

Compressing natural speech - PROJECT REPORT -

1. Introduction

The idea for this project came a few months ago when I was thinking how useful it will be if our computers will be able to read the help files to us. But reading a help file is a challenging task because, in order to make his reading more comprehensible, the reader will have to use all sorts of prosodic variations (emphasize some of the words, change the timing). Using a speech synthesis system for this task will require a very complicated and powerful prosody model. This model will require a lot of training and one may expect that it will probably make mistakes in some cases. To avoid such an arduous task, one can just have someone (probably an artist) read the desired text and extract from the speech signal enough information to reproduce the original signal with the same quality. Preferably, when this information will be integrated into a speech synthesizer, the result will be as close as possible to the original interpretation of the text. The biggest advantage of this approach is that it will probably provide a *very good compression* of the original sound file. Instead of delivering the WAV or MP3 version of the sound, one will deliver only the information extracted from the speech signal and the text. This makes it very suitable for delivery over the Internet for example.

My project will try to materialize the above idea. To argue that the prosodic model approach may not be the most useful approach, consider the case when we have to read poetry, interpret a show or any type of text reading that requires some kind of artistic interpretation. Creating a prosodic model for these situations will probably fail miserably. On the other hand, having an artist do the “hard stuff” will be much easier and we can offer various interpretations of a text. Moreover, such an approach will probably allow us to use the interpretation with a different voice than the original one.

The project goal will be to find a mechanism to extract the prosody (the interpretation of the text) and then study if this information, when fed into a speech synthesis system, will yield a satisfactory reproduction of the original signal. Succeeding in this approach has many applications. First of all, a very good compression of the original signal will be obtained making it feasible to distribute news broadcast, books and any other type of text to devices with limited bandwidth access (computers connected by modem, cellphones, PDAs). In some domains (like news broadcast) the interpretations of the text are probably already available making their distributions even less costly.

Here is formal version of the problem. We have a text and an interpretation of that text by someone. The problem is to extract enough information from the original signal so that it can be accurately reproduced using a speech synthesizer. We (why I keep saying “we” since there is only me?; probably it sounds better ☺) will focus in this project only in

extracting the prosodic information which, at least intuitively, seems to be enough. There are three dimensions that need to be investigated:

- *Compression* – will the extracted information provide a good compression of the original signal? One will expect that since we will only have to store the text and the accompanying prosody, the compression will be very good.
- *Quality* – we also care about how good the original signal is reproduced especially since for some application obtaining the original signal may prove to be costly.
- *Portability* – will the extracted information apply to other voices than original? What do we have to do to transform it (for example transform from male to female)? Devising methods to do this transformation will lead to a lot of fun stuff (who will not want to use his voice and the president style of speech to utter something☺).

2. Experiment setup

In order to study the above dimensions, I collected the interpretation of seven texts (the artist was mYseLf - see Appendix for the texts uttered):

- Four of them were the same text from the limited time domain with different prosody (“The time is now exactly five past one in the morning”). As we will see later, the text was chosen from the limited domain so that an “upper bound” in quality can be investigated. They accounted for variation in:
 - duration - longer vowels in “time” and “now” and faster speech for the rest of the text : “The tiiiiime is noooooow exactly five past one in the morning”.
 - pitch - end of question prosody (rise in pitch) for “now”, “one” and “morning” : “The time is now? exactly five past one? in the morning?”.
 - both pitch and duration – by singing the same text – 2 versions (impossible to reproduce in words – yes, my singing was so marvelous!!! ☺)
- Two of them were jokes. I choose them because good joke tellers employ all sort of prosodic variation to tell the joke. Both of them are a two person situation. Thus, the interpreter of the joke will have to somehow mark that not only at the semantic level but also at the prosodic level.
- One was a fragment from the Microsoft Excel help file (academic purposes – please don’t sue) just to follow the original idea. Also, this text will try to capture the effect of size on the prosodic information extraction and give a more reliable estimation of the compression.

3. Extracting the information from the original signal

When interpreting something, people use various methods to change their style of speech. The most frequent ones are the change in duration and the change in pitch. Of course, there are other methods like changing the amplitude (speaking louder or softer), whispering, shouting, changing the word/syllable stress. We will choose to extract only duration and pitch because our tools, Festival and Festvox, have support for extracting them (I have limited experience with them, so they might have support for other things mentioned above).

Extracting duration and pitch follows the same path as building a unit selection synthesizer:

1. Prepare the text to be uttered,
2. Generate the synthesized prompts using a default synthesizer,
3. Record the prompts,
4. Automatic phonetic labeling,
5. Extract pitch marks and pitch contour
6. Build utterance structure

After executing all these steps, we can extract the necessary information by using `dumpfeats`. We extracted for every phoneme its name, its duration and the pitch in the middle of the phoneme. Since for longer phones (especially vowels), having the pitch only in the middle of the phone may prove not to be enough because of specific variation in pitch inside the phone (for example when singing), we decided to experiment by extracting the entire pitch information. Using the full pitch information may pose problems because usually the pitch contour extraction is not very reliable. Using just the middle of the phone value can be more precise due to averaging and interpolation used to get this value. Since full pitch information is provided every 5 milliseconds (which is too fine-grained), the full pitch information used was the average of pitch over a 100 millisecond interval (average over 20 pitch values). We will experiment with both options (using only the middle-of-the-phone pitch or using the full pitch information) to see what is their effect on the quality of the generated speech.

The prosodic information extracted above was used to generate utterance description that was then fed to Festival to generate the speech signal. Utterance descriptions require three fields: the phone, the phone duration and a list of pitch target point and pitch value in that point. Below, there is an example of the beginning of an utterance description using only the middle-of-the-phone pitch or full pitch information.

Middle-of-the-phone F0	Full F0
(set! utt1 (Utterance Segments ((pau 0.2) (dh 0.035 (0.0175 125.017)) (ax 0.5 (0.25 125.017)) (t 0.83 (0.415 117.514)) (ay 0.455 (0.2275 105.898))	(set! utt1 (Utterance Segments ((pau 0.2) (dh 0.035 (0.0175 125.01651)) (ax 0.5 (0.05 125.01651)(0.1475 125.01651)(0.2425 125.01651)(0.3375 125.01651)(0.4325 125.01651)(0.49 125.01651))

These descriptions were then used in Festival using the default diphone voice (`kal_diphone`) to generate the synthesized interpretation.

4. Results

As described in the end of Section 1, there are three dimensions to be taken in account: the quality of the generated speech, the compression achieved and the portability of the information extracted. We will analyze every aspect in part.

Quality

After extracting the prosodic information and generating the utterance descriptions, the Festival speech synthesis system was used to generate the synthesized interpretation. We used the default diphone voice (kal_diphone).

When compared with the default uttering of the text by Festival, the quality of the speech generated using natural prosody was much better, though there are some errors caused by misalignment (some of the phones extend to the following space) and F0 extraction (this was due to error correction used in F0 extraction algorithm that will lower the F0 if it exceeds a certain range). All this errors can be corrected by minimal hand labeling and tuning the F0 extraction (which was not done at all by me).

But when compared with the original speech, the quality of the generated speech has worse. This was especially due to the synthesized voice quality (you can clearly notice problems at phones transitions). In order to factor out the quality of the default voice, we performed another experiment. Given the fact that the quality of the voice in the “time” limited domain synthesizer was very good, we used the same artist (yes, me!) to build such a voice and generated the synthesized interpretation using this new voice. The quality of the generated signal was much better than the default voice and it was very close to the original signal. The errors caused by the labeling and pitch extraction are even easier to notice in this experiment.

We also tried to see if there is any advantage in using middle-of-the-phone pitch or full pitch information by generating the signal using these values. There were no audible differences between the two options, though full pitch information may prove to be helpful in domains where longer phones are uttered (like singing).

In conclusion, our model has the potential to extract most of the information required to reproduce the original signal and, given a (very) good synthesized voice, it can generate a speech signal very close to the original one.

Compression

Since there is very few information extracted from the speech signal, we can expect an excellent compression rate. Given the fact that there was no clear difference in quality when using middle-of-the-phone pitch and full pitch information, we experimented only with the middle-of-the-phone pitch. We used a very simple representation scheme of the extracted prosody though more complicated ones can be easily devised. For every phone, we represented the phone name on 7 bits (allowing up to 128 phones), duration on 14 bits (as milliseconds – up to 16 seconds long phones), and the pitch on 12 bits (as tenth of a unit – up to a pitch value of 400).

	Original		Prompt file	Synthesized	Compressed	% of MP3 original
	WAV	MP3	MP3	MP3		
1	256,044	24,408	12,312	24,408	168	0.69%
2	256,044	24,408	12,312	24,516	178	0.73%
3	320,044	30,348	12,312	30,456	178	0.59%
4	617,322	58,212	45,144	58,320	681	1.17%
5	256,044	24,408	12,312	24,516	173	0.71%
6	962,284	90,612	73,440	90,612	994	1.10%
7	1,277,164	120,096	107,028	120,204	1,628	1.36%

The above table summarizes our experiments with compression. For all our sound files we show the size (in bytes) of the original WAV, the size of the MP3 version of the original sound, the size of the MP3 version of the prompt file (notice half the size required when MP3-ing synthesized speech), the size of the MP3 version of the signal we generated (comparable size with original) and the compressed version. The reduction in size is very large, the size required being between 0.5% and 1.2% of the MP3 version of the original sound. If we can improve the quality of the generated signal even more, this technique/model can prove to be very efficient in disseminating existing speech on low-bandwidth connections.

Portability

We also wanted to see if the information we extracted is portable to other voices than the original one. We performed the same upper-bound experiment (the one in the “time” limited domain) using a different voice (Alan Black’s voice). Since the voices were relatively close, there was no clear need to transform the pitch values. The results show very good portability, the resulting sound being very natural. Of course, when porting to other voices with different pitch range (especially from male to female), there has to be a mechanism for transforming the values. The simplest one will be a linear mapping between the two speakers pitch ranges. Given the fact that my “time” limited domain female voice (my wife) was not recorded properly, I did not have the chance to test this transformation and its effect on the generated speech.

5. Conclusions

Our experiment indicates that by extracting only the duration and the pitch from natural speech and using a higher quality synthesized voice, the original speech can be reproduced with relative high quality and excellent compression rates can be achieved by representing only that information. Future work needs to address better techniques for phonetic labeling and pitch extraction, and the effect of the information that was left out (like power, shouting, whispering) on the reproduced signal.

Acknowledgements

I would like to thank Alan for his support and patience; also, for providing his voice in the time limited domain and for his stoicism when I was playing his voice using my interpretation in front of the class (though I never asked for permission).

Appendix

Here is the text we used in our experiments.

1. The time is now exactly five past one in the morning
2. The time is now exactly five past one in the morning
3. The time is now exactly five past one in the morning
4. An atom goes to a police station. It goes to the counter and says I have to report a theft. What was stolen? asks the policeman. I have just lost an electron. says the atom. Are you sure? Then the atom replies I'm positive.
5. The time is now exactly five past one in the morning

6. What can I get for you? Yeah, I would like a spicy chicken with garlic sauce. And then? And a chicken fried rice? And then? A coke. And then? That's all. And then? I said, that's all! And then? That's all! And then? And then, I'm gonna come in there and I'm gonna put my foot in your face if you say AND THEN again! And then, and then, and then, and then.
7. Help topic: Set the default font. Step one. If your document already contains text formatted with the properties you want to use, select that text. Step two. On the Format menu, click Font. Step three. Select the options you want to apply to the default font. If you selected text in step 1, the properties you want will appear in the dialog box. For Help on an option, click the question mark , and then click the option. Step four. Click Default. Any new document you open will use the font settings you selected.