ISSN: 2348 - 2117
International Journal of Engineering Technology and Computer Research (IJETCR)
Available Online at www.ijetcr.org
Volume 3; Issue 1; Page No. 87-92

A STUDY OF ARTIFICIAL INTELLIGENCE AS A REAL TIME HOST

Prakash Kawar Asawat[1], Girdhar Gopal Singh[2]

[1]Assistant Professor, Computer Science and Engineering, Jodhpur Institute of Engineering & Technology, Jodhpur, Rajasthan Technical University, Kota

[2]Computer Science and Engineering, Jodhpur Institute of Engineering & Technology, Jodhpur, Rajasthan Technical University, Kota

ABSTRACT

This paper deals with application of the electronic-dj which brings out a friendly relationship between the listener and the tuner. Off course, these both sequences of actions are done by a single individual. In this, the selection of the song is been accessed through speech recognition. The main purpose of implementing this model is, to favor and provide entertainment to the user. This paper presents a Novel approach to speech recognition. Currently, most speech recognition system is based on Hidden Markov models (HMMs), a statistical framework that supports both acoustic and temporal modeling. HMMs make a number of suboptimal modeling assumptions that limit their potential effectiveness. Neural Networks avoid many of these assumptions, while they can learn complex functions and generalize effectively. Thus this method is tested over a standard speech data base and the results are presented. In this paper we describe a speaker independent speech recognition system. The module performs recognition using microphone. This model establishes the environment where the user can interact with the system for his favorite song selection from the songs listed in the database by his oral communication. Speaker independent speech recognition is important for successful development of speech recognizers in most real world applications like an E-DJ. While speaker dependent speech recognizers have achieved close to 100% accuracy, the speaker independent speech recognition systems have poor accuracy not exceeding 75%. This model gave 85% of true results.

Index Terms: *DJ, Neural Network, Music retrievals, Speech recognition, Intelligent Systems and Approach.*

I. INTRODUCTION

Our cornerstone for this model is to favour the user in selection of a song on his choice. The one where the user can control the system. We are doing this to favor the user in listening to songs on his choice on the base of speech recognition. With this, user can have a friendly atmosphere where he can tune over to the song of his selection and enjoy the moment through his speech. This is how the model helps to the user in his song selection using speech recognition.

The Audience

The audience is the most important of all. They are the final judge of what you're doing. A party without people is no party. It is as simple as that. People can be at a party for a number of reasons; they are there because they are organizing the party.

- People can come for social contact.
- To drink & forget their problems.
- To dance and have a good time.
- These people are the ones which will be your judge.
- People can also be there because they came along with other people. These people aren't expecting anything.

So, a party is not that difficult at all. Nevertheless some DJ's have quite a strange picture of what's going on at parties. Playing music is not only using your intuition. You know what to play will be very small. If DJ plays 300 songs overnight, 80% will be based on ratio. Only 20% on 'feeling'. Especially in the beginning because you will be nervous and will need to fall back to your technical skills. For most DJ's the following holds true: you are playing for the audience.

II. EXISTING SYSTEM

"DJ is a person who plays the song on user's choice. This is the manual procedure which is been existing in the present world at all the parties". The Disadvantage In this system, the selection of a song is a three way operation where, the chain process is as, Time taken is more in this mechanism for searching the song and then making it to play. Not efficient for the present technical world. This is the block diagram for manual DJ.

III. DJ THEORY

A. Selecting Your Music

Whether you have CD's, vinyl or MP3's, have an index at hand, sorted by style, annotate with the BPM and marked with the 'sound-color'. This list should contain cross references between

*Corresponding author: Prakash Kawar Asawat

styles: 'switch to this style using this song'. On top of this style list, also have a full index by name available every time you play. Creating such a list takes a lot of time. You can easily spend months to create it, but when you have such a list it is your treasure. This will be half the money you make with DJ'ing.

Simple thing is whatever the above said is belongs to Manual DJ. Here is we avoid the months of time to create such a list as Dj treasure. The E –Dj system consisting of no. of songs in the database. In E- Dj, selecting the songs which the audience wants to listen is simply based on Speech recognition. In my system the selection of song can takes place based on some logic or fundamental identification for each and every song. When any audience spoke that word, the control will switch to that song then play. This is fundamental step in Electronic Dj.

A good strategy to play music for a specific audience is to rely on a number of prototype people you know that like the music that is typically played at a certain kind of party. Think: 'would this person like this music?' This works quite well!

B. Playing Different Kinds of Popular Music

When playing different kinds of popular music, the most important is to know what is popular with the audience. On top of this there are a number of rules.

- Play every song between 2.0 - 2.5 minutes. If you play songs longer people will find it boring. If you play songs too short people will become irritated. Of course, a mistake in the 'short' direction is not that bad.
- Minimum 4 songs of the same style in a row.
- Work your tempo down until you reach a suitable tempo for a slow.
- Always play two slows. After the first not everybody has the girl/the boy he/she wants. After a slow, kick in a beat again. No point in messing around with a 'good' build-up. Some (lonely) people are waiting to dance, and the people slowing will leave the floor anyway when you switch to a non-slow.
- In the beginning of the night choose your end style of music. After 3:00, 4:00 o'clock people go home when you switch style, so stay to the same style after that.

Switching from song A to B

- Can be done when A has the same connotation as B. E.g., Red Zebra after the Sisters of Mercy is quite possible in Belgium for people who likes Gothic.
- Can be done when A has the same 'color' as B.
- Can be done when A has the same 'tempo and style' as B.
- Is done with a cross fading over 5 to 10 seconds.

At every moment have a list of the three/four/five next songs you will play, this should ensure continuity. If people ask something, don't switch immediately, put them at the end of your list, and eventually adapt your list. Trusts people's opinion only when they are happy. Otherwise neglect them. Don't play killer music. Killer music is music where you lose a lot of people. For example. If you have 32 diagonal spread, seriously drunk people with 16 men and 16 women, at 6:00 'o clock in the morning do not play a slow. They will go home afterward. OK maybe that was the intention.

C. Tempo

This is second technique which we are incorporating in my system. It is also a very good idea to accurately measure the tempo of all songs (that is up to 1/100 of a BPM). Programs such as BpmDj, BpmCount or BpmLive can help with this. The tempo in general is necessary to

a) Match the tempo of the new song to the old song and

b) Set the tempo of various effect boxes exactly to the current playing tempo.

This above technique can be executed in E- Dj very easily it's just check tempo of the old and new songs adjust the timing between them, for that we have take help from predefined programs like BpmDj, BpmCount and BpmLive.

D. Finding Cool Music

Just ask producers for previews of music and songs that will come out. The sound quality, timbre, color and immediate recognizability belongs to the song, not to the DJ. The night belongs to the party organizers. This means that if the songs are not good or boring that it is not Manual DJ responsibility to fix it. DJ should select songs that are already good in the first place. How you weave them together in your set is on the other hand your task. Even 5 minutes of crappy songs can ruin your set, so be sure to use the best music.

So my system avoids this type of discrepancy for finding the cool music. First of all we place occasion related songs in our database and provide the index to the audience. Whatever the song want to listen by anyone just spoke that clearly. That's enough to E- Dj.

E. Flangers

Add space to stuff. Euhm, Well, yes (DJ Words) they add space in general.

- useful at the end of phrase before the next phrase starts
- Flangers are a form of auto filter, so the depth of the effect is dependent on the spectrum of the sound. Therefore adding a reverb to the sound before the flanger can help to have a deeper effect.
- Flangers don't work well on low frequencies (well they do their work, but it is difficult to use them properly at low frequencies).

F. Mixing Two Songs

When we go for mixing two songs, Manual Dj goes for the following steps. Meanwhile the E-Dj process all the tasks based on simple programs those are adjust all the options which needed in the function. Technically this is not difficult at all. However, this scheme should be remembered very accurately and practice

is necessary. Otherwise, one of the steps is easily looked over. In the E-Dj we overcome all the problems and economically save the money while we work with the E- Dj. Now, even if they are not accurately enough, it is possible you want to use things like AlsaPlayer (it is good software after all).

G. Beat Mixing

Now, something more difficult: Beat Mixing. Beat mixing is mixing two beats exactly over each other during a certain period. The difficulty with this is that different songs have different tempos. In the upcoming discussion we refer to song B as the one which will be mixed over song A. Synchronizing B with A is the first problem, keeping them synchronized is the second.
In general beat-mixing is only possible when the two songs are playing at the same speed. Therefore, one needs to bring the tempo of one of both songs to the tempo of the other song.

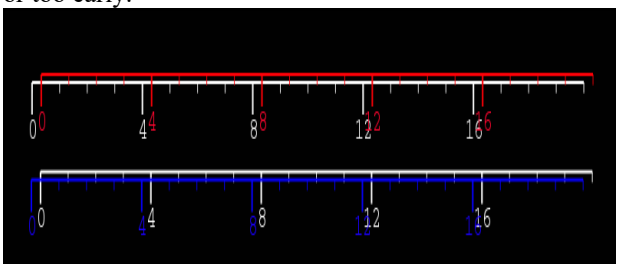
This however forms a problem because

- The tempo of most acoustic songs have is not perfectly constant.
- Depending on the technique used, the tempo can be measured slightly wrong.

Therefore, during playing one needs the ability to shift a playing song a bit forward or a bit backward, such that they stay synchronized. This is called nudging. A nudge typically consists of shifting the song 5 to 10 ms. this is around 1/64 note.

H. Syncing

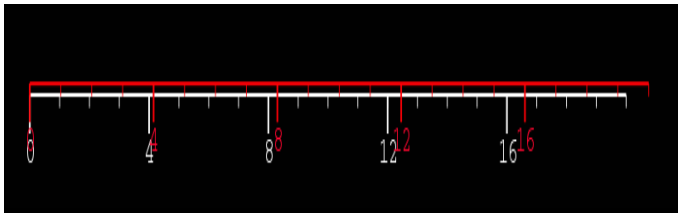
When a suitable song has been selected and it is playing at the correct tempo one needs to start the song at the correct moment. Typically this moment is at the beginning of a phrase (that is the beginning of 8 measures). Normally, when the song is started it won't start exactly at the moment you intended it. Therefore, you will need to nudge a little bit. This however is not easy because it is difficult to decide whether the song you threw in started too late or too early. For instance, in the figure below, the white line is the time-line of the main song. The red line is the monitor song which has been started too late. The blue line is the same monitor song but started too early. As can be seen, if we only listen to the beats, it is impossible to distinguish whether the song is too late or too early.



Nevertheless, we do not necessarily need to listen only to the bass-drums, we can also listen to the entire song. Another pragmatic way to solve this problem is to nudge forward, if the problem becomes worse, nudge two times backward.

I. Nudging

During the time the two songs overlap the tempo difference between the two songs (even if it is a very small tempo difference) will result in a slight synchronization drift. This is pictured in the figure below



To solve this one needs to know beforehand which song is the slowest one of both. before a mix is done. Solve this problem is easy. Make sure both song are synchronized, now wait until the two beats sound double. Nudge forward. If it becomes better, you should keep on nudging forward since the second song is going a bit too slow. The direction determined by this technique is the direction you need to use to keep them synchronized once they have been synchronized.

J. Cross fading

When you finally have the two beats exactly over each other in your headphones you want to switch slowly to song B. Before you do this be sure to cut off the bass drum with the equalizer.

K. Take Your Time

Most songs are in a 4/4 rhythm and it is in general a good idea to respect this pattern: multiples of 4. 4 beats in one measure & 4 (or 8) measures in a sequence. If you respect this you will find that you get easily into the flow of mixing. Of course, this requires some practice, but after a while you will actually start using this scheme. Each 4 measures you can change something like cutting away the bass-drum of one song in favor of the other or using the 4 beats/4 measures knowledge to add breaks and gaps in the music at appropriate places. Such breaks will also ensure that the audience does not loose track of the underlying synchronization.

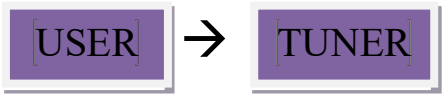
L. Some of the other techniques performed by DJ are:

- Breaking
- Sound Effects
- Reverbs
- Pan/Bouncer Effects
- Voice Samples
- Delay Effect
- Keep them dancing but offer bar breaks.

IV. EASE OF USE

As per the present generation we have to change, make and use things technically for as per our convince. So, we have implemented the electronic DJ where elimination is done to the

"DJ" and direct interaction is made with the system itself. This has been done through oral communication. The Advantage of E-dj is the system where the interaction is only between the user and the tuner.



User can just select the song within few seconds from the database. Relaxation is even provided without any clumsy environment. It seems to be an efficient one for present generation.

A. Problem statement

The system automatically classifies the songs based on drums played in that particular song and administrator's choice. The Neural network classifies the entire songs database to the different categories of moods. Then extract the features of songs then select the appropriate song from database which is requested by the audience, then play the song which is represented in the Figure1.

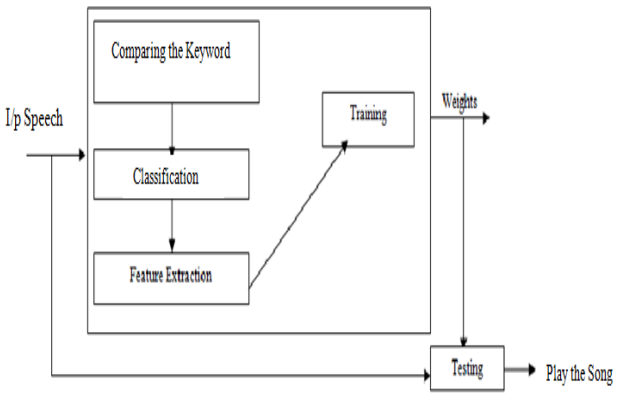


Figure 1. Block Diagram of identification System

In users point of view the module is to select the song from the database by giving the keyword to the system and then for playing that particular song he has to utter the play action.

B. ProcDJ Actions

First thing is playing the song which is requested by the audience. Then next task is to apply the different techniques based on audience response.

Once song is playing mode then the machine is in Idle for two minutes. Machine again take the input from the audience after immediate completion of 2 minutes, Now the input is reach the critical frequency then the system increase the Tempo, it makes enjoy the audience.

Again completion of another 2 mins, the machine will collect audience response, again it reaches the significant frequency - The system goes for mixing the different songs from the same category.

Maximum limit of song duration is four minutes. So, after completion of next 2 mins the machine will collect audience response, it still exceeds the significant frequency - The system switches some other song from the same category. To synchronize the melody we have to apply the nudging and synch to particular actions. Add space to stuff. Euhm, Well, yes (DJ Words) they add space in general.

Likewise the E-Dj is very useful in functions with low cost compared to manual Dj.

TABLE I. LIST OF CATEGORIES IN DATABASE

S.No	Category Keyword	Melody Movie Name	Hard Rock & Energetic Movie Name
1	Birthday	Darling	Shivagi
2	Marriage	Murari	Mr. Perfect
3	Love	Orange	Arya -2
4	College	Happy Days	Happy Days
5	Happy	Jalsa	Super
6	Romance	Titanic	Titanic

C. Speech Recognition

There are many ways to design a classification network for speech recognition. Designs vary along five primary dimensions: network architecture, input representation, speech models, training procedure, and testing procedure. In the experiments described so far, all of the time delays were located between the input window and the hidden layer. However, this is not the only possible configuration of time delays in an MLP. Time delays can also be distributed hierarchically, as in a Time Delay Neural Network. A hierarchical arrangement of time delays allows the network to form a corresponding hierarchy of feature detectors, with more abstract feature detectors at higher layers (Waibel et al, 1989); this allows the network to develop a more compact representation of speech (Lang 1989). The TDNN has achieved such renowned success at phoneme recognition that it is now often assumed that hierarchical delays are necessary for optimal performance. We performed an experiment to test whether this assumption is valid for continuous speech recognition.

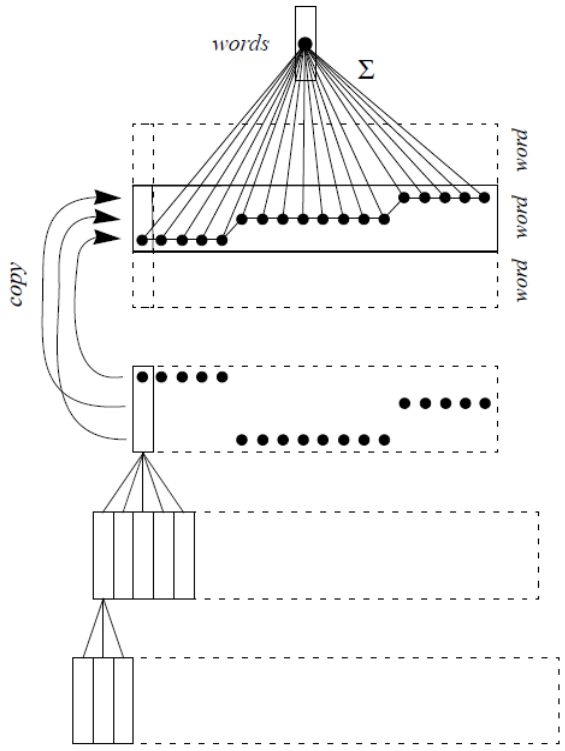


Figure 2. Time Delay Neural Network for Speech Recognition

A simple extension to this is to allow non-uniform sampling:

$$\bar{x}_i(t) = x(t-\omega_i)$$

Where ω_i is the integer delay associated with component *i*. Thus if there are n input units, the memory is not limited simply the previous n timestamps. Another extension that deals is for each "input" to really be a convolution of the original input sequence.

$$\bar{x}_i(t) = \sum_{\tau=1}^{t} c_i(t-\tau)x(\tau)$$

In the case of the delay line memories:

$$c_i(t) = \begin{cases} 1 & \text{if } t=\omega_i \\ 0 & \text{otherwise} \end{cases}$$

V. RESULTS

In our system initially speaking the some key words those are stored in database. Here we can use the mobile as a machine to perceive input from the audience. After executing the task immediately it displays the speech recognition screen.

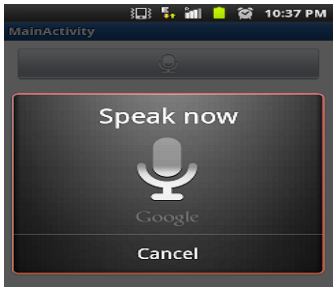


Figure 3. Speech Recognition

Here is the first command

$rec -t alsa default ./dsr.flac trim 0 2

This command will records a flac format file for 2 seconds from the inbuilt microphone. Our system will work based on some commands uttered by the audience.

First playing the song at that time for switching the song user has to utter command. Then the system will wait for any category name utter by audience. The system records the voice from the audience then sends it to the Google server. This can be done by using the python Script is given below.

```
#!/usr/bin/env python2
# -*- coding: utf-8 -*-
import httplib
import json
import sys
def speech_to_text(audio):
    url = "www.google.com"
    path = "/speech-
api/v1/recognize?xjerr=1&client=chromium&lang=e
n&maxresults=10"
    headers = { "Content-type": "audio/x-flac;
rate=48000" };
    params = {"xjerr": "1", "client":
"chromium"}
    conn = httplib.HTTPSConnection(url)
    conn.request("POST", path, audio, headers)
    response = conn.getresponse()
    data = response.read()
    jsdata = json.loads(data)
    return data
#    return
jsdata["hypotheses"][0]["utterance"]
if __name__ == "__main__":
    if len(sys.argv) != 2 or "--help" in
sys.argv:
        print "Usage: stt.py <flac-audio-file>"
        sys.exit(-1)
    else:
        with open(sys.argv[1], "r") as f:
            speech = f.read()
        text = speech_to_text(speech)
        print text
```

Here second command to run python script

$python stty.py ./dsr.flac

This command will post the recorded file to the Google speech recognizer and get the nearby matched utterances in the form of text. So, first we have to record the audio file and utter the word.

I have uttered the word color and the Google speech recognizer will return a set of matched words after we post the sound file in the FLAC (free Lossless Audio CODAC) format. Here the first word uttered was '**color',** the result can show below.

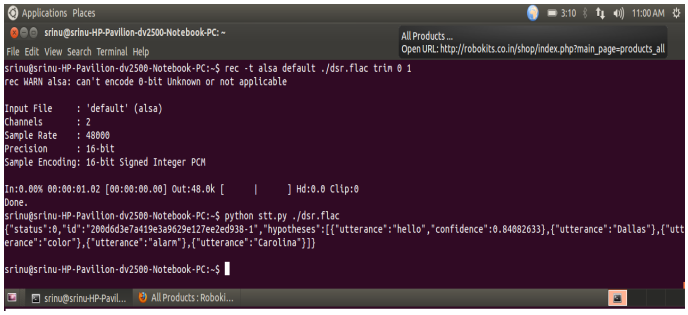


Figure 4. Return a set of matched words

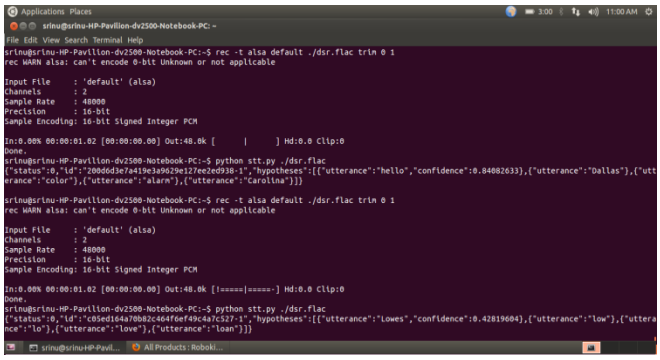


Figure 5. Return a set of matched words for the word LOVE

The returned words are compared with the words in the Songs Database arranged in category wise and the event was trigged to DJ means randomly select and plays the song. Then after receiving the some keywords as input it checks then extract the features finally play the song as part of e-dj as shown in the figure 4.

Figure 5. Playing Song

VI. CONCLUSION

We conclude by giving assurance that the user feel comfort with our model and enjoy the environment by listening to his favorite song. In addition to this he can add the songs to the database and perform the play action as per his choice. This task provides us a user friendly environment. Selection is made easy by this module and addition is also provided. Finally, NN-HMM hybrids offer several theoretical advantages over standard HMM speech recognizers. Specifically: Modeling accuracy, Context sensitivity, Discrimination, and Economy.

VII. FUTURE SCOPE

We extend the scope of this paper to when the requested song is not available in the databases then connect to the web, play the song which is requested by user. We have to enhance the composition of song based on user's mood. Suppose the environment is party then add some rock beats to the actual song track and mixing the two different songs with some DJ words. Likewise we have to train the Neural Network for composition also.

REFERENCES

[1] http://bpmdj.yellowcouch.org/djskills.html

[2] Austin, S., Zavaliagkos, G., Makhoul, J., and Schwartz, R. (1992). Speech Recognition Using Segmental Neural Nets. In Proc. IEEE International Conference on Acoustics,Speech, and Signal Processing, 1992.

[3] Bodenhausen, U., and Manke, S. (1993). Connectionist Architectural Learning for High Performance Character and Speech Recognition. In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993.

[4] D. Yuk, C. Che, L. Jin, and Q. Lin. Environment-independent continuous speech recognition using neural networks and hidden Markov models. IEEE International Conference on Acoustics, Speech, and Signal Processing, 6:3358–3361, May 1996.

[5] D. Yuk, C. Che, P. Raghavan, S. Chennoukh, and J. Flanagan. N-best breadth search for large vocabulary continuous speech recognition using a long span language model. 136th meeting of Acoustical Society of America, October 1998. N. Morgan and H. Bourlard. Neural networks for statistical recognition of continuous speech. Proceedings of the IEEE, 83(5):742–770, May 1995.

[6] Bodenhausen, U. (1994). Automatic Structuring of Neural Networks for Spatio-Temporal Real-World Applications. PhD Thesis, University of Karlsruhe, Germany.

[7] Bourlard, H., Morgan, N., Wooters, C., and Renals, S. (1992). CDNN: A Context Dependent Neural Network for Continuous Speech Recognition. In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, 1992.

[8] Franzini, M., Witbrock, M., and Lee, K.F. (1989). Speaker-Independent Recognition of Connected Utterances using Recurrent and Non-Recurrent Neural Networks. In Proc. International Joint Conference on Neural Networks, 1989.

[9] Hild, H. and Waibel, A. (1993). Connected Letter Recognition with a Multi-State Time Delay Neural Network. In Advances in Neural Information Processing Systems 5,

[10] Hanson, S., Cowan, J., and Giles, C.L. (eds), Margan Kaufmann Publishers.

[11] Jacobs, R., Jordan, M., Nowlan, S., and Hinton, G. (1991). Adaptive Mixtures of
Local Experts. Neural Computation 3(1), 79-87.

[12] Lee, K.F. (1988). Large Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System. PhD Thesis, Carnegie Mellon University.

[13] C. Chen and R. Miikkulainen. Creating Melodies with Evolving Recurrent Neural Networks. In Proceedings of the Internacional Joint Conference on Neural Networks, IJCNN 01. p.2241-2246, Washington, DC 2001.

J. A. Franklin. Recurrent Neural Networks for Musical Pitch Memory and Classi_cation. Journal on Computing, Vol. 18, No. 3, Summer 2006, pp. 321-338.