Naturalization of Text [Status Report 9/3/2022]

1 Tasks Performed

1. Preprocessing

   At the time of this writing, we are still currently working on preprocessing the corpuses in order to create the training data set. In order to do so, we needed to come up with a tagging list of disfluencies that can be abstracted to any given corpus.

   Since our end goal is to transform a given sentence into a more natural sounding one for potential use in technologies such as Amazon Alexa or Google Home, we ended up basing our choice of disfluencies on parameters accepted by the SSML (Speech Synthesis Markup Language) of modern text to speech technologies.

   Here are a selection of a few of the most prevalent disfluencies from our set:

   - (long_pause)
   - (med_pause)
   - (short_pause)
   - (no_pause)
   - (inhale)
   - (exhale)
   - (um)
   - (umm)

   Every tag in our set matches to a text pattern in our data. For example '...' would become a (long_pause) and '..' would become a (med_pause) and so forth.

   Then in the preprocessing step, we wrote a simple regex expression to replace occurrences of these disfluencies with our chosen tags.

   LENORE: ... So uh ... you don't need to go ... borrow equipment from anybody, .. to – do the feet?

   Would transform after preprocessing into:

   LENORE: (long_pause) So (uh) (long_pause) you don't need to go (long_pause) borrow equipment from anybody, (med_pause) to (short_pause) do the feet?

   This task has been more or less completed for the Santa Barbara Corpus. The only thing left to do being finishing up this preprocessing task for other potential datasets we would like to include in our project.

2. Transformation

   We have set up two initial models for adding disfluencies to existing input sentences. The models are prototypes at the moment, and are expected to change as the project develops.

   (1) The first is a bi-gram model that learns the frequency of each word and disfluency pairing in the training dataset. The model takes in an input sentence as well as a percentage modifier which determines what percentage of the input sentence words will have a disfluency 'attached' to them (either before of after the word). If two disfluencies are added in a row, then only the most probably of the two is kept, with the least probable discarded. For example, with an input sentence of "I'm tired, I'll sleep." and modifier of 0.50, we can encounter the output sentence "I'm tired (p=0.35: short_pause) (p=0.42: long_pause) I'll sleep". In that case, only the most probable of the two disfluencies would be kept: "I'm tired (long_pause) I'll sleep."

   (2) The second model is an auto-regressive transformer model, which takes an input sentence, and at each auto-regressive step uses the words so far to predict the most likely token to come next. The possible tokens are filtered so that only disfluencies are available to be selected, and the most probably disfluency is then added to the partial sentence. The next word from the input sentence is appended to the partial sentence (as we are not trying to re-predict the original sentence, only modify it) and the auto-regressive step begins again. This process continues until all of the words in the input sentence have disfluencies predicted for them. Then, based on the input modifier value, a subset of the predict disfluencies are removed.

3. Evaluation

   Recognizing the overheads of employing a group of people to evaluate the translations made by our model, we focus on automatic evaluation methods. We have implemented two methods:

   - Insertion distance in similar insertions - Consider a transformed sentence. Using the test corpus, identify sentences with similar insertions as the transformed sentence. Compare the distance of the insertion in the test sentence and the transformed sentence with an anchor word. Averaging this distance returns a score.

- Insertion distance in similar sentences - Preprocess the test corpus by calculating the sentence embedding. Now, consider a transformed sentence. Calculate its sentence embedding in the same vector space and compute similarity scores across the test corpus. Pick top n sentences and compare the distance of the insertion in the test sentence and the transformed sentence with an anchor word. Averaging this distance returns a score.

2 Risks and challenges

One of the biggest challenges in preprocessing our data currently is that every corpus in non-standardized and contains different levels of detail when it comes to labeling their text. For example, the Santa Barbara Corpus contains intonation units whereas something like the Movie Corpus does not. Therefore if we wished to include intonation as a disfluency label then using both corpuses could potentially result in a biased model.

Another challenge that we are facing is choosing appropriate evaluation metrics and methodology. The subject of our study is to naturalize speech. Therefore, the ideal method of evaluation would be to set up a survey that can get real humans to assess how well our model performs. However, that is not feasible in the scope and timeline of this project.

We have since then chosen to employ automated evaluation methods. However, that presents its own challenges. There is no accurate way to validate our results since embeddings could be arbitrary or there could be more than one 'correct' result. What sounds unnatural to one person could sound natural to another. Getting an objective measure of performance can be difficult due to the subjective nature of the domain.

3 Plan to mitigate the risks and challenges

In order to tackle the issue of non-standardized datasets, we have to settle the debate of quality over quantity. If we want a lot of data to train our model, we would have to select disfluency tags that appear in all the datasets. For example disfluencies such as '...' and 'uh' will almost be guaranteed to appear in all corpuses selected for this task. However having only those two tags would most likely result in a less accurate depiction of natural sounding text.

If we were to select a more robust corpus with higher levels of detail such as the Santa Barbara Corpus that contains tags such as intonation, tone, and phonetic transcriptions, then we would be limiting ourselves to fewer datasets to train our model.

To deal with this issue, we decided to find a middle ground between these two. We will not select every corpus we initially decided on in the planning stages while also not selecting every possible disfluency tag. Our plan is to limit ourselves to what we consider the most important tags, and then choose the appropriate corpora based on that.

Next, in order to overcome the challenge of subjective measures of success, we came up with objective evaluation methods. We plan to measure the performance of our model by using the metrics we proposed: insertion distance in similar insertions and similar sentences.

We are also considering using synthetic speech detection models as a test oracle to determine whether our model's outputs are 'natural'. We achieve this by adding a text-to-speech layer to generate speech data from the output before passing it to the oracle. The oracle will then determine whether the output sounds synthetic or not.