A Framework for Automatic Generation of Spoken Question-Answering Data

Merve Ünlü Menevşe
Boğaziçi University
merve.unlu1@boun.edu.tr

Yusufcan Manav
Boğaziçi University
yusufcan.manav@boun.edu.tr

Ebru Arisoy
MEF University
saraclare@mef.edu.tr

Arzucan Özgür
Boğaziçi University
arzucan.ozgur@boun.edu.tr

Abstract

This paper describes a framework to automatically generate a spoken question answering (QA) dataset. The framework consists of a question generation (QG) module to generate questions automatically from given text documents, a text-to-speech (TTS) module to convert the text documents into spoken form and an automatic speech recognition (ASR) module to transcribe the spoken content. The final dataset contains question-answer pairs for both the reference text and ASR transcriptions as well as the audio files corresponding to each reference text. For QG and ASR systems we used pre-trained multilingual encoder-decoder transformer models and fine-tuned these models using a limited amount of manually generated QA data and TTS-based speech data, respectively. As a proof of concept, we investigated the proposed framework for Turkish and generated the **Tur**kish **Qu**estion **As**wering (TurQuAse) dataset using Wikipedia articles. Manual evaluation of the automatically generated question-answer pairs and QA performance evaluation with state-of-the-art models on TurQuAse show that the proposed framework is efficient for automatically generating spoken QA datasets. To the best of our knowledge, TurQuAse is the first publicly available spoken question answering dataset for Turkish. The proposed framework can be easily extended to other languages where a limited amount of QA data is available.

1 Introduction

Spoken question answering (SQA) is the task of finding the answer of a question from a given spoken document. A typical approach in SQA is to use a cascade of ASR and QA systems. ASR outputs transcriptions of spoken documents and QA searches these potentially erroneous transcriptions for the answers of given questions. Additionally, end-to-end SQA systems that jointly train audio and text have been proposed (Chuang et al., 2019; Lin et al., 2022). Compared to QA on text documents, SQA has been less explored, partly due to the limited amount of spoken datasets.

Spoken SQuAD (Li et al., 2018), which was generated from SQuAD (Rajpurkar et al., 2016, 2018) using the Google TTS and CMU Sphinx (Walker et al., 2004) ASR systems, is one of the largest SQA datasets. Another example of a TTS-based spoken dataset is Spoken-CoQA (You et al., 2022) which was generated from CoQA (Reddy et al., 2019). Open-Domain Spoken Question Answering (ODSQA) (Lee et al., 2018) is a large SQA dataset that contains the recordings of a machine reading comprehension dataset by native Chinese speakers.

In this paper, we propose a framework to automatically generate SQA data. Our framework contains (i) QG to automatically obtain question-answer pairs from given text documents; (ii) TTS to convert text into spoken documents; (iii) ASR to transcribe spoken documents. For each module in our framework, we use state-of-the-art systems – mT5 for QG (Xue et al., 2021), Google Text-to-Speech[1] for TTS and XLSR (Conneau et al., 2021) for ASR. Since both mT5 and XLSR are multilingual pre-trained models and Google TTS supports various languages, the proposed framework can be easily utilized for different languages to generate spoken QA datasets. Only the pre-trained models need to be fine-tuned with data from the language of interest. Fine-tuning the QG and ASR models requires limited amount of QA data and TTS-based speech data, respectively. Even though our framework follows a similar strategy with spoken SQuAD in generating SQA data, the textual QA data in our framework is also generated automatically. To the best of our knowledge, our work is the first study on automatic generation of SQA data from scratch.

As a proof of concept, we explored the application of the proposed framework to Turkish, where

[1] https://cloud.google.com/text-to-speech

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 4659–4666
December 7-11, 2022 ©2022 Association for Computational Linguistics

there are limited textual (Soygazi et al., 2021) and spoken (Ünlü and Arisoy, 2021; Ünlü et al., 2019) QA datasets. A **Tur**kish **Qu**estion **A**n**s**w**e**ring (TurQuAse) dataset was automatically generated using Wikipedia articles and QA performance on this dataset was evaluated with state-of-the-art models. Our main contributions can be summarized as (i) an easily extensible framework for automatic generation of an SQA dataset in a language of interest and (ii) the first publicly available Turkish SQA dataset, TurQuAse. We publicly share our code, model, and datasets as open source [2].

This paper is organized as follows. Recent work is summarized in Section 2. Section 3 presents the proposed framework. Section 4 describes the experimental setups and reports the results on Turkish datasets. Section 5 concludes the paper.

2 Related Work

2.1 Question Generation

Research in question generation has shifted from RNN or LSTM based models (Du et al., 2017; Song et al., 2018; Duan et al., 2017; Du and Cardie, 2018) to transformer encoder-decoders. These encoder-decoders take advantage of large pre-trained language models as starting point and then fine-tune the models with the dataset of interest (Lopez et al., 2020; Dong et al., 2019). With the idea of combining NLP tasks in a single framework, a text-to-text transfer transformer (T5) (Raffel et al., 2020) model was proposed. T5 allows the same architecture to be used for multiple NLP tasks. Its multilingual version, mT5 (Xue et al., 2021) has extended this idea to various languages. In our research, we utilize the pre-trained mT5 model to automatically generate questions from given text documents.

2.2 Automatic Speech Recognition

Recently proposed ASR models exploit the idea of large pre-trained models (Schneider et al., 2019; Baevski et al., 2020; Conneau et al., 2021). To be able to generalize the speech representations across different languages, XLSR model (Conneau et al., 2021) which is based on Wav2Vec 2.0 was proposed. In our research, we use XLSR for ASR.

2.3 Spoken Question Answering

A typical SQA system relies on a cascade of ASR and textual QA models to find answers to questions in spoken documents (Tseng et al., 2016; Lee et al., 2019; Ünlü and Arisoy, 2021; Li et al., 2018). To improve SQA performance, incorporating additional information from sub-words (Li et al., 2018; Lee et al., 2018), contextualized word representations (Su and Fung, 2020), ASR confusion networks (Ünlü and Arisoy, 2021) and knowledge distillation using text and speech domains (You et al., 2021a) have been investigated. Recent research on SQA has focused on using large pre-trained models in which acoustic and text data can be trained jointly (Chuang et al., 2019) or a self-supervised learning followed by contrastive multi-task manner can be used to learn the multi-modality representations (You et al., 2021b). To utilize the unlabeled data, an ASR transcript-free model pretrained with unpaired text and acoustic data was proposed (Lin et al., 2022).

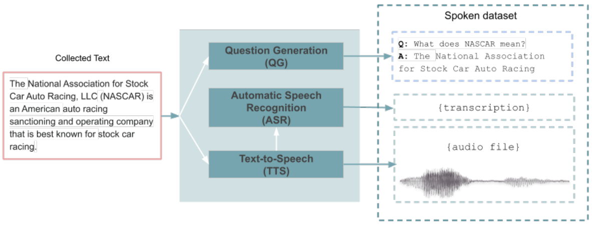
Collected Text
The National Association for Stock Car Auto Racing, LLC (NASCAR) is an American auto racing sanctioning and operating company that is best known for stock car racing.
Question Generation (QG)
Automatic Speech Recognition (ASR)
Text-to-Speech (TTS)
Spoken dataset
Q: What does NASCAR mean?
A: The National Association for Stock Car Auto Racing
{transcription}
{audio file}

Figure 1: The proposed framework. Collected paragraphs are taken as input and automatically generated question-answer pairs, TTS-based audio files of the paragraphs and corresponding ASR transcriptions are produced as output.

In our research, we evaluate the performance of the generated textual data using BERT (Devlin et al., 2019), mT5 (Xue et al., 2021) and Electra (Clark et al., 2020) QA models. We also evaluate the performance of the generated spoken data using BERT QA model on ASR transcriptions.

3 Framework

In this section, we describe the proposed framework for generating a spoken QA dataset from scratch. Figure 1 shows the framework where the input is text documents and the output is the dataset containing automatically generated question-answer pairs, TTS-based audio files obtained from the input texts and corresponding ASR transcriptions.

3.1 Question Generation

For question generation, we utilized mT5 (Xue et al., 2021), a multilingual encoder-decoder trans-

[2] https://github.com/mmerveunlu/Framework-QA-Dataset.git

former model. The encoder takes the input text and generates vectors as inputs to the decoder. The outputs of the decoder are generated in an autoregressive manner and passed to a softmax layer.

The mT5 model was fine-tuned in a multi-task manner on the answer extraction, question generation, and question answering tasks. We modified the QA dataset used for fine-tuning the mT5 model to generate training data for all tasks. The answer extraction task takes the context and predicts an answer span. The QG task uses the predicted answer span as input to generate a question. The QA task takes the question and the context as input to predict an answer span from the context.

In our framework, a single paragraph is given as input to the QG model as the context. The model first extracts possible answer spans and then uses the extracted answer spans with the given context to generate questions. For fine-tuning the QG model, we used a limited amount of manually generated QA data from the language of interest.

3.2 Text-to-Speech

We used the Google Text-to-Speech (TTS) framework to generate audio data. The input paragraphs were divided into smaller segments (10-word windows) to allow XLSR to be trained with a large batch size. Although Google TTS has an internal text normalizer, we normalized the text before using it as input to TTS to fairly evaluate ASR performance. Normalization involves converting numbers to letters and removing punctuation. Quality of the normalized text affects the quality of the synthesized audio and this may improve ASR performance.

3.3 Automatic Speech Recognition

The TTS-based audio files were fed into ASR to generate transcriptions. For ASR, we used the pretrained multilingual XLSR model (Conneau et al., 2021). The ability to learn speech representations across different languages allows this model to be utilized for ASR in various languages.

4 Experiments

This section explains how we used the proposed framework to generate the **Tur**kish **Qu**estion **A**n**s**w**e**ring (TurQuAse) dataset, and presents our Turkish QA and SQA experiments and results.

4.1 Turkish Text Data

For generating the TurQuAse data, we collected 460K Wikipedia pages using an XLM parser (Vardar et al., 2019). Each page contains a title, a subject, a table, and several paragraphs. The title indicates who/what the page is about. The table contains structured information about the page. For our framework, we used the first paragraph of each page in our Wikipedia dataset, since the first paragraph is usually a summary of the article with more general information. Then, we filtered out the paragraphs containing non-Turkish characters for better TTS performance, the paragraphs with missing subject field to diversify the data based on subjects and the paragraphs containing less than 40 words to provide longer context to the QG module. Finally, we ended up with 20.4K paragraphs.

4.2 Question Generation

The QG module was implemented in Python using the HuggingFace library (Wolf et al., 2020). We used the small pre-trained mT5 model with a batch size of 8 and 32 gradient accumulation steps to achieve an effective batch size of 256. The model was fine-tuned for 30 epochs with a learning rate of 1e^{-4} using two Turkish QA datasets, ThQuAD (Soygazi et al., 2021) and an English to Turkish machine translated version of SQuAD. These datasets contain 15.4K and 64.8K question-answer pairs, respectively. After fine-tuning, the QG model resulted in 83.6K question-answer pairs on the Turkish Wikipedia data explained in Section 4.1. Each paragraph has on average 4 questions, and the average question and answer lengths are around 7 and 3 words, repectively.

The performance of the QG model was evaluated on the development set of ThQuAD and XQuAD Turkish (Artetxe et al., 2020) with the BLEU and ROUGE metrics. For the evaluation, we compared the original and generated questions using both lemmatized and surface form representations of words. Table 1 shows that the results on ThQuAD are better compared to XQuAD. The reason why

		Rouge L	**BLEU 1**	**BLEU 2**
ThQuAD	Lemmatized	0.478	0.485	0.297
	Surface	0.443	0.390	0.235
XQuAD	Lemmatized	0.397	0.425	0.192
	Surface	0.328	0.307	0.116

Table 1: QG performance evaluation.

the model gives better results in ThQuAD than XQuAD may be that the training set of ThQuAD was also used to pre-train the QG model together with the machine translated SQuAD[3].

Among the 83.6K question-answer pairs generated from the Turkish Wikipedia articles, we manually evaluated 2.8K paragraphs with 12.3K question-answer pairs. This subset represents about 14% of the total data. A manual evaluation revealed that 55% of the questions were annotated as grammatically correct and sensible, and among these questions 11% had incorrect answer spans. In order to better understand the generated questions, we analyzed 168 randomly selected incorrect questions generated by the QG module and found the following distribution of errors: 37% factually inaccurate, 18% semantically incomplete, 39% grammatically incomplete, and 6% incorrectly formed entities.

4.3 TTS and ASR

The TTS and ASR models were implemented in Python using the TTS library [4] and the HuggingFace library (Wolf et al., 2020). Using TTS, we generated the audio files for all 20.4K paragraphs used as input to our framework and ended up with 223 hours of speech data. Then this data was decoded using the XLSR model to obtain ASR transcriptions. To fine-tune the XLSR model, we used a small amount of set apart text data from the collected Turkish Wikipedia articles. After generating the audio files with TTS for this subset, we ended up with 8 hours of speech data for fine-tuning the XLSR model and 2 hours of dev set for tuning the hyperparameters. The model was fine-tuned with an initial learning rate of $5e^{-4}$ for 30 epochs with a batch size of 2. Note that the articles used in QG and in fine-tuning the ASR model were disjoint. The ASR model yielded 14.8% WER on the paragraphs used as input in QG. By using the QG, TTS and ASR systems, we generated the TurQuAse dataset. To sum up, TurQuAse contains 83.6K automatically generated question-answer pairs from 20.4K paragraphs, as well as TTS-based audio files and ASR transcriptions corresponding to these paragraphs.

4.4 Question Answering

For QA experiments, we trained three models: BERT, mT5 and Electra. BERT and Electra models were trained with a batch size of 16 and a learning rate of $2e^{-5}$ for 20 epochs. The mT5 model was trained with a batch size of 8, 32 gradient accumulation steps and a learning rate of $1e^{-3}$ for 20 epochs. All models were fine-tuned on ThQuAD, TurQuAse, and a combination of these two datasets. Models were evaluated on the ThQuAD test set and the XQuAD Turkish set.

		ThQuAD		**XQuAD**	
		EM	**F1**	**EM**	**F1**
BERTurk	ThQuAD	0.57	0.76	0.47	0.64
	TurQuAse	0.46	0.67	0.43	0.58
	Combined	0.57	0.76	0.50	0.64
mT5	ThQuAD	0.45	0.64	0.33	0.51
	TurQuAse	0.32	0.52	0.32	0.47
	Combined	0.47	0.66	0.39	0.55
Electra	ThQuAD	0.58	0.77	0.46	0.64
	TurQuAse	0.45	0.66	0.44	0.58
	Combined	0.57	0.77	0.52	0.66
BERTurk	Zero-Shot	0.00	0.07	0.00	0.04
mBERT	SQuAD	0.37	0.57	0.36	0.51

Table 2: Scores of different QA setups on ThQuAD test set and XQuAD.

The Exact Match (EM) and F1 scores for the QA experiments are given in Table 2. The first column represents the models used in evaluation (BERTurk (Schweter, 2020), mT5 and Turkish Electra), the second column represents the QA data used in fine-tuning the models (ThQuAD, TurQuAse and combination of these two) and the remaining columns show the QA results on ThQuAD and XQuAD test sets.

In all experiments fine-tuning the models with ThQuAD alone leads to better results than fine-tuning the models with TurQuAse alone. This might be due to TurQuAse being a noisy QA dataset. Note that TurQuAse was generated automatically whereas ThQuAD was generated manually by human annotators. However, the combination of the ThQuAD and TurQuAse (Combined in Table 2) improves the results especially for XQuAD which is a QA test set from Wikipedia articles. For XQuAD, the EM improvements are 6.4% (from 0.47 to 0.50) with the BERTurk model, 18% (from 0.33 to 0.39) with the mT5 model and 13% (from 0.46 to 0.52) with the Electra model. Even though fine-tuning with the combined data did not improve F1 for the BERTurk model, we obtained 7.8% (from 0.51 to 0.55) and 3.1% (from 0.64 to 0.66) F1 improvements with the mT5 and Electra

[3] https://github.com/boun-tabi/SQuAD-TR

[4] https://github.com/pndurette/gTTS

	ThQuAD		XQuAD	
	EM	F1	EM	F1
ThQuAD	0.51	0.74	0.35	0.55
TurQuAse	0.47	0.69	0.40	0.58
Combined	0.53	0.74	0.47	0.63

Table 3: SQA Performance of the BERTurk model.

	ThQuAD		XQuAD	
	EM	F1	EM	F1
Reference	0.62	0.80	0.48	0.63
ASR	0.46	0.71	0.32	0.54

Table 4: QA Performance of the BERTurk model fine-tuned on the reference transcriptions of ThQuAD.

models. Fine-tuning with the combined data did not really improve the performance on ThQuAD. This might be due to the domain mismatch with the ThQuAD and TurQuAse datasets.

For further analysis, we evaluated the BERTurk model without any fine-tuning for the zero-shot experiments (Second to the last row in Table 2). However, the results have revealed that the Turkish BERT model without any fine-tuning can not be utilized for the given QA task. Additionally, we tested a multilingual BERT model fine-tuned on English SQuAD on Turkish datasets (The last row in Table 2). The results of these experiments show that using cross-lingual capabilities in QA models can be a viable research direction.

For SQA experiments, we only used the BERTurk model after fine-tuning with the ASR transcriptions. In order to investigate the SQA performance with the ThQuAD and the combined datasets on the ThQuAD and XQuAD test sets, we applied the TTS and ASR frameworks also to ThQuAD and XQuAD and obtained the ASR transcriptions for the paragraphs. Note that for a fair evaluation we removed the question-answer pairs from the training and test sets if the ASR system did not correctly transcribe the answer.

The SQA results are given in Table 3. The first column of the table represents the QA data used in fine-tuning the BERTurk model and the remaining columns represent the SQA results on the ThQuAD and XQuAD test sets. Similar to the QA experiments reported in Table 2, we did not observe improvements on ThQuAD even with the combined dataset. This might be again due to the noise introduced by the automatically generated TurQuAse data. However, we obtained improvements on top of the model fine-tuned with ThQuAD by using TurQuAse alone and in combination with ThQuAD. For XQuAD, the EM improvements are 14.3% (from 0.35 to 0.40) and 34.3% (from 0.35 to 0.47) and the F1 improvements are 5.5% (from 0.55 to 0.58) and 14.5% (from 0.55 to 0.63) for the models fine-tuned with TurQuAse alone and with the combined data, respectively. The best SQA performance on XQuAD was obtained when ThQuAD was combined with the TurQuAse dataset which shows the effectiveness of the proposed framework for the SQA task.

Additionally, we performed an experiment to investigate the effect of ASR errors on SQA. The BERTurk model fine-tuned on the reference transcriptions of ThQuAD was evaluated on the reference and ASR transcriptions of the ThQuAD and XQuAD test sets. The results are reported in Table 4. The first row shows the QA results on reference transcriptions and the second row shows the QA results on the ASR transcriptions. The EM and F1 scores have decreased when ASR transcriptions were used in the test set. This is an expected performance drop due to the ASR errors in the transcribed data. Comparing the first row of Table 3 (both training and test data are ASR transcriptions) with the last row of Table 4 (training data are reference transcriptions and test data are ASR transcriptions) shows that the performance drop can be alleviated to some extent by using ASR transcriptions both in training and test data.

5 Conclusion

In this paper we proposed a framework for generating SQA data from scratch. The framework outputs automatically generated question-answer pairs, audio data, and ASR transcriptions for a given input text. We demonstrated the effectiveness of the proposed framework by creating TurQuAse, the first publicly available SQA dataset for Turkish. Experimental results showed that the TurQuAse dataset improves SQA performance. The framework presented in this paper can be easily extended to other languages. As future work, we plan to improve the quality of the automatically generated question-answer pairs by including additional information to QG. We are also planning to collect real speech data for a subset of our Turkish dataset to compare TTS and real speech performances in SQA.

6 Ethics

The input text data used in this paper comes from publicly available Wikipedia pages. The input data, automatically generated questions and audio files, do not contain any personal information. The annotators participated in manual annotations voluntarily. The Wikipedia pages used to generate the dataset were compiled to cover as homogeneous topics as possible to avoid any bias towards a particular topic. We will make the generated Turkish dataset publicly available along with the implementation to ensure reproducibility.

7 Limitations

The empirical results reported herein should be considered in light of some limitations. The first limitation is in the collection of the speech data. Google TTS system is free and easy to use, but there is a daily limit on the requests submitted. This limit caused the audio data collection process to drag on. As a result, we could only collect audio data for a subset of large amounts of textual data. The second limitation is in computational resources. Multilingual state-of-the-art pretrained models require GPU support and large memory sizes during fine-tuning, even with small data. We utilized the models with small number of parameters because of our limited computational resources. The third limitation is in working with a limited resource language for QA. Due to the lack of Turkish QA datasets, we used the same dataset (ThQuAD) to fine-tune both the QG and QA models, which might impose a bias toward this dataset. However, this bias was alleviated to some extent when we used the spoken versions of the datasets (noisy datasets due to ASR errors).

Acknowledgements

The authors would like to thank Şeniz Demir for providing the Turkish Wikipedia dataset, Emrah Budur for providing the English to Turkish machine translated SQuAD dataset and the anonymous reviewers for their valuable feedback.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460. Curran Associates, Inc.

Yung-Sung Chuang, Chi-Liang Liu, Hung-Yi Lee, and Lin-shan Lee. 2019. SpeechBERT: An audio-and-text jointly learned language model for end-to-end spoken question answering. *INTERSPEECH*.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. 2021. Unsupervised cross-lingual representation learning for speech recognition. In *Interspeech*, pages 2426–2430.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *Advances in Neural Information Processing Systems*, 32.

Xinya Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from Wikipedia. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1907–1917, Melbourne, Australia. Association for Computational Linguistics.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874, Copenhagen, Denmark. Association for Computational Linguistics.

Chia-Hsuan Lee, Hung-yi Lee, Szu-Lin Wu, Chi-Liang Liu, Wei Fang, Juei-Yang Hsu, and Bo-Hsiang Tseng. 2019. Machine comprehension of spoken content: Toefl listening test and Spoken SQuAD. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(9):1469–1480.

Chia-Hsuan Lee, Shang-Ming Wang, Huan-Cheng Chang, and Hung-Yi Lee. 2018. ODSQA: Open-domain spoken question answering dataset. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 949–956. IEEE.

Chia-Hsuan Li, Szu-Lin Wu, Chi-Liang Liu, and Hung-yi Lee. 2018. Spoken SQuAD: A study of mitigating the impact of speech recognition errors on listening comprehension. *INTERSPEECH*.

Guan-Ting Lin, Yung-Sung Chuang, Ho-Lam Chung, Shu-wen Yang, Hsuan-Jui Chen, Shuyan Dong, Shang-Wen Li, Abdelrahman Mohamed, Hung-yi Lee, and Lin-shan Lee. 2022. DUAL: Discrete spoken unit adaptive learning for textless spoken question answering. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Korea , 18-22 September 2022*. ISCA.

Luis Enrico Lopez, Diane Kathryn Cruz, Jan Christian Blaise Cruz, and Charibeth Cheng. 2020. Transformer-based end-to-end question generation. *arXiv preprint arXiv:2005.01107*, 4.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. In *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pages 3465–3469. ISCA.

Stefan Schweter. 2020. BERTurk - BERT models for Turkish.

Linfeng Song, Zhiguo Wang, Wael Hamza, Yue Zhang, and Daniel Gildea. 2018. Leveraging context information for natural question generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 569–574, New Orleans, Louisiana. Association for Computational Linguistics.

Fatih Soygazi, Okan Çiftçi, Uğurcan Kök, and Soner Cengiz. 2021. THQuAD: Turkish historic question answering dataset for reading comprehension. In *2021 6th International Conference on Computer Science and Engineering (UBMK)*, pages 215–220. IEEE.

Dan Su and Pascale Fung. 2020. Improving spoken question answering using contextualized word representation. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8004–8008.

Bo-Hsiang Tseng, Sheng-Syun Shen, Hung-Yi Lee, and Lin-Shan Lee. 2016. Towards machine comprehension of spoken content: Initial toefl listening comprehension test by machine. *INTERSPEECH*.

Uluç Furkan Vardar, İlkay Tevfik Devran, and Seniz Demir. 2019. An XML parser for Turkish Wikipedia. In *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4. IEEE.

Willie Walker, Paul Lamere, Philip Kwok, Bhiksha Raj, Rita Singh, Evandro Gouvea, Peter Wolf, and Joe Wölfel. 2004. Sphinx-4: A flexible open source framework for speech recognition. *Sun Microsystems*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Chenyu You, Nuo Chen, Fenglin Liu, Shen Ge, Xian Wu, and Yuexian Zou. 2022. End-to-end spoken conversational question answering: Task, dataset and model. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1219–1232, Seattle, United States. Association for Computational Linguistics.

Chenyu You, Nuo Chen, and Yuexian Zou. 2021a. Knowledge distillation for improved accuracy in spoken question answering. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7793–7797. IEEE.

Chenyu You, Nuo Chen, and Yuexian Zou. 2021b. Self-supervised contrastive cross-modality representation learning for spoken question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 28–39, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Merve Ünlü and Ebru Arisoy. 2021. Uncertainty-aware representations for spoken question answering. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 943–949.

Merve Ünlü, Ebru Arisoy, and Murat Saraclar. 2019. Question answering for spoken lecture processing. In *Proc. ICASSP*, pages 7365–7369, Brighton, UK.