6/30/19

On Sunday 6/23, I suddenly received a message from my friend from Brandeis. Having some few chats with him, he suddenly introduced me to a job opportunity.

A team in Korea was apparently trying to find a software engineer to make their idea to real life.

Though I was interested at the beginning, I initially had to say no. This was because I had no knowledge on what they wanted me to make and more importantly, they had a strict deadline of finishing it in 3 days.

However, being myself (with sometimes a bit too much courage), I ended up telling my friend that I would need to look into what is available and I could decide if I can do it or not the next day.

Basically, their product was to transcribe meeting logs digitally. As I headed towards Yokohama to go to Karaoke with my friends, I quickly searched on Google for APIs since it was probably impossible to engineer some NLP speech recognition package in a day. I quickly resolved to Google Cloud Speech API. Reading the documentation, I figured that Google has our lives so much easier and immediately told my friend that this was going to be possible for me to complete. With his quick response, I was added to their Slack team and was explained about their product in further detail.

And this how my 2 day journey of creating the prototype of Dolly began!

First, since their UI seemed to look like it was transcribing speech to text as the meeting went on, I decided to go with the streaming option of the Google Cloud Speech-to-Text API. Finding out that it was easy to configure with multiple languages, I decided to support four languages in this prototype: English, Korean, Japanese, and Chinese. After doing some rewrites, I was able to get this function to work and it was now writing from speech to text. The other function the team wanted me to implement was the ability to recognize different speakers. However, looking through the documentation, there was no such functionality implemented in the API yet. Luckily, I found out that they had a beta version that would detect different voices although honestly, the accuracy isn’t that high yet. With this functionality also implemented, my final step was to output some interesting results including words that were repeated over 30 times and 10 times over the meeting. Another functionality that the Korean team wanted was “Dolly’s random suggestions”. Rather than completely “random”, I thought I could use some feature/keyword extraction technique from NLP and decided to output 5 of the most relevant words that may not have been repeated overly during the meeting. This finally concluded the development of the prototype of Dolly! However, it wasn’t the end of my job because I had to make sure that the Korean team can run this program locally. I made a whole instruction list from the very beginning of how to install Python so I will be making several posts in the future about this too.

In the end, I am glad that my friend introduced me to this opportunity and I was able to meet new people through work. I am also thankful that I was able to expand my knowledge in this new field of speech recognition and NLP.

Keep reading to see how I actually implemented Dolly!