



GSQA: An End-to-End Model for Generative Spoken Question Answering

Min-Han Shih^{*1}, Ho-Lam Chung^{*1}, Yu-Chi Pai^{*1}, Ming-Hao Hsu¹,
Guan-Ting Lin¹, Shang-Wen Li², Hung-yi Lee¹

¹Electrical Engineering, National Taiwan University, Taiwan

²Meta AI, USA

mhshih0405@gmail.com, b09504006@ntu.edu.tw

Abstract

In recent advancements in spoken question answering (SQA), end-to-end models have made significant strides. However, previous research has primarily focused on extractive span selection. While this extractive-based approach is effective when answers are present directly within the input, it falls short in addressing abstractive questions, where answers are not directly extracted but inferred from the given information. To bridge this gap, we introduce the first end-to-end Generative Spoken Question Answering (GSQA) model that empowers the system to engage in abstractive reasoning. The challenge in training our GSQA model lies in the absence of a spoken abstractive QA dataset. We propose using text models for initialization and leveraging the extractive QA dataset to transfer knowledge from the text generative model to the spoken generative model. Experimental results indicate that our model surpasses the previous extractive model by 3% on extractive QA datasets. Furthermore, the GSQA model has only been fine-tuned on the spoken extractive QA dataset. Despite not having seen any spoken abstractive QA data, it can still closely match the performance of the cascade model. In conclusion, our GSQA model shows the potential to generalize to a broad spectrum of questions, thus further expanding the SQA capabilities of abstractive QA.

spoken Question Answering, Textless NLP, Generative Models, Transfer learning, Representation Learning

1. Introduction

Question Answering (QA) tasks have consistently emerged as one of the foundational challenges [1–4]. As a touchstone for measuring machine comprehension of textual content, QA presents unique complexities that have long captivated researchers. If answers are directly in the text and just need to be found, we categorize it as extractive QA. In contrast, when the answers are not explicitly present, needing a deeper dive involving inference and synthesis to craft a response, we refer it to as abstractive QA. With the rise of smart speakers and voice assistants, an emerging challenge in QA is Spoken Question Answering (SQA) to facilitate interactions between humans and these assistants. SQA requires both the questions and answers to be vocalized. Unlike text, speech encompasses richer details such as the identity of the speaker, their tone, and more, which introduces added intricacies to the SQA tasks.

An intuitive approach to tackle this challenge is to cascade models. Here, an Automatic Speech Recognition (ASR) system first transcribes the input speech of both questions and passages and then feeds transcriptions into a textual question answering language model that predicts the corresponding answers. Last, answers are transformed back into speech through a Text-To-Speech (TTS) model. The cascaded method's primary advantage

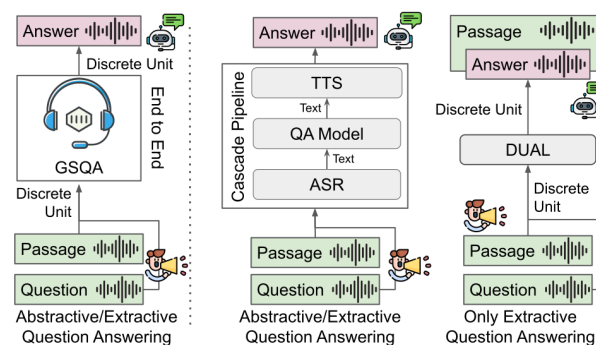


Figure 1: *GSQA compared to other baselines: The Cascade Method accommodates both abstractive and extractive QA but risks error propagation. DUAL is an end-to-end textless approach, exclusive to extractive QA. GSQA is a textless, end-to-end generative method, capable of handling both extractive and abstractive QA.*

is its ability to harness the great power of text-based language models trained on vast textual datasets, ensuring it can tackle both extractive and abstractive QA tasks. However, the cascaded model has its drawbacks. The primary issue stems from error propagations, since the text-based language models are not typically trained on ASR-erroneous data, the presence of ASR errors can notably undermine the performance of cascaded models. Moreover, some natural languages do not possess writing text, which leads to the absence of language models that are capable of acquiring these languages. Thus, to tackle these issues, an End-to-End speech model is needed.

¹ Recently, end-to-end, textless approaches (e.g., DUAL [5]) have emerged as a solution to the cascade method's limitations. These works [5–7] rely on the encoded representations of acoustic models, like HuBERT [8], and apply K-means clustering to transform speech representations into discrete units. These units, encapsulating both questions and passages, are fed into the model that predicts the span of the answer in the passage using start and end positions. By adopting this textless approach, DUAL effectively sidesteps the error propagation from ASR systems [9–11]. It improves the model robustness in noisy speech. However, there is a notable limitation to the DUAL model: it is not able to handle abstractive QA scenarios, as it is designed to only output start and end positions within the input spoken passage.

As Fig.1 shows the difference between our proposed method, cascaded method, and DUAL, we combine the advan-

¹* Equal Contribution.

tages of both the cascaded method and end-to-end (E2E) textless approach to introduce the first-ever textless E2E model capable of handling both extractive and abstractive QA. Mirroring DUAL’s methodology, we utilize HuBERT to convert the input speech (questions and passages) into discrete units [6]. A sequence-to-sequence model then processes these units to generate answers in the form of discrete units. Motivated by the strengths and limitations of the approaches above, we propose GSQA (Generative Spoken Question Answering), the first textless end-to-end model capable of handling both extractive and abstractive QA.

Furthermore, given the current absence of a spoken abstractive QA corpus, we begin by initializing GSQA with a generative Text QA (TQA) model. This TQA model has been trained on several extractive and abstractive datasets using a generative approach. Subsequently, we fine-tune the TQA model on NMSQA, a dataset dedicated to extractive spoken QA. The primary objective of GSQA is to impart the Natural Language Understanding prowess of the TQA model to GSQA. We also use Microsoft’s Azure TTS service to create a synthesized test set for evaluation. It’s noteworthy that, during the transfer learning phase, GSQA exhibited a significant enhancement, registering an 18% relative gain in the BLEU1 score in a zero-shot abstractive spoken QA setting.

The contributions of this paper are summarized below:

1. The introduction of spoken generative question answering model GSQA, the first end-to-end textless generative model.
2. The establishment of a method where pre-training on textual QA, followed by fine-tuning on extractive spoken QA, leads to zero-shot abstractive spoken QA.
3. Demonstrating that our proposed model has competitive performance against textless extractive models.

2. Related Work

The existing SQA methods primarily utilize Automatic Speech Recognition (ASR) [12, 13] transcripts, processed by a large language model (LLM) for QA tasks. A key challenge is that ASR errors significantly impact the LLM’s performance. For instance, less robust ASR systems produce incorrect transcripts, leading to erroneous LLM predictions. In contrast, more advanced ASR systems, such as Whisper-large with 1550M parameters, offer lower error rates and provide more accurate input for the LLM, improving QA task accuracy. However, these high-parameter models also result in longer inference times. In extractive SQA, models identify answer spans within spoken documents. DUAL [5] is noteworthy for its ability to determine these spans, but it fails when answers are not present in the input audio. TWIST [14] addresses SpeechLM’s [15] limitations, particularly its lack of semantic understanding, which often results in irrelevant or incorrect responses. While TWIST improves SpeechLM’s initialization, it still shows limited progress in transferring semantic knowledge from text to speech.

3. Methodology

3.1. Overview & Formulation

In the realm of Question Answering (QA), we divide the problem into three components: the Question (Q), the Passage (P), and the Answer (A). These entities can manifest in two formats: textual data (Q_t, P_t, A_t) and speech discrete unit-based data (Q_u, P_u, A_u).

Our method lies in the ambition to create a textless generative end-to-end language model that is capable of question an-

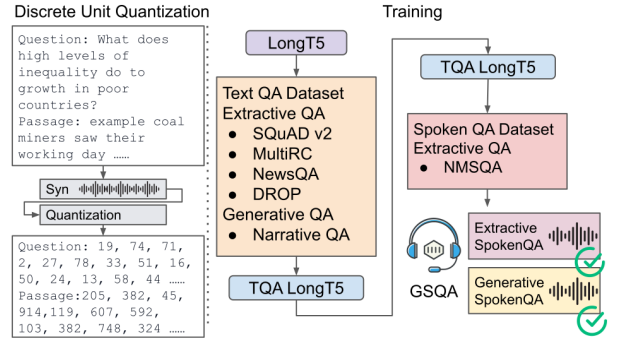


Figure 2: **Left:** The process of discrete unit quantization from synthesis data. **Right:** Model Training Procedure: A depiction of the transition from textual QA pretraining to spoken QA fine-tuning.

swering. Initially, we follow [6, 7] to quantize speech representations into discrete units. We then utilize a discrete unit-based sequence generative model to craft the answer units, which would be subsequently transformed into speech via a Unit-based HiFi-GAN vocoder [16]. To boost the logical understanding capabilities of the discrete unit-based generative model, we initialize this generative SQA model using weights from a textual model [14]. Fig.2 shows the entire model architecture and training pipeline.

3.2. Speech Quantization

HuBERT [8] is used for efficient speech representation, converting speech signals into discrete units through mask segment prediction training. This involves clustering each 20ms frame into one of K categories using k -means clustering, where the centroid ID of each cluster represents the frame’s discrete unit. To achieve this conversion, we utilize the HuBERT-base model with a 100-cluster configuration, following Lee *et al.*’s methods [6]. Additionally, we employ run-length encoding to condense consecutive identical units into a single unit, thus reducing the length of input unit sequences.

3.3. Text Question Answering Pretraining

Due to the scarcity of training data in abstractive spoken QA, our model’s performance could be limited. To address this, we propose using a pretrained textual model to enhance semantic understanding. This model is initially trained as a generative language model and then adapted for SQA, facilitating the transfer of textual QA capabilities to speech units.

Handling the longer sequences in speech, even after deduplication, presents a challenge, as many generative textual models struggle with such long inputs [17]. Therefore, our chosen initialized model must be generative and capable of processing sequences longer than 4096 tokens [18, 19].

We select LongT5 [19] for this purpose. LongT5, an extension of the T5 encoder, incorporates global-local attention mechanisms to efficiently manage longer inputs. It combines the attention efficiency of ETC with PEGASUS’s summarization-focused pre-training, offering improved performance. We further pretrain LongT5 on textual QA datasets, denoted as **LongT5-TQA**.

The primary goal during this phase is to train the model to process a textual question Q_t and passage P_t and generate the corresponding answer A_t . This pretraining is crucial, equipping our model with foundational QA skills for the subsequent

Model	Num. Param.	NMSQA-dev		NMSQA-test		Spoken-NarrativeQA-test	
		F1 Score	EM Score	F1 Score	EM Score	BLEU1	ROUGE-L
Cascade model (w/ ASR transcriptions)	1025M	49.1	32.0	47.3	30.4	13.5	19.9
DUAL	452M	39.4	21.9	33.6	21.2	-	-
Unit-LongT5	312M	25.5	12.6	20.1	9.4	6.8	10.4
Unit-LongT5-TQA (Proposed)	312M	41.8	24.9	36.0	24.0	8.0	11.8

Table 1: The table presents the speech-to-speech evaluation of our models and the number of parameters for each method. For extractive QA datasets, we utilized F1 and exact match (EM) as evaluation metrics. On the other hand, for abstractive QA datasets, we employed BLEU1 [21] and Rouge-L [22] for assessment. The cascaded model in this table includes an ASR system and an LM, which is *Whisper-medium.en* [13] and *LongT5-TQA* [19], respectively. DUAL [5] is combined a *HuBERT-base* [8], *K-means* model for 128 clusters, and a *Longformer* [23].

phases of our training methodology.

3.4. GSQA: Generative Spoken QA Model

Fig.2 illustrates our GSQA training procedure. As mentioned in Sec.3.3, we pretrain the LongT5 on Text Question Answering (TQA) tasks to produce LongT5-TQA, which is capable of text-level extractive and abstractive QA. This model is then fine-tuned on the Extractive Spoken QA dataset, NMSQA [5]. To unify the dataset format in abstractive QA form, we modify the labels in NMSQA from time spans to the answers’ waveform. Here, the given spoken passages, questions, and answers would go through a quantization model and would be converted into discrete units. Since LongT5-TQA does not recognize discrete units, we introduced them as new tokens. The embeddings of units are initialized by random sampling from other text token embeddings, ensuring proper tokenization and a seamless transition from text to speech. By these modifications, LongT5-TQA would be fine-tuned to predict answers in discrete units instead of text, where we denote the fine-tuned model as **Unit-LongT5-TQA**. Note that if the model’s denotation is not appended with TQA, it means that we don’t pretrain the model on TQA tasks. Last, to achieve speech-to-speech, we employ a HiFi-GAN vocoder [20] to convert discrete units back to speech. This vocoder includes a duration prediction module for deduplicated unit synthesis.

4. Experimental Setup

4.1. Dataset

For TQA pretraining data, We utilized four extractive QA datasets: SQuAD [24], MultiRC [25], NewsQA [26], and Drop [27], along with one abstractive dataset, NarrativeQA [28], to pretrain LongT5-TQA, and use it as foundation model of GSQA. On the other hand, for SQA training, we use NMSQA [5] as our downstream dataset for training Unit-LongT5. The NMSQA is derived from SQuAD v1.1 [29] and consists of Wikipedia paragraphs with human-written questions. While its train and dev sets feature synthesized speech using Amazon Polly, the test set contains audio from 60 human speakers (30 males and 30 females). In addition, we propose a new spoken abstractive SQA evaluation set, named **Spoken-NarrativeQA**. This dataset is built upon the NarrativeQA [28] test set, an abstractive text question answering dataset for reading comprehension from books and movie scripts with complex narrative understanding. Note that many answers in this dataset are not directly extractable from passages. We transform the text to speech using the TTS system and specifically select test data where answers are absent in the passage. There are 1626 testing samples in the Spoken-NarrativeQA dataset.

Method	Speech-to-(text/unit) model	QA Model	Total
Cascade model	Whisper-medium.en (764M)	LongT5-TQA (261M)	1025M
DUAL	HuBERT-large-128-layer ₁₋₂₂ (304M)	LongFormer (148M)	452M
Unit-LongT5-TQA	HuBERT-base-100-layer ₁₋₆ (51M)	LongT5-TQA (261M)	312M

Table 2: The respective number of parameters in our experiment. Note that the E2E method, like DUAL and our work, does not need an ASR system but needs a Unit-Extraction model to generate units, and the number after the name of that is the number of clusters.

NMSQA-dev	NMSQA-test	Spoken-NarrativeQA-test
8.1	14.8	6.3

Table 3: Word Error Rate (WER) of the pretrained *Whisper-medium.en* ASR system on different datasets for Spoken QA tasks.

Question	What did Tancred’s destiny seem to be?
Answer	To live the life of a normal member of the British ruling class.
Unit-LongT5	live the life of any
Unit-LongT5-TQA	live the life of any conventional member of the British ruling class

Table 4: Comparison of Results

4.2. Cascaded Pipeline

The conventional approach for SQA tasks typically follows a cascaded pipeline, as illustrated in the middle of Fig. 1. The passages and questions are first transcribed by an ASR system, followed by a text-based language model that generates text-based answers, and a TTS system converts the text-based answers into speech answers. To be more specific, we use *Whisper-medium.en* [13] as the ASR system, the fine-tuned LongT5-TQA model as the text-based language model, and the Azure TTS service as the TTS system. We record the cascaded pipeline’s experiment result in the Cascade Model column in Table 1. The total number of parameters of the cascaded model is shown in Table 2, and the WERs of the ASR system in given datasets are listed in Table 3.

4.3. Automatic Performance Evaluation

Since all our models ultimately output in speech, what we focus on is whether the content of this waveform contains the answer. We then use an ASR model to convert the generated waveform back to text, allowing us to use text-based metrics to evaluate the content of the model. To reduce errors from the ASR during evaluation, we utilize the state-of-the-art ASR model,

whisper-large-v2, during automatic performance evaluation. We then compare the transcriptions of the outputs from whisper-large-v2 with the text ground truth. We calculate the F1-score and Exact-matched (EM) scores for extractive QA tasks. The BLEU1 score [21] and the Rouge score [22] are used for abstractive QA tasks. Lastly, the experiment results of all models are reported in Table 1.

4.4. Implementation details

As the previous section mentioned, we pretrain LongT5 on the 5 TQA datasets for 13 epochs, the pretraining learning rate is set to 0.0005, and the weight decay is 0.01. On the other hand, we fine-tune the pretrained LongT5 on the downstream unit-based dataset, NMSQA, for 25 epochs, the downstream learning rate is set to 0.0003, and the weight decay is 0.001. Last, when we inference our model on the abstractive test set (e.g., **Spoken-NarrativeQA**-test), to encourage GSQA output longer answers, we use beam search for decoding with beam size of 5, and set length penalty to 2.

5. Results

As illustrated in Table 1, we evaluate our models on both extractive QA dataset NMSQA-dev, NMSQA-test, and on abstractive QA dataset Spoken-Narrative QA. Among the models evaluated, the proposed **Unit-LongT5-TQA** exhibits remarkable performance. On the NMSQA-dev set, it achieves an F1 score of 41.8% and an EM score of 24.9%. Its performance remains consistent on the NMSQA-test dataset, registering scores of 36.0% and 24.0% for F1 and EM, respectively. Furthermore, this model outperforms the others on the generative **Spoken-NarrativeQA**-test set, securing a BLEU1 score of 8.0% and a ROUGE-L score of 11.8%.

5.1. The effectiveness of TQA pretraining

From Table 1, we can observe that Unit-LongT5-TQA has huge improvements on both NMSQA-dev and NMSQA-test, which gets 16% F1-scores gain than Unit-LongT5 and gets two times than it on EM scores. It shows that pretraining language model on TQA tasks brings significant advantages for unit-based model initialization to learn semantic information in speech. On the other hand, in the Spoken-NarrativeQA-test, an abstractive spoken QA dataset, Unit-LongT5-TQA also outperforms Unit-LongT5 on both BLEU1 and ROUGE-L scores, which gets 1.2 and 1.4 increase, respectively. In our showcase, we have observed that TQA pretraining could further help the model generating more fluent sentences. The following results in Table 4 are from our test set. Without TQA pretraining, our prediction is "live the life of any," where the sentence is truncated in the middle, resulting in an incoherent meaning. However, after TQA training, the model would predict complete sentences.

5.2. Performance at different WERs

Fig. 3 illustrates the superior stability and efficiency of our E2E model for SQA, especially compared to the cascaded model's sensitivity to ASR WERs. The cascaded model, using ASR systems **Whisper-medium.en** and **Whisper-small.en**, demonstrates marked performance degradation with increasing WERs, whereas our E2E model maintains consistency even in high WER scenarios. Additionally, the cascaded model, particularly with **Whisper-medium.en**, requires significantly larger parameters, escalating computational resource demands. While enhancing the ASR system improves cascaded model perfor-

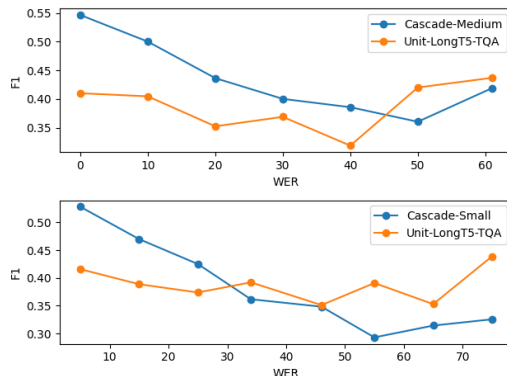


Figure 3: We sample the 8000 data within NMSQA-dev to verify the impact on the cascaded model under different Word Error Rates (WERs) with different ASR systems.

mance, this approach requires extensive paired text-speech data, larger model scales, and considerable training costs for both LLMs and the TTS system, making it extremely expensive. In contrast, our smaller E2E model potentially offers comparable performance with greater training efficiency. Furthermore, the E2E model circumvents the limitations of languages without written forms, which are insurmountable for ASR-based systems. These advantages position our E2E model as a more stable, efficient, and generally applicable solution for SQA tasks.

5.3. Models parameters

For a fair comparison, we strive to ensure the number of parameters across all models remains relatively consistent. Upon examination, the cascaded method is found to have a large number of parameters. This is primarily because it requires stacking three huge models. Since the performance of each model should not be compromised, opting for smaller models is not feasible, leading to a larger overall parameter size. In contrast, end-to-end models rely on a single model to accomplish SQA tasks, resulting in a reduced number of parameters. We observed that for abstractive QA, there isn't a stringent requirement for a compelling speech-to-unit model. It is sufficient to encode the content's information into discrete units. Instead, a more potent QA model is needed to infer the answers to the questions. Detailed parameter counts are presented in Table 2. Our model has the fewest parameters among all models. Notably, our model outperforms the others across various metrics.

6. Conclusion

In conclusion, we introduce the first end-to-end Generative Spoken Question Answering (GSQA) model, bridging the gap between spoken extractive and abstractive QA tasks. Combining the advantages of cascaded and textless models, our GSQA model demonstrates superior performance on extractive QA datasets, outperforming the previous E2E extractive SQA models. Furthermore, in the challenging abstractive zero-shot domain, our model exhibits competitive capabilities. We also highlighted the importance of pretraining the language model on textual data to enhance the unit-to-unit model's capability of semantic understanding. Our future research would concentrate on improving the model's performance with human narrations and exploring training strategies to enhance generalization ability across diverse SQA tasks.

7. References

- [1] M. Lewis and A. Fan, “Generative question answering: Learning to answer the whole question,” in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.
- [2] S. S. et al., “End-to-end synthetic data generation for domain adaptation of question answering systems,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, November 16-20, 2020*.
- [3] Z. Y. et al., “Contrastive domain adaptation for question answering using limited text corpora,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*.
- [4] —, “Domain adaptation for question answering via question classification,” in *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*.
- [5] G. L. et al., “DUAL: discrete spoken unit adaptive learning for textless spoken question answering,” in *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*.
- [6] H. I. et al., “Unity: Two-pass direct speech-to-speech translation with discrete units,” in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, ACL 2023, Toronto, Canada, July 9-14, 2023*.
- [7] A. L. et al., “Textless speech-to-speech translation on real data,” in *NAACL 2022, Seattle, WA, USA, July 10-15, 2022*.
- [8] W. H. et al., “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” *IEEE ACM Trans. Audio Speech Lang. Process.*, pp. 3451–3460, 2021.
- [9] C. L. et al., “Spoken squad: A study of mitigating the impact of speech recognition errors on listening comprehension,” in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*.
- [10] C. Lee, Y. Chen, and H. Lee, “Mitigating the impact of speech recognition errors on spoken question answering by adversarial domain adaptation,” in *ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*.
- [11] C. Y. et al., “Knowledge distillation for improved accuracy in spoken question answering,” in *ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*.
- [12] A. B. et al., “wav2vec 2.0: A framework for self-supervised learning of speech representations,” in *NeurIPS 2020, December 6-12, 2020*.
- [13] A. R. et al., “Robust speech recognition via large-scale weak supervision,” in *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, ser. Proceedings of Machine Learning Research.
- [14] M. H. et al., “Textually pretrained speech language models,” 2023.
- [15] Z. Zhang, S. Chen, L. Zhou, Y. Wu, S. Ren, S. Liu, Z. Yao, X. Gong, L. Dai, J. Li, and F. Wei, “Speechlm: Enhanced speech pre-training with unpaired textual data,” 2023.
- [16] J. Kong, J. Kim, and J. Bae, “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.
- [17] M. L. et al., “BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020*.
- [18] H. T. et al., “Llama: Open and efficient foundation language models,” 2023.
- [19] M. G. et al., “Longt5: Efficient text-to-text transformer for long sequences,” in *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*.
- [20] A. e. a. Lee, “Direct speech-to-speech translation with discrete units,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 3327–3339. [Online]. Available: <https://aclanthology.org/2022.acl-long.235>
- [21] P. et al., “Bleu: a method for automatic evaluation of machine translation,” in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.
- [22] C.-Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: <https://aclanthology.org/W04-1013>
- [23] I. Beltagy, M. E. Peters, and A. Cohan, “Longformer: The long-document transformer,” 2020.
- [24] P. R. et al., “Know what you don’t know: Unanswerable questions for squad,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018*.
- [25] A. W. et al., “Superglue: A stickier benchmark for general-purpose language understanding systems,” in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*.
- [26] A. T. et al., “Newsqa: A machine comprehension dataset,” in *Proceedings of the 2nd Workshop on Representation Learning for NLP, Rep4NLP@ACL 2017, Vancouver, Canada, August 3, 2017*.
- [27] D. D. et al., “DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019*.
- [28] T. K. et al., “The narrativeqa reading comprehension challenge,” *Trans. Assoc. Comput. Linguistics*.
- [29] P. R. et al., “Squad: 100, 000+ questions for machine comprehension of text,” in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*.