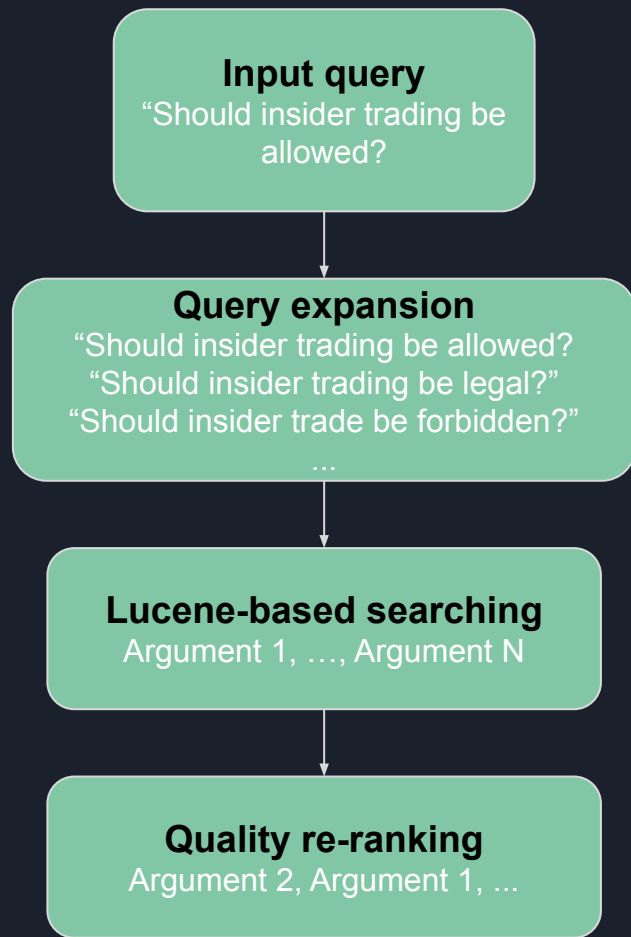
Example of user query: “Should insider trading be allowed?”



Our approach

After indexing, given a user query:

1. Expand user query: try to expand query scope
2. Search for each expanded query and merge retrieved arguments
3. Run argument quality re-ranking: boost arguments with higher “quality”





Indexing: Args.me arguments

We considered the following properties for each document, i.e. argument:

```
{  
  "id": "abc-123",  
  "text": "I think climate change is fake because I feel cold during winter",  
  "stance": "CON",  
  "context": {  
    "discussionTitle": "Is climate change real?"  
  }  
}
```



Indexing: Args.me arguments

A few documents were discarded due to:

- **Empty body** (text field): no reason to include them
- **Repeated ID**: the argument IDs are computed hashing the argument's body (plus other things), therefore same ID means same argument

Original dataset size: **8.2 GB**

Final index size: **457 MB**



Indexing: analyzer

We used to following filters (for title and argument body):

- **Standard tokenizer**
- **Lower case filter**

We tried and removed:

- **Stop words filter**: slightly lower score
- **Stemming filter**: any kind of stemmer significantly downgraded the performance
- **English possessive filter**: slightly reduced performance



Searching: similarity

- **Similarity:** we used **LMDirichletSimilarity** which almost doubled the score obtained with BM25
- **Baseline nDCG@5 score: 0.8279**
(using the corrected .qrels file)



Searching: query parser

Idea: the “discussionTitle” should briefly describe the topic of the argument, therefore we tried to match the query terms both in the body and title of the arguments.

We used **MultiFieldQueryParser**:

- Run the same boolean query on more fields (title and body)
- Assign a boost to each field

Parametrizing the boost given to a match of a query term with a title term, we found that the higher was the title boost the lower was the final score.

Possible reason: “bad” arguments may be pushed up in the final rank just because they inherit the discussion title.



Query Expansion: definition

- Query Expansion consists in enriching a user query:
 - to increase its effectiveness in the search process.
 - to improve the recall of the retrieval system.
- Transformer-based models -> Masked Language Model -> BERT¹
 - generates substitute terms depending on context of the query.
- No labelled data or external resources (e.g. WordNet)

[1] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Query Expansion: our approach

Sounds like a job for me!



1. Tokenization and PoS tagging.
2. Replace specific tokens with BERT's [MASK] .
3. Generate the best 10 tokens that fit in place of each [MASK]
-> exploit BERT's bidirectional attention mechanism.
4. Compute the BERT embeddings of these 10 tokens and compare them, using cosine similarity, to the embedding of the original token.
5. Perform a two-phase screening: if generated tokens are not good enough
-> use BERT again to generate new candidates.
6. Compose all the possible new queries and take a set of 10 random queries.

Query Expansion: some examples

- “Should insider trading be <allowed>?”

permitted (+)
tolerated (+)
forbidden (-)
illegal (-)



- “Is homework beneficial?”

“Is homework acceptable?”
“Is homework great?”
“Is homework appropriate?”

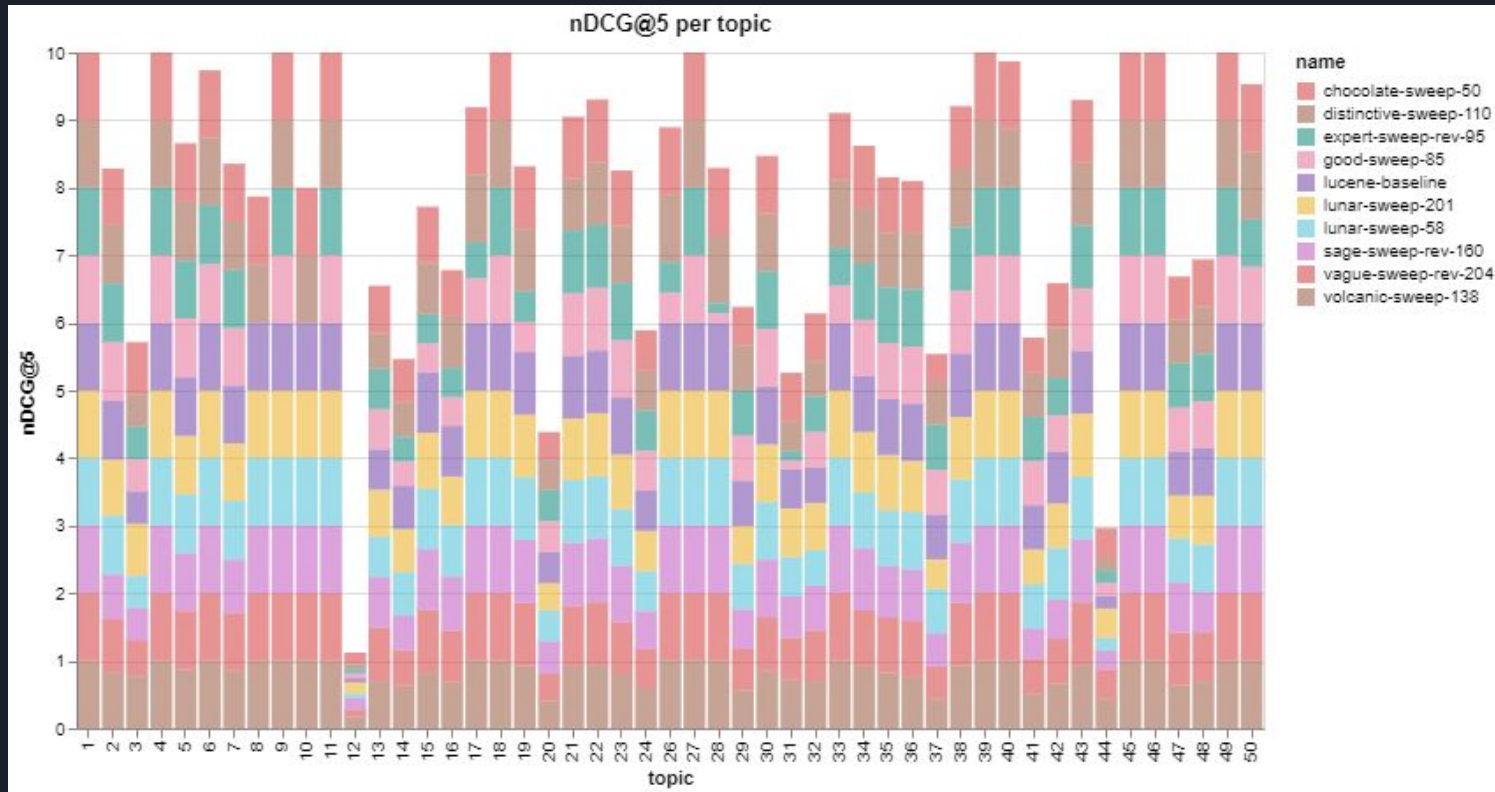


- “Is vaping with <e-cigarettes> safe?”

e-book
half-readers
mini-com (?)
z-commerce (?)



Query Expansion: hardest topics





Query Expansion: hardest topics

- Topic 8: “Should abortion be legal?”
 - it is a very short topic.
 - “abortion” and “divorce” are very similar according to BERT.
 - bias in BERT’s pre-training dataset.
- Topic 10: “Should any vaccines be required for children?”
 - “required” is equally replaced with “mandatory” and “recommended”.
 - there are contexts where lexical nuances make the difference.



Argument Quality re-ranking

- Objective: provide the user with relevant but also “good” arguments
- What makes an argument good? We can distinguish between¹
 - Logical quality
 - Rhetorical quality
 - Dialectical quality

[1] Wachsmuth, Henning, et al. “Computational Argumentation Quality Assessment in Natural Language.” *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, vol. 1, 2017, pp. 176–187.

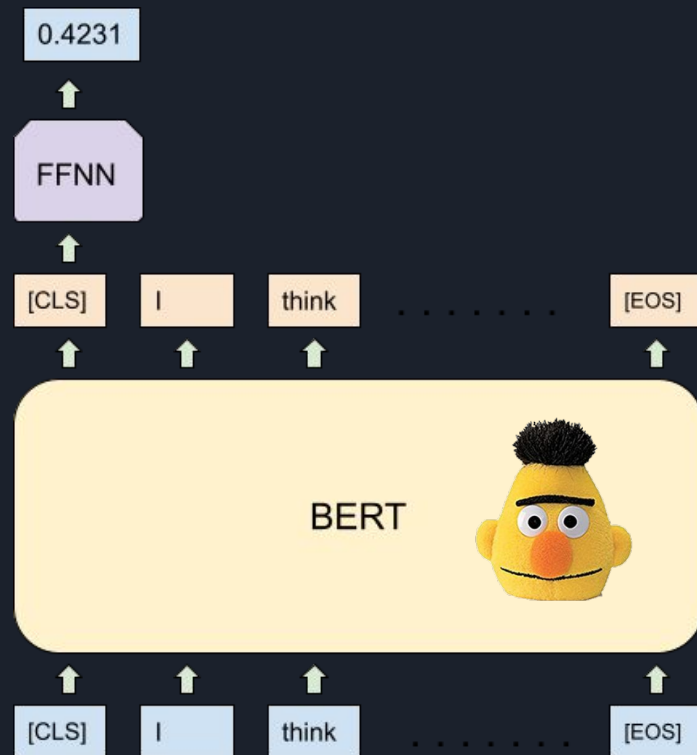


The Argument Quality Dataset

- To train our models, we used the Argument Quality Dataset
- It contains 1271 arguments from *args.me*, each having:
 - Relevance Score
 - Rhetorical Quality
 - Logical Quality
 - Dialectical Quality
 - Combined Quality

BERT for Argument Quality Prediction

- Idea: use models from the BERT family to have an argument quality regressor
- Explored 4 different models¹:
 - BERT
 - DistilBERT
 - RoBERTa
 - ALBERT



[1] huggingface.co/models



Parameter Space

- α : regulates the importance of quality for the re-ranking
- β : controls sigmoid steepness
- n_{rerank} : number of documents that are re-ranked according to quality
- *Query expansion*: boolean value to activate QE
- *Quality model*: BERT, DistilBERT, ALBERT and RoBERTa.
- $R(q,d)$: ranking function



Ranking Functions

- Normalization Function

$$R(q, d) = (1 - \alpha) r_{norm} + \alpha q_{norm}$$

- Sigmoid Function

$$R(q, d) = (1 - \alpha) \sigma(\beta r(d)) + \alpha \sigma(\beta q(d))$$

- Hybrid Function

$$R(q, d) = (1 - \alpha) r_{norm} + \alpha \sigma(\beta q(d))$$

Hyperparameter Study and Results

- We studied several combination of parameters using Weights and Biases¹
- We selected 10 runs, 5 of which were sent to Touchè



Run	AQE	α	β	n_{rerank}	quality model	$R(q, d)$	nDCG@5
lunar-sweep-201	no	0.75	-	5	BERT	normalize	0.8279
chocolate-sweep-50	no	0.1	2	5	BERT	sigmoid	0.8273
volcanic-sweep-138	no	0.5	0.8	5	BERT	sigmoid	0.8271
swordsman baseline	no	-	-	-	-	-	0.8266
lunar-sweep-58	no	0.1	0.2	5	RoBERTa	hybrid	0.8230
vague-sweep-rev-204	no	0.75	1.1	5	ALBERT	sigmoid	0.8229
lucene-baseline	no	0	-	-	-	normalize	0.8224
sage-sweep-rev-160	no	0.5	1.5	5	RoBERTa	sigmoid	0.8093
distinctive-sweep-110	no	0.1	0.8	15	ALBERT	hybrid	0.7992
good-sweep-85	yes	0.1	0.3	15	DistilBERT	hybrid	0.6857
expert-sweep-rev-95	yes	0.1	0.3	20	RoBERTa	hybrid	0.6801

[1] wandb.ai