An End-to-End Generative System for Smart Travel Assistant

Miraç Tuğcu[1], Begüm Çıtamak Erdinç[1], Tolga Çekiç[1], Seher Can Akay[1], Derya Uysal[1], Onur Deniz[1] and Erkut Erdem[2]

[1] *Natural Language Processing Department, Yapı Kredi Teknoloji, Istanbul, Turkey*

[2] *Department of Computer Engineering, Hacettepe University, Ankara, Turkey*

{*mirac.tugcu, begum.citamakerdinc, tolga.cekic, seher.akay, derya.uysal, onur.deniz*}*@ykteknoloji.com.tr,*

Keywords: Generative AI, Voice Assistant, Text-to-Speech, Speech-to-Text, Chatbot, Language Models, Deep Learning, Natural Language Processing.

Abstract: Planning a travel with a customer assistant is a multi-stage process that involves information collecting, and usage of search and reservation services. In this paper, we present an end-to-end system of a voice-enabled virtual assistant specifically designed for travel planning in Turkish. This system involves fine-tuned state-of-the-art models of Speech-to-text (STT) and Text-to-speech (TTS) models for increased success in the tourism domain for Turkish language as well as improvements to chatbot experience that can handle complex, multifaceted conversations that are required for planning a travel thoroughly. We detail the architecture of our voice-based chatbot, focusing on integrating STT and TTS engines with a Natural Language Understanding (NLU) module tailored for travel domain queries. Furthermore, we present a comparative evaluation of speech modules, considering factors such as parameter size and accuracy. Our findings demonstrate the feasibility of voice-based interfaces for streamlining travel planning and booking processes in Turkish language which lacks high-quality corpora of speech and text pairs.

1 INTRODUCTION

When planning their trips, users encounter a range of options and constraints. Traditionally, the planning process relies mostly on online search engines and user interactions via an interface which can be cumbersome. Voice assistants offer a more flexible and accessible way for users to express their needs. Improving human-computer interaction is possible by developing such an interface with a virtual assistant to offer a natural and intuitive way to provide information. A voice-enabled assistant has the potential to significantly improve this experience by allowing users to verbally convey their needs, and receive both textual and spoken confirmations about booking details.

The main objective of such an assistant system is understanding the requests of user and perform an action related to a travel topic. Therefore, intent classification and slot-filling, which are two crucial NLU components, are used to decide which travel related function to perform e.g. searching for tours, booking a hotel, cancelling reservations and so on. In (Dündar et al., 2020), a robust intent classifier for Turkish language is proposed with a similar objective for the banking domain. However, a slot-filling module is also needed to perform an action related to intention based on the preferences of a user. To meet this need, a named entity recognition (NER) model can be integrated into the chatbot. A recent work (Stepanov and Shtopko, 2024) demonstrates a specialized transformer model that outperforms ChatGPT and fine-tuned LLMs in zero-shot cross-domain NER benchmarks for various languages except Turkish. Users might specify their preferences in a more natural manner where contextual relation and domain knowledge are required. With this purpose, slot-filling can be even more successful in a few-shot setting with LLMs (Brown et al., 2020) instead of zero-shot.

To understand the user's intention, we utilized a BERT (Bidirectional Encoder Representations from Transformers) classifier (Devlin et al., 2019), which has been specifically fine-tuned for Turkish (Schweter, 2020). BERT is well-suited for understanding context and nuance in a language due to its deep bidirectional architecture. This allows the model to consider the full context of a word by looking at words that come before and after it. This is partic-

Tuğcu, M., Erdinç, B., Çekiç, T., Akay, S., Uysal, D., Deniz, O. and Erdem, E.
An End-to-End Generative System for Smart Travel Assistant.
DOI: 10.5220/0013064200003838

In *Proceedings of the 16th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2024) - Volume 1: KDIR*, pages 473-479
ISBN: 978-989-758-716-0; ISSN: 2184-3228

ularly beneficial for agglutinative languages such as Turkish.

There are notable challenges to developing a voice-enabled travel assistant in Turkish, due to the lack of natural voice generation and Automatic Speech Recognition (ASR) models which are also robust to noise and low-quality voice sources. This is mostly because of the limited availability of high-quality parallel data for training robust speech recognition and synthesis models in Turkish. There are multi-lingual TTS models successful in generating natural speech or speech recognition e.g. XTTS (Casanova et al., 2024) and Whisper (Radford et al., 2023), however, the models either have licences not available for commercial use or demand high computational resources. The latency of a response generated by a smart assistant directly affects the user experience. Therefore, a mono-lingual and single speaker but a robust, small architecture satisfies the high-throughput need such as MMS (Pratap et al., 2024) and FastSpeech2 (Ren et al., 2020). For automatic speech recognition task, there are successful models introduced in recent years such as Wav2Vec 2.0 (Baevski et al., 2020) and Whisper (Radford et al., 2022) with multi-lingual foundation models available. However, this foundation models has only rudimentary capabilities in some languages such as Turkish and they require further fine tuning to perform well enough for active usage.

In this study, a chatbot with NLU modules such as intent classification and slot filling in the travel domain for searching, booking and purchasing purposes of hotels and tours is developed. We approached the slot-filling problem with a hybrid approach by using few-shot prompting technique with an LLM where context matters the most for user messages. We further trained robust and lightweight STT and TTS models for Turkish language in the tourism domain to develop a voice interface for the chatbot which completes the virtual assistant experience.

2 SYSTEM ARCHITECTURE

Visual representation of our developed system is illustrated at Figure 1. The user is able communicate with the assistant through speech modules or written chat. Conversation flow manager is a multi module system that understands the intention of the user, leverages generative slot-filling and pattern matching to perform an action with given information through travel services. Through the function calling component located in the conversation flow manager, intent classification, slot filling, and pattern matching components elucidated in Section 2.1 enable the semantic interpretation of transcribed sentences. The generative slot-filling and pattern matching components address different needs by employing distinct methodologies. Generative slot filling leverages generative models to identify entities that cannot be easily expressed through predefined rules, whereas the pattern matching component uses regular expressions and fuzzy match scores based on predefined dictionaries to detect entities with fixed formats, such as hotel names, cities, districts, and dates. By leveraging the information extracted from these sentences within travel planning services, the system facilitates user interaction, ensuring the effective execution of functions such as system utilization and information retrieval from services. Moreover, with the function calling component, we enabled dynamic modification of endpoints and service variables directly from the interface we designed. This approach allowed for the seamless integration of new services with intents and facilitated the rapid adaptation to changing service requirements without the need for additional coding.

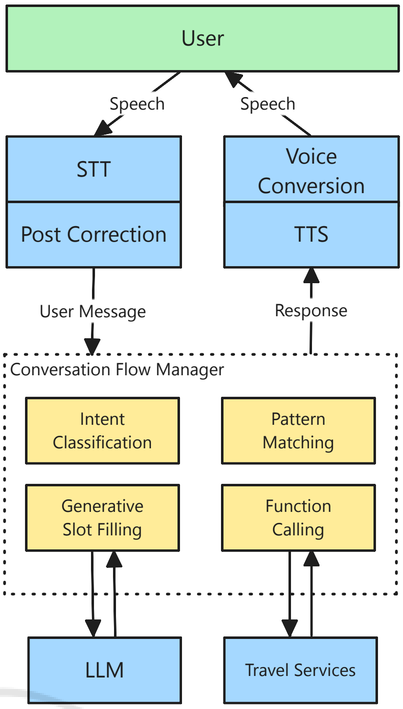

Figure 1: Overall virtual assistant architecture.

2.1 NLU Module

The Natural Language Understanding (NLU) component of our chatbot has two major components: intent classification and entity recognition. For intent classification, we utilized the BERT model based on the methodology outlined in (Dündar et al., 2020). We employed BERTurk model (Schweter, 2020), which is a model trained in Turkish corpora. This BERT based classifier met our demands and surpassed few shot training with LLMs, hence it is finetuned to be used as an intent classifier in travelling domain for this work as well.

On the other hand, we have observed that the entity collection for tourism can be challenging. The words we consider as entities can vary significantly in terms of subject matter and type. It may be necessary to perform entity extraction for a diverse range of entities, such as `spa`, `sport`, `aquapark`, `nature`, `outdoor pool`, `child/baby-friendly`, `pet-friendly`, `honeymoon`, and `seafront`. A user may wish to specify multiple features of the desired hotel within a single sentence to obtain results based on those criteria. Since it is more appropriate to consider these words as features rather than distinct entities, they were tagged as `feature1`, `feature2` and so forth, before being transmitted to the relevant services. Additionally, the sentences constructed by users do not adhere to specific rhetorical patterns. Due to these problems, we decided that the use of large language models is more appropriate for this problem because of their capability of understanding complex patterns with fewer training examples.

To achieve this, we utilize ChatGPT from OpenAI to extract entities from sentences with our generative slot-filling component. By engineering a dynamic prompt, we were able to receive a JSON-formatted output that parsed the specified types and numbers of entities from the given sentences. Additionally, through the modification capabilities provided within the application, we enabled the addition or removal of new entities without the need for further development.

Moreover, working with large language models inherently posed the risk of receiving outputs in irregular formats. To mitigate this, we provided JSON examples within the prompts and implemented checks to ensure the outputs adhered to JSON rules. Additionally, since user prompts were directly fed into the large language model for entity extraction, this opened the possibility for the system's outputs to be manipulated. To address this, we refined the prompts to prevent users from altering the system prompt and obtaining distorted results.

Additionally, as mentioned in Section 2, we ensured that entities, which could be defined by rules, were identified for travel services by comparing them against regular expressions and words in our custom dictionaries, using calculated fuzzy match scores. The system we developed is depicted in Figure 2.

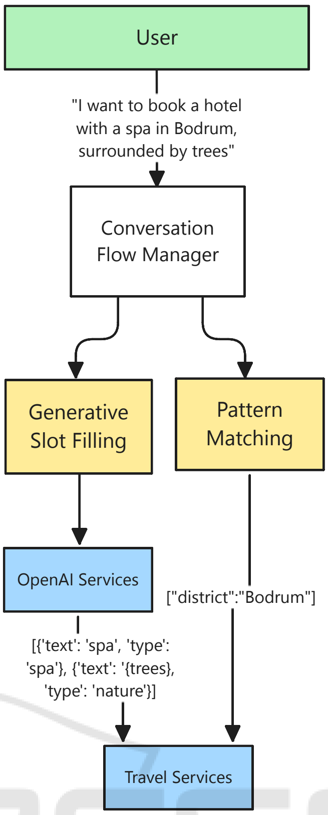
User
"I want to book a hotel with a spa in Bodrum, surrounded by trees"
Conversation Flow Manager
Generative Slot Filling
Pattern Matching
OpenAI Services
["district":"Bodrum"]
[{'text': 'spa', 'type': 'spa'}, {'text': '{trees}, 'type': 'nature'}]
Travel Services

Figure 2: Extracting entities from user prompts.

2.2 Text-to-Speech Module

The presented generation pipeline, as shown in Figure 3, contains a phoneme encoder, the LightSpeech TTS model, a vocoder and a voice conversion model. The text is converted to phoneme sequence by using an open-source Turkish grapheme-to-phoneme model and dictionary (McAuliffe et al., 2017). Speech synthesis with low latency is necessary for a seamless user experience which is why we use LightSpeech (Luo et al., 2021) and Parallel WaveGAN (PWG) (Yamamoto et al., 2020) vocoder that proved its efficiency. LightSpeech model is based on FastSpeech 2 but its architecture is designed more lightweight and more efficient via Neural Architecture Search. The audio quality is on par with FastSpeech2 while having a remarkable inference speed up. The generated mel-spectrograms are transformed into audio waveform by using a Parallel WaveGAN vocoder that is pre-trained on LibriTTS (Zen et al., 2019) which is capable of high-fidelity speech generation for Turkish. Finally, the OpenVoice (Qin et al., 2023) model is utilized for zero-shot cross-lingual Voice Cloning to specifically convert the speaker of the generated audio waveform. This pipeline allows alternative models to be used in

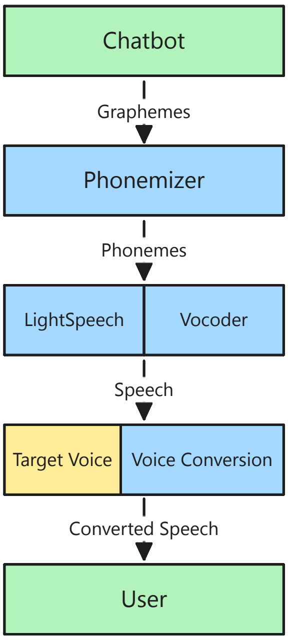

Figure 3: Text-to-speech pipeline for generation and voice cloning.

any part. For example, a vocoder model or a different voice cloning model can be easily implemented to replace respective parts.

2.3 Speech-to-Text Module

For the speech-to-text module Wav2Vec 2.0 speech recognition model is used and in order to correct potential errors in transcripts a post-correction method using an N-gram language model is used as shown in Figure 4. Wav2Vec 2.0 is a transformer based model that can be trained with raw audio data without any need for preprocessing (Baevski et al., 2020). Using raw audio data helps both with managing training data and with inference in the software pipeline as it does not introduce another layer that increases complexity. Using a multi-lingual foundation model with this architecture we have fine-tuned the model Turkish data and tourism related data. We have also implemented another layer for post-correction using N-gram based language model KenLM (Heafield, 2011). KenLM is a fast language modelling tool that can be used to create N-gram language models efficiently and also can be adapted to work with the Wav2Vec 2.0 model. Post-correction layer is used not only because it helps with correcting transcription errors that may arise due to similar sounding words and external noises; but also, recent studies have shown that using N-gram language model based post-correction can improve performance in low resource languages

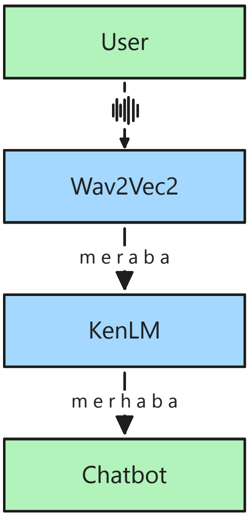

Figure 4: Speech-to-text pipeline that contains sequence modelling and post-correction models. As a side note, `merhaba` means `hello` in Turkish.

(Avram et al., 2023) and it can help with better adaptation on specific domains (Ma et al., 2023). We have created 5-gram language models to use in our experiments from general domain and tourism domain texts.

The other model we experimented on is W2v-BERT which combines the language model post-correction aspect into the trained model (Chung et al., 2021). This model uses a BERT encoder model as a language model instead of an N-gram language model. The advantage of using BERT is that it keeps a larger contextual information and also semantic knowledge of the words as well but compared to using an N-gram model it is a more resource demanding approach.

3 EXPERIMENTS & RESULTS

3.1 Text-to-Speech Experiments

Experimental Setup. We evaluate the LightSpeech model trained on a dataset that contains 5,131 audio samples with approximately 6 hours of novel reading without their text pairs. The average duration of the audio samples is 4.1 seconds. The transcriptions of audio samples are generated using our STT method. The errors in synthetic data consist mostly of similar-sounding words. Therefore, the effects of these errors are very little. Moreover, the errors in the synthetic transcriptions are expected to be minimal due to audio quality. The speech dataset and its phonemes are generated and aligned with the Montreal Forced Aligner public tool (McAuliffe et al., 2017) following (Ren et al., 2020). The audio waveforms are transformed into mel-spectrograms following (Luo et al., 2021), differently, we set the frame size and hop size to 300 and 1200 concerning the sampling rate of 24000. We train the model for 100k steps on a single NVIDIA V100 GPU. Models in our TTS pipeline other than the LightSpeech model are utilized as pre-trained models with their public weights.

Evaluation Methodology. There is no straightforward approach to evaluate speech generation. Most of the speech features like timbre or prosody may vary in the generated speech of a text compared to the ground truth utterance and it is an even harder challenge for multi-speaker datasets. Therefore, it is meaningful to evaluate a system by the aspect or the feature needed. We decided to evaluate intelligibility and pronunciation by transcribing the generated speech with ASR. We specifically choose a well-known and capable multi-lingual model Whisper (Radford et al., 2023), and 743 audio-text pairs from the Turkish subset of multi-lingual ASR benchmark known as FLEURS (Conneau et al., 2023) which is considered as an out-of-domain evaluation with respect to our training domain. The subset is approximately 2.6 hours long and the average duration of samples is 12.6 seconds. We generate speech of the texts from the dataset to create synthetic audio and original text pairs for each TTS model. ASR models transcribe TTS outputs into text hypotheses, allowing us to calculate the Word Error Rate (WER) and Character Error Rate (CER) by comparing them to the original transcript. For measuring the error rates, we apply the normalization of Whisper on references and hypotheses. In most cases, another preferable evaluation method is to evaluate naturalness and audio fidelity by Mean Opinion Score (MOS) metric but it's not an automatic evaluation strategy and the reliance on human raters presents a challenge. However, we decided to use a subset of 100 utterances generated for each model from the ASR benchmark we mentioned. We compare our results with public models successful in Turkish speech synthesis: 1) pre-trained Turkish MMS TTS model (Pratap et al., 2024) which is an end-to-end model with VITS (Kim et al., 2021) architecture, and 2) multi-lingual XTTS (Casanova et al., 2024) model with zero-shot voice-cloning feature that has a novel architecture based on Tortoise (Betker, 2023) and a HiFi-GAN vocoder (Kong et al., 2020) with 26M parameters. Parameter sizes of models are shown in Table 1.

Table 1: Text-to-speech models that is used in experiments and their parameter size.

Model	#Params
LightSpeech	1.8M
LightSpeech + PWG	3.1M
MMS	36.3M
XTTS	466.9M

Table 2: Evaluation results of Turkish speech synthesis by using Whisper models and FLEURS benchmark dataset. Original denotes the results of ASR models from Whisper paper (Radford et al., 2023).

Model	WER(↓)	CER(↓)
Whisper-medium		
LightSpeech	13.0	2.9
MMS	18.4	4.4
XTTS	10.1	2.5
Original	10.1	-
Whisper-large-v2		
LightSpeech	10.8	**2.5**
MMS	15.3	3.7
XTTS	8.3	**2.5**
Original	8.4	-

Table 3: MOS scores from a human study with regard to naturalness on a subset of Turkish FLEURS dataset.

Model	MOS(↑)
LightSpeech	2.98 ±0.081
MMS	3.34 ±0.082
XTTS	4.43 ±0.055

Results. In our experiments, the LightSpeech model (our setup) is able to generate utterances that preserve the text content better than the MMS TTS model, as shown in Table 2. LightSpeech performs less well than XTTS which has equal or better accuracy on ASR evaluation than the original utterances in the FLEURS dataset. However, our setup is as accurate as XTTS on the CER metric evaluation with Whisper-large-v2. Also, there is a slight difference with XTTS on the CER metric evaluation with Whisper-medium. However, the MOS score of LightSpeech is less natural than MMS and far from XTTS on naturalness as shown in Table 3. This is mostly due to the size of the training data and its recording quality. Also, our observations show that our model is less natural on long input sequences of the FLEURS benchmark because it is trained on a dataset with short sequences and is not able to generalize long sequences in terms of naturalness. Note that MMS and XTTS models have nearly 12x and 150x more parameters than LightSpeech + PWG respectively, as shown in Table 1. Therefore, the results show that the model is robust in comprehensibility but needs improvement on naturalness, considering the constraints imposed by its size and limited training data.

Table 4: Performance Comparison of Speech Recognition Models.

General Test Set		
Model	WER(%)	CER(%)
Wav2Vec 2.0	14.038	4.070
W2v-BERT	16.636	4.252
Wav2Vec 2.0 + kenLM	**8.106**	**1.669**
Tourism Test Set		
Model	WER(%)	CER(%)
W2v-BERT	13.112	2.229
Wav2Vec 2.0+kenLM	**8.888**	**1.974**

3.2 Speech-to-Text Experiments

Experimental Setup. The evaluation was done on a 6 hours dataset that is obtained from the public Turkish Common Voice dataset in the general domain and 1 hours dataset from tourism domain. For evaluation of Speech-to-text tasks, generally used metrics are word error rate (WER) and character error (CER). These metrics measure the error rates of transcriptions compared to the actual transcriptions of audio files. The lower error rates mean the model is more successful.

Results. Our experiments have demonstrated that using Wav2Vec 2.0 model together with kenLM post-correction outperforms using it without the language model and W2v-BERT model. The results are shown in shown in table 4. It is unsurprising for the N-gram language model post-correction to surpass the performance of the base model as the previous studies have shown similar results. The low scores from W2v-BERT model may be due to the multi-lingual foundation model's BERT model not being too successful in Turkish language.

4 CONCLUSION & FUTURE WORK

In this paper, we introduced the pipeline for a voice assistant in Turkish, that is capable of helping users in the tourism domain. This assistant leverages an intuitive voice interface by enabling users to seamlessly request information, access travel services, and complete their entire travel planning experience through spoken interactions. For slot-filling task of the assistant, a hybrid approach that combines regular expressions with few-shot LLM prompting is utilized. Additionally, lightweight and robust models for our NLU and speech modules are implemented to ensure a conversation at a natural pace. Our findings have demonstrated that the speech-to-text and text-to-speech models we trained achieved high intelligibility in spite of the scarcity of Turkish speech resources.

For future work, to improve the performance of text-to-speech models we intend to increase the quality and the quantity of our training data by speech enhancement and denoising techniques. We also aim to implement a zero-shot prosody cloning feature to the TTS pipeline to control the emotion emphasized in synthesized speech. For speech recognition, an additional post-correction model will be used to correct transcriptions of foreign words that can often be encountered in the tourism domain. For the NLU component, which constitutes the chatbot's understanding functions, we aim to leverage generative methods further to provide the user with more diverse and varied responses.

ACKNOWLEDGEMENTS

This work was done as a part of Smart Travel Digital Ecosystem project which was supported by TUBITAK (no: 9220043) and Celtic-NEXT (no: CE-2020-2-3) for the international industrial R&D projects program.

REFERENCES

Avram, A.-M., Smădu, R.-A., Păiş, V., Cercel, D.-C., Ion, R., and Tufiş, D. (2023). Towards improving the performance of pre-trained speech models for low-resource languages through lateral inhibition.

Baevski, A., Zhou, H., Mohamed, A., and Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations.

Betker, J. (2023). Better speech synthesis through scaling. *arXiv preprint arXiv:2305.07243*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. *CoRR*, abs/2005.14165.

Casanova, E., Davis, K., Gölge, E., Göknar, G., Gulea, I., Hart, L., Aljafari, A., Meyer, J., Morais, R., Olayemi, S., et al. (2024). Xtts: a massively multilingual zero-shot text-to-speech model. *arXiv preprint arXiv:2406.04904*.

Chung, Y.-A., Zhang, Y., Han, W., Chiu, C.-C., Qin, J., Pang, R., and Wu, Y. (2021). W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training.

Conneau, A., Ma, M., Khanuja, S., Zhang, Y., Axelrod, V., Dalmia, S., Riesa, J., Rivera, C., and Bapna, A. (2023). Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805. IEEE.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (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.

Dündar, E. B., Kiliç, O. F., Cekiç, T., Manav, Y., and Deniz, O. (2020). Large scale intent detection in turkish short sentences with contextual word embeddings. In *KDIR*, pages 187–192.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. In Callison-Burch, C., Koehn, P., Monz, C., and Zaidan, O. F., editors, *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland. Association for Computational Linguistics.

Kim, J., Kong, J., and Son, J. (2021). Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*, pages 5530–5540. PMLR.

Kong, J., Kim, J., and Bae, J. (2020). Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033.

Luo, R., Tan, X., Wang, R., Qin, T., Li, J., Zhao, S., Chen, E., and Liu, T.-Y. (2021). Lightspeech: Lightweight and fast text to speech with neural architecture search. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5699–5703. IEEE.

Ma, R., Wu, X., Qiu, J., Qin, Y., Xu, H., Wu, P., and Ma, Z. (2023). Internal language model estimation based adaptive language model fusion for domain adaptation.

McAuliffe, M., Socolof, M., Mihuc, S., Wagner, M., and Sonderegger, M. (2017). Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Interspeech*, volume 2017, pages 498–502.

Pratap, V., Tjandra, A., Shi, B., Tomasello, P., Babu, A., Kundu, S., Elkahky, A., Ni, Z., Vyas, A., Fazel-Zarandi, M., et al. (2024). Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.

Qin, Z., Zhao, W., Yu, X., and Sun, X. (2023). Openvoice: Versatile instant voice cloning. *arXiv preprint arXiv:2312.01479*.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. (2022). Robust speech recognition via large-scale weak supervision.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. (2023). Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Ren, Y., Hu, C., Tan, X., Qin, T., Zhao, S., Zhao, Z., and Liu, T.-Y. (2020). Fastspeech 2: Fast and high-quality end-to-end text to speech. *arXiv preprint arXiv:2006.04558*.

Schweter, S. (2020). Berturk - bert models for turkish.

Stepanov, I. and Shtopko, M. (2024). Gliner multi-task: Generalist lightweight model for various information extraction tasks. *arXiv preprint arXiv:2406.12925*.

Yamamoto, R., Song, E., and Kim, J.-M. (2020). Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6199–6203. IEEE.

Zen, H., Dang, V., Clark, R., Zhang, Y., Weiss, R. J., Jia, Y., Chen, Z., and Wu, Y. (2019). Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*.