Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2021, 8(2):23-25**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Clean and Noisy Datasets Generation for DeepSpeech Open-Source Speech-To-Text Engine Based on Google Translate API

Varuzhan Harutyun Baghdasaryan

Bachelor of Computer Systems and Informatics, National Polytechnic University of Armenia, Armenia.
www.varuzh2014@gmail.com

Abstract Speech-to-text engines use both clean and noisy datasets to train models for best performance. But for some languages (for example, Armenian language) there is no enough data for training. The purpose of this article is to design a tool that can generate both clean and noisy(additive white Gaussian noise(AWGN) and real-world noise(RWN)) datasets for DeepSpeech speech-to-text engine using Google Translate's text-to-speech API feature that can convert text to normal and slow speech.

Keywords Speech-to-text engine, clean and noisy dataset, Google Translate's text-to-speech API, additive white Gaussian noise (WGN), real-world noise (RWN)

Introduction

DeepSpeech is a speech-to-text engine based on Baidu's Deep Speech research paper. DeepSpeech uses end-to-end deep learning. Project DeepSpeech uses datasets provided by Mozilla's other project calling Common Voice.

Common Voice is a crowd sourcing project started by Mozilla to create a free database for speech recognition software. Volunteers record sample sentences with a microphone and review recordings of other users. Voice databases are available under the public domain license CC0. This license ensures that developers can use the database for speech-to-text applications without restrictions or costs.

The Common Voice corpus consists of the following components:

- The *.tsv files output by CorporaCreator for the downloaded language.
- The *.mp3 audio files they reference in a clips sub-directory.

After DeepSpeech's database import and integration process, the clips sub-directory will contain for each required .mp3 an additional .wav file. It will also add the following .csv files:

- clips/train.csv.
- clips/dev.csv.
- clips/test.csv.

The .csv files have the following fields:

- wav_filename - the path of the sample, either absolute or relative. Here, the importer produces relative paths.
- wav_filesize - samples size given in bytes, used for sorting the data before training. Expects positive integer.
- transcript - transcription target for the sample.

According to Wikipedia, Google Translate supports 109 languages. From this, it is supposed that Google Translate's text-to-speech API can be used to create datasets for speech-to-text engines in 109 languages. The

main problem is that Google Translate's text-to-speech API for languages provides only female or not-human voices.

This article describes the general principles and steps of text data processing, converting it to speech, adding noises to speech and generating clean and noisy datasets corresponding to the Common Voice datasets structure.

2 types of noises can be added to audio data:

- additive white Gaussian noise (AWGN)
- real-world noises

Adding noise to a neural network during training can improve the robustness of the network, resulting in better generalization and faster learning. Besides this, it is a common approach to combine clean and noisy data. First, pre-train a network using the large noisy dataset and then fine-tune with the clean dataset. That is why need to generate additional 2 datasets with additive white Gaussian and real-world noises.

Additive white Gaussian noise (AWGN) is a basic noise model used in information theory to mimic the effect of many random processes that occur in nature. The modifiers denote specific characteristics:

- Additive - because it is added to any noise that might be intrinsic to the information system.
- White - refers to the idea that it has uniform power across the frequency band for the information system. It is an analogy to the colour white which has uniform emissions at all frequencies in the visible spectrum.
- Gaussian - because it has a normal distribution in the time domain with an average time-domain value of zero.

Real-world noises can be extracted from the environment. There are many types of real-world noises. For example: arctic wind noise, radio or tv static noise, etc.

Materials and Methods

Text data that will be used for the process must be normalized. According to Wikipedia article, text normalization is the process of transforming text into a single canonical form that it might not have had before. Normalizing text before storing or processing it allows for separation of concerns since the input is guaranteed to be consistent before operations are performed on it. Text Normalization rules are:

- Numbers, dates, acronyms, and abbreviations are non-standard "words" that need to be pronounced differently depending on the context. For example, "$200" would be pronounced as "two hundred dollars" in English.
- The text need to be normalized by removing non-alphanumeric characters, diacritical marks and regular expressions.
- The text need to be normalized by converting to a lower case.
- The text need to be normalized by converting multiple whitespace characters to a single whitespace character.
- The text need to be normalized by removing punctuation marks except apostrophe(').
- The text need to be normalized by containing one sentence in each line.

After importing text, system must give each line of text to Google Translate's text-to-speech API as input and get back normal and slow speech in following audio file specifications:

- Audio file format must be .wav.
- Channels number must be 1(mono).
- Sampling rate must be 16000 Hz.

To avoid Google Translate's text-to-speech API problems(for example, IP addressblocking) can simply add time out period and additional batch processing featurefor increasing performance speed. Also, it is maximum characters limit that Google Translate's text-to-speech API takes at a time.

Another important note is that DeepSpeech can only process audios that are longer than 0.5 seconds, shorter than 20 seconds and are not too short for transcript.

The next step is to random shuffle and split the generated audio data to create train, dev and test .csv files in the same folder where .wav files are (clips folder) with wav_filename, wav_filesize, transcript fields. Common

Voice corpus additionally contains validated.csv file. Audio files (clips) that are included in the official training, development, and testing sets must be included in validated.csv file too.

According to Common Voice paper, the number of clips is divided among the three datasets according to statistical power analyses. Given the total number of validated clips in a language, the number of clips in the test set is equal to the number needed to achieve a confidence level of 99% with a margin of error of 1% relative to the number of clips in the training set. The same is true of the development set. But it can simply be divided by a ratio of 80-10-10 or 60-20-20.

Adding noise to audio samples can evaluate the performance of machine learning models under these noisy conditions.

Additive white Gaussian noise is easier to model for analytical analyzes and it's easier to generate. But it may not represent realistic noise. There are many open-source tools that can add additive white Gaussian noise to audio data.

For adding real-world noise can be used another audio clip which contains real-world noise. It must just be removed from audio clip and add to target clip.

For these 2 types of noises, we can generate dataset containing audio clips and .csv files corresponding to Common Voice datasets structure.

Conclusion

The suggested method can be used to generate clean and noisy datasets for speech-to-text engines. One of the disadvantages of this method is that result depends on the voice gender and voice timbre provided by Google Translate's text-to-speech API. For example, a database created by this method in the English language will only contain female voices, because for English language Google Translate's text-to-speech API provides only female voice. But this problem can be solved by using third party software to change voice gender and voice timbre of audio file(clips) before generating .csv files.

References

[1]. Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, Andrew Y. Ng. (2014). Deep Speech: Scaling up end-to-end speech recognition.

[2]. Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, Gregor Weber. (2020).Common Voice: A Massively-Multilingual Speech Corpus.

[3]. Vineel Pratap, Qiantong Xu, Anuroop Sriram, Gabriel Synnaeve, Ronan Collobert. (2020). MLS: A Large-Scale Multilingual Dataset for Speech Research.