Applications of the BBN Sage Speech Processing Platform

Ralf Meermeier, Sean Colbath

Raytheon BBN Technologies, Cambridge, Massachusetts, USA
rmeermei@bbn.com, scolbath@bbn.com

Abstract

As a follow-up to our paper at Interspeech 2016 [1], we propose to showcase various applications that now all use BBN’s Sage Speech Processing Platform, demonstrating the platform’s versatility and ease of integration.

In particular, we will showcase 1) BBN TransTalk: A turn-based speech-to-speech translation program running entirely on an Android smartphone, alongside a custom 3D-printed peripheral for it. 2) A continuous transcription and translation application running on a Raspberry Pi 3) An offline OCR application utilizing Sage, running on a COTS Windows laptop.

Index Terms: speech-to-speech translation, continuous speech recognition, optical character recognition

1. Introduction

The Sage Speech Processing Platform, and in particular the Godec framework that is part of it, has quickly shown its utility at BBN by unifying many of our formerly disparate applications into one underlying core framework. All three applications showcased here were created by integrating with the same Godec library (compiled for the platform at hand), but configured with a different processing graph and using models trained for their specific purpose. Since this graph creation happens at the lowest level in C++, very low latencies can be achieved even on platforms such as Android and the Raspberry Pi.

2. BBN TransTalk

BBN TransTalk is a two-way speech-to-speech (S2S) translation capability that runs entirely on an off-the-shelf Android smartphone. It uses a vocabulary customized for military-style applications such as security checkpoints.

Combined with the custom 3D-printed peripheral, the system is used in an “interview style” interaction where the person will speak a question in English, the system translates it, and the translation is output via TTS on a speaker. All this happens in a matter of seconds, entirely running on the smartphone. Similarly, the foreign speaker can respond in her own language, and the English speaker will hear the translation shortly after. This interview-style of interaction has proven to be very effective since it needs no introduction to foreign speakers, who are usually familiar with this type of interaction through television exposure.

Internally, TransTalk runs two Sage ASRs and two Sage MTs, all in separate processes as Android services.

3. Low-latency transcription and translation

This application, running on an off-the-shelf Raspberry Pi 3, is meant to illustrate with how little effort a low-latency, yet state-of-the-art speech application can be created by means of the Godec library, and still run on a resource-constrained platform. The application combines three technologies: Speech activity detection (SAD), speech recognition (ASR), and machine translation (MT), all in one configured graph (Figure 4). Note the use of a dummy (no CPU overhead) ASR decoder for the non-speech audio. This is an outcome of Godec requiring all parts of the incoming audio stream to be accounted for, as described in [1]. Also of note, unlike many other solutions, all updates down the decoding path are incremental. That is, the decoder will feed evolving utterances to the MT engine, so the translation is updated in real-time while it is being spoken, resulting in the lowest possible latency.

Copyright © 2017 ISCA
4. BBN OCR

This laptop-based OCR application had the previous custom-tailored Byblos engine [3] replaced with a Godec instance, in this case configured to decode a stream of text lines. While image analysis and line finding still happens separately for now, the actual decoding is entirely processed by the Godec graph. The application runs entirely on the laptop, but a server-based, web-enabled version exists as well.

5. Conclusion

The applications showcased here perform vastly different and complex NLP processing tasks, yet all rely on the exact same underlying framework. Because the processing is specified by a flexible graph, the applications can be quickly updated with latest research developments, an important consideration in commercial application development. In the future we plan to replace most of our remaining applications’ cores to use the Godec library to maximize its impact.

6. References

