In this write up, I will try to cover the span of my research keeping certain requisites and philosophy in mind. I am also the only PhD student working with Alan in core speech right now. Therefore, I will also align my writing with my aspirations for the prospective students.

Overview of current projects:

Let me start with the short term goal: *code mixing*.

Code-switching (or mixing) refers to the phenomenon where bilingual speakers alternate between the languages while speaking. It occurs in every multilingual society such as India, Singapore, etc. It is used both to express opinions as well as for personal and group communications. This can go beyond simple borrowing of words from one language in another and is manifested at lexical, phrasal, grammatical and morphological levels. The technology today - from speech processing systems through conversational agents - assume monolingual mode of operation and do not process code-switched content. However, the mixed content is intuitively the most important part in the content. Since the systems are now handling conversations, it becomes important that they handle code-switching. In simple terms, we need the following:

(1) Speech Synthesis systems that can synthesize codemixed content.
(2) Speech Recognition systems that can recognize codemixed content.

Speech Synthesis of codemixed content:

We can do this in multiple ways. First let's look at this problem with applications in mind:

While code mixing happens across different scenarios, there are two semi formal scenarios that might make sense to target as first applications: (a) News paper headlines where the content is primarily in native language (say, Hindi) with English words interspersed and (b) Navigation instructions where the content is primarily in English with named entities in the native language. We have handled (a) here and (b) here.

Now let's look at this from the perspective of available data. Speech synthesis typically uses clean recordings from a speaker in a controlled settings. Given that codemixing happens in social scenarios, it is difficult to get speaker data. There will be three scenarios here: (a) When we have data only from one language (b) When we have data from both the languages but monolingual in the language - One records data first in Hindi and later in English (c) When we have data that is truly mixed - YouTube videos with interviews of contemporary stars. (a) is handled here. (b) is handled here.

Finally let's look at this from the perspective of algorithms. Speech synthesis has at least these three modules: (a) Text processing (b) Acoustic Modeling and (c) Prosody Modeling. I have explained the techniques for each here and here.

Speech Recognition of codemixed content:

Code Mixing falls under low resource. So let's start with how to go about recognizing such content. Obviously, we can do a bunch of tricks. Here, we cover 4 such tricks:

(a) Augmenting content at data, feature and model levels
(b) Incorporating extra information into the existing data
(c) Modifying acoustic models
(d) Enhancing training procedure by using linguistic cues about the content.

Following this spirit, this work highlights how we can incorporate external information into the model in the context of code mixing.

Let's move onto the medium term goal: *Conversational Agents*

For building natural conversational agents, we need speech synthesis systems that can speak with intent and use filled pauses. For this, we did a bunch of tricks in our submission to Blizzard challenge 2018:

(a) We used Rhetorical Structure Theory to identify contrastive sentences.
(b) We tried binding acoustics and text components using a Tacotron style approach.

I will update the full paper in some time. (Its due in July)

We should also identify the intent. For this, we are participating in three challenges. The write up is here.