



Report

**Title: Study Implemented Systems on Question
Answering and Build A System**

CSCE 798: Directed Study and Research

By: Kunal Ajay Dhavale

Instructor: Dr. Biplav Srivastava

Abstract

Question Answering (QA) is a notable exploration issue of Natural Language Processing (NLP). NLP is territory which comprises of tackling different issues like Sentiment Analysis, Text Classification, Text Extraction, Machine Translation, and so on. The focal point of this investigation is explicitly Question Answering.

Question Answering has become a premier errand in nowadays. It is a test to take the client query(question) as an information and give wanted yield. QA errand can be delegated, QA on even information, QA on text based. At the point when the inquiry is in regard to plain information, it is compulsory to give the specific answer as the appropriate response is now present to us as even information. The lone test is to deal with the client question and give the right answer. Effective approach to take care of this kind of issue can be use of Structured Query Language (SQL) to acquire the appropriate responses from the information. With replying on text-based information, it depends whether the appropriate response is available in the given content. Effective approach to tackle this kind of issue is utilize Bidirectional Encoder Representations from Transformers (BERT).

In this investigation, we explored about existing frameworks, for example, Google TAPAS[1], NeuralQA[2], WikiSQL[3] and attempted to accomplish comparable outcomes. For Question Answering over plain information the base engineering utilized was of Google TAPAS. I attempted to perform QA on our dataset. For Question Answering over text, an option improved on technique to NeuralQA was executed which yields comparable outcomes to NeuralQA.

Acknowledgements

This investigation and execution have taken significant measure of time and assets. I might want to thank my instructor **Dr. Biplav Srivastava** for his time, tolerance, direction and for permitting me to chip away at the subject of my advantage. His consolation propelled me a great deal.

I might likewise want to thank each one of those individuals who supported me and gave me significant counsel.

Contents

Abstract	2
Acknowledgements	3
Contents	4
Introduction.....	5
Problem.....	6
Related Work.....	7
Data Sources	8
Method/ Solution Steps/ Algorithm.....	10
Demonstration.....	13
Experiments	16
Significance of Work/ Discussion.....	21
Reference.....	22

Introduction

This record is a report for singular undertaking "Study Implemented Systems on Question Answering and Implement a System". This undertaking is an endeavor to build a system which can Answer inquiries of the clients dependent on the printed reference passage. Q/A dependent on literary section contains a reference passage which should contain the response to the mentioned question. The point is to return fitting responses to the mentioned inquiry (question) by the client. Aside from the carried out system, we will likewise attempt to perform question answering on our tabular data utilizing existing executed system; Google TAPAS [1].

One can think about various approaches to respond to address dependent on plain information or text-based information. For replying on literary information, we are following BERT [4] based approach to return the significant answers from the text. Utilizing BERT, the point is to make a rearranged at this point powerful approach to accomplish comparable usefulness as NeuralQA [2].

We would be implementing a system for Question Answering over text data for now. But the implementation of Q/A on tabular data using Google Collaboratory would be shared in a GitHub repository.

Problem

The inspiration driving this venture is to implement a system which can adequately address the inquiries posed by clients. With Q/A systems, you should be exact while responding to address. At the point when client poses an inquiry, they anticipate sensible and right answer. It is a test to make a machine answer the inquiries dependent on the reference information we have. As people, we can without much of a stretch read an English passage and point where is the appropriate response. Yet, for machines it isn't that basic assignment. It's lumbering to respond to the inquiries proficiently.

To beat this trouble, we can utilize existing techniques to cause the machine to comprehend human language and answer likewise. The methodology used to carry out such assignment is: BERT based methodology utilizing the pretrained model on Stanford SQUAD dataset [5], utilizing pre prepared tokenizers on SQUAD dataset. Another methodology is utilizing Google TAPAS [1] to perform question answering over tabular data. Pretrained model is utilized on our dataset to respond to the inquiries. We simply need to manipulate our tabular data to fit it as per the TAPAS requirements.

Related Works

Below is the list of works which were referred:

I TAPAS: Weakly Supervised Table Parsing via Pre-Training^[1]

II NeuralQA: A Usable Library for Question Answering on Large Datasets with BERT^[2]

III WikiSQL^[3]

IV Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning^[6]

Above is the rundown from which I got my inspiration to execute such a system. I additionally attempted to carry out an improved-on adaptation for one of the connected works and accomplish comparable outcomes. This subject has helped me a great deal to get further understanding about the point and propelled me to continue to deal with comparable themes.

Data Sources

Data for textual question answering

The data was taken from different sources. For answering questions related to water safety, we combined data from different sources and stored it in a single file. The corpus we made is by explicitly extricating applicable data from the existing documents. The answer to question 'what is limit for copper in water in South Carolina' in the files is '1.3 mg/L'. Modification is done so that we have the answer in paragraph format like: 'In South Carolina, the allowed limit for copper in drinking water is 1.3 mg/L'.

Here is the drive link to the documents referred to get the answers to desired questions.

<https://drive.google.com/drive/folders/1H23Afgb3VS1yUe9uKiYH8--RoqBRZ9aV?usp=sharing>

Also, to demonstrate the working of Q/A on text, we have taken short paragraph from the web to check the efficiency of our system.

To compare the performance with NeuralQA, same paragraph is used to answer the questions. Below is the demo link of NeuralQA. The paragraph is taken from the demo.

<https://neuralqa.fastforwardlabs.com/#/>

Data for Tabular Question Answering

- Covid Data: New York Times github
<https://github.com/nytimes/covid-19-data/blob/master/live/us-counties.csv>

County	Deaths	Cases
Abbeville	24	1735
Aiken	157	11179
Allendale	10	694
Anderson	373	16849

Fig 1: COVID Dataset

- Cricket Dataset: Created by self

Position	Player	Team	Span	Innings	Runs	Best Score	Average	Strike Rate
1	Sachin Tendulkar	India	1989-2012	452	18426	200	44.83	86.23
2	Kumar Sangakkara	Sri Lanka	2000-2015	380	14234	169	41.98	78.86
3	Ricky Ponting	Australia	1995-2012	365	13704	164	42.03	80.39
4	Sanath Jayasuriya	Sri Lanka	1989-2011	433	13430	189	32.36	91.2

Fig 1.1: Cricket Dataset

Method

Question Answering Over Textual Data

The aim for building a Q/A system which can answer over data was to achieve similar performance as compared to NeuralQA [2]. The methodology which is utilized for carrying out our framework resembles NeuralQA. The benefit is that our methodology is an improved on form to what NeuralQA is attempting to do. For this methodology we have, retriever, indicator model.

Retriever

For the given query(question), retriever returns relevant document from the corpus with the highest likelihood of containing the specific answer. To accomplish this usefulness, rank-bm25[7] is utilized. It contains various algorithms for querying documents and returning the most important records. For our execution we are utilizing BM25L.

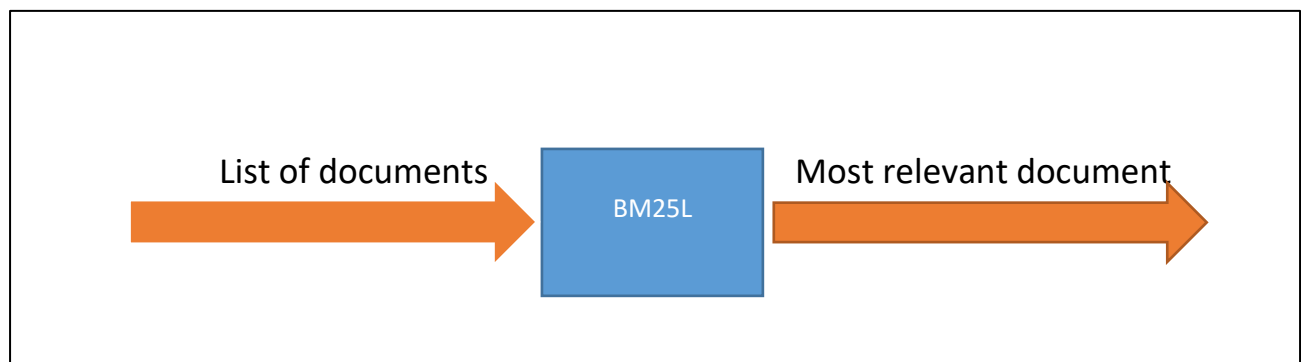


Fig 2: Working of BM25L Retriever

Predictor

When the relevant document is obtained from the retriever, it is then passed to a predictor model which returns the relevant document corresponding to the query. BERT is used to predict the answer for the query from the relevant document. The BERT model returns the span of the answer(if available);which is the start position of our answer till the end position. For the implementation, pre-trained model is used. The pre-trained model is bert-large-uncased-whole-word-masking-finetuned-squad. Also, the tokenizer is of bert-large-uncased-whole-word-masking-finetuned-squad.

When the systems relevant paragraph from the retriever, the query and the retrieved passage are tokenized to get the tokens. Once the data is fitted according to BERT criteria i.e. Segment A consist of query and Segment B consists of passage in which the answer is supposed to be; then we pass the tokens representing the input text tokens and pass the segment id to differentiate between the segments.

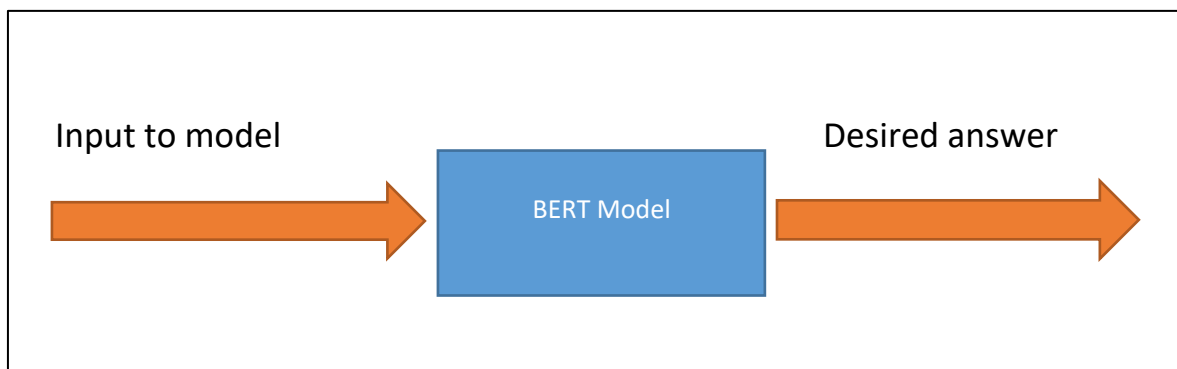


Figure 3: Working of BERT Predictor

Input to model: The tokens representing input text. The segment id to differentiate the question from answer text.

Question Answering Over Tabular Data

The rationale is to carry out Q/An on our custom information utilizing Google TAPAS. Our work is just to accommodate our dataset into the organization needed by the model utilized by TAPAS. The approach which the authors used is of using neural networks to answer query based on table.

At the point when a question is asked, BERT is utilized to encode the inquiry just as the table containing information column by line. Special embeddings are used. Important point is the usage of the special embedding used to encode the structured input. For column embedding, the authors use learned embeddings, the special embedding is used for rows. The model produces two outputs: 1) For each table, a score representing probability that the answer is in this particular table and 2) Indication of aggregation operator (if any) used to get the output of query.

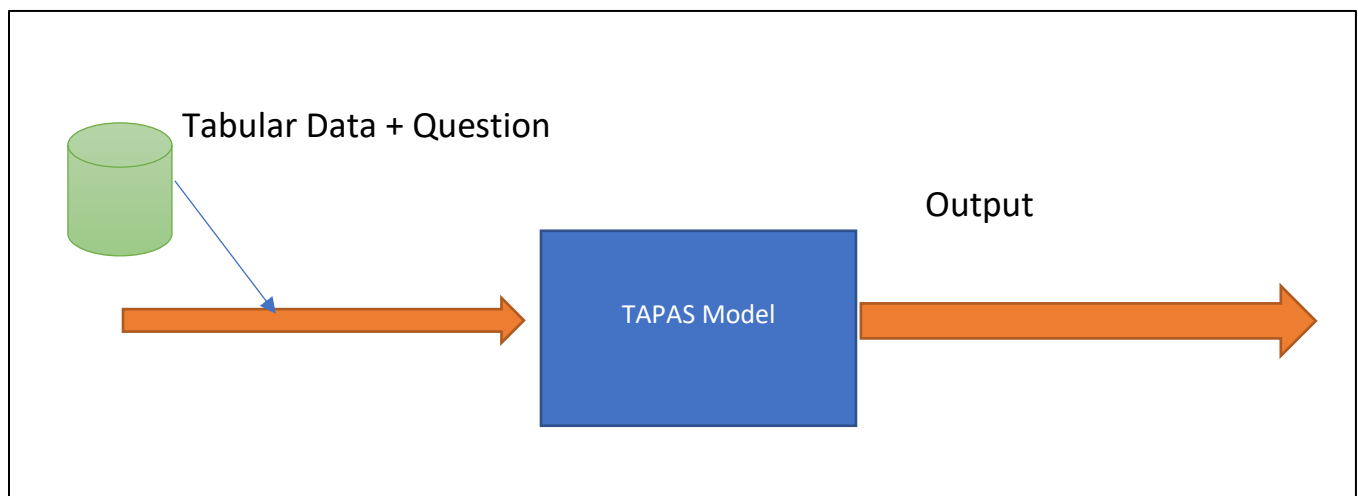




Fig 4: Working of TAPAS


Demonstration

Demonstration of Q/A System



Welcome to Question Answering System





About the System

This is a Question Answering System over Textual data. The user inputs query(question) and the model will predict the answer. The answer is predicted from the reference paragraph. The approach used to implement this system is *BERT* based approach. First based on the query the Retriever returns the relevant document. Then the query and relevant document are passed to bert model to predict the answer. BERT expects input as *tokenized inputs* and *segment id*. The tokenized inputs are nothing but the tokenized id of the query text and answer text. The segment id is used to distinguish among the query text and the answer text.

Implemented by : Kunal Ajay Dhavale


Instructor : Dr. Biplav Srivastava [About](#)

Below is the paragraph. You can ask questions based on the paragraphs


In South Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. In North Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. The Environmental Protection Agency (EPA) kept safe level for lead in drinking water as 0.015 mg/L safe level for copper in drinking water as 1.3 mg/L. The World Health Organization set safe level for lead in drinking water to 0.01 mg/L and safe level for copper in drinking water to 2 mg/L. In Michigan, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.2 mg/L. California has set action level of lead to 0.015 mg/L and for copper the action level is 0.0013 mg/L. Illinois has set action level of lead to 0.005mg/l and for copper the action level is 0.200 mg/L.

Ask your question :

[Get Answer](#)



Welcome to Question Answering System



Below is the paragraph. You can ask cross check your answers.

In South Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. In North Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. The Environmental Protection Agency (EPA) kept safe level for lead in drinking water as 0.015 mg/L safe level for copper in drinking water as 1.3 mg/L. The World Health Organization set safe level for lead in drinking water to 0.01 mg/L and safe level for copper in drinking water to 2 mg/L. In Michigan, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.2 mg/L. California has set action level of lead to 0.015 mg/L and for copper the action level is 0.0013 mg/L. Illinois has set action level of lead to 0.005mg/l and for copper the action level is 0.200 mg/L.

The answer to your question is

0 . 005mg / l

[Go Back](#)



About the System

This is a Question Answering System over Textual data. The user inputs query(question) and the model will predict the answer. The answer is predicted from the reference paragraph. The approach used to implement this system is *BERT* based approach. First based on the query the Retriever returns the relevant document. Then the query and relevant document are passed to bert model to predict the answer. BERT expects input as *tokenized inputs* and *segment id*. The tokenized inputs are nothing but the tokenized id of the query text and answer text. The segment id is used to distinguish among the query text and the answer text.

Implemented by : Kunal Ajay Dhavale

Instructor : Dr. Biplav Srivastava [About](#)

Below is the paragraph. You can ask questions based on the paragraphs

In South Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. In North Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. In New York, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. The Environmental Protection Agency (EPA) kept safe level for lead in drinking water as 0.015 mg/L safe level for copper in drinking water as 1.3 mg/L. The World Health Organization set safe level for lead in drinking water to 0.01 mg/L and safe level for copper in drinking water to 2 mg/L. In Michigan, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.2 mg/L. California has set action level of lead to 0.015 mg/L and for copper the action level is 0.0013 mg/L. Illinois has set action level of lead to 0.005mg/l and for copper the action level is 0.200 mg/L.

Ask your question :

[Get Answer](#)

Below is the paragraph. You can ask cross check your answers.

In South Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. In North Carolina, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. In New York, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.3 mg/L. The Environmental Protection Agency (EPA) kept safe level for lead in drinking water as 0.015 mg/L safe level for copper in drinking water as 1.3 mg/L. The World Health Organization set safe level for lead in drinking water to 0.01 mg/L and safe level for copper in drinking water to 2 mg/L. In Michigan, the action level for lead in drinking water is 0.015 mg/L and action level for copper in drinking water is 1.2 mg/L. California has set action level of lead to 0.015 mg/L and for copper the action level is 0.0013 mg/L. Illinois has set action level of lead to 0.005mg/l and for copper the action level is 0.200 mg/L.

The answer to your question is

0 . 015 mg / l

[Go Back](#)

Demonstration of Google TAPAS

```
▶ result = predict(list_of_list, ["for which team sachin tendulkar play", "how many runs were scored by him",  
                                "among dravid and sachin who scored highest runs"])
```

```
is_built_with_cuda: True  
is_gpu_available: True  
GPUs: [PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU')]  
Training or predicting ...  
Evaluation finished after training step 0.
```

```
> for which team sachin tendulkar play  
India  
> how many runs were scored by him  
18426  
> among dravid and sachin who scored highest runs  
Sachin Tendulkar
```

```
▶ result = predict(list_of_list, ["How many cases of covid were in county of Abbeville",  
                                " Among Charleston and York, who has more number of deaths"])
```

```
↗ is_built_with_cuda: True  
is_gpu_available: True  
GPUs: [PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU')]  
Training or predicting ...  
Evaluation finished after training step 0.
```

```
> How many cases of covid were in county of Abbeville  
1735  
> Among Charleston and York, who has more number of deaths  
Charleston
```

Experiments

Test on Q/A over Textual Data

To evaluate our system 57 questions were selected. Same questions were asked before implementing the actual system and during the testing stage. The testing before implementation was done on Google Colab. Of the 57 questions, the implemented system was able to answer 56 questions accurately i.e. accuracy of 98.2%. Whereas, during testing stage the model was able to answer 56 questions, giving accuracy of 98.2% as well. To compare the answer returned during testing stage and actual implemented system, Jaccard similarity of the two answer strings was calculated. Result of 0 means the strings are exactly same and 1 indicates the strings are totally different without any similarity. Among the 57 questions, all 57 had Jaccard similarity of 0.

The accuracy for document retrieval for testing stage and actual implemented system is 100%.

The experiment details can be found using the below link:

https://drive.google.com/file/d/1pDHfKybhXBUaIQoTRHO_wP2VACICXo-5/view?usp=sharing

Test on TAPAS

Correctly Answers

> Did Sachin Tendulkar scored more runs or Rahul Dravid
Sachin Tendulkar

> Who scored highest between sachin sendulkar and rahul dravid
Sachin Tendulkar

> players who play for INDIA
Sachin Tendulkar, Rahul Dravid, Saurav Ganguly, Yuvraj Singh, Mohammad Azharuddin ,
Mahendra Singh Dhoni, Virender Sehwag, Virat Kohli

> Australian player with highest runs
Ricky Ponting

> Which team has highest number of players
India

> Which team has more number of occurences among India,Pakistan and Australia
India

> Which team has more number of occurences among south africa,Pakistan and Australia
South Africa

> who has highests average of all other players
Virat Kohli

> which team has highest score of best score
India

> players having average over 50
AB de Villiers, Virat Kohli

> Maximum runs scored between sachin and dravid
18426

Wrong Answers

> Who scored more runs among Sachin Tendulkar and Rahul Dravid
Rahul Dravid

> Total combined runs in the data
14234

> Combined total addition of Runs
13704, 18426, 11867

> player name with maximum strike rate
Sachin Tendulkar, Virat Kohli

> Player with maximum strike rate

> Player having highest strike rate among all
Sachin Tendulkar

> who played highest number of innings between ponting,gayle,david,sehwag
Sachin Tendulkar

Special Case

> Who scored maximum runs between sachin and sehwag
Kumar Sangakkara

> players who play for countries but australia
Adam Gilchrist, Steve Waugh, Ricky Ponting, Michael Waugh

> players who play for teams but not australia
Sachin Tendulkar, Stephen Fleming, Mohammad Yousuf , Shahid Afridi, Jacques Kallis, Saurav Ganguly, Inzamam ul Haq, Aravinda de Silva, Mohammad Azharuddin , AB de Villiers, Tilakratne Dilshan, Michael Clarke, Herschelle Gibbs, Brian Lara, Virender Sehwag, Virat Kohli, Kumar Sangakkara, Chris Gayle

> Among India and West Indies, which team has highest runs
India, West Indies

The answer predictions returned by the model are classified into three categories: Correct Answers, Wrong Answers, Special Case.

Correct Answers

Based on the query, these are the answers to the questions which were correctly returned by the model. These are straightforward questions and the model was able to correctly predict the answer.

Wrong Answers

These are the answers which are not correct. The word 'among' is supposed to play a important role. The same question was asked in a different manner and the system was able to correctly return the correct answer. But, when the word 'among' was used inside the question which means the same, the answer returned was totally opposite. The example is the first question of the wrong answered ones. The question is "Who scored more runs among Sachin Tendulkar and Rahul Dravid ". The real answer is Sachin Tendulkar. Also, the system was able to give correct answer for the same question "Did Sachin Tendulkar scored more runs or Rahul Dravid". The question which mean same, but the words used to phrase question play a role in predicting the answer. When the combined total was asked, the model only returned the highest runs scored. The model is having issues when adding the values because we have to convert all our data to string format as it is required for the model.

Special Case

This category is interesting as over here we can expect some unexpected outcomes. The question asked is “who scored maximum runs between Sachin and Sehwag. The correct answer is Sachin, but surprisingly the model returns Kumar Sangakkara. In the dataset, after Sachin it was Kumar Sangakkara who scored most runs. It is interesting to see that the model is analyzing full data and returned the player name who score most runs after Sachin correctly. Also the question was” players who played for countries but Australia”. Surprisingly the model returned only the players who play for Australia. “Among India and West Indies, which team has highest runs”. The model answered the name of both countries. It was surprising to see that the top scorer country was India and the country with second top scoring was West Indies.

Significance of Work/ Discussion

The system was able to answer questions accurately based on the query and the retrieved document from the retriever. Our system provided promising results; accuracy of 98.2%. In the future, we plan to experiment with more architectures, regions (US states) and document sizes. By enhancing the document reader, the retriever functionality will be enhanced as it will be able to return the documents even more precisely. Our system performs similarly to NeuralQA. It is a simpler version of NeuralQA. Our system can provide key information from regulations to facilitate the compliance efforts of EPA, keep communities safe, and highlight attention to areas where gaps still exist.

Reference

- [1] <https://ai.googleblog.com/2020/04/using-neural-networks-to-find-answers.html>
- [2] <https://github.com/victordibia/neuralqa>
- [3] <https://github.com/salesforce/WikiSQL>
- [4] <https://www.aclweb.org/anthology/N19-1423.pdf>
- [5] <https://huggingface.co/bert-large-uncased-whole-word-masking-finetuned-squad>
- [6] <https://arxiv.org/pdf/1709.00103.pdf>
- [7] <https://pypi.org/project/rank-bm25/>
- [8] Water safe limits, Biplav Srivastava, Github: <https://github.com/biplav-s/water-info>