Improving Retrieval-Based Question Answering with Deep Inference Models

George-Sebastian Pîrtoacă, Traian Rebedea, Ștefan Rușeți

University Politehnica of Bucharest
george.pirtoaca@stud.acs.upb.ro,{traian.rebedea, stefan.ruseti}@cs.pub.ro

Abstract. Question answering is one of the most important and difficult applications at the border of information retrieval and natural language processing, especially when we talk about complex science questions which require some form of inference to determine the correct answer. In this paper, we present a two-step method that combines information retrieval techniques optimized for question answering with deep learning models for natural language inference in order to tackle the multi-choice question answering in the science domain. For each question-answer pair, we use standard retrieval-based models to find relevant candidate contexts and decompose the main problem into two different sub-problems. First, assign correctness scores for each candidate answer based on the context using retrieval models from Lucene. Second, we use deep learning architectures to compute if a candidate answer can be inferred from some well-chosen context consisting of sentences retrieved from the knowledge base. In the end, all these solvers are combined using a simple neural network to predict the correct answer. This proposed two-step model outperforms the best retrieval-based solver by over 3% in absolute accuracy.

Keywords: Question Answering, Natural Language Processing, Deep Learning, Natural Language Inference, Information Retrieval.

1 Introduction

We live in a world where Artificial Intelligence is becoming more and more a part of everyday life and the ability to answer questions expressed in natural language is very important in order to deploy successful products requiring natural language interaction. More and more commercial systems that handle human interaction encompass a form of question answering (QA). Thus, Google Assistant and Amazon Alexa manage to answer simple factoid questions like *"What is the most viewed song on YouTube in the current month?"*. All these systems have a common trait as they are designed to answer only a specific category of factoid questions that rely on extracting information from a large collection of sources (e.g. proprietary knowledge bases).

However, to make question-answering more realistic and difficult at the same time, a system should be able at least to answer multi-choice questions that are related to grade-level science classes (e.g. *"Which organisms contribute most to the decomposition of dead organisms?"*). The premise is that these questions are much more difficult to answer (even if the system is given a list of possible answers) and require models that should be able to (partially) understand the context in which the question is asked and to perform (some kind of) inference in order to arrive at the correct answer. The datasets on which we train and evaluate the models proposed in this paper contain both factoid (e.g. „*What is the population of South Africa*?") and non-factoid questions. Usually, this classification is important since factoid and non-factoid questions require different information and inference models in order to determine the correct answer. Ever since the Turing test has been proposed as a possible assessment for machine intelligence [1], there has been a continuous debate about what makes a machine intelligent. Is it enough that a human evaluator, just by hearing the conversation between a machine and a human, would not be able to reliably distinguish the machine from the human? First of all, the Turing test can be tricked, it the sense that a robot can reply with general answers that do not tell anything in particular. Secondly, the Turing test proposes no clear, formal rules about what questions can be asked during a conversation. Therefore, the Turing test is difficult to formalize as it can lead to various interpretations. Schoenick et al. [2] have argued that answering science questions with high accuracy can be a better way to assess machine intelligence than the Turing test. In some sense, a lot of "intelligence tests" can be transformed into an instance of question answering.

Thus, the problem approached in this paper is answering multi-choice questions from a field of science studied in grade school (e.g. chemistry, biology, physics, astronomy, etc.). All questions are expressed solely in natural language (without diagrams or equations) and each question has four possible answers from which only one is guaranteed to be correct. The system can use any publicly available resource (e.g. Wikipedia, science books, ad-hoc data corpora) as a knowledge base. The input of the system is the question itself along with four candidate answers (also expressed in natural language). The output of the system is the answer predicted to be correct and the confidence in that answer being correct.

In this paper, we propose to improve standard retrieval based techniques used for question answering with deep learning methods which have been shown to provide good results for assessing (simple) natural language inference. Deep learning models are top performers on various reading comprehension and natural language inference datasets, such as SQuAD [16], SciTail [21], or MultiNLI [23]. Moreover, deep learning has greatly improved machine translation performance, which is also a problem that requires a form of reading comprehension.

The paper continues with an overview of the domain, focusing on solutions proposed for solving the multi-choice science question answering problem described above. Then, we present our proposed method in detail and evaluate the model's performance on relevant datasets.

2 Related work

A lot of work has been done in order to improve retrieval-based baselines for question answering and to give a better sense of what a QA system could achieve with proper, structured knowledge and more complex ranking strategies for the candidate answers.

In this section, we provide a short overview of the most important approaches along with their strengths and limitations, showing where our proposed method fits within the global context and existing solutions.

Khot et al. [4] proposed a way to formulate the multi-choice question answering problem as a Markov Logic Network (MLN) problem. Each question along with its four candidate answers is translated into four questions that can be answered with either “True” or “False”, thus transforming the QA problem into a binary classification. Then, the system has to decide what questions can be answered with „True” given a knowledge base (they use a set of 4th-degree science books available on the Internet and general Web crawled information). The correct answer is the one with the greatest confidence (probability) as predicted by the system, given the knowledge base. Rules are automatically extracted from the natural language corpus (collected a priori) and represented as IF-THEN clauses, as described by Clark et al. [5]. Three MLN formulations have been proposed and evaluated: *First-order MLN*, *Entity Resolution Based MLN*, and *Probabilistic Alignment and Inference* (PRALINE).

Khashabi et al. [6] describe a method that allows formulating multiple-choice QA as a sub-graph optimization problem. It uses semi-structured information represented in tables where each row is a predicate of arity k (number of columns) over a short natural language sentence. The knowledge base has been constructed using both automated tools and manual work. The QA problem is viewed as an optimal sub-graph selection problem, where the algorithm tries to find the pair *(question, answer*) that best fits the knowledge base.

A different approach was suggested by Jansen et al. [7]. They used potential answer justifications for each candidate answer not only to improve the system’s accuracy but also to provide simple, natural language explanations for the predicted answer. These are useful when one needs very high confidence in the answer elected by the system as the correct one. For example, in the medical domain, the answer and its justification may be reviewed by a human expert in order to validate that it is, indeed, correct. Information is extracted from various sources (like study guides) and decomposed into smaller sentences (called *information nuggets*). Combining those *information nuggets* results in potential answer justifications. A perceptron is trained to order the list of justifications and to choose the most reliable one. The computed answer is the one that corresponds to the best justification.

Nicula et al. [8] proposed a model for predicting correct answers based on candidate contexts extracted from Wikipedia. Using Lucene-based indexing and retrieval, each paragraph in the English Wikipedia has been indexed and used as a candidate context for questions and corresponding answers. Each *(question, answer)* pair has been searched in the index using the standard BM25 score and the top 5 retrieved documents, along with the question and candidate answer, are concatenated and fed into a deep neural network that computes a score for the *(question, candidate, context)* triple. Two neural network architectures have been tested, that use different ways of combining the question, candidate, and context. For each architecture, two different encoders have been evaluated: bidirectional long-short term memory network (BiLSTMs) [9] and convolutional neural networks (CNNs) [10, 11].

The most similar approach to the one presented in this paper has been described by Clark et al. [12]. The proposed model combines an *information retrieval* solver, statistical information using Pointwise Mutual Information (PMI), text similarity using word embeddings to represent the lexical semantics and a simple Support Vector Machine ranker, and a *structured knowledge* solver. All of these solvers have been combined using a logistic regression classifier. The method has been tested on the NY Regents 4th Grade Science exams and it outperforms any other solutions on that dataset.

Our proposed method is different from the described approaches in the way that it tries to combine solvers that work at different representation levels: information retrieval and natural language inference using deep learning. Following this strategy, our model is able to answer both easy, factoid questions and more complex questions.

3 Proposed Method

Similar to other recent studies [7, 8, 12], we also propose to improve the retrieval-based baseline by supplementing it with other solvers that provide an alternative view for the QA problem. Thus, we combine multiple solvers that work at different representation levels in order to improve the performance of the QA system. Our solution relies solely on natural language, plain text information (we use various corpora available on the Internet: Wikipedia, CK12 books) and not on structured knowledge at all. This is an important advantage as plain text corpora are much easier to collect and do not require human intervention to extract structured rules and entities. However, it is clear that the system has to be trained to automatically encode information and to extract relevant knowledge from the plaintext, which is a difficult task given the ambiguity of natural language. Based on the latest attempts to improve multiple choice question answering with candidate contexts [8], we aim to build a supervised system that is able to predict whether an answer is correct or not given a tuple *(question, answer, relevant context)*. In order to extract a relevant context, we are going to use an efficient indexing engine backed by various collections of documents (English Wikipedia, science books) similar to other recent solutions [8, 12].

To improve the results of the QA system, we have divided the main QA task into two sub-tasks, which can be designed and tested independently:

1. Extract relevant contexts for each *(question, candidate answer)* pair using an information retrieval approach.
2. Construct various (more complex) models to predict if an answer is correct based on additional information which can be inferred from the context.

Each model or solver from the second step should be able to tackle the QA problem at different levels of abstraction. Later, the results provided by all these solvers, including the retrieval-based one for fetching relevant contexts, are combined using an ensemble regression or voting mechanism.

In our research, we decided to use deep neural networks for the second stage solvers, as they have been recently shown to solve (simple) textual inference problems [21]. The combining mechanism is another neural network, however, any other non-linear regression could have been used to compute the score for a candidate answer to be the correct answer for a given question. In the following sections, we describe each component of the proposed solution, focusing on the solvers in the second step. In the end, we evaluate both the system as a whole and the performance of each solver in particular.

3.1 Extracting relevant contexts

As mentioned, the first step requires to generate relevant contexts for each candidate answer and a given question, which is basically a retrieval problem. We use Lucene to index a large collection of documents (entire English Wikipedia, science books collected over the Internet[1], ARC Corpus[2]) pertinent for the science QA task at hand and later to retrieve relevant information for candidate answers.

The documents have been filtered to include only affirmative sentences without references to images or tables. The queries contain *(question, candidate answer)* pairs and they are looked up in the Lucene index and ranked based on the default BM25 score used by Lucene [13] with term boosting. The top-scoring document is chosen as the most relevant context for each candidate answer. This method also offers a simple baseline for testing the accuracy of the system: given a question and all its candidate answers, pick the answer with the largest score retrieved by Lucene as the correct answer.

The question and the retrieved contexts will be fed in the second step into multiple solvers. These are built based on several neural network architectures, each with a different task as will be further detailed in this section. Term boosting scores are computed using another simple neural network trained to predict the essentialness [14, 24] of each term in a question. Essentialness has been defined as the degree to which removing that term from the question makes it impossible to answer by a human and has shown to improve standard retrieval-based methods used for QA. The neural network has been trained on the dataset collected by Khashabi et al. [14]. In Fig. 1 we provide an example of the scores generated by this model for a sample question. These essentialness scores are very important in improving the performance of the IR model as shown in [24]. On the ARC Easy dataset, they can increase the accuracy by up to 4%. Furthermore, this improvement indicates that the extracted candidate contexts are more relevant and the neural networks that perform inference on those contexts are likely to benefit from this.

In total, we have indexed about 15 GB of raw text, most of which was generated from the Wikipedia's March 2018 dump (> 85%). However, the science book collection and ARC Corpus turned out to be very useful in increasing the accuracy of the system. The reason behind this behavior is that both corpora contain information very concentrated and aimed at the science field, whereas the Wikipedia corpus covers a wider spectrum of information that is not always relevant to science questions.

This IR component serves two purposes. First, it provides the context for all *(question, candidate answer)* pairs that is required in order to perform textual inference. Second, it defines the first solver of the ensemble – the answer with the highest TF-IDF score is elected as the right one. This kind of simple solver helps the system to decide upon the correct answer in the situation where the inference solvers output almost equal

[1] https://github.com/SebiSebi/AI2-Reasoning-Challenge-ARC-Data (as of August, 2018)

[2] http://data.allenai.org/arc/arc-corpus/ (as of September, 9th, 2018)

probabilities for all candidate answers. This is a tie that can be solved by looking at the TF-IDF scores. For the interested reader, Appendix A gives examples of contexts extracted using the Information Retrieval component described in this subsection.

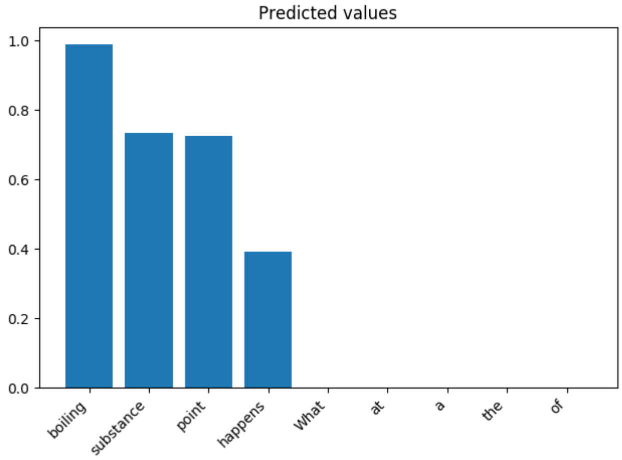

Fig. 1. Essentialness term scores computed for the question: "*What happens at the boiling point of a substance?*"

3.2 Predicting the correct answer based on the extracted context

In the second stage, solvers reason about the triplet *(question, candidate answer, context)* on different levels such as computing an enhanced semantic similarity or determining whether the triplet contains a natural language inference relation.

To compute an enhanced semantic similarity between the question, answer and contest, we propose to use a deep neural network that takes as input a question, a candidate answer, and the context and outputs the predicted probability that the current candidate answer is correct. The answer with the highest probability of being correct is chosen as the final answer. The neural network's architecture is designed around the Bidirectional attention flow (BiDAF) architecture proposed by Seo et al. [15] for reading comprehension, a special case of QA.

The BiDAF model proposes a couple of important ideas that combined can improve the performance of vanilla LSTM models when exposed to large input sequences with long dependencies. We have modified the BiDAF architecture to fit our needs by altering the final output layer and also introducing the answer as an additional input in the network. The attention flows bidirectionally from the context to the question and from the question to the context. This mechanism allows the network to focus on the most important parts of the context. In general, the contexts extracted with the information retrieval engine can be large (~300 words, on average), however, only some shorter text spans are used to actually find the correct answer. Consider the following scenario, in which the context is: *„The general theory of relativity which was developed by Einstein between 1907 and 1915 is one of the most important theories in physics. ... [Theory* *description here]... It explains the inner workings of the gravitational force and how it is linked to other forces in nature.”*. Let us consider the following question: *„Between what years did Einstein develop the theory which explains the source of the gravitational field?”*. It seems natural to skip the part of the context which provides details about the general theory of relativity because it does not offer any information helpful in answering the question above.

Furthermore, some science questions are intentionally injected with extra information that is not helpful in answering the question. These longer questions may cause problems for QA systems that have not been trained to avoid this situation. In this set of circumstances, the BiDAF architecture comes in handy by providing the bidirectional attention flow – it allows the neural network to “skip” parts from the context based on the question and, also, to “skip” parts of the question given the context.

Training a BiDAF network requires a large dataset due to its large number of parameters. Unfortunately, all the available multi-choice science QA datasets contain too few examples to train this model. On the other hand, reading comprehension datasets like SQuAD [16] are large enough to allow complex neural networks to be trained. The only problem is that SQuAD contains open questions, not multi-choice questions. We need to transform this problem into a multi-choice QA instance. One important aspect of the SQuAD dataset is that for each question a good context is given (the document to choose a shorter span from) and it is guaranteed that the correct answer can be found within that context. We use this property to generate wrong answers and transform the SQuAD dataset into one suitable for multi-choice question answering. However, generating random wrong answers (even random answers from within the same context) will be an over-simplistic problem. Indeed, using that approach, a simple LSTM network achieves 97% accuracy. The method we propose is to use the fact that multiple questions refer to the same article and their correct answers can be considered relevant wrong answers to other questions. Therefore, in order to generate wrong answers to a given question we look at all other questions based on the same document and randomly pick correct answers for these questions. This assures that, to some extent, wrong answers to a question, are not trivially “wrong”. For example, we may have the following two questions given the same article: *“Which NFL team represented the AFC at Super Bowl 50?”* and *“Which NFL team represented the NFC at Super Bowl 50?”*. If we look at all the possible correct answers to the second question these are clearly wrong answers for the first one. Apart from that, a system which pretends to answer both questions needs to have some comprehend ability in order to distinguish between answers.

We use the modified SQuAD v1.1 dataset to train the adapted BiDAF architecture and then fine-tune the model’s parameters by using continuous training on smaller multi-choice science QA datasets that we are interested in (check the results section for a description of these datasets). The hypothesis is that the neural network has learned on the larger dataset to encode core characteristics that are used in general and can be reused to answer new, unseen questions even of a slightly different nature (domain, method of generation). These core characteristics are inputs to the decision layers of the neural network and a combination of these dictates if the answer is correct or not.

In our implementation of the model, we used pre-trained GloVe 50D word embeddings [17]. One may argue that GloVe captures similarities between words from the same semantic field, but with slightly different interpretations (e.g. “man” and “woman”, “queen” and “king”). This issue has been previously mentioned by Pennington et al. [17] and can be crucial for questions answering, where such subtle differences in meaning are important for determining the correct answer. This is one of the reasons for which the BiDAF model also uses char-level embeddings and a special layer capable of altering some word embeddings if necessary by using highway networks [18].

3.3 Recognizing Textual Entailment (RTE)

We also investigate how to transform the plain QA task into a natural language inference task to add a new solver to the second stage. Given a question and a candidate answer, the question is first translated into an affirmative statement containing the term *@placeholder* where the answer needs to be filled in. Then we replace *@placeholder* with the answer itself and this forms the hypothesis of the entailment. The premise is the context extracted using the information retrieval system in the first stage. In this scenario, we can actually use multiple contexts (not only the context with the highest TF-IDF score) and consider the average result.

We run a standard RTE model on each premise and the average score is considered as the probability that the entire set of premises entails the hypothesis. Following this strategy, we first need a way to translate questions into statements and then a well-performing (neural) model for recognizing textual entailment.

Using an empiric approach over the SQuAD v1.1 dataset, we have manually identified 36 different question types. For each question type, a set of rules (mostly based on the syntactic dependency tree generated using spaCy[3]) has been proposed in order to translate between the question and the corresponding statement. On average, each question type has four rules (applied in order of their priority). Therefore, a set of around 150 rules is used. Together, the rules cover over 90% of the questions available in the ARC dataset (as evaluated by the authors) and produce natural, human-readable statements (which are grammatically correct). See Appendix B for a complete list of the question types and Table 1 for translation examples. For questions consisting of several sentences, out of which only one is interrogative, the algorithm only translates the interrogative sentence into affirmative form.

The actual model that performs RTE is a bidirectional attention model based on the BiDAF architecture mentioned earlier, with the following modifications:

a) The context is considered the premise and the translated *(question, answer)* pair represents the hypothesis;
b) The final layer is a 3-way softmax function used for the determining the type of textual entailment: entailment, neutral, or contradiction.

[3] https://spacy.io/ (as of September, 9th, 2018)

Table 1. Question to statement translation examples.

Original question	Statement
Which of these is a greenhouse gas?	@placeholder is a greenhouse gas.
What do plant roots prevent?	Plant roots prevent @placeholder.
What does FIFA stand for?	FIFA stands for @placeholder.
Where is corruption most noticeable?	Corruption is most noticeable in @placeholder.
Who is the Microsoft owner?	@placeholder is the Microsoft owner.
When was ENIAC fully operational?	ENIAC was fully operational on @placeholder.
What year did Chopin die?	@placeholder (year) Chopin died.
Which ocean does Portugal border?	@placeholder (ocean) Portugal border.

For training the neural network, we used three publicly available datasets for natural language inference, each with its special characteristics:

a) The Stanford Natural Language Inference (*SNLI*) corpus [19] – a collection of 550,000+ English sentence pairs manually labeled with entailment, contradiction, or neutral. All sentences are based on image captions and thus are not representative to science-like questions.

b) The Multi-Genre NLI (*MultiNLI*) corpus [20] is similar to the SNLI corpus but offers a wider range of genres (not only visual, image captions). The dataset contains over 415,000 English sentence pairs from fiction, travel, and government sources.

c) The *SciTail* dataset [21] consists of 27,000 entailment pairs of sentences (like SNLI or MultiNLI) but created from multiple-choice science questions. This dataset is particularly important since it contains science related sentences.

The models resulted from training the adapted BiDAF architecture on those three particular datasets are three independent solvers due to the core differences in the dataset structure, genre, and semantic content.

3.4 Combining all solvers

So far, we described the three main approaches proposed in this paper to solve the multi-choice QA problem: using information retrieval (optimized with essentialness terms), using enhanced semantic similarity computation starting from reading comprehension datasets and using natural language inference (NLI). In the last stage, we combine all these models into a single one using another simple fully connected network taking the role of an ensemble. To summarize, for a tuple *(question, answer, context)* multiple predictors give the following results:

a) The TF-IDF score from Lucene – normalized with the softmax function;

b) The QA score from the reading comprehension model;

c) The NLI scores from the natural language inference model – trained separately on *SNLI*, *MultiNLI*, and *SciTail* datasets;

Each of the solvers outputs a single real number: TF-IDF scores or probabilities. It is important for the neural network to receive information about the scores of the other candidates. Most of the time, absolute values do not matter: is an answer with a score of 0.45 (out of 1) correct? It depends: there may be another answer with 0.45 (and the other two 0.05 and 0.05) or it may be the case that the remaining answers have low scores like 0.15; 0.20, 0.20; It seems reasonable to input into the neural network a score that is relative to the second best score (that is the difference between the current score and the best one out of the other candidates).

4 Results

We evaluate our model[4] on the AI2 Reasoning Challenge (ARC) dataset by Clark et al. [22] which contains 7,787 science related questions, partitioned into a challenge set and an easy set based on questions difficulty. The neural network that combines the results of all solvers is trained on the ARC Easy/Challenge datasets and its hyperparameters (e.g. number of neurons in the hidden layers, dropout probabilities) are fine-tuned on the dev datasets. We report results on the ARC test sets, as well as on the dev set. However, as hyper-parameters were tuned on the dev set, the results are affected by over-fitting and only the test results are relevant.

In Table 2, ”IR Single” refers to the best information retrieval model (with essential term boosting) using a single knowledge base. On the other hand, ”IR Multiple” refers to retrieval-based model applied to multiple knowledge bases (ARC Corpus, science book collection, and Wikipedia) and taking the average score over them. It is important to notice that the model with Multiple KBs actually has lower accuracy than the best Single KB model on the Challenge dataset. This can be explained considering the fact that on complex questions each IR component can output irrelevant contexts. When taking the average score, even if the score from one source (e.g. ARC Corpus which provides the best single IR score) is reliable, the total effect is reduced due to averaging.

Table 2. P@1 of the two-stage combined model and individual solvers on the ARC datasets

Model	ARC Easy Dev	ARC Easy Test	ARC Challenge Dev	ARC Challenge Test
Random choice	25.00%	25.00%	25.00%	25.00%
QA (SQuAD)	25.40%	26.89%	24.75%	22.15%
BiDAF (SNLI)	27.34%	29.94%	27.12%	24.64%
BiDAF (MultiNLI)	29.98%	32.18%	25.76%	25.67%
BiDAF (SciTail)	30.16%	30.57%	30.51%	23.00%
IR Single KB	57.31%	56.99%	25.42%	24.72%
IR Multiple KBs	61.38%	61.10%	22.71%	23.78%
Two-stage model	**61.90%**	**61.10%**	**31.18%**	**26.86%**
Difference	***+0.52%***	***+0.0%***	***+8.47%***	***+3.08%***

[4] Source code at https://github.com/SebiSebi/AI2-Reasoning-Challenge-ARC

By comparing to the Multiple KBs solver, our proposed two-stage model increases the accuracy by 3.08% on the Challenge test dataset. However, on the Easy test dataset, the performance is not boosted by the two-stage model due to the nature of the questions and the fact that IR outperforms all inference solvers. With respect to the IR Single KB model, however, our proposed model improves the performance by over 4% on the Easy test set.

It is important to notice that on the Challenge dataset, the best inference model (BiDAF trained on MultiNLI), outperforms both the IR Single KB and the IR Multiple KBs models. Furthermore, the accuracy of the two-stage model is 1.19% higher than the BiDAF (MultiNLI) model, suggesting that it has learned to combine the scores in such a way that the end-to-end performance is increased even though the local components individually don't perform that well.

It is also useful to take a look at some of the questions that are answered incorrectly by the information retrieval models but are answered correctly by the combined model:

- „*All organisms depend on the transfer of energy to survive. Which best shows the energy transfer between animals in a shoreline ecosystem?*" A) Fish -> Plants -> Birds; B) Plants -> Birds -> Fish; C) Plants -> Fish -> Birds; D) Fish -> Birds -> Plants;
- „*Patricia and her classmates are visiting different rocky areas in the city. They want to know what kind of rocks can be found in each area. If the investigation is done correctly, what will Patricia and her classmates do each time they visit an area?*" A) draw the rock shapes; B) count the rock numbers; C) record the rock types and locations; D) measure the rock masses and lengths;
- „*Dr. Wagner is investigating a newly discovered, disease-causing agent. She determines that one structure in the agent is double-stranded RNA. What kind of agent is Dr. Wagner studying?*" A) a virus; B) a protis; C) a fungus ; D) a bacterium;

All these questions require some form of simple textual inference to determine the correct answer. Our model inference solvers are trained to perform this kind of logical deduction and thus improve the accuracy compared to the IR solvers on the Challenge dataset. It is important to note that the 3% absolute improvement means more than 10% relative improvement compared to IR Multiple and over 8% relative improvement compared to the best IR single model.

In comparison to other state-of-the-art models (as given by the ARC Leaderboard[5]), our combined model is ranked second on the Easy dataset and eighth on the Challange dataset. Moreover, our model is the only one that performs well on both Easy and Challenge datasets. This is because it combines solvers that work at different levels and tackles the questions from various directions: simple information retrieval and more complex textual inference and enhanced semantic similarity.

If we do a quick error analysis on a subset of the ARC Challange dataset, we observe that about 50% of the questions are answered incorrectly due to insufficient support from the knowledge base and the IR component. This means that the for about half of the questions the extracted contexts are not helpful in finding the correct answer (e.g. even a human would not be able to pick the correct answer given only the information in the context). Therefore, using the proposed two-stage strategy, the second stage

[5] http://data.allenai.org/arc/ (as of October 1st, 2018)

solvers can only improve the results for about half of the questions in the dataset. However, more complex solvers are still needed to improve the current results.

One important advantage of the proposed two-stage model is that it brings together different natural language solvers that reason about the contexts generated by the retrieval-based models. The overall performance can be improved in both stages: better retrieval methods can generate more relevant contexts, while adding more solvers that work on different levels can improve the second stage inference processes. The current solvers highlight that retrieval-based QA can be easily improved by adding deep learning solvers for more advanced natural language processing (inference, similarity) without adding any complex feature engineering, rules, or linguistic expertise. Moreover, this is one of the first papers that shows transfer learning can be useful for improving QA systems on specific domains (e.g. science) with small datasets.

5 Conclusions

In this paper, we described a two-stage model that combines different solvers in order to tackle the multi-choice science question answering problem. In the first stage, we deploy an information retrieval solver with essentialness term boosting working on different knowledge bases to generate relevant candidate contexts for each (q*uestion, candidate answer)* pair. In the second stage, we employ more complex models based on deep neural networks to further analyze each triplet *(question, candidate answer, context)* to determine whether there is a natural language inference relation present or a more complex semantic similarity between the context, question, and candidate answer. In the end, all solvers including the IR one are combined to determine the most probable answer.

The model has been trained using continuous training so as to overcome the issue of small datasets available for the multi-choice science QA task. Our system outperforms the retrieval-based models on both the easy, factoid questions (only on dev) and on more challenging questions (both on dev and test) where an information retrieval solver alone is not powerful enough to determine the correct answer. Attention mechanisms allow deep neural networks to focus only on the important aspects of the extracted candidate contexts and ignore large, irrelevant spans.

The proposed model can potentially be improved by using a better knowledge base to find candidate contexts and by adding additional solvers to the ones used in this paper. For example, we aim to add new solvers that use some sort of structured information to tackle even more difficult questions. Moreover, our model would highly benefit from larger datasets (science questions in particular) that could be used to successfully train elaborate neural network architectures and learn more complex hypothesis for natural language inference.

References

1. Turing, A. M. (2009). Computing machinery and intelligence. In Parsing the Turing Test (pp. 23-65). Springer, Dordrecht.
2. Schoenick, C., Clark, P., Tafjord, O., Turney, P.D., & Etzioni, O. (2017). Moving Beyond the Turing Test with the Allen AI Science Challenge. Commun. ACM, 60, 60-64.
3. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436.
4. Khot, T., Balasubramanian, N., Gribkoff, E., Sabharwal, A., Clark, P., & Etzioni, O. (2015). Markov logic networks for natural language question answering. arXiv preprint arXiv:1507.03045.
5. Clark, P., Balasubramanian, N., Bhakthavatsalam, S., Humphreys, K., Kinkead, J., Sabharwal, A., & Tafjord, O. (2014, December). Automatic construction of inference-supporting knowledge bases. In 4th Workshop on Automated Knowledge Base Construction (AKBC).
6. D. Khashabi, T. Khot, A. Sabharwal, P. Clark, O. Etzioni, and D. Roth. 2016. Question answering via integer programming over semi-structured knowledge (extended version). In Proc. 25th Int. Joint Conf. on Artificial Intelligence (IJCAI).
7. Jansen, P., Sharp, R., Surdeanu, M., & Clark, P. (2017). Framing qa as building and ranking intersentence answer justifications. Computational Linguistics, 43(2), 407-449.
8. Nicula, B., Ruseti, S., & Rebedea, T. (2018, March). Improving Deep Learning for Multiple Choice Question Answering with Candidate Contexts. In European Conference on Information Retrieval (pp. 678-683). Springer, Cham.
9. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
10. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
11. Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.
12. Clark, P., Etzioni, O., Khot, T., Sabharwal, A., Tafjord, O., Turney, P. D., & Khashabi, D. (2016, February). Combining Retrieval, Statistics, and Inference to Answer Elementary Science Questions. In AAAI (pp. 2580-2586).
13. Manning, C. D., & Raghavan, P. H.(2008) Introduction to Information Retrieval. Cambridge University Press, New York, NY, USA, 3, 38-48.
14. Khashabi, D., Khot, T., Sabharwal, A., & Roth, D. (2017). Learning What is Essential in Questions. In Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017) (pp. 80-89).
15. Seo, M., Kembhavi, A., Farhadi, A., & Hajishirzi, H. (2016). Bidirectional attention flow for machine comprehension. arXiv preprint arXiv:1611.01603.
16. Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. arXiv preprint arXiv:1606.05250.
17. Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).
18. Srivastava, R. K., Greff, K., & Schmidhuber, J. (2015). Highway networks. arXiv preprint arXiv:1505.00387.
19. Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015). A large annotated corpus for learning natural language inference. arXiv preprint arXiv:1508.05326.

20. Williams, A., Nangia, N., & Bowman, S. R. (2017). A broad-coverage challenge corpus for sentence understanding through inference. arXiv preprint arXiv:1704.05426.
21. Khot, T., Sabharwal, A., & Clark, P. (2018). SciTail: A textual entailment dataset from science question answering. In Proceedings of AAAI.
22. Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., & Tafjord, O. (2018). Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. arXiv preprint arXiv:1803.05457.
23. Williams, A., Nangia, N., & Bowman, S. R. (2017). A broad-coverage challenge corpus for sentence understanding through inference. arXiv preprint arXiv:1704.05426.
24. *** Reference removed for anonymity ***

Appendix A

Below, we give some examples of contexts extracted using the Information Retrieval solver for the correct answer only. That is, we always look for the pair (question, *correct answer*). It is clear that some system equipped with the right inference engine should be able to mark the candidate answer as the right answer with a fairly high probability.

1. Question: "*To express the distance between the Milky Way galaxy and other galaxies, the most appropriate unit of measurement is the?*" – Correct answer: "light-year" - Generated context: "*Light-Years (we need a really big unit to measure distances out in space because distances between stars are so great. 9 trillion miles), is the distance that light travels in one year.*"

2. Question: "*Volume and mass are properties of?*" – Correct answer: "matter" – Generated context: "*The mass of an object is a measure of the amount of matter that an object contains. A small sample of a certain type of matter will have a small mass, while a larger sample will have a greater mass. The volume of an object is a measure of the space that is occupied by that object. Volume and mass are common properties of matters.*"

3. Question: "*In which of the following organs is an acid produced?*" – Correct answer: "stomach" – Generated context: "*Hydrochloric acid (HCl), which is released from cells in the lining of the stomach, is a strong acid because it releases all of its H+ in the stomach's watery environment*"

4. Question: "*What event occurs because of the rotation of Earth?*" – Correct answer: "night and day" – Generated context: "*As Earth turns, the Moon and stars change position in our sky. Earth's Day and Night Another effect of Earth's rotation is that we have a cycle of daylight and darkness approximately every 24 hours. As Earth rotates, the side of Earth facing the Sun experiences daylight, and the opposite side (facing away from the Sun) experiences darkness or nighttime. Since the Earth completes one rotation in about 24 hours, this is the time it takes to complete one day-night cycle.*"

5. Question: "*Which part of a sound wave measures loudness?*" – Correct answer: "amplitude" – Generated context: "*The amplitude, or height of the sound wave, determines how much energy it contains and is perceived as loudness (the degree of sound volume). Loudness is measured using the unit of relative loudness known as the decibel. Zero decibels represent the absolute threshold for human hearing, below which we cannot hear a sound.*"

Appendix B

Table 3. A complete list of the question types used in our model.

Question type	Example
WHICH_OF	*Which of these is a greenhouse gas?*
IN_WHICH_OF	*In which of the following types of cells does cellular respiration occur?*
REPLACE_UNDERSCORES	*The atoms in a can of soda are* __________.
WHAT_BE	*What is an example of a natural satellite?*
WHAT_DO	*What does vastenavond mean?*
WHERE_BE	*Where is the biggest city in the world located?*
WHERE_DO	*Where did Sheptycki study police cooperation?*
WHO	*Who was the founder of the Kirata dynasty?*
HOW_MANY	*How many seats are in the stadium?*
IN_WHAT	*In what structure is photosynthetic tissue found?*
WHEN_DO	*When did Greece adopt the Euro?*
WHEN_BE	*When is the genitive case used?*
WHAT_NOUN	*What type of galaxy is the Milky Way?*
WHICH_NOUN	*Which stadium is in the South Bronx?*
WHICH_BE	*Which is Melbourne's largest dam?*
IN_WHICH_NOUN	*In which structure is the Sun located?*
WHY	*Why is the Sun orbiting the Earth?*
WHAT_VERB	*What caused the loss to Steaua in Seville?*
WHAT_GENERAL	*What in tobacco can hurt dogs?*
HOW_MUCH	*How much land did Bronck eventually own?*
HOW_LONG	*How long is the Suez canal?*
HOW_DO	*How do producers get energy?*
HOW_BE	*How is adolescence defined socially?*
THE	*The Permian is an example of what?*
ALONG_WITH	*Along with Charles, who was the son of Pippin?*
ACCORD_TO	*According to Avicenna, what always exists?*
ON_WHAT	*On what date was the Aviation School founded?*
IN_SMTH_QUESTION	*In 1995, who managed the girl's group?*
START_WITH_NOUN	*Methodism is part of what movement?*
START_WITH_PROPER_NOUN	*Mrigavyadha means what?*
START_WITH_A	*A natural camo pattern is known as what?*
HOW_ADVJ	*How large is Notre Dame in acres?*

WHICH_VERB	*Which can be regarded as an interesting project?*
WHICH_GENERAL	*Which famous landmark did Mark see in China?*
START_WITH_BE	*Is balsa a softwood or a hardwood?*
START_WITH_THIS	*This is an example of...?*