DisfluencyFixer: A tool to enhance Language Learning through Speech To Speech Disfluency Correction

Vineet Bhat, Preethi Jyothi, Pushpak Bhattacharyya

Department of Computer Science, IIT Bombay, India

180260042@iitb.ac.in, pjyothi@cse.iitb.ac.in, pb@cse.iitb.ac.in

Abstract

Conversational speech often consists of deviations from the speech plan, producing disfluent utterances that affect downstream NLP tasks. Removing these disfluencies is necessary to create fluent and coherent speech. This paper presents DisfluencyFixer, a tool that performs speech-to-speech disfluency correction in English and Hindi using a pipeline of Automatic Speech Recognition (ASR), Disfluency Correction (DC) and Text-To-Speech (TTS) models. Our proposed system removes disfluencies from input speech and returns fluent speech as output along with its transcript, disfluency type and total disfluency count in source utterance, providing a one-stop destination for language learners to improve the fluency of their speech. We evaluate the performance of our tool subjectively and receive scores of 4.26, 4.29 and 4.42 out of 5 in ASR performance, DC performance and ease-of-use of the system. Our tool can be accessed openly at the following link[1].

Index Terms: speech recognition, disfluency correction, human-computer interaction, language learning

1. Introduction

Speech-based interactive features are rapidly expanding into appliances such as AI assistants, chatbots, autonomous driving, customer service, etc. However, they are challenging due to variations in speech, such as dialects, pronunciations and impairments. In addition, humans often do not have a well-defined speech plan that introduces filler words, repetition or correction of uttered phrases, making it difficult for the downstream understanding of the spoken utterance [1]. Words that are part of the spoken utterance but do not add meaning to the sentence are termed disfluencies, and removing them becomes an essential post-processing task for speech technologies [2].

Previous work has used ASR in English language learning for foreign speakers [3]. TTS has been applied for language learning by providing a method to aid learners in understanding correct pronunciations and prosody of the English language [4]. In this work we propose DisfluencyFixer, a tool that integrates ASR, DC and TTS models for performing Speech To Speech Disfluency Correction. In contrast to existing related technologies for speech-based language learning, we extend our tool to two languages - English and Hindi and provide an easy-to-use interface for deploying the system in any noisy environment.

2. DisfluencyFixer: A tool for Speech To Speech Disfluency Correction

DisfluencyFixer is an online web service created using ReactJS frontend and Python backend. It consists of an interactive user interface for automatic disfluency correction in input speech generating fluent output speech with transcription, information about the total disfluency count and type of disfluency. We describe the salient features of our tool below.

2.1. Features

1. **User Interface:** We provide a simple drop-down button to select between the two languages for input. The interface has two main components, as shown in Figure 1. The first component is the source dialog box, allowing users to start or stop recording their speech for analysis and fluent speech generation. They can also view the raw ASR transcript of their spoken utterance along with the audio file to hear their input. The second component is the target box, which shows the fluent transcript of the utterance and the type of disfluency identified. Users can also directly hear the fluent speech as output in this box.
2. **Topic generation module:** This feature allows the user to generate a random prompt for the conversation, mimicking the environment of standard English learning exams like TOEFL or IELTS. After reading the prompt, users can speak into the tool using the source dialog box.
3. **ASR Module:** Our backend ASR is the XLSR-wav2vec 2.0 [5] model finetuned on the CommonVoice [6] English and Hindi datasets. We chose this model because of its low latency and word error rate in noisy settings.
4. **DC Module:** Our DC system is based on the MuRIL transformer [7] architecture and trained on annotated disfluent English sentences from the Switchboard corpus [8]. For Hindi, we train the system in a zero-shot setting using synthetically generated data [9]. Since MuRIL is pretrained on many Indian languages, the representations it produces are of high quality. Finetuning the model for token classification enables it to classify every word in the sentence as disfluent or fluent. The final fluent text is obtained by removing all the disfluent words marked by the model.
5. **TTS Module:** We train a Forward Tacotron model[2] with a CARGAN [10] vocoder on the IndicTTS [3] dataset in English and Hindi and deploy it for TTS synthesis.
6. **Disfluency Classifier:** After performing manual annotation of real and synthetic data from [9], we finetune MuRIL transformer [7] for multi-class classification using a softmax
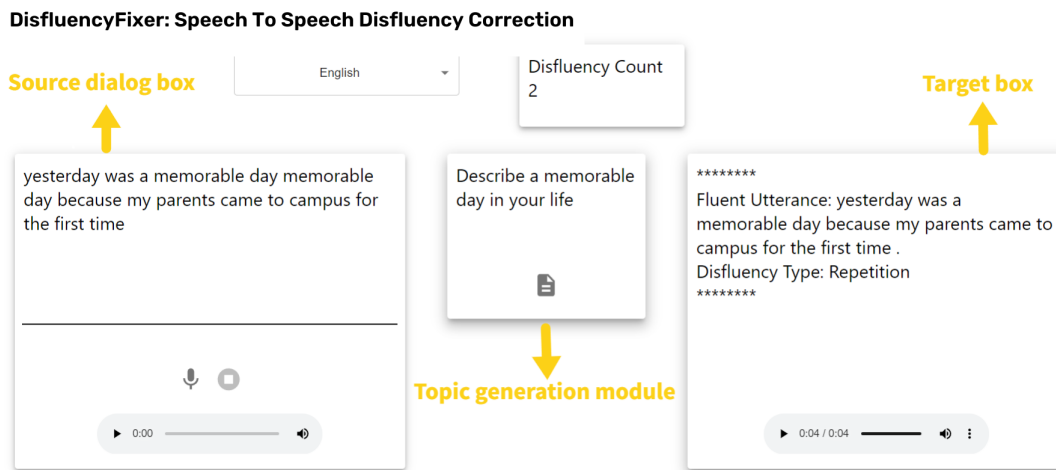

Figure 1: *Screenshot of DisfluencyFixer tool*

classifier over the following disfluency types: Filler, Repetition, Correction, False Start and Fluent.

7. **Disfluency Count Indicator:** The disfluency count indicator on the top right side of the interface measures the number of words removed from source utterance to create fluent speech. A higher disfluency count implies incoherent speech providing a way of automatic user feedback.

2.2. User Study

We asked 48 language learners to evaluate our tool on three parameters - Performance of ASR, Performance of DC and Ease of Use. Of these 48 participants, 30 were students potentially preparing for language testing exams. The remaining surveyors were language learners, either in English or Hindi. The results of the survey are described in Table 1 below.

Evaluation Metric	Score (out of 5.0)
Performance of ASR	4.26
Performance of DC	4.29
Ease of Use	4.42

Table 1: *Results of our subjective survey to evaluate the DisfluencyFixer tool; We report average score across both languages for the first two evaluation metrics.*

The survey demonstrates scope of improvement for our ASR and DC models. Our ASR model makes phonetic mistakes that are propagated to the DC model. An interesting observation is that our tool performs better when users test it on their laptop/desktop than on mobile phones/tablets, implying that the tool performance correlates significantly with the type of microphone used to record speech.

3. Conclusion and Future Work

In this paper, we introduce DisfluencyFixer, a tool which uses disfluency identification and correction to aid language learners improve the fluency of their speech. Our designed interface integrates ASR and TTS models to allow users to interact with the system through speech. The raw transcripts of disfluent and fluent speech, disfluency count and its type provide users with feedback on their speech and how to improve their conversational skills.

We are working on improving our backend ASR and DC models to improve the tool's overall performance. We also plan to add more insights and metrics in the web service to improve user experience based on the feedback we received from our survey.

4. References

[1] A. Gupta, J. Xu, S. Upadhyay, D. Yang, and M. Faruqui, "Disfl-QA: A benchmark dataset for understanding disfluencies in question answering," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3309–3319.

[2] M. Honal and T. Schultz, "Correction of disfluencies in spontaneous speech using a noisy-channel approach," *8th European Conference on Speech Communication and Technology*, 2003.

[3] H. Morton, N. Gunson, and M. Jack, "Interactive Language Learning through Speech-Enabled Virtual Scenarios," *Advances in Human-Computer Interaction*, vol. 2012, pp. 1–14, 2012.

[4] W. Cardoso, G. F. Smith, and C. G. Fuentes., "Evaluating text-to-speech synthesizers." 2015.

[5] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised cross-lingual representation learning for speech recognition," 08 2021, pp. 2426–2430.

[6] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*. European Language Resources Association, May 2020.

[7] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, B. Gopalan, D. K. Margam, P. Aggarwal, R. T. Nagipogu, S. Dave, S. Gupta, S. C. B. Gali, V. Subramanian, and P. P. Talukdar, "Muril: Multilingual representations for indian languages," *ArXiv*, vol. abs/2103.10730, 2021.

[8] J. J. Godfrey, E. Holliman, and J. McDaniel, "Switchboard: telephone speech corpus for research and development," *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1992.

[9] R. Kundu, P. Jyothi, and P. Bhattacharyya, "Zero-shot disfluency detection for Indian languages," in *Proceedings of the 29th International Conference on Computational Linguistics*, 2022.

[10] M. Morrison, R. Kumar, K. Kumar, P. Seetharaman, A. C. Courville, and Y. Bengio, "Chunked autoregressive gan for conditional waveform synthesis," *ArXiv*, vol. abs/2110.10139, 2021.