Capstone Proposal

Simple words speech recognition system

The project's domain background

Speech recognition systems try to imitate the human capability of understanding and extract information about what is being said, using sounds as the main input. The field of speech recognition has been a major subject of study since the early days of artificial intelligence, throughout this period the world has witnessed the development of a diverse range of speech-related applications, from the single digits' recognizer system “Audrey”, developed by “Bell Laboratories” in the early 50s, to a variety of popular virtual assistants, such as Siri, Google Assistant or Alexa, powered by state-of-the-art algorithms and models capable of receive instructions and perform a wide range of actions. These applications have had an important impact on several aspects of modern life and industry and will continue to play an important role in the coming years as the performance of the hardware increase with even lower costs and the accuracy of algorithms is improved every day.

Furthermore, been able to execute commands through spoken words is key in the development of human-machine interfaces, as this is the most natural way for human to issue instructions, consequently, get experience in the development of NLP applications and understand the underlines of such solutions will certainly empower my skills and the base knowledge to apply these innovations on potential products particularly those related with IoT applied to the manufacturing industry, which is the main subject of research in my job.

As an interesting starting point, in 2017 Google’s teams TensorFlow and AIY released to the public the dataset “Speech Commands Dataset”, containing 65,000 one-second long utterances of 30 short words [1]. This dataset was aimed to provide a simple but robust base for the development and testing of speech recognition systems. Alongside this, they also released a set o example models [2] and sponsor the Kaggle competition “TensorFlow Speech Recognition Challenge” [3] so developers or enthusiast with low knowledge could start right away working and learning.

A problem statement

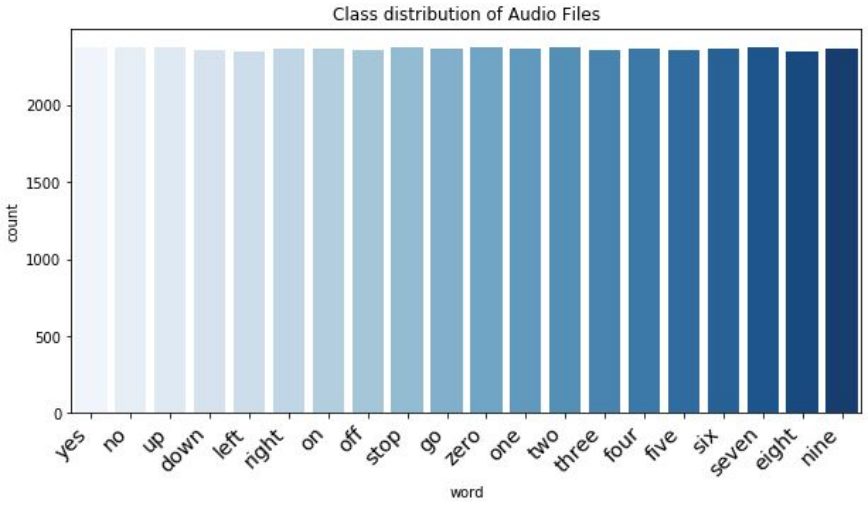
Following the bases of the Kaggle competition, the aim of the proposed project is to build a functional speech recognition system able to spot the following set of words: *yes, no, up, down, left, right, on, off, stop,* and *go,* plus two labels for *unknown* words and *silence*, so there are 12 possible labels [4].

The datasets and inputs

The dataset to be used in the project include the “Speech Commands Dataset, released by google in 2017. As described before, this dataset contains 65,000 one-second long utterances of 30 short words, by thousands of different people, contributed by members of the public through the AIY website.

The dataset is organized into 31 folders containing the audio files for each word. There are 20 core words ("Yes", "No", "Up", "Down", "Left", "Right", "On", "Off", "Stop", "Go", "Zero", "One", "Two", "Three", "Four", "Five", "Six", "Seven", "Eight", and "Nine"), and 10 auxiliary words ("Bed", "Bird", "Cat", "Dog", "Happy", "House", "Marvin", "Sheila", "Tree", and "Wow") to help to distinguish between unrecognized words. There is also included a set of files with background noise.

As show in the graph below, the distribution of the target classes in the given dataset is balanced, since all core words, included those we want to classify, have a similar number of samples, around 2370.



A detailed description of this dataset can be found at
https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/data

A solution statement

The approach for the solution for the given problem statement is to build, train and test a TensorFlow model for a classification task, having as input a one-second clip and as output one of the 12 possible labels.

Benchmark model

As a reference, the accuracy of the top ranked 200 models on the Kaggle competition range from 0.87 to 0.91, with the first place having a score of 0.91060. On the other hand, the example model used in the Google’s TensorFlow tutorial [2] could reach an accuracy of between 85% and 90%.

Set of evaluation metrics

Having a balanced dataset, as showed previously, and following the same metric used to evaluate submissions in the Kaggle's competition, this project will use Multiclass Accuracy, which is simply the average number of observations with the correct label.

Outline of the project design

As an initial step, I will do some data exploration to gain meaningful insights about the information contained in the dataset, some visualization in this phase will be helpful.

The first approach for the solution of this project will be to follow the model proposed by Google's TensorFlow tutorial [2] which uses a CNN based on the paper Convolutional Neural Networks for Small-footprint Keyword Spotting [5]. In this model audio clips are transform into spectrograms, which is a visual way to represent the spectrum o frequencies of the file over time, from this point on the input could be treated as a single-channel image, making the process more familiar since I have some experience with image recognition. Having this model as a framework the next step will customize it, evaluating the performance of variations for the possible hyperparameters in the training phase and trying different architectures, as light-weight CNN or even RNN.

Finally, having a proper understanding of the solutions it would be interesting to prepare better the input of the model using algorithms to preprocess the audio, trying to handle background noise or the different volume levels could be helpful. Using data augmentation could also provide a make the model stronger against

References

[1]. Warden P. (2017, August 24). Launching the Speech Commands Dataset. Retrieved from: https://ai.googleblog.com/2017/08/launching-speech-commands-dataset.html

[2] TensorFlow tutorials, Simple Audio Recognition. 2017. https://github.com/tensorflow/docs/blob/master/site/en/r1/tutorials/sequences/audio_recognition.md#simple-audio-recognition

[3] TensorFlow Speech Recognition Challenge. 2017. https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/overview

[4] TensorFlow Speech Recognition Challenge, Evaluation. 2017. https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/overview/evaluation

[5] Sainath T, Parada C. Convolutional Neural Networks for Small-footprint Keyword Spotting. INTERSPEECH 2015.