Natural Language Engineering (2024), **30**, pp. 73–95
doi:10.1017/S1351324922000286

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

Artificial fine-tuning tasks for yes/no question answering

Dimitris Dimitriadis* and Grigorios Tsoumakas

Aristotle University of Thessaloniki, Thessaloniki, Greece
*Corresponding author. E-mail: dndimitri@csd.auth.gr

(Received 2 March 2021; revised 24 May 2022; accepted 30 May 2022; first published online 30 June 2022)

Abstract

Current research in yes/no question answering (QA) focuses on transfer learning techniques and transformer-based models. Models trained on large corpora are fine-tuned on tasks similar to yes/no QA, and then the captured knowledge is transferred for solving the yes/no QA task. Most previous studies use existing similar tasks, such as natural language inference or extractive QA, for the fine-tuning step. This paper follows a different perspective, hypothesizing that an artificial yes/no task can transfer useful knowledge for improving the performance of yes/no QA. We introduce three such tasks for this purpose, by adapting three corresponding existing tasks: candidate answer validation, sentiment classification, and lexical simplification. Furthermore, we experimented with three different variations of the BERT model (BERT base, RoBERTa, and ALBERT). The results show that our hypothesis holds true for all artificial tasks, despite the small size of the corresponding datasets that are used for the fine-tuning process, the differences between these tasks, the decisions that we made to adapt the original ones, and the tasks' simplicity. This gives an alternative perspective on how to deal with the yes/no QA problem, that is more creative, and at the same time more flexible, as it can exploit multiple other existing tasks and corresponding datasets to improve yes/no QA models.

Keywords: Question answering; Machine learning; Yes/no question answering; Transfer learning

1. Introduction

Question answering (QA) is one of the most challenging tasks in natural language processing (NLP). Unlike other NLP tasks, such as morphological and syntactical parsing, where computational models have been proposed years ago (Jurafsky and Martin 2000), only recent studies in QA show a systematic way of solving this problem.

In the past, QA was mainly treated as an engineering task, combining a lot of resources and computational models in a pipeline fashion (Athenikos and Han 2010; Lally *et al.* 2012; Gupta and Gupta 2012; Bouziane *et al.* 2015; Pundge, Khillare, and Mahender 2016), while it was not clear how to jointly encode the question and the relevant passages into a learning framework. Based on their expertise on QA, researchers were typically engineering features, which combine the resources with statistics, for machine learning algorithms (Peng *et al.* 2015; Yang *et al.* 2016; Dimitriadis and Tsoumakas 2019a).

The systematic use of word embeddings in NLP and the rise of deep learning changed the scene. Researchers focused their attention on designing complex neural network models that are able to learn the regularities of the question and the relevant passages (Weissenborn, Wiese, and Seiffe 2017; Xiong, Zhong, and Socher 2017; Rondeau and Hazen 2018). Remarkable results have been presented to the community following this perspective. During that period, the attention

mechanism (Vaswani *et al*. 2017) led to the transformers, a new family of neural networks. The results became even better and easy variations of the QA task were solved. In the SQuAD challenge (Rajpurkar *et al*. 2016; Rajpurkar, Jia, and Liang 2018), for example, the best learning model achieved an F1 score of 93% in the reading comprehension task, outperforming the human performance of 89.45%.[a] Empirical results have shown that transfer learning is a key concept for improving the performance of learning models both in QA as well as in other NLP tasks.

This paper focuses on the last component in an end-to-end QA system, where a learning model decides the answer to a given question based on a reference text. We assume that an information retrieval model has already returned the most relevant documents (Kolomiyets and Moens 2011). Then, a second model, such as an answer sentence selection model (Yu *et al*. 2014), has already filtered those documents and returned a part of the text that is more relevant to the given question.

We address the yes/no QA task, that is, questions that can be answered with either a *yes* or a *no*. Transformer-based models and transfer learning are the cutting-edge technologies in yes/no QA (Yin *et al*. 2020; Ignatov 2021). The main existing research insights are that: i) adapting pre-trained language models to other tasks improves the accuracy in yes/no QA and ii) the higher the similarity of these tasks to yes/no QA, the better the results.

To the best of our knowledge, all previous studies focus on existing real tasks that are quite similar to yes/no QA. This puts a limit on the opportunities for fine-tuning and to the corresponding transfer of knowledge from such tasks. Our key novel idea in this work is to instead explore the more flexible pathway of constructing artificial yes/no tasks based on other existing tasks. Our main research question is " *can the fine-tuning of pre-trained language models on artificial yes/no tasks improve the performance in yes/no QA?*".

To answer this question, we introduce three artificial yes/no tasks by adapting three corresponding existing tasks, namely candidate answer validation, sentiment classification, and lexical simplification. We experiment with three different variations of the BERT model to gather quantitative empirical evidence on our research question, which we accompany with a qualitative analysis based on the representations learned by BERT. In summary, the main contributions of this paper are as follows:

(1) A new perspective on dealing with the yes/no QA task. Instead of focusing on similar existing tasks, we propose the construction of similar artificial tasks by adapting other existing tasks. This general perspective could be applicable to other QA and NLP tasks.

(2) Three novel yes/no tasks that can be used for transferring knowledge to language models toward yes/no QA.

(3) An empirical study that shows the effectiveness of constructing artificial yes/no tasks for enhancing the performance on yes/no QA, along with corresponding insights.

The rest of this article is organized as follows. Section 2 introduces the new yes/no tasks. Section 3 describes the experimental setup, Section 4 presents the results, and Section 5 the qualitative analysis. Section 6 reviews related work in yes/no QA. Section 7 discusses the results in the context of relevant past work. Finally, Section 8 concludes this work and proposes future directions.

2. Tasks

This section introduces three novel yes/no transfer tasks for yes/no QA, each addressing a corresponding challenge of yes/no QA. The first challenge we consider is the limited guidance offered by the binary annotations of the training examples about how to reason for the answer to a given question. A yes/no QA model is responsible for learning patterns considering only the fact that

[a]Scores taken from the leaderboard at https://rajpurkar.github.io/SQuAD-explorer/.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

the answer is yes or no. Contradictory, in extractive QA, a model learns a probability distribution over the input reference passage guided by stronger supervision about the start and the end of the answer in the passage text. We hypothesize that adapting a language model on a yes/no task with additional supervision about the truth of a question within the referenced passage could benefit the performance of the model in yes/no QA. We expect the final model to learn this way where to pay attention for reasoning about the truth of the question. Toward this direction, we create an artificial yes/no task based on the existing task of answer validation (Section 2.1).

A second challenge in yes/no QA is recognizing whether the question agrees or contradicts the referenced passage. This objective is different from those of other question formats. For instance, a model for answering factoid questions has to understand the difference between the context of a possible answer and the answer itself. This is possibly a reason why models fine-tuned on the entailment task achieved better results in yes/no QA, in contrast to those fine-tuned on extractive QA (Clark *et al*. 2019a). The polarity of the words in a question and a passage constitutes useful information about the truth of a question. Terms in the question with contradictory polarity compared to terms in the referenced passage could hint that the answer is probably no. On the other hand, if the polarities of words in both question and passage are in agreement, this could hint that the answer is yes. A sentiment classifier for a novel yes/no task considering pairs of sentences could provide such knowledge about the agreement or contradiction between the polarities of the two sentences. We introduce this task in Section 2.2.

The third challenge of yes/no QA that we consider is that often there is lack of direct evidence in the reference passage about answering a question with a yes or a no. Contradictory, in multiple-choice QA, there is always at least one true answer to be selected, while most of the extractive QA systems assume that the answer is part of the referenced passage. To simulate this lack of evidence with training examples, we construct a new yes/no task based on lexical simplification, called simplification validation (Section 2.3).

2.1 Answer validation

Before the rise of neural networks in QA, several approaches were including a post-processing phase after the answer processing step in order to validate the answer. In specific, an *answer validator* was responsible for getting a collection of candidate answers and for deciding whether one of them is the correct answer for a given question or not (Magnini *et al*. 2002; Pakray *et al*. 2011). Nowadays, this process is covered by the same model, which both extracts the candidate answers and scores them based on their probability of being correct.

Instead of considering the input of the answer validator to be the question and a candidate answer, we assume that a transformer-based model gets as input the question and the relevant passage, with the addition of two *special tokens*, [SA] and [EA], surrounding a piece of text in the passage. The model's goal is to learn to predict if the designated piece of text corresponds to the answer or not. Figure 1 illustrates the differences of the new task with respect to the original answer validation task.

For example, consider the following pair of question (Q) and passage (P) from the SQuAD dataset, where the correct answer is "Xerox":

Q: Beyonce's father worked as a sales manager for what company?

P: Beyonc Giselle Knowles was born in Houston, Texas, to Celestine Ann "Tina" Knowles (ne Beyinc), a hairdresser and salon owner, and Mathew Knowles, a Xerox sales manager . . .

We create a positive training example by surrounding the answer with the special tokens:

P: Beyonc Giselle Knowles was born in Houston, Texas, to Celestine Ann "Tina" Knowles (ne Beyinc), a hairdresser and salon owner, and Mathew Knowles, a **[SA]** Xerox **[EA]** sales manager . . .

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

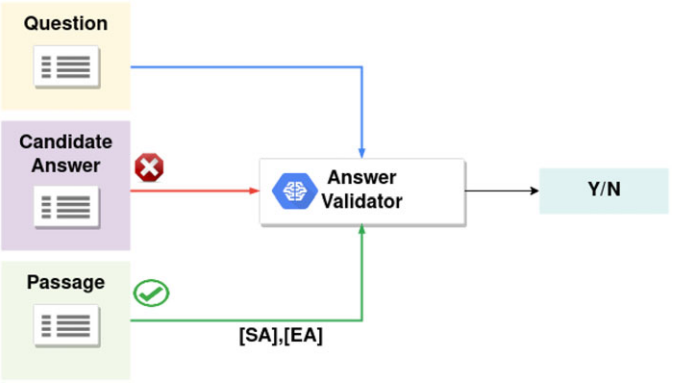

Figure 1. The adaptation of the answer validation task. An answer validator gets as input a question and an enriched passage, instead of a candidate answer, to predict if the special tokens are in the correct position. The check mark (green color) indicates the new input to the answer validator, while the X mark (red color) indicates the input in the original task that we removed. The line without mark (blue color) indicates the part that we keep the same before and after the task adaptation.

We create negative training examples in two different ways. In the first one (AV1), we put the special tokens around a random sequence of words from the passage, different from the actual answer, but with the same number of words as the actual answer. An example of a passage of such a negative training example is:

P: Beyonc Giselle Knowles was **[SA]** born **[EA]** in Houston, Texas, to Celestine Ann "Tina" Knowles (ne Beyinc), a hairdresser and salon owner, and Mathew Knowles, a Xerox sales manager . . .

In the second one (AV2), we put the special tokens around the first sequence of words from the passage that has the same number of words and the same part-of-speech tags as the actual answer, excluding the actual answer. We revert to AV1, if such a sequence is not found in the passage. An example of a passage in such a negative training example is:

P: **[SA]** Beyonc **[EA]** Giselle Knowles was born in Houston, Texas, to Celestine Ann "Tina" Knowles (ne Beyinc), a hairdresser and salon owner, and Mathew Knowles, a Xerox sales manager . . .

A transformer-based model will discover patterns to validate that the text span inside the special tokens is the correct one, considering both the context of the span text and the given question. Although the special tokens will be missing in the yes/no QA task, the model will have learned to pay attention to text that is useful for answering the questions. We hypothesize that such an adjustment of attention weights will benefit the yes/no QA task and improve the models' performance.

2.2 Sentiment classification in pairs

A typical sentiment classifier categorizes a given text as either positive or negative. Transformer-based models are capable of predicting the polarity of a text with high accuracy (Gao *et al.* 2019; Munikar, Shakya, and Shrestha 2019). One way to modify the task is to consider pairs of sentences with the same polarity as positive examples and pairs of sentences with opposite polarity as negative examples. The model could learn whether there is a contradiction or agreement between the sentences. However, the model will not be able to distinguish the polarity of the words.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

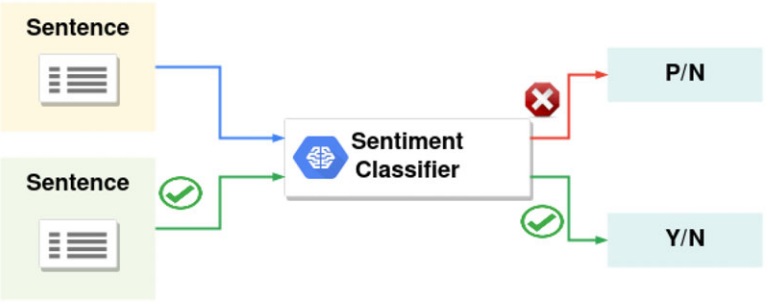

Figure 2. The adaptation of the sentiment classification task. Instead of classifying a sentence as negative or positive, the model answers the question *are the two sentences both positive?* The check mark (green color) indicates the new input and output of the sentiment classifier, while the X mark (red color) indicates the original output that we removed. The line without mark (blue color) indicates the part that we keep the same before and after the task adaptation.

Consider the following two positive (same polarity) examples as input to such a model, the first one containing two positive and the second one two negative sentences:

S1: Quick delivery and item is working!
S2: Got it. Excellent job of packing! Everything works perfectly! Thanks!
S1: Slow delivery and item is not working!
S2: Got it. Poor job of packing! Nothing works perfectly!

During training, the model has to discover patterns considering the direct relation between the sentences in each example and the indirect relation between the examples. Since the examples have common words, and there are similar relations between sentences in both examples (e.g., *delivery* of S1 and *packing* of S2), the model can consider that the words with different polarity are noisy terms or terms without useful information. Thus, the model will not learn patterns based on the words' polarity. Although this is a simple example, we expect that the model will not be able to recognize words with the same/different polarities in all cases, since negative/positive words can be found both in positive/negative examples. As a result, the model will learn whether there is a contradiction or agreement in an example, learning the patterns which probably ignore the words' polarity. However, it is not clear if these patterns can improve the QA model's performance. Preliminary results showed that this method negatively affects the QA model's performance. The results were worse than the results considering the baseline models without extra fine-tuning on this task.

Instead, we modify this task so as to consider two texts as input to a model with the goal of recognizing whether they are both positive or both negative. We assume that texts of the same polarity have common characteristics, which the learning model can capture at its last layers. This is a valid hypothesis, since analysis of empirical results has shown that NLP transformer-based models capture a substantial amount of linguistic knowledge in the hidden states and the attention maps (Clark *et al*. 2019b). Therefore, we expect the models to be able to understand the relation between the two sentences by learning which words have the same polarity. This knowledge can then be transferred to answering yes/no questions. Figure 2 illustrates the new task and its difference with the existing task.

Starting from a sentiment classification dataset, we create positive (negative) examples by sampling randomly with replacement pairs of sentences of positive (negative) sentiment. We create an equal number of positive and negative examples. The actual number is a parameter of the process.

Suppose that a positive example for this task is the following pair of sentences:

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

S1: This is a good camera.

S2: It's fine!

The classifier will learn which are those parts that are important for understanding the polarity of the sentences. For example, the classifier may focus on the words *good* and *fine*. Now consider the following pair of a question and a related sentence:

- Q: Is this a good camera?
- S: It's fine!

The yes/no QA model will be able to recognize that the polarities of the question and the sentence are the same. Such knowledge is useful for yes/no QA, as similar (opposite) polarity implies absence (presence) of a contradiction between the question and the sentence, which in turn offers evidence in support of a yes (no) question.

2.3 Simplification validation

Consider the following example from the BoolQ dataset:

- Q: Is the show Bloodline based on a true story?
- P: Bloodline was announced in October 2014 as part of a partnership between Netflix and Sony Pictures Television, representing Netflix's first major deal with a major film studio for a television series. The series was created and executive produced by Todd A. Kessler, Glenn Kessler, and Daniel Zelman, who previously created the FX series Damages. According to its official synopsis released by Netflix, Bloodline "centers on a close-knit family of four adult siblings whose secrets and scars are revealed when their black sheep brother returns home.

The answer to this question is *no*, since there is not a shred of evidence that *Bloodline* is a true story. The passage does not mention anything about the genre of the show. To be able to answer such questions, a yes/no QA model must learn to look in the passage for important parts of the question and understand whether they are missing or not. This is by no means an easy task. To help yes/no QA models learn such knowledge, we introduce a simpler artificial task, based on the lexical simplification task, which we call simplification validation.

The lexical simplification task is concerned with the production of a simplified version of an input sentence, which may differ in structure and content, but preserves the same meaning (Silpa and Irshad 2018). We adapt the task so that a transformer-based model gets a pair of sentences and predicts whether the second sentence is a simplification of the first sentence. Figure 3 illustrates the new task and its difference with the existing task.

All training examples of a lexical simplification task can serve as positive examples in the simplification classification task. For constructing negative examples, a straightforward approach is to take random pairs of the original sentences. However, this would lead to an easy task, since the models will be able to discover simple patterns to classify the instances. To avoid this case, we assume that the simplified sentence is the same as the original, except for one noun phrase. This way, a correct simplified sentence is syntactically and semantically similar to the incorrect one, while the models need to put more effort to separate the positive from the negative examples. Therefore, we randomly select a noun phrase of the original sentence to be removed to formulate the simplified version. We discarded negative examples that happened to be identical with the positive ones. This can occur when the deleted noun phrase is not critical to the meaning of the sentence.

The learning model is expected to attend to the missing or extra parts of the simplified version and to learn whether these parts are necessary with respect to the context. A yes/no QA model can

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

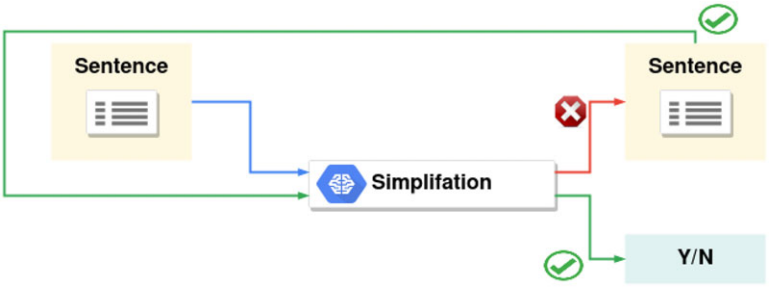

Figure 3. The adaption of the lexical simplification task. A classifier gets as input a pair of sentences and predicts if one sentence is a simplified version of the other, instead of getting an original sentence to generate the simplified version. The check mark (green color) indicates a new input/output to the classifier and the X mark (red color) indicates an output of the original task. The line without mark (blue color) indicates the part that we keep the same before and after the task adaptation.

benefit from such knowledge to focus on key parts of the text that are evidence for answering yes or no.

Consider the following example, where S1 and S2 are examples from the WSW dataset (Kauchak 2013) and *S2 is a negative example:

- S1: He subsequently modeled the establishment of King's and Eton College upon the successful formation of Wykeham's institutions.
- S2: He copied a lot of Wykeham's ideas when building King's and Eton College.
- *S2: He subsequently modeled the establishment of College upon the successful formation of Wykeham's institutions.

S2 is a simplified version of S1, since the knowledge about the inspiration for the establishment of King's and Eton College is preserved. However, *S2 cannot be considered as a simplified version, since an important part of the sentence is missing (King's and Eton). The model will be able to predict which of the two sentences is the correct simplified version.

Now, consider the following yes/no question given to a model fine-tuned on the simplification classification task:

- Q: Do mice have small rounded **eyes**?
- S: Mice are known to have a pointed snout, small rounded **ears**, a body-length scaly tail, and a high breeding rate.

We expect that the answer of the model will be *no*, due to the fact that the word *eyes* is missing from the passage and therefore there is no sufficient evidence in the given sentence that can answer the question.

3. Experimental setup

In this section, we first present the datasets that we use for the artificial tasks, as well as the main yes/no QA task. Then, we discuss the models and the fine-tuning process.

3.1 Datasets

We conduct experiments using the BoolQ (Clark *et al.* 2019a) yes/no QA dataset. BoolQ comprises a collection of yes/no questions gathered from anonymized, aggregated queries to the

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Google search engine, selecting only questions that can be answered by a Wikipedia page. Human annotators select the most relevant passage from the corresponding page and specify whether the answer is yes or no. Each instance of the dataset is thus a triple consisting of a question, a passage, and a yes/no answer. The dataset has been split into train, development, and test sets with 9427, 3270, and 3245 instances, respectively. We used the train set for training and the development set for testing, as the currently unavailable test set will be unlabeled and will serve the purpose of a leaderboard.[b]

For the answer validation transfer task, we experimented with SQuAD (Rajpurkar *et al*. 2016; Rajpurkar *et al*. 2018) and BioASQ (Tsatsaronis *et al*. 2015). SQuAD has two versions. SQuAD 1.0 contains more than 100K questions posed by crowdworkers on a set of Wikipedia articles, where the answer is a segment of text from the corresponding reading passage. SQuAD 2.0 enriches the collection of SQuAD 1.0 with over 50,000 unanswerable questions written adversarially by crowdworkers to look similar to answerable ones. QA systems addressing SQuAD 2.0 have to first decide whether there is an answer or not in the passage, and if yes, then to provide the answer. We selected 5000 (500) answerable questions as positive training (test) examples and created 5000 (500) negatives training (test) examples as explained in Section 2.1, for a total of 10k (1k) training (test) examples.

The BioASQ dataset contains triples of questions, passages, and answers from the biomedical domain. It contains questions of four types: list, factoid, summary, or yes/no. We focus on factoid questions that fit the needs of our task. The dataset consists of questions with relevant passages. Each question corresponds to more than one passage, while the answer is in at least one of them. Since the dataset is small, containing only 941 questions, we decided to create pairs of questions and passages mapping each passage to a question. So, it is possible to have pairs of questions and passages, where the question of one instance is the same with the question of another instance(s). We extracted 4k questions and we created 4k negative examples (8k in total). We further constructed 1k examples for testing as described above.

For the sentiment classification task, we experimented with the Sentiment Labelled Sentences Data Set (SLSD) and the Stanford Sentiment Treebank (SST). SLSD (Kotzias *et al*. 2015) contains sentences extracted from reviews of movies, products, and restaurants in three corresponding web resources, namely imdb.com, amazon.com, and yelp.com. For each web resource, there exist 500 positive and 500 negative sentences, for a total of 3K sentences. We constructed 4.5k (500) positive pairs and 4.5k (500) negative ones to form a training (test) set.

SST (Socher *et al*. 2013) contains 10,605 snippets from the original pool of Rotten Tomatoes HTML files, split into phrases and labeled using five classes (very negative, negative, neutral, positive, and very positive). We joined the first two and last two classes, while neutral instances were omitted. We also removed small phrases with less than 40 characters. We constructed the same number of instances for training and testing as in SLSD.

Finally, for the simplification task, we used the Wikipedia-Simple Wikipedia (WSW) and Sscorpus (Kajiwara and Komachi 2016) datasets. WSW contains 67,689 aligned sentences from 60,000 aligned articles in English by pairing Simple English Wikipedia,[c] a Wikipedia in which the authors use the *Simple English* words and grammar, with English Wikipedia. Four thousand five hundred positive instances and 4.5k negative instances are used for training. Five hundred positive instances and 500 negative instances are used for testing. The Sscorpus dataset contains 492,993 aligned sentences using the Simple English Wikipedia and English Wikipedia as in the WSW dataset. For each pair in the dataset, a score indicates how similar the two sentences are. Four thousand five hundred positive instances and 4.5k negative instances are used for training.

[b] https://github.com/google-research-datasets/boolean-questions.
[c] https://simple.wikipedia.org/wiki/Main_Page.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Table 1. The datasets for each new constructed task

Tasks	Datasets	Train	Test
Answer	SQuAD	10k	1k
validation	BioASQ	8k	1k
Sentiment	SLSD	9k	1k
classification	SST	9k	1k
Simplification	WSW	9k	1k
validation	Sscorpus	9k	1k
Yes/no QA	BoolQ	9.4k	3.2k

Five hundred positive instances and 500 negative instances are used for testing. Table 1 presents all datasets per task, along with the size of their training and test sets.

3.2 Learning phase

We experimented with three models from the BERT (Devlin *et al*. 2019) family using the transformers library provided by Hugging Face (Wolf *et al*. 2019). Each one has 12 repeating layers, 768 hidden size, and 12 attention heads. In detail:

- The BERT base model with 110M parameters is pretrained on lower-cased text from the BooksCorpus (800M words) (Zhu *et al*. 2015) and English Wikipedia (2500M words) using the masked language modeling and next sentence prediction objectives.
- The RoBERTa (Liu *et al*. 2019) base model with 125M parameters is pretrained on large English corpora (BooksCorpus, CC-NEWS, OPENWEBTEXT and STORIES) in a self-supervised fashion with the masked language modeling objective.
- The ALBERT (Lan *et al*. 2019) base model with 11M parameters is pretrained on the same datasets as the BERT model with the masked language modeling and sentence ordering prediction objectives.

For a fair comparison between the models, we used the same hyperparameters during training. The learning rate was set to 10^{-5} to avoid the catastrophic forgetting problem (McCloskey and Cohen 1989) and the batch size was set to 24. The maximum sequence length was set to 256, applying padding to the sequence length. We also added the special symbols [CLS] and [SEP]. During training, we followed the setup of Clark *et al*. (2019a). In specific, we trained the models on the BoolQ dataset training set for five epochs and then estimated their accuracy on the dev set.

When we fine-tune on a transfer task, we train the models for five epochs using the same hyperparameters. For the answer validation task, we also include the special tokens [SA] and [EA], which were introduced in Section 2, in the tokenization phase with randomly initialized vectors. The models are then fine-tuned on the yes/no QA task. For each neural network architecture, each task and each dataset, we estimate the mean accuracy of five models trained with five different random seeds.

Figure 4 presents the learning process. In detail, we firstly fine-tune a base model on a transfer task and then we fine-tune more on yes/no QA task. However, we also present results fine-tuning the base model directly on yes/no QA. The models trained with this process are then used for classifying the questions in the BoolQ development dataset.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Figure 4. The transitions for training the question answering classifier. A transition indicates the fine-tuning process of a model on a task.

4. Results

Table 2 presents the experimental results. We first notice that the RoBERTa model outperforms significantly the other two models, both with and without the use of the transfer tasks. ALBERT is slightly better than BERT, both with and without the use of transfer tasks, with the exception of the SST dataset, where BERT answers one more question correctly than ALBERT.

The baseline models without additional transfer learning have worse performance than the models considering firstly the artificial tasks. For example, BERT fine-tuned on the answer validation task using the SQuAD dataset and the AV1 approach for creating negative examples overcomes the baseline BERT model (71.41% over 69.51%). The same happens also for ALBERT with 73.36% accuracy over 71.90% and RoBERTa with 79.91% accuracy over 79.23%. The best overall results are obtained by RoBERTA using the sentiment classification transfer task with the SST dataset.

Among the transfer tasks, it is not clear which one is more appropriate than the others. The sentiment classification task gives the best results for BERT and RoBERTa, while the answer validation task gives the best results for ALBERT. For all three models, however, the simplification validation task typically leads to worse results compared to the other two tasks.

Which dataset is more appropriate for a specific transfer task is not clear too. For example, within the sentiment classification task, the SLSD dataset leads to the best results for BERT and ALBERT, while the SST for RoBERTA. Similarly, within the answer validation task, SQuAD (AV2) gives the best result for ALBERT and RoBERTA, while SQuAD (AV1) for BERT. In fact, SQuAD (AV2) is the worst answer validation dataset for BERT.

We can also notice that the transfer tasks are easier than the yes/no QA task (see *Source Acc.* column in Table 2), while some of them are easier than others. For example, the simplification validation transfer task using the WSW dataset is harder than the sentiment classification transfer task using any of the two datasets.

Besides the accuracy of the learning models, we are interested to know the agreement of the predictions between two models trained on two corresponding tasks. To measure their agreement ratio, we used the RoBERTa model, fine-tuned on the SQuAD (AV2), SST, and Sscorpus datasets where the highest accuracy is observed. Table 3 shows three confusion matrices, each of which presents the number of questions answered correctly by two models, the number of questions answered only by one of the two models, and the number of questions that none of the models was able to answer. As we can see from the sum of the numbers in the main diagonals of these matrices, the SC and SV models answered most questions in the same way, while the AV and SV models had the lowest agreement in their predictions.

To further improve the results, we combined the three transfer tasks. In Table 4, we show the results from adapting the RoBERTa model to all transfer tasks (all-in-one scheme). We also present the results from separately fine-tuning the RoBERTa model to each transfer task and then combining the three models using a voting scheme to answer the questions. In the all-in-one scheme, the results are better than the results of the corresponding baseline model but worse than the models fine-tuned on the SST dataset of the sentiment classification task and on

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Table 2. Results on BoolQ development dataset with and without transfer learning. The results are the mean accuracy of 5 runs with different random seeds. Underline scores indicate better performance over the same baseline model architecture. Bold indicates better performance over all the baseline models. Last column corresponds to the questions that are correctly answered by the corresponding model, transfer task and transfer data

Model	Transfer task	Transfer data	Source Acc.	BoolQ Acc.	#Questions
BERT	N/A	–	–	0.6951	2273
	Answer	SQuAD(AV1)	0.9261	0.7141	2335
	validation	SQuAD(AV2)	0.8581	0.7103	2323
		BioASQ	0.9470	0.7140	2335
	Sentiment	SLSD	0.9978	0.7206	2356
	classification	SST	0.9877	0.7204	2356
	Simplification	WSW	0.8838	0.7078	2315
	validation	Sscorpus	0.9707	0.7106	2324
ALBERT	N/A	–	–	0.7190	2351
	Answer	SQuAD(AV1)	0.9552	0.7336	2399
	validation	SQuAD(AV2)	0.8821	0.7358	2406
		BioASQ	0.9545	0.7299	2387
	Sentiment	SLSD	0.9952	0.7284	2382
	classification	SST	0.9761	0.7202	2355
	Simplification	WSW	0.9036	0.7224	2362
	validation	Sscorpus	0.9818	0.7212	2358
RoBERTA	N/A	–	–	0.7923	2591
	Answer	SQuAD(AV1)	0.9962	0.7991	2613
	validation	SQuAD(AV2)	0.9982	0.8021	2623
		BioASQ	0.9972	0.7979	2609
	Sentiment	SLSD	0.9966	0.7986	2611
	classification	SST	0.9910	**0.8049**	**2632**
	Simplification	WSW	0.8937	0.7946	2598
	validation	Sscorpus	0.9909	0.7959	2603

Table 3. A confusion matrix corresponds to a comparison between two models fine-tuned on a task (AV, SC, SV) regarding their predictions on BoolQ dev dataset. *A* corresponds to the number of correct answers and ¬*A* to the number of wrong answers for a model

		SC				SV				SV	
		A	¬*A*			*A*	¬*A*			*A*	¬*A*
AV	*A*	2408	216	AV	*A*	2388	236	SC	*A*	2405	218
	¬*A*	215	431		¬*A*	209	437		¬*A*	192	455

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Table 4. Results on the BoolQ development set combining the three transfer tasks either by adapting the RoBERTa model to all transfer tasks (all-in-one scheme) or by fitting the RoBERTa model separately to a transfer task and then combining the models using the voting scheme

Model	BoolQ Acc.	#Questions
All-in-one scheme	0.8001	2617
Voting scheme	0.8113	2653

Table 5. (Un)Expected outcome using RoBERTa model fine-tuned on AV task. Examples from BoolQ development dataset

Expected Outcome
Q: Does the antagonist always have to be a person?
P: An **antagonist is a character, group of characters, institution or concept** that . . .
Q: Is social studies and social science the same?
P: In the United States education system, **social studies is the integrated study of multiple fields of social science** and . . .
Q: is it illegal to buy organs in the us?
P: All other nations have some form of legislation meant to **prevent the illegal trading of organs**, whether by . . .
Unexpected Outcome
Q: Is a kippah the same as a yamaka?
P: A kippah (. . .) or yarmulke (. . .) is a brimless cap, usually made of cloth, worn by Jews to fulfill the customary requirement held by Orthodox halachic authorities that the head be covered. It is usually worn by men in Orthodox communities at all times. Most synagogues and Jewish funeral services keep a ready supply of kippot.

the SQuAD(AV2) dataset of the answer validation task. On the other hand, the voting scheme improves the QA model achieving the best accuracy (81.13%).

5. Qualitative analysis

In this section, we present case studies from the BoolQ development dataset, where the models fine-tuned on a corresponding task can correctly answer questions that another model cannot. Next, we visualize the patterns that the transformer-based models discover with a couple of examples.

5.1 An analysis of models' predictions in BoolQ examples

We expect that the model fine-tuned on the answer validation task can answer questions, where the reference text has a part that indicates the truth of the question. In Table 5, we present a sample of questions that are correctly answered only by this model. As we can see, the evidence about the truth of the question is in a specific part in the reference text for the first three questions. The model correctly answered the last question, but we are not sure of the reason. Perhaps the tokenization method used by the model brings closer the words "yamaka" and "yarmulke."

For the model fine-tuned on the sentiment classification task, we expect that it can answer questions where the referenced passage contains terms with different/same polarity with the

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Table 6. (Un)Expected outcome using RoBERTa model fine-tuned on SC task. Examples from BoolQ development dataset

Expected Outcome
Q: Is it goaltending if the ball hits the backboard **below the rim** ?
P: . . . goaltending is the violation of interfering with the ball downward flight, (b) entirely **above the rim**
Q: Is there a seat belt **law** in new hampshire?
P: . . . New Hampshire is the only state that has **no enforceable laws** for the wearing of seat belts in a vehicle
Q: Do you **have to tag up** on an infield fly rule?
P: . . . the runners **must tag up**. On the other hand, if "infield fly" . . .
Unexpected Outcome
Q: Did tom hardy won an oscar for the revenant?
P: The following is a list of awards and nominations received by English actor Tom Hardy. He was nominated for the Academy Award for Best Supporting Actor for the 2015 film The Revenant. He also won the 2011 BAFTA Rising Star Award, and has twice won the British Independent Film Award for Best Actor, for Bronson (2009) and Legend (2015).

terms in the question. Table 6 presents some examples. In the first three examples, the agreement/contradiction is clear (below—above, law—no law, have to tag up—must tag up). Although the model correctly answers the last question, the reason is not clear. One possible explanation is that "oscar" is an "Award" and between "Award" and "Revenant" there isn't any contradictory term, consequently the answer is yes.

Finally, we present some examples for the outcome of a model trained on the simplification validation task (Table 7). We expect that the model can answer questions, even though some terms of the question may not be included in the referenced passage. In the first example, despite the phrase "part of" does not exist in the referenced passage, the model answered the question correctly. The same happens in the second example, where the word "sequel" is not involved in the reference text. In the last example, although the phrase "same as" is not involved in the referenced passage, we expect that the answer would be yes, since the two noun phrases in the question are close enough in the referenced passage, and the model has no evidence that they are not the same given the way that it has been trained.

5.2 Examples for visualizing the differences between the baseline and other models

The models fine-tuned firstly on a transfer task exhibit different behavior than the models trained only on the QA task. We experiment with the BERT base model trained on uncased English texts. We visualize the attentions scores using the open-source tool BertViz (Vig 2019) to show the differences between the models firstly trained on a transfer task and the models trained only on yes/no QA task. For the rest of this section, the models firstly fine-tuned on a transfer task and then on a QA task are denoted as *TM*, while the models trained only on QA task as *M*.

5.2.1 Answer validation

One of our main concerns was if BERT is affected by the appearance of special tokens in the given passages. We examined the following case for this purpose.

- Q: What currency is used in China ?
- P: The [SA] renminbi [EA] is the official currency of China and one of the world's reserve currencies.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Table 7. (Un)Expected outcome using RoBERTa model fine-tuned on SV task. Examples from BoolQ development dataset

Expected Outcome
Q: Was romania part of the austro hungarian empire?
P: Austria-Hungary was one of the Central Powers in World War I. It was already effectively dissolved by the time the military authorities signed the armistice of Villa Giusti on 3 November 1918. The Kingdom of Hungary and the First Austrian Republic were treated as its successors de jure, whereas the independence of the West Slavs and South Slavs of the Empire as the First Czechoslovak Republic, the Second Polish Republic and the Kingdom of Yugoslavia, respectively, and most of the territorial demands of the Kingdom of Romania were also recognized by the victorious powers in 1920.
Q: Is batman forever a sequel to batman returns?
P: Batman Forever's tone is significantly different from the previous installments, becoming more family-friendly since Warner Bros. believed that the previous Batman film, Batman Returns (1992), failed to outgross its predecessor due to parent complaints about the film's violence and dark overtones. Schumacher eschewed the dark, dystopian atmosphere of Burton's films by drawing inspiration from the Batman comic book of the Dick Sprang era, as well as the 1960s television series. Keaton chose not to reprise the role due to failing to negotiate with studio executives Terry Semel and Bob Daly about the overall approach to the script. William Baldwin and Ethan Hawke were initially considered for Keaton's replacement before Kilmer joined the cast. Rene Russo was originally set to play Chase Meridian, based on her chemistry with Keaton in One Good Cop, but was replaced with the much younger Nicole Kidman after being deemed "too old" for Kilmer.
Unexpected Outcome
Q: Is los cabos the same as cabo san lucas?
P: Cabo San Lucas (Spanish pronunciation: (. . .), Cape Saint Luke), commonly called Cabo in English, is a resort city at the southern tip of the Baja California Peninsula, in the Mexican state of Baja California Sur. As of 2015, the population of the city was 81,111 inhabitants. Cabo San Lucas together with San Jos É del Cabo is known as Los Cabos. Together they form a metropolitan area of 305,983 inhabitants.

- *P: The renminbi is the [SA] official [EA] currency of China and one of the world's reserve currencies.

The positive instance is (Q, P) and the negative one is (Q, *P). The model can correctly classify both instances. To find pieces of evidence about the classification process, we illustrate the attention from the question to the passage for both instances (Figure 5). As we see most of the heads (different colors in the figure) of [CLS] attend to [EA] (subfigures (a),(d)), while in lower layers, we observe that [CLS] also attends to [SA] (subfigures (b),(c)). An interesting observation is that when the model classifies the example as positive, the attention scores are higher from [CLS] to [EA] (vivid colors in subfigure (a)). This shows the effect of special tokens in the learning process.

We also have to notice that the model can not always recognize the incorrect passage. For example, the following negative examples have been classified as positive:

- The renminbi is the official currency of China and one of the [SA] world's reserve currencies [EA].
- The Louvre is the largest [SA] art museum [EA] and a historic monument in Paris, France.

However, the same patterns have been found, that is, the attention scores are higher from [CLS] to [EA].

After fine-tuning on the QA task, we expect to find similar patterns. In detail, we expect that the [CLS] will attend to the context of a possible answer. This attention will be stronger if the model recognizes that the instance is yes.

Let's consider the below example:

- Q1: Is Paris the capital city of France?
- Q2: Is Berlin the capital city of France?
- P: Paris is the capital and most populous city of France.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

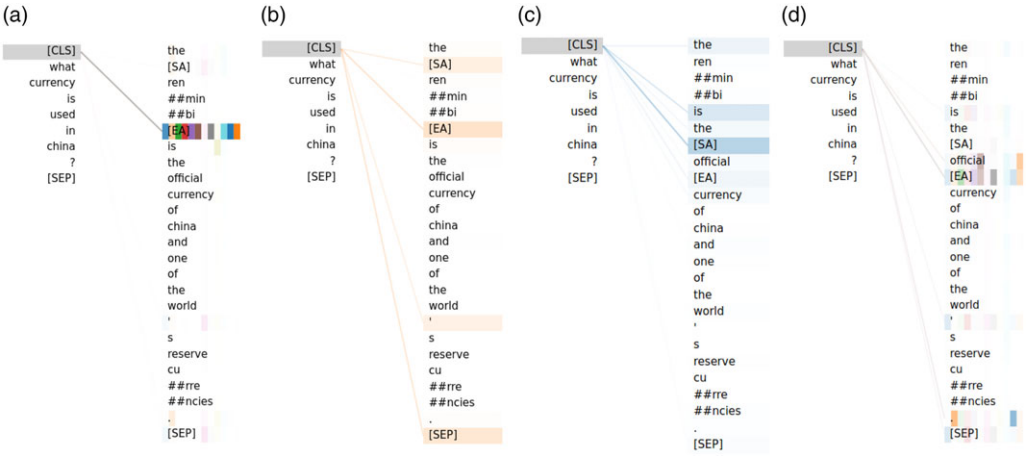

Figure 5. The attentions between the question and the correct passage (left side (a),(b)) and the question and wrong passage (right side (c),(d)). The weight of the arrows indicates the magnitude of the attention scores, while the different colors correspond to the 12 attention heads of the BERT base model.

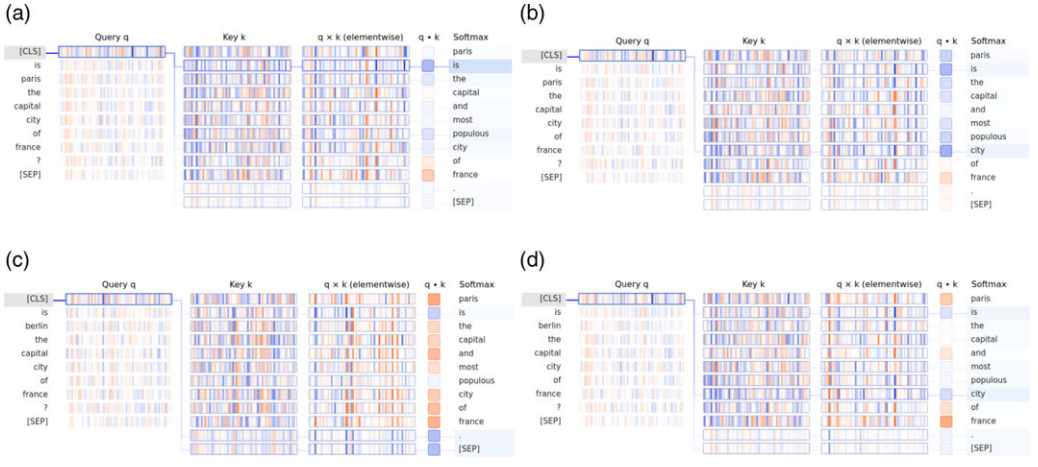

Figure 6. The attention of [CLS] token for the pair (Q1,P) of the models TM (a) and M (b), and the pair (Q2,P) (c),(d). Positive and negative values are colored blue and orange, respectively, with color saturation based on magnitude of the value.

The pairs are (Q1, P) and (Q2, P). TM can correctly answer both questions, but M considers both instances as no. In Figure 6 (subfigures (a),(b)), we illustrate the attentions of the first pair. The [CLS] strongly attends to a smaller window of words. On the other hand, the [CLS] of M strongly attends to much other passage's words. In detail, the [CLS] of TM strongly attends the word *is*, and the values of the rest attentions scores are low, while the [CLS] of M strongly attends the words *is* and *city* and the values of some of the other attentions scores (e.g., *paris*, *populous*) are high. The distinction is clear in the second pair (subfigures (c),(d)). The values of TM attention scores are negative except for the word *is*.

5.2.2 Sentiment classification

The concern here was if the sentiment classifier can learn useful patterns during training. A case study is presented in Figure 7. When the two sentences are positives, we observe that the word

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Figure 7. The attentions between the sentences of a positive instance (a),(b) and a negative one (c),(d).

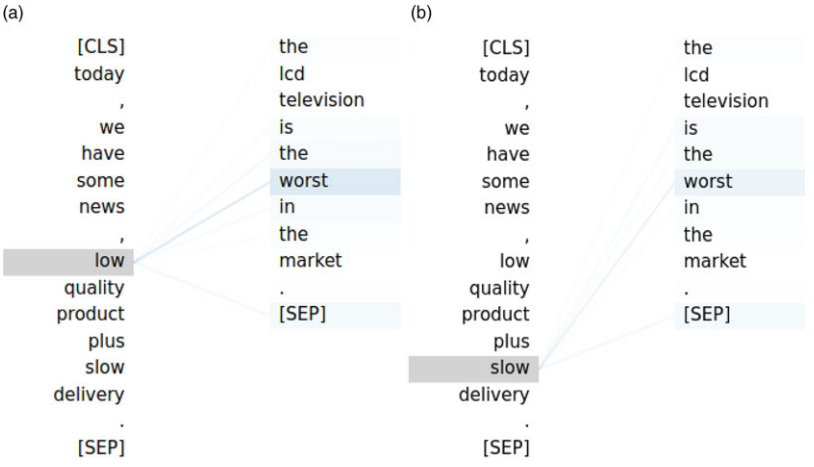

Figure 8. The attentions between the sentences of a negative instance repositioning the word *worst*.

high attends to the word *best* which are both positive words (subfigure (a)). The same happens with the word *fast* and *best* (subfigure (b)). In the second example, we replaced the positive words with negatives. The negative words *low* and *slow* attend the word *worst* (subfigures (c),(d)). The model correctly predicts both two cases.

If we slightly modify the negative example, the model still predicts correctly the class of the instance and also discovers the same patterns (Figure 8).

To present the effect of the sentiment classification transfer task, we examine the following example:

- Q: Is this a good camera ?
- P: It's fine!
- *P: It's not fine!

The pairs are (Q, P) and (Q, *P). TM can correctly classify the two instances as yes and no, respectively. However, M considers both instances as no. In Figure 9, we illustrate the attentions of the [CLS]. As we can see, the [CLS] of TM strongly attends the negative word *not* (subfigure (d)), while the [CLS] of M is not affected by the existence of this word (subfigure (a)). Furthermore, M attends all the words in the second sentence of the positive instance (subfigure (b)). In contrast, TM attends more to the word *fine* and the exclamation mark (subfigure (c)).

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

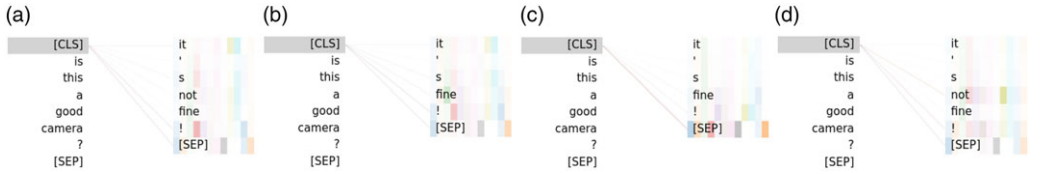

Figure 9. The attention of the [CLS] token for the M (a),(b) and TM (c),(d) for the pairs (Q,P) and (Q,*P).

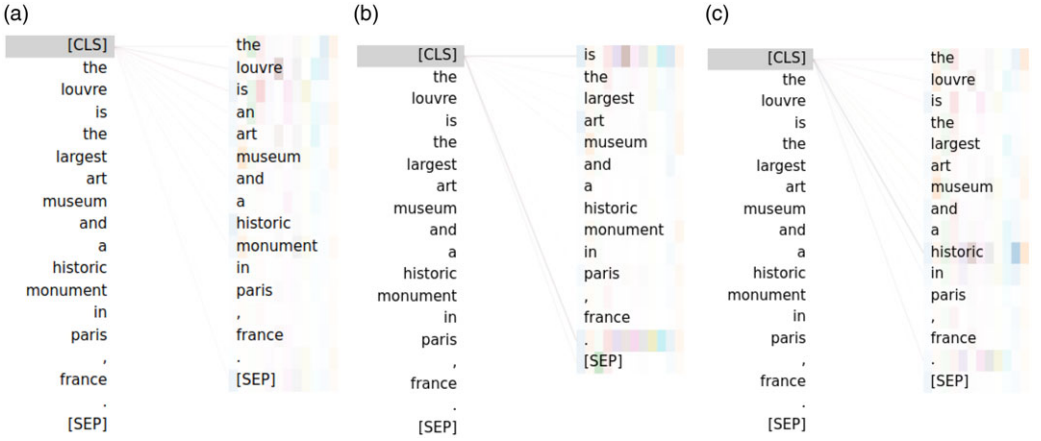

Figure 10. The attentions between the sentences of a correct instance (a) and two incorrect ones (b)(c). The model trained on simplification validation task.

5.2.3 Simplification validation

Simplification validation improves the results over the baseline. We examine the following example:

- S1: The Louvre is the largest art museum and a historic monument in Paris, France.
- S2: The Louvre is an art museum and a historic monument in Paris, France.
- *S2a: is the largest art museum and a historic monument in Paris, France.
- *S2b: The Louvre is the largest art museum and a historic in Paris, France.

The pairs are (S1, S2), (S1, *S2a), and (S1, *S2b). We assumed that the S2 sentence is the correct simplified sentence due to the process followed to create the task. Specifically, we considered that a negative example is an original sentence skipped a noun phrase. The missing part of the S2 is not a noun phrase, since the head of a noun phrase is not missing but an adjective. Thus, it can be considered as a correct simplified sentence.

In Figure 10, we present some useful patterns. The model classifies correctly the three instances. An interesting finding is that the [CLS] attends the context of the missing part of the S2 for all three instances. The intuition here is that the model learns where to focus in order to decide if the given pair is a positive instance or not. To further validate that the behavior of the model is affected by the transfer task, we fine-tune on QA task instead of simplification validation one. In Figure 11, the [CLS] also attends to other parts of the sentence. The most clear case is presented in subfigure (c). The [CLS] of the model trained on the simplification task strongly attends the word *historic* which is close enough to the missing word (*museum*). On the other hand, the [CLS] of the model trained on QA focuses on other words. We also notice that the model trained on QA incorrectly classifies the *S2a and *S2b as correct simplified sentences.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

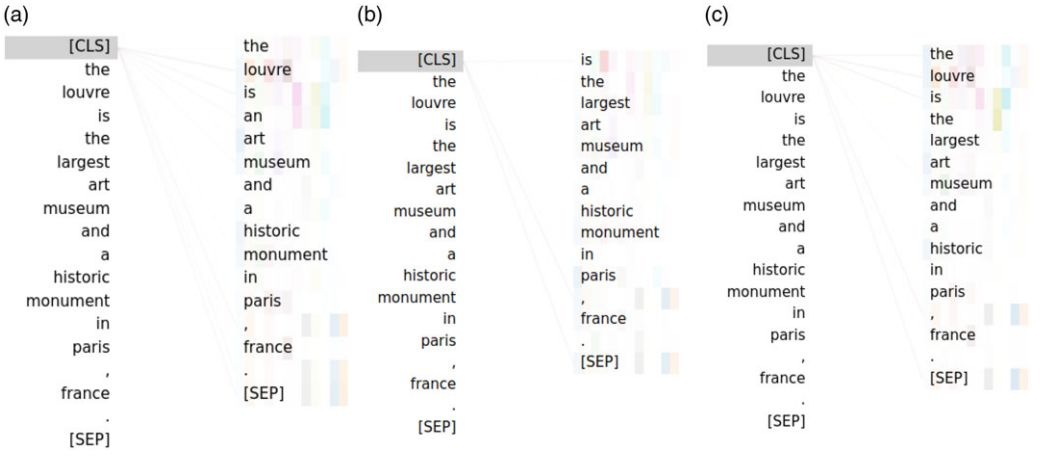

Figure 11. The attentions between the sentences of a correct instance (a) and two incorrect ones (b)(c). The model trained on QA task instead of simplification validation task.

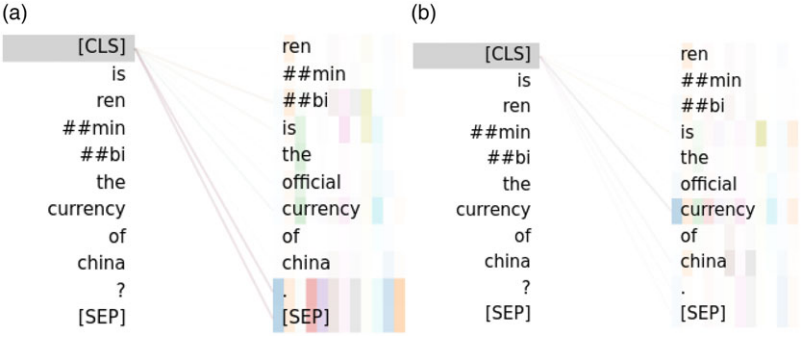

Figure 12. The attentions of TM (left side) and the attentions of M (right side).

To show that the model fine-tuned firstly on simplification validation and then on QA, we examine the below example:

- Q: Is renminbi the currency of China?
- P: Renminbi is the official currency of China.

M cannot classify the question as yes. Again, TM has a different behavior than M (Figure 12). The [CLS] of TM also attends to the extra word *official*, while the [CLS] of the M model gives more attention to the word *currency*.

6. Related work

The most common way to deal with the yes/no QA problem is to consider it as a binary classification task. Traditional machine learning algorithms with handcrafted features have been used to solve the problem. Yang *et al*. (2016) showed that the selected features can help for improving the accuracy of the model overcoming the strong baseline of selecting the majority class as response to the user. Kim *et al*. (2013) used deep linguistic features to train an unsupervised classifier to answer legal bar exams outperforming SVMs. Pasca and Harabagiu (2001) used several features

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

to build a high-performance QA system for TREC-9 evaluations. He and Dai (2011) combined opinion features together with two weighting scores to classify the answers as yes, no, or neutral and conduct experiments on a real-word dataset.

Many approaches are focused on complex neural networks. Dzendzik, Vogel, and Foster (2019) used various representations for encoding the questions and customer reviews, including bag-of-words, word2vec, ELMo, and BERT showing that BERT base model overcome all the other models even the large one. Shen *et al.* (2018) proposed a hierarchical matching network using self-attention on pairs of questions and sentences to predict if the final answer is yes or no in QA-style sentiment classification. Sharma *et al.* (2018) treated the yes/no QA problem as natural language inference (NLI) task and used hierarchical convolutional neural network based on inference models to answer the question in biomedical domain achieving 65% accuracy.

Other approaches do not treat the yes/no questions independently of other question types (Baradaran, Ghiasi, and Amirkhani (2020)). For instance, in a baseline for the natural questions (Alberti, Lee, and Collins 2019), five target classes are defined including *short*, *long*, *yes*, *no*, and *no-answer*, while the learning model is responsible for recognizing the type of the question and also for giving an answer for different question types.

Most of the current methods focus their attention on transfer learning techniques on tasks similar to the yes/no QA and transformer-based models to improve the results. Clark *et al.* (2019a) introduce the BoolQ dataset and experimented with several similar QA tasks to improve the accuracy of the yes/no model. The results showed that transferring from MultiNLI, as well as the unsupervised pretraining in BERT, had the highest impact. In a similar manner, Ignatov (2021) introduced the DaNetQA dataset and also experimented with similar yes/no QA tasks, overcoming the base models. Yin *et al.* (2020) fine-tuned the models by multitask learning, achieving comparative results. A unified learning framework has been proposed by Raffel *et al.* (2020) for text-to-text learning, achieving state-of-the-art results in several tasks including yes/no QA (91.2% acc. in BoolQ). Aghajanyan *et al.* (2021) experimented with several preexisted tasks to improve the performance of the QA models.

Finally, there are also plenty of approaches that do not follow the machine learning paradigm. Question inversion and factoid QA have been used to answer yes/no questions (Kanayama *et al.* 2012). Particularly, they convert the task to a set of factoid-style questions and used an existing QA system as a subsystem. Then, they aggregated the answers and confidence values from a factoid-style QA system determining the correctness of the entire proposition or the substitutions that make the proposition false. One important issue of this work is that the question inversion was a manual task, which means that the approach was not tested for its scalability. On the other hand, the performance of the system heavily depends on the performance of factoid QA. Thus, it is not clear which are the factors that affect the performance on yes/no questions. Sentiment information was used by Sarrouti and El Alaoui (2017) to answer yes/no questions. They used SentiWordnet to obtain the sentiment score of each word and then aggregated the scores to decide if the answer is yes or no. An end-to-end QA system was implemented for answering Arabic yes/no questions in Bdour and Gharaibeh (2013). The main idea is the use of a logical representation of the question and the selected answer (a span text inside a sentence of the retrieved documents related to the question) for deciding if the response is yes or no. Kano (2016) suggested a penalized scoring method assigning scores to parts of documents that include terms, which indicate that the answer is no. Kano, Hoshino, and Taniguchi (2017) built a system using linguistic analysis to find correspondences of predicates and arguments from the given problem sentences and knowledge source sentences. Although these approaches are interpretable, the experience has shown that more complex systems are necessary for solving the QA problem, while the machine learning approaches are necessary for generalization purposes.

Our approach is based on transfer learning and transformer-based models, since this paradigm significantly outperforms all the other methods. The main difference in our work, in contrast to the previous ones, is the fact that we experimented with new tasks that have not been previously

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

studied to the best of our knowledge. This gives an alternative perspective on how to deal with the QA problem, that is more creative, and at the same time more flexible, as it can exploit multiple other existing tasks and corresponding datasets to improve yes/no QA models.

7. Discussion

RoBERTa base is a strong baseline on this task overcoming the BERT and ALBERT baseline models. RoBERTa outperforms significantly the models discussed in Clark *et al*. (2019a), which are fine-tuned on several transfer tasks. The results are even better when we further fine-tune the RoBERTa base model with an adapted task. The best model presented by the BoolQ creators achieves 82.20% accuracy using the large BERT model with millions of parameters and fine-tuned on MultiNLI dataset with 392k instances. The RoBERTa model fine-tuned on sentiment classification transfer task using 9k instances of SST dataset has comparative performance with that model (80.49% over 82.20%). Furthermore, the experimentation showed that there is a run where the accuracy of the model is higher than 81%, while the combination of all transfer tasks using a voting scheme achieves 81.13% accuracy. This is interesting since the model is smaller than the large BERT and also the training data is significantly less. We expect that using larger models with much more parameters, the results will be better. We also expect the same if we experimented with much more transfer data.

The results showed that the tasks affect much more the BERT and ALBERT models rather than RoBERTa. For example, the BERT model fine-tuned on the SST dataset answers 83 more questions than the baseline, and the ALBERT model fine-tuned on SQUAD answers 55. Contradictory, RoBERTa with SST answers 41 more questions. We believe that the RoBERTa baseline model, as an optimized version of BERT, was able to recognize patterns for cases that were hard for the other two baseline models with and without fine-tuning on artificial tasks. However, the RoBERTa model still needs improvements. The performance of the model is better considering the tasks and the combination of them.

An interesting finding is that despite the differences between the transfer tasks, the differences between the datasets, the decisions that we made to adapt the original ones, and their simplicity, both three tasks can improve all the baseline models. Thus, a reasonable question is that if the baseline models are improved when further fine-tuned on a transfer task because they manipulate much more data and not due to the transfer tasks. The qualitative analysis above showed that the transfer tasks share useful knowledge to the QA problem. The most obvious example was the one presented on the sentiment classification transfer task (Figure 9). The model attends to the negative word *not* to decide if the answer is yes or no. However, more experimentation is necessary to find more appropriate patterns to show clearly the usefulness of the transfer tasks.

Another finding is that the models fine-tuned using the answer validation transfer task with the two methods described in the experimental setup section have different results based on the source accuracy. The second method seems to make the problem harder. That was expected because we selected parts of texts that are similar to the gold span text as negative examples. However, the RoBERTa and ALBERT models were positively affected by this change. The effect of using alternative methods on the same tasks and the same datasets can be positive, and the models can be further improved.

The previous works showed that sentiment is a factor that influences the performance of a learning model in yes/no QA. There are studies that focus their attention on this direction. In our previous work (Dimitriadis and Tsoumakas 2019b), we showed that polarity can improve the performance of a model trained on ELMo embeddings. Indeed, the sentiment can affect the performance of the models. The use of sentiment classification transfer task improved the performance of yes/no QA on BoolQ dataset.

Finally, our study has similar outcomes with previous studies about the effectiveness of transfer learning during training the learning models. However, we show this effect from a different

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

perspective assuming transfer tasks that are created with simple modifications and are easier to be solved than the original ones (e.g., extractive QA on SQuAD dataset vs. answer validation with extra special tokens).

8. Conclusions and future work

Transfer learning and transformer-based models have been empirically proved that improve the results on QA. Toward this direction, we proposed a different perspective about transferring knowledge from artificial yes/no tasks to QA. The main point of our work is that we assumed that every task that can be reconsidered as a yes/no task can improve the performance. The results showed on three tasks that this assumption holds even with the selection of small datasets. Although the tasks are simple and easier than the yes/no QA, the knowledge that is transferred can be necessary. This perspective can be useful for other tasks where there are limited data, or it is difficult to find other similar tasks that have an impact in the real world.

Further work is necessary for this direction. Firstly, it would be useful to be defined new datasets for the artificial transfer tasks. Furthermore, more experimentation is necessary to establish this technique for transferring knowledge between tasks. Finally, systematic experimentation between large transformer-based models and several similar tasks or artificial ones is necessary for a fair comparison.

Acknowledgment. This work was supported by computational time granted from the National Infrastructures for Research and Technology S.A. (GRNET S.A.) in the National HPC facility—ARIS—under project ID pa181002-NEBULA.

References

Aghajanyan A., Gupta A., Shrivastava A., Chen X., Zettlemoyer L. and Gupta S. (2021). Muppet: Massive multi-task representations with pre-finetuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5799–5811.

Alberti C., Lee K. and Collins M. (2019). A BERT baseline for the natural questions. arXiv.

Athenikos S.J. and Han H. (2010). Biomedical question answering: A survey. *Computer Methods and Programs in Biomedicine* **99**(1), 1–24.

Baradaran R., Ghiasi R. and Amirkhani H. (2020). A survey on machine reading comprehension systems. *Natural Language Engineering*, 1–50.

Bdour W.N. and Gharaibeh N.K. (2013). Development of yes/no Arabic question answering system. *International Journal of Artificial Intelligence & Applications* **4**(1), 51–63.

Bouziane A., Bouchiha D., Doumi N. and Malki M. (2015). Question answering systems: Survey and trends. *Procedia Computer Science* **73**(Awict), 366–375.

Clark C., Lee K., Chang M.W., Kwiatkowski T., Collins M. and Toutanova K. (2019a). Boolq: Exploring the surprising difficulty of natural yes/no questions. In *NAACL HLT 2019-2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, vol. 1, pp. 2924–2936.

Clark K., Khandelwal U., Levy O. and Manning C.D. (2019b). What does bert look at? an analysis of bert's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 276–286.

Devlin J., Chang M.W., Lee K. and Toutanova K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL HLT 2019-2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, vol. 1(Mlm), pp. 4171–4186.

Dimitriadis D. and Tsoumakas G. (2019a). Word embeddings and external resources for answer processing in biomedical factoid question answering. *Journal of Biomedical Informatics* 92, 103118.

Dimitriadis D. and Tsoumakas G. (2019b). Yes/no question answering in bioasq 2019. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, pp. 661–669.

Dzendzik D., Vogel C. and Foster J. (2019). Is it dish washer safe? automatically answering "yes/no" questions using customer reviews. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pp. 1–6.

Gao Z., Feng A., Song X. and Wu X. (2019). Target-dependent sentiment classification with bert. *IEEE Access* **7**, 154290–154299.

https://doi.org/10.1017/S1351324922000286 Published online by Cambridge University Press

Gupta P. and Gupta V. (2012). A survey of text question answering techniques. *International Journal of Computer Applications* **53**(4), 1–8.

He J. and Dai D. (2011). Summarization of yes/no questions using a feature function model. *Journal of Machine Learning Research* **20**, 351–366.

Ignatov D.I. (2021). Danetqa: A yes/no question answering dataset for the russian language. In *Analysis of Images, Social Networks and Texts: 9th International Conference*, *AIST* 2020, *Skolkovo, Moscow, Russia, October 15–16, 2020, Revised Selected Papers*, vol. 12602. Springer Nature, p. 57.

Jurafsky D. and Martin J.H. (2000). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 1st Edn. USA: Prentice Hall PTR.

Kajiwara T. and Komachi M. (2016). Building a monolingual parallel corpus for text simplification using sentence similarity based on alignment between word embeddings. In *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers*, pp. 1147–1158.

Kanayama H., Usuke Y., Iyao M. and Rager J.P. (2012). Answering yes/no questions via question inversion. In *Proceedings of COLING 2012*, pp. 1377–1392.

Kano Y. (2016). Answering yes-no questions by penalty scoring in history subjects of university entrance examinations. In *Proceedings of the Open Knowledge Base and Question Answering Workshop (OKBQA 2016)*, pp. 91–96.

Kano Y., Hoshino R. and Taniguchi R. (2017). Analyzable legal yes/no question answering system using linguistic structures. In *COLIEE@ ICAIL*, pp. 57–67.

Kauchak D. (2013). Improving text simplification language modeling using unsimplified text data. In *ACL 2013 - 51st Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, vol. 1, pp. 1537–1546.

Kim M.-Y., Xu Y., Goebel R. and Satoh K. (2013). Answering yes/no questions in legal bar exams. In *JSAI International Symposium on Artificial Intelligence*. Springer, pp. 199–213.

Kolomiyets O. and Moens M.-F. (2011). A survey on question answering technology from an information retrieval perspective. *Information Sciences* **181**(24), 5412–5434.

Kotzias D., Denil M., De Freitas N. and Smyth P. (2015). From group to individual labels using deep features. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015-August, pp. 597–606.

Lally A., Prager J.M., McCord M.C., Boguraev B.K., Patwardhan S., Fan J., Fodor P. and Chu-Carroll J. (2012). Question analysis: How Watson reads a clue. *IBM Journal of Research and Development* **56**(3–4), 1–14.

Lan Z., Chen M., Goodman S., Gimpel K., Sharma P. and Soricut R. (2019). Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L. and Stoyanov V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

Magnini B., Negri M., Prevete R. and Tanev H. (2002). Is it the right answer? exploiting web redundancy for answer validation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 425–432.

McCloskey M. and Cohen N.J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. *In Psychology of Learning and Motivation*, vol. **24**. Elsevier, pp. 109–165.

Munikar M., Shakya S. and Shrestha A. (2019). Fine-grained sentiment classification using BERT. arXiv, pp. 2–5.

Pakray P., Bhaskar P., Banerjee S., Pal B.C., Bandyopadhyay S. and Gelbukh A.F. (2011). A hybrid question answering system based on information retrieval and answer validation. In *CLEF (Notebook Papers/Labs/Workshop)*.

Pasca M.A. and Harabagiu S.M. (2001). High performance question/answering. In *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, pp. 366–374.

Peng S., You R., Xie Z., Wang B., Zhang Y. and Zhu S. (2015). The Fudan participation in the 2015 BioASQ Challenge: Large-scale biomedical semantic indexing and question answering. In *CEUR Workshop Proceedings*.

Pundge A.M., Khillare S. and Mahender C.N. (2016). Question answering system, approaches and techniques: A review. *International Journal of Computer Applications* **141**(3), 0975–8887.

Raffel C., Shazeer N., Roberts A., Lee K., Narang S., Matena M., Zhou Y., Li W. and Liu P.J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* **21**, 1–67.

Rajpurkar P., Jia R. and Liang P. (2018). Know what you don't know: Unanswerable questions for SQuAD. In *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, vol. 2, pp. 784–789.

Rajpurkar P., Zhang J., Lopyrev K. and Liang P. (2016). Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392.

Rondeau M.-A. and Hazen T.J. (2018). Systematic error analysis of the stanford question answering dataset. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pp. 12–20.

Sarrouti M. and El Alaoui S.O. (2017). A yes/no answer generator based on sentiment-word scores in biomedical question answering. *International Journal of Healthcare Information Systems and Informatics (IJHISI)* **12**(3), 62–74.

Sharma V., Kulkarni N., Pranavi S., Bayomi G., Nyberg E. and Mitamura T. (2018). Bioama: Towards an end to end biomedical question answering system. In *Proceedings of the BioNLP 2018 Workshop*, pp. 109–117.

Shen C., **Sun C.**, **Wang J.**, **Kang Y.**, **Li S.**, **Liu X.**, **Si L.**, **Zhang M. and Zhou G.** (2018). Sentiment classification towards question-answering with hierarchical matching network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3654–3663.

Silpa K.S. and Irshad M. (2018). A survey of lexical simplification. In *Emerging Trends in Engineering, Science and Technology for Society, Energy and Environment - Proceedings of the International Conference in Emerging Trends in Engineering, Science and Technology, ICETEST 2018*, vol. 60, pp. 785–791.

Socher R., **Perelygin A.**, **Wu J.**, **Chuang J.**, **Manning C.D.**, **Ng A.Y. and Potts C.** (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642.

Tsatsaronis G., **Balikas G.**, **Malakasiotis P.**, **Partalas I.**, **Zschunke M.**, **Alvers M.R.**, **Weissenborn D.**, **Krithara A.**, **Petridis S.**, **Polychronopoulos D.**, **Almirantis Y.**, **Pavlopoulos J.**, **Baskiotis N.**, **Gallinari P.**, **Artiéres T.**, **Ngomo A.-C. N.**, **Heino N.**, **Gaussier E.**, **Barrio-Alvers L.**, **Schroeder M.**, **Androutsopoulos I. and Paliouras G.** (2015). An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics* **16**, 138.

Vaswani A., **Shazeer N.**, **Parmar N.**, **Uszkoreit J.**, **Jones L.**, **Gomez A.N.**, **Kaiser Ł. and Polosukhin I.** (2017). Attention is all you need. In *The 31st Conference on Neural Information Processing Systems (NIPS 2017)*.

Vig J. (2019). A multiscale visualization of attention in the transformer model. In *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of System Demonstrations*, pp. 37–42.

Weissenborn D., **Wiese G. and Seiffe L.** (2017). Making neural QA as simple as possible but not simpler. In *CoNLL 2017 - 21st Conference on Computational Natural Language Learning, Proceedings*, pp. 271–280.

Wolf T., **Debut L.**, **Sanh V.**, **Chaumond J.**, **Delangue C.**, **Moi A.**, **Cistac P.**, **Rault T.**, **Louf R.**, **Funtowicz M.**, **Davison J.**, **Shleifer S.**, **Platen P.**, **Ma C.**, **Jernite Y.**, **Plu J.**, **Xu C.**, **Scao T.**, **Gugger S.**, **Drame M.**, **Lhoest Q. and Rush A.** (2019). Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771.

Xiong C., **Zhong V. and Socher R.** (2017). Dynamic coattention networks for question answering. In *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, pp. 1–14.

Yang Z., **Zhou Y. and Nyberg E.** (2016). Learning to answer biomedical questions: Oaqa at bioasq 4b. In *Proceedings of the Fourth BioASQ Workshop*, pp. 23–37.

Yin H., **Zhou F.**, **Li X.**, **Zheng J. and Liu K.** (2020). Transfer learning on natural yes/no questions. In *2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, pp. 1–6.

Yu L., **Hermann K.M.**, **Blunsom P. and Pulman S.** (2014). Deep learning for answer sentence selection. arXiv preprint arXiv:1412.1632.

Zhu Y., **Kiros R.**, **Zemel R.**, **Salakhutdinov R.**, **Urtasun R.**, **Torralba A. and Fidler S.** (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 19–27.

Cite this article: Dimitriadis D and Tsoumakas G (2024). Artificial fine-tuning tasks for yes/no question answering. *Natural Language Engineering* **30**, 73–95. https://doi.org/10.1017/S1351324922000286