Speech Processing with Deep Learning

Leo Ebert, Tuananh Vu, Moritz Kirchner

September 2021

Table of Contents

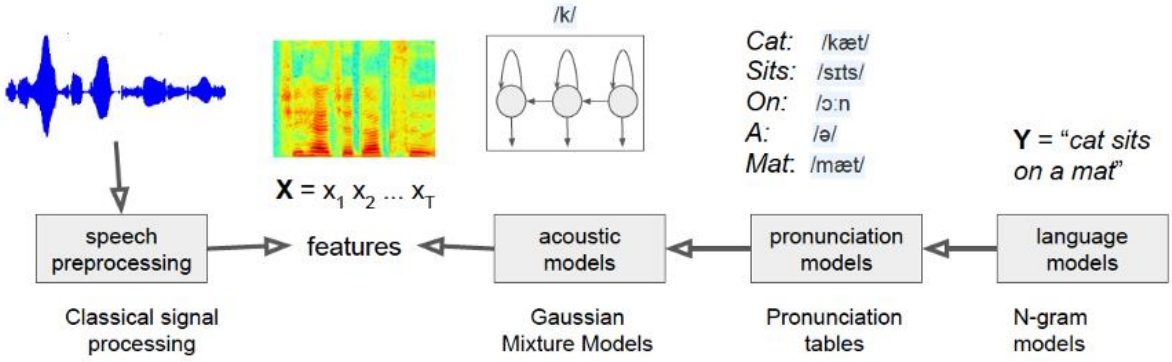
Introduction

The first question which we have to ask about speech processing is, why is that topic so important for the current time and the future?

Until around 2007 the most common way to communicate with machines was a keyboard and a mouse. However, the introduction of the iPhone with its touchscreen changed this significantly. According to the research of perficient mobile devices drove 61% of visits to U.S. websites only 35.7 % from desktop computers. The most important interface between man and machine has therefore changed drastically. But is speech processing the next big thing in this human machine interaction?

The first language was created about 100.000 years ago. Since then, language is the medium of mankind to exchange knowledge and communicate. But is speech recognition faster and more accurate than keyboard input? The answer is no. According to studies from the British Journal of Educational Technology and the Carnegie Mellon University speech recognition is quite as fast as typing but there are other benefits which are decisive for the usage of speech recognition. For example, it is not necessary to use hands or eyes. That is overall a benefit for people with disabilities and people do other things at the same time like using machines or driving a car for example. That's the main argument why speech recognition will be so important for the future.

In the following we will discuss an overview of speech recognition with deep learning. The reasons why neuronal networks are mandatory for a good speech recognition are that not only language is so complicated overall there are also big challenges in acoustics, pronunciation and context. The following image describes the procedure of speech recognition.



So from the one side we need to preprocess the audio signal and get it from an audio wave to digital numbers. Which we could do with some classical signal processing tools and algorithms.

On the other side we have to deal with the language, pronunciation and acoustic models. which must be learned by deep learning. To learn we need a lot of training data i.e. letter to voice and word to voice data sets. But let first talk about preprocessing.

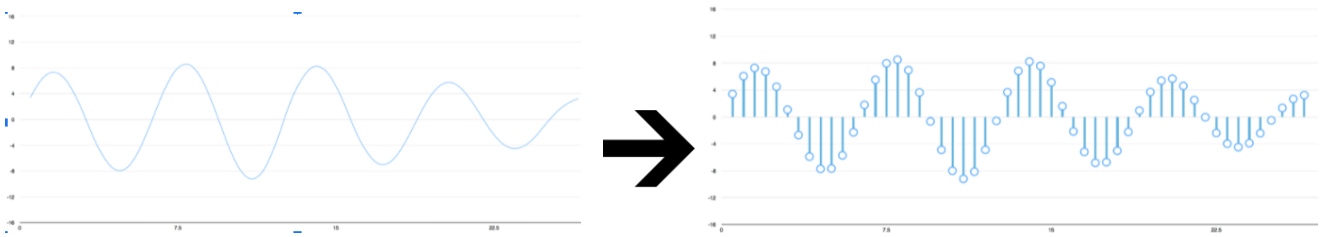
Preprocessing

The sound signal which comes i.e. from a microphone is usually delivered as a sound wave. To process the soundwave, it must be converted into bits. Since the conversion is based on known approaches from signal processing and has nothing to do with artificial intelligence, we will only go into this step roughly.

Sound waves

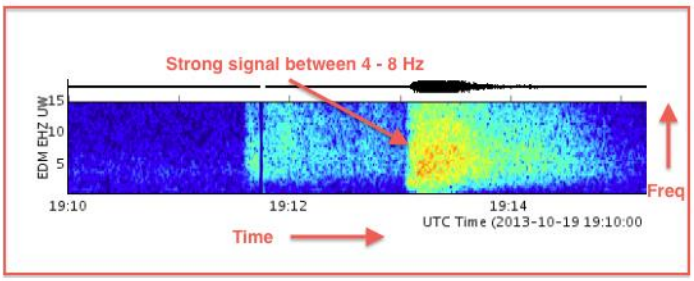
1. Step:

Sampling: getting analog wave signals into digital numbers by recording the height of the wave at equally spaced points (Image):



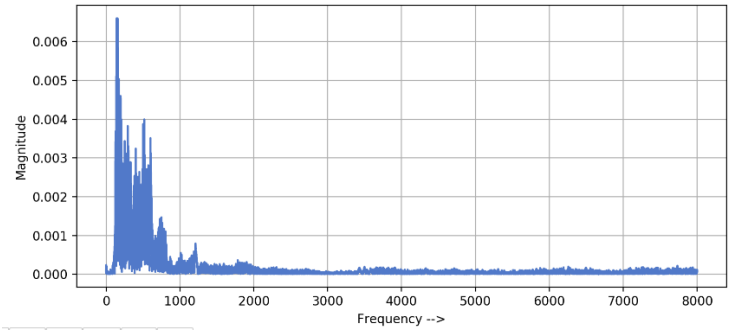
2. Step

To get an individual Footprint of each word the digital signal transformed to a spectrogram

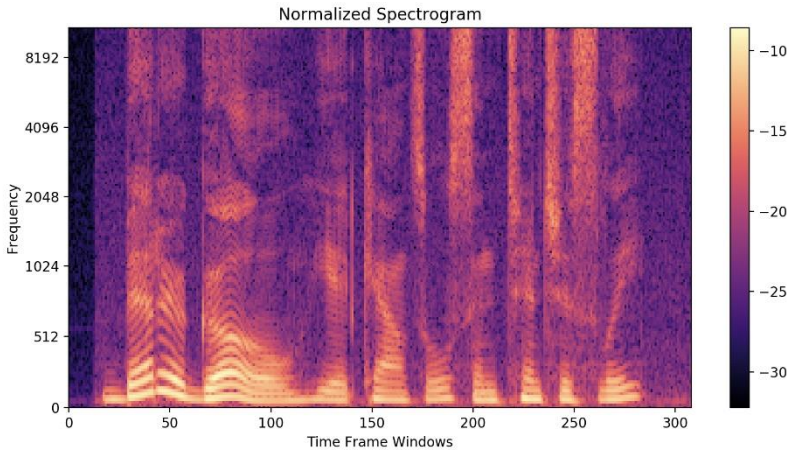


Spectrograms

In the image below you can see a „3-second long" signal. The signal is composed of thousand different frequencies, and every frequency value has a corresponding magnitude. You can see that in the first 2000 frequency values the magnitude is high and the frequencies after staying low. Unfortunately, there is not enough information for a neural network to predict text. The reason for that is we miss the time information.



For example, if we have an audio file with the phrase „how are you “, the system should be capable of predicting these words in the right order. With the information given in the image above, we don’t have any information about what was spoken first. Therefore, we need a different approach to calculate features. That’s why we need spectrograms.

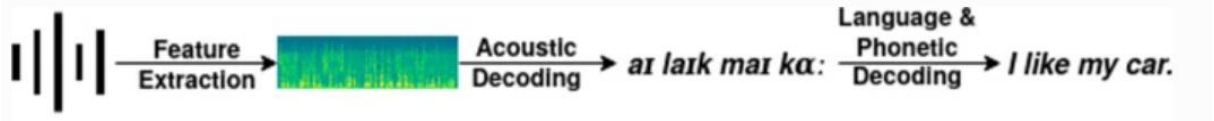


In the image above you can see a spectrogram. It provides us three information’s. One axis gives information about time, the second about frequencies and the colors about magnitude of the observed frequency at a particular time. Bright colors are strong frequencies.

With the given information from the spectrogram, we can feed a deep learning model. Audio was converted to an image. Now we have to change the perspective to an image classification problem. The deep learning model identifies english characters from left to right.

Automatic Speech Recognition

A typical way creating a service for Automatic Speech Recognition (ASR) is following a pipeline approach. The following image shows how such a pipeline would look like.

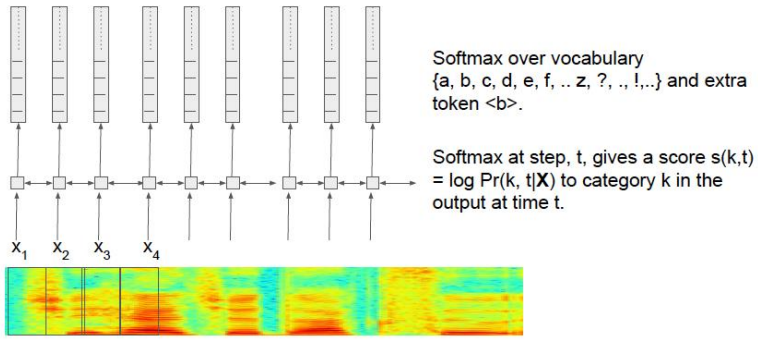


The previous section already showed how features from audio are extracted. After extracting the features, models are required which identify the context dependent phones. Typical a model for that is called acoustic model. So for the phrase „I like my car“ we first get „ai laik mai ka“. In order to get a phrase, which is understandable for humans a model is used to form valid words and sentences. This model is typically called language model.

Acoustic Model

Like already mentioned the acoustic model deals with the audio. To be precise, it deals with the features represented in the spectrogram. With the spectrogram the acoustic model predicts what phoneme each feature corresponds to, typically at the character or subword

level. Usually, the acoustic model is some kind of neural network. As seen in the following image a model tries to predict the characters from the spectrogram.



Language Model

In contrast to the acoustic model, the language model is typical a probabilistic model. The output from the acoustic model is the input for the language model and with the output the language model computes a probability for a word. It's common to build a language model on N-grams.

N-grams are just sequence of words. For example, „love you" is a 2-gram or bigram. And after Markov's assumption it is possible to predict the probability of some future unit without looking too far in the past. So, for a bigram model, the word is predicted based on the last word. Below you can see Markov's assumption, whereby w_{n-1} is the previous word and w_1^{n-1} are all the previous words.

$$P(w_n|w_1^{n-1}) \approx P(w_n|w_{n-1})$$

Postprocessing

Experiments show that the output of speech recognition is still not perfect. Many redundant white spaces, unnecessary special characters and case sensitivity problems exist. These errors can result in mistakes further down the development in the analysis, so they should be fixed. Especially for string analysis and matching processes, a correction is necessary. Unexpected special characters like a period or a question mark change the meaning of sentences. This can be seen in the following image:

`Let's eat, kids.´ vs. `Let's eat kids.´

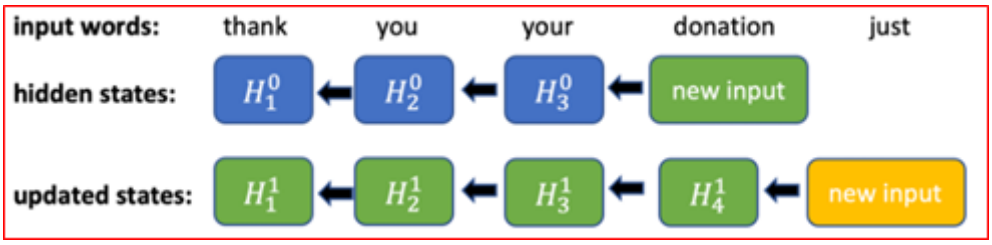
A post-processing system should…:

- have a very good performance on punctuation and capitalization from read in data
- should occupy little resources, since speech recognition is already computationally intensive
- have the ability to process unknown words

The old-fashioned way of handling the postprocessing is with a n-gram based approach. They

have a reliable quality and a fast conclusion. On the negative side, these approaches can need multiple gigabytes of space. Furthermore it´s not possible to handle words that weren't in the training set.

More up-to-date approaches use bidirectional RNN (Recurrent Neural Networks) or transformer-based neural networks. These are very accurate. The downside of these models is that the hidden states of the whole input must be updated with every new token. This approach is depicted in the image below.



A solution to that problem could be a TruncBiRNN. Since it is important to have future context when fixing punctuation, updating hidden states for following words is necessary. In this model the focus lays only on a specified window. So, in the forward direction, it is just a normal RNN. Going backwards however only the hidden state of a specified number is updated. This results in a constant time inference for every new token. Following image shows this:

Conclusion

After this short overview of speech recognition and processing the question occurs: if speech recognition is still working, what are the main challenges and tasks for the future and why is this theme still such a big topic in AI research. Leslie Pound the CEO of Tada Labs i.e. is certain that "voice connected to real-query data" will be the next big challenge in future voice technology. Also, the lack of trust and privacy issues are still problems in speech processing. Therefore, a connection to the internet is mostly needed to use services like Siri, Alexa or Google Home. Summarized speech recognition is still a hot topic and still has the potential to change our world. The foundation is created now it is in the hands of the companies to find new ways of usage, make this technology safer and more accessible to everyone.

References

1. Introduction
    - Paper: SPEECH-UNDERSTANDING SYSTEMS
    - Paper: Speaking versus typing: a case-study of the effects of using voice-recognition software on academic correspondence
    - Article: Mobile vs. Desktop Usage in 2020

2. Preprocessing
    - Article: What is a spectrogram?
    - Article: Understanding Audio data, Fourier Transform, FFT and Spectrogram features for a Speech Recognition System

3. Automatic Speech Recognition
    - Article: What is an Acoustic Model in Speech Recognition?
    - Article: Language Models: N-Gram
    - Paper: LSTM, GRU, Highway and a Bit of Attention: An Empirical Overview for Language Modeling in Speech Recognition
    - Paper: Performance vs. hardware requirements in state-of-the-art automatic speech recognition

4. Postprocessing
    - Paper: POST-SPEECH-RECOGNITION PROCESSING IN DOMAIN-SPECIFIC TEXT-CORPUS-BASED DISTRIBUTED LISTENING SYSTEM ANALYSIS, INTERPRETATION AND SELECTION OF SPEECH RECOGNITION RESULTS
    - Article: Post-processing in automatic speech recognition systems
    - Article: A Humorous Look at how Punctuation can Change Meaning
    - Demo: Simple audio recognition: Recognizing keywords

5. Conclusion
    - Article: Voice Recognition Technology Challenges In 2020, Possibilities For The Future