OPEN – DOMAIN QUESTION AND ANSWER

AI for DS Report
Submitted in the partial fulfillment of the requirements for
the award of the degree of

Bachelor of Technology
in
Department of Computer Science and Engineering

By

Tahseen Begum	2010030168
E. Pravallika	2010030046
N. Sowgna	2010030344
P. Keerthana	2010030445

under the supervision of

Dr. Arpita Gupta

Assistant Professor

Department of Computer Science and Engineering

K L University Hyderabad,

Aziz Nagar, Moinabad Road, Hyderabad – 500 075, Telangana, India.

March 2022

DECLARATION

The AI for DS Report entitled "Open - Domain Question and Answer" is a record of Bonafede's work of Tahseen begum – 2010030168, E. Pravallika – 2010030046, N. Sowgna – 2010030344, P. Keerthana – 2010030445, submitted in partial fulfillment for the award of B.Tech in the Department of Computer Science and Engineering to the K L University, Hyderabad. The results embodied in this report have not been copied from any other Departments/Universities/Institutes.

Signature of the Student's

TAHSEEN BEGUM

E. PRAVALLIKA

N. SOWGNA

P. KEERTHANA

CERTIFICATE

This is to certify that the AI for DS Report entitled "Open - Domain Question and Answer" is being submitted by Tahseen begum (2010030168), E. Pravallika (2010030046), N. Sowgna (2010030344), P. Keerthana (2010030445) submitted in partial fulfillment for the award of B.Tech in Dr. Arpita Gupta to the K L University, Hyderabad is a record of bonafide work carried out under our guidance and supervision.

The results embodied in this report have not been copied from any other departments/universities/institutes.

Signature of the Supervisor

Dr. Arpita Gupta

Signature of the HOD

Signature of the External Examin

ACKNOWLEDGEMENT

First and foremost, we thank the lord almighty for all his grace & mercy showered upon us, for completing this AI for DS successfully.

It is great pleasure for me to express my gratitude to our honorable President **Sri. Koneru Satyanarayana**, for giving the opportunity and platform with facilities in accomplishing the project-based laboratory report.

I express my sincere gratitude to our Principal **Dr. L. Koteswara Rao** for his administration of our academic growth.

I express sincere gratitude to our Coordinator for her leadership and constant motivation provided in the successful completion of our academic semester. I record it as my privilege to deeply thank you for providing us with the efficient faculty and facilities to make our ideas into reality.

I express my sincere thanks to our project supervisor **Dr. Arpita Gupta** for her novel association of ideas, encouragement, appreciation, and intellectual zeal which motivated us to venture into this project successfully.

We wholeheartedly thank all the teaching and non-teaching staff of our department without whom we won’t have made this Social Internship a reality. We would like to extend our sincere thanks, especially to our parents, our family members, and our friends who have supported us to make this AI for DS a grand success.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to making this project report success.

ABSTRACT

Open-domain question answering (QA) is the task of identifying answers to natural questions from a large corpus of documents. The typical open-domain QA system starts with information retrieval to select a subset of documents from the corpus, which are then processed by a machine reader to select the answer spans.

Aiming to answer an open domain question based on the knowledge base, we suggest a TANDA algorithm that can automatically extract an adverbial pearl from a simple question and translate it into a KB query. similarity preferences are used to exclude a candidate's start after an easy way to link business. Our method obtained an F1 score of 82.47% in test data. In addition, there is also a series of full bug testing and analysis that can identify features and disabilities of a new data set.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Questionnaire (QA) is a field of computer science within the fields of information retrieval and natural language processing (NLP), which deals with building programs that automatically answer questions asked by people in the native language. To achieve this, we will use the Python Library and TensorFlow wrapper which enables in-depth learning and AI.

Intends to answer the question in the form of a natural language based on large informal texts.

The purpose of the QA is to produce concise answers to summarized questions asked in the original language. This type of retrieval is required with the growth of digital knowledge. Previously QAS was designed for a specific domain and has limited functionality.

Introduce QAS Target to the types of questions most frequently asked by users, the features of the data source, and the correct answer generated. We aim to build a QA web-scale system. Many QA systems prior to issuing an answer perform quiz settings to predict the type of question response.

One significant concern with this approach is that the lexical overlap will make sentence selection easier for the QASENT dataset and might inflate the performance of existing systems in additional natural settings. Our WIKIQA dataset differs from the present QASENT dataset in both question and candidate answer sentence distributions. Questions in QASENT were originally employed in TREC 8-13 QA tracks and were a combination of questions from query logs (e.g., Excite and Encarta) and from human editors.

The questions may well be outdated and do not reflect actuality information needed of a QA system user. against this, questions in WIKIQA were sampled from real queries of Bing without editorial revision. On the sentence side, the candidate sentences in QASENT were selected from documents returned by past participating teams in the TREC QA tracks, and sentences were only included if they shared content words from the questions.

These procedures make the distribution of the candidate sentence skewed and unnatural. compared, 20.3% of the answers in the WIKIQA dataset share no content words with questions. Candidate sentences in WIKIQA were chosen from relevant Wikipedia pages directly, which may be closer to the input of a solution sentence selection module of a QA system.

Open-domain Question Answering (OpenQA) is a vital task in tongue Processing (NLP), which aims to answer an issue within the type of tongue supported large-scale unstructured documents. Recently, there has been a surge in the amount of research literature on OpenQA, particularly on techniques that integrate with neural Machine Reading Comprehension (MRC).

While these research works have the need performance to new heights on benchmark datasets, they need to be rarely covered in existing surveys on QA systems. during this work, we review the most recent research trends in OpenQA, with particular attention to systems that incorporate neural MRC techniques. Specifically, we start with revisiting the origin and development of OpenQA systems.

We then introduce modern OpenQA architecture named "Retriever-Reader" and analyzed the varied systems that follow this architecture yet because of the specific techniques adopted in each of the components. We then discuss key challenges to developing OpenQA systems and offer an analysis of benchmarks that are commonly used. We hope our work would enable researchers to be told of the recent advancement and also the open challenges in OpenQA research, so on stimulate further progress in this field.

The “open-domain” part refers to the lack of the relevant context for any arbitrarily asked factual question. In the above case, the model only takes as the input the question but no article about “why Einstein didn’t win a Nobel Prize for the theory of relativity” is provided, where the term “the law of the photoelectric effect” is likely mentioned. In the case when both the question and the context are provided, the task is known as Reading comprehension (RC).

2. LITERATURE SURVEY

s.no	Authors	Title	Publishing	Techniques & Dataset	Pros	Cons
1	Sewon Min, Danqi Chen, Luke Zettlemoyer, Hannaneh Hajishirzi	Knowledge guided text retrieval and reading for open domain question answering	arXiv preprint arXiv:1911.03868, 2019	Qualitative analysis to illustrate which components contribute the most to the overall system performance. outperforms competitive baselines on three opendomain QA datasets, WEBQUESTIONS, NATURAL QUESTIONS and TRIVIAQA.	We proposed a general approach for text-based open-domain question answering that integrates graph structure at every stage to construct, retrieve and read a graph of passages. Our retrieval method leverages both text corpus and a knowledge base to find a relevant set of passages and their relations. Our reader then propagates information according to the input graph, enabling knowledge-rich crosspassage representations.	when dealing with the out-of-scope reasoning target, and are unaware of explainable structured information
2	Yunlin Zhan, Yinya Huang, Xiao Dong, Qingxing Caoan, Xiaodan Liang	Explainable reasoning paths for commonsense question answering	Received 16 May 2021, Revised 13 October 2021, Accepted 16 October 2021, Available online 29 October 2021	A reasonable and explainable framework is proposed to explicitly incorporate external reasoning paths with structured information to explain and facilitate commonsense QA.	A path finder and a hierarchical path learner. To answer a commonsense question, the path finder first retrieves explainable reasoning paths from a large-scale knowledge graph, then the path learner encodes the paths with hierarchical encoders and uses the path features to predict the answers.	when dealing with the out-of-scope reasoning target, and are unaware of explainable structured information

Table. 2.1 Survey for ODQA1

s.no	Authors	Title	Publishing	Techniques & Dataset	Pros	Cons
3.	Zechen Guo	Research and Implementation of Open Domain Question Answering System Based on DuReader Dataset and BIDAF Model	2021 doi:10.1088/1742-6596/1769/1/012033	This article embeds deep learning technology into the system and uses intelligent chat to show them.	An open domain question answering system aims at returning an answer in response to the user's question. The returned answer is in the form of short texts rather than a list of relevant documents.	when dealing with the out-of-scope reasoning
4.	Sharon Levy	Open-Domain Question-Answering for COVID-19 and Other Emergent Domains	**[v1]** Wed, 13 Oct 2021 18:06:14 UTC (6,650 KB)	we incorporate effective re-ranking and question-answering techniques, such as document diversity and multiple answer spans. Our open-domain question-answering system can further act as a model for the quick development of similar systems that can be adapted and modified for other developing emergent domains.	which we can use to efficiently find answers to free-text questions from a large set of documents.	when dealing with the out-of-scope reasoning

Table. 2.2 Survey for ODQA2

Technique

S. No	Title of the Study	Model	Dataset	Evaluation Criteria	Results	Ref.
1.	TANDA: Transfer and Adapt Pre-Trained Transformer Models for Answer Sentence Selection	TANDA-Roberta	WikiQA, TREC-QA,QNLI	We demonstrate the benefits of our approach for answer sentence section, which is a well-known inference task in Question Answering.	TANDA produces an intermediate model with three main features: (i) it can be more effectively used for fine-tuning on the target NLP application, being more stable and easier to adapt to other tasks; (ii) it is robust to noise, which might affect the target domain data; and (iii) it enables modularity and efficiency, i.e., once a Transformer model is adapted to the target general task, e.g., AS2, only the adapt step is needed for each targeted domain	1175

Table. 2.3 Technique survey

Dataset

S. No	Name of the Dataset	Characteristics	Model	Publisher
1.	SQA: Sequential Question Answering	Natural language text to meaning representations informal logic has emerged as a key technical component for building question answering systems. Once a natural language question has been mapped to a formal query, its answer can be retrieved by executing the query on a back-end structured database	Sequential question answering task, we propose a novel dynamic neural semantic parsing framework trained using a weakly supervised reward-guided search. Our model effectively leverages the sequential context to outperform state-of-the-art QA systems that are designed to answer highly complex questions.	Iyyer, Mohit, Wen-tau Yih, and Ming-Wei Chang 2019

Table. 2.4 Dataset survey

3. METHODOLOGY

The system is a web application that helps the user to get the answer of the specific domain. We have given the text box where the user can enter his/her question, it gives the answer to it. All the user gives data to the application may save for further use to update the status of the model, and data analysis in the future. We also help users by giving some guidance on how to get the answer.

4. FLOWCHART

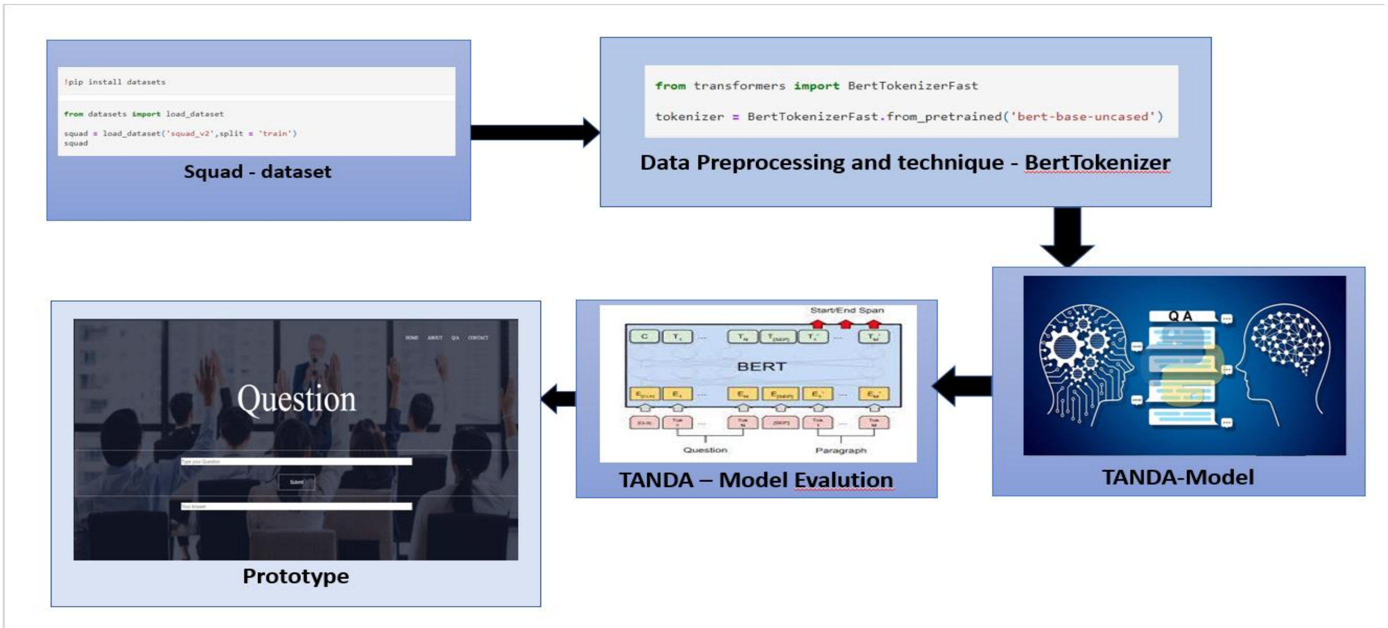


Fig. 4.1 work progress

5. IMPLEMENTATION

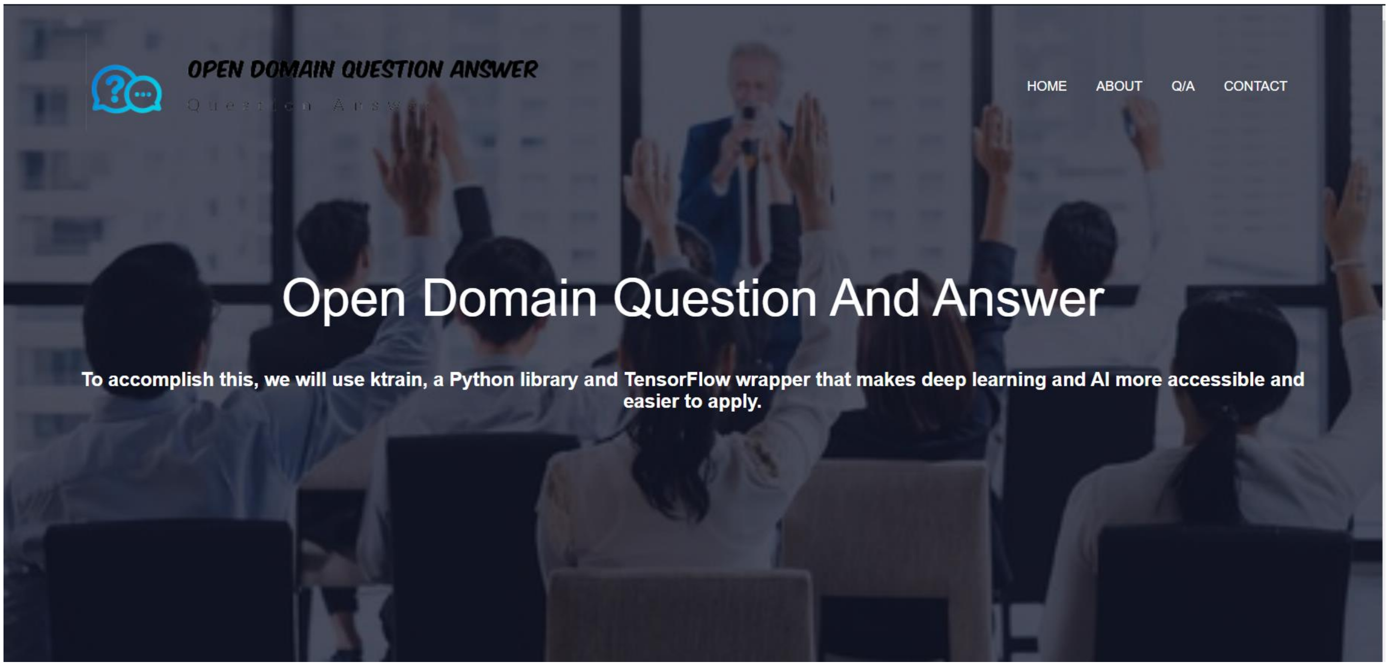
First, we download the SQuAD v2 dataset. This training process needs us to pass a context as input and add optimize on two integer values, answer_start, and answer_end as labels.
We have context and answer_start.
Here, the answer_start marks the character position within the context where we can find the start of our answer. But we also need an answer_end, which does not exit vet.
That looks good, however, we will find that some answers do not include answers, like this:
The reason we are not returning an answer here is that there is no answer, so we can either ignore this attempt to train our QA model to identify when there is not an answer by setting answer_start and answer_end to a 'no answer values like 1 and 0. we will keep the samples and zero no answers.

for all other samples, we will find answer_end as answer_start+length of the answer txt.
Loading cached processed dataset

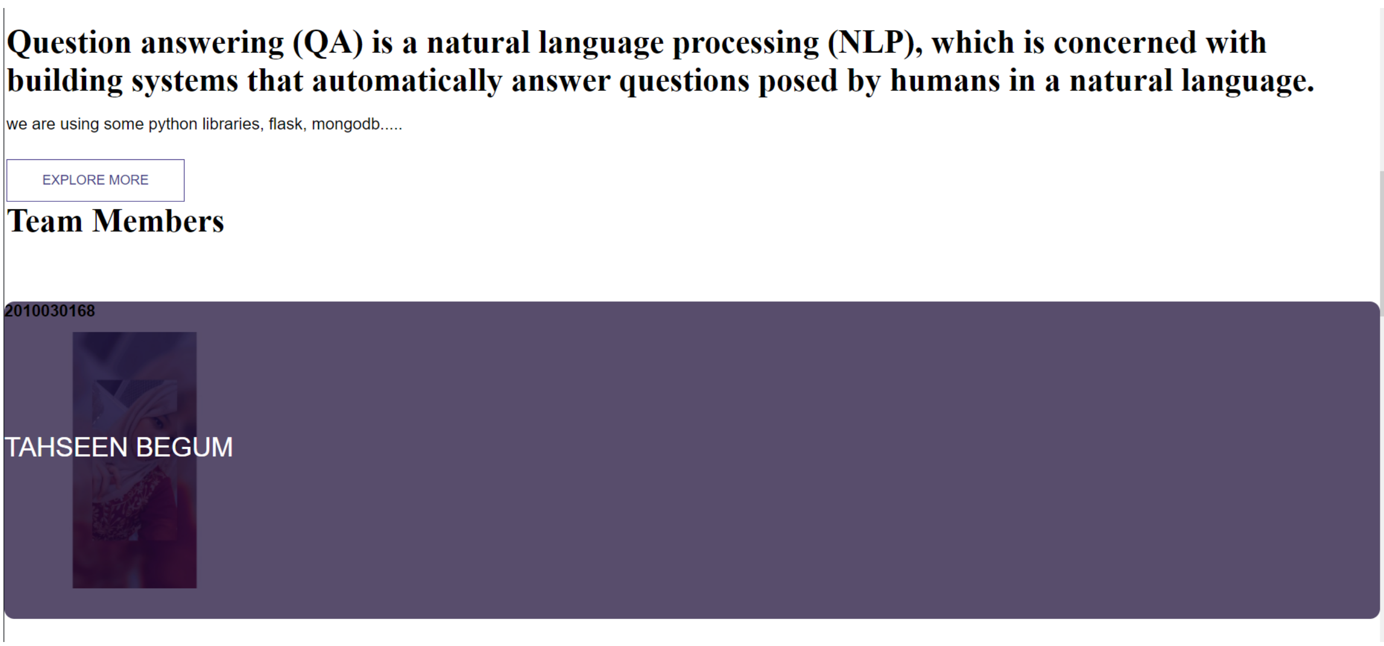
This tokenizes the text in the format [CLS] question [SEP] context [SEP].
The only problem we have now is that our transformer model will be processing tokens, not strings, and our answer_start/answer-end position is based on character position in the context tokens string. So, we need to update these positions instead, and consider the context tokens.

Fortunately, the offsets_mapping tensor can help us here, this tells us for each input ID, the start and end character in the original text of that token. So we get the token position of the context tokens here. But before doing this we must use the token_type_ids tensor to identify the length of the preceding question tokens which we use to shift the ID position.
The first set of 0 tokens is our question, the following 1 tokens are our context.
Here we can see that the question_len allows us to shift between our question and context segments. let's use this along the offsets_mapping tensor to get our answer start and end tokens.
Now we drop unnecessary, all we need are input_ids, attention_mask, token_type_ids start_position, and end_position.

6. RESULTS



6.1 Home page



6.2 about page

Space Question

```
[9] answers = qa.ask('When did the Cassini probe launch?')
    qa.display_answers(answers[:5])
```

	Candidate Answer	Context	Confidence	Document Reference
0	in october of 1997	cassini is scheduled for launch aboard a titan iv / centaur in october of 1997 .	0.819034	59
1	on january 26,1962	ranger 3, launched on january 26,1962 , was intended to land an instrument capsule on the surface of the moon, but problems during the launch caused the probe to miss the moon and head into solar orbit.	0.151228	8525
2	- 10 / 06 / 97	key scheduled dates for the cassini mission (vvejga trajectory)---10 / 06 / 97-titan iv / centaur launch 04 / 21 / 98-venus 1 gravity assist 06 / 20 / 99-venus 2 gravity assist 08 / 16 / 99-earth gravity assist 12 / 30 / 00-jupiter gravity assist 06 / 25 / 04-saturn arrival 01 / 09 / 05-titan probe release 01 / 30 / 05-titan probe entry 06 / 25 / 08-end of primary mission (schedule last updated 7 / 22 / 92) - 10 / 06 / 97	0.029694	59
3	* 98	cassini * * * * * * * * * * * * * * * * * * 98 ,115 * * * *	0.000026	5356
4	the latter part of the 1990s	scheduled for launch in the latter part of the 1990s , the craf and cassini missions are a collaborative project of nasa, the european space agency and the federal space agencies of germany and italy, as well as the united states air force and the department of energy.	0.000017	18684

6.3 Space Question

As shown above, the top candidate's answer of October 1997 is the correct one. (This won't always be the case, but it is here.)

Technical Support Question

```
[10] answers = qa.ask('What causes computer images to be too dark?')
     qa.display_answers(answers[:5])
```

	Candidate Answer	Context	Confidence	Document Reference
0	if your viewer does not do gamma correction	if your viewer does not do gamma correction , then linear images will look too dark, and gamma corrected images will ok.	0.937990	13873
1	is gamma correction	this, is gamma correction (or the lack of it).	0.045166	13873
2	so if you just dump your nice linear image out to a crt	so if you just dump your nice linear image out to a crt , the image will look much too dark.	0.010337	13873
3	that small color details	the algorithm achieves much of its compression by exploiting known limitations of the human eye, notably the fact that small color details are not perceived as well as small details of light and dark.	0.002114	6987
4	that small color details	the algorithm achieves much of its compression by exploiting known limitations of the human eye, notably the fact that small color details are not perceived as well as small details of light and dark.	0.002114	12344

6.4 Technical Support Question

It looks like a lack of gamma correction is a cause of this technical problem.

Religious Question

```
answers = qa.ask('Who was Mohammed Prophet?')
qa.display_answers(answers[:5])
```

	Candidate Answer	Context	Confidence	Document Reference
0	anwar mohammed	unfortunately not all think like this, we have cases like : anas omran, hamza saleh, jle, mohammed reza, mehmed abu abed, anwar mohammed and others who think that jihad is the only solution.	0.834524	18764
1	prophet isaiah	17] this was to fulfil what was spoken by the prophet isaiah , " he took our infirmities and bore our diseases.	0.094471	913
2	accept brigham young	" the rest " were apostates and excommunicated members of the church, while the great majority of the membership, the twelve, and the various auxiliary organizations, chose to accept brigham young as the new prophet and leader of the church.	0.027686	7242
3	" rushdie	[this was in response to the claim that " rushdie made false statements about the life of mohammed ", with the disclaimer " (fiction, i know, but where is the line between fact and fiction ?)-i stand by this distinction between fiction and " false statements "]	0.021582	8475
4	barnabas	barnabas was a prophet, acts says, before he was even sent out as an apostle.	0.016268	8118

6.5 Religious Question

Here, we see different views on Mohammed Prophet buried within this dataset.

7. CONCLUSION

We will build a fully-functional, end-to-end open-domain QA system in only 3 lines of code. To accomplish this, we will use ktrain, a Python library, and TensorFlow wrapper that makes deep learning and AI more accessible and easier to apply. ktrain is a free, open-source.

1. Uses the search index to locate documents that contain words in the question
2. Extracts paragraphs from these documents for use as contexts and uses a BERT model pretrained on the SQuAD dataset to parse out candidate answers
3. Sorts and prunes candidate answers by confidence scores and returns results

8. FUTURE WORK

In future work, we plan to improve the Bert modules with extended contexts (i.e. more than one sentence) and add some spatial reasoning. We also want to improve the retrieval by crawling relevant documents from web search engines instead of using snippets. This could be a good method to find more sentences with supported answers.

9. REFERENCES

Liu, Qian, Bei Chen, Jiaqi Guo, Zeqi Lin, and Jian-guang Lou. "TAPEX: Table pre-training via learning a neural SQL executor." arXiv preprint arXiv:2107.07653 (2021).

Hu, Zhe, Zuohui Fu, Cheng Peng, and Weiwei Wang. "Enhanced Sentence Alignment Network for Efficient Short Text Matching." In Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020), pp. 34-40. 2020.

Soldaini, Luca, and Alessandro Moschitti. "The cascade transformer: an application for efficient answer sentence selection." arXiv preprint arXiv:2005.02534 (2020).

Shi, Qi, Qian Liu, Bei Chen, Yu Zhang, Ting Liu, and Jian-Guang Lou. "LEMON: Language-Based Environment Manipulation via Execution-Guided Pre-training." arXiv preprint arXiv:2201.08081 (2022).

Matt Crane. 2018. Questionable answers in question answering research: Reproducibility and variability of published results. Transactions of the Association for Computational Linguistics, 6:241–252.

Guo, Zechen, Fucheng Wan, and Ning Ma. "Research and Implementation of Open Domain Question Answering System Based on DuReader Dataset and BIDAF Model." In Journal of Physics: Conference Series, vol. 1769, no. 1, p. 012033. IOP Publishing, 2021.

Levy, Sharon, Kevin Mo, Wenhan Xiong, and William Yang Wang. "Open-Domain Question-Answering for COVID-19 and Other Emergent Domains." arXiv preprint arXiv:2110.06962 (2021).