BOHR

BOHR International Journal of Smart Computing
and Information Technology
2023, Vol. 4, No. 1, pp. 1–6
DOI: 10.54646/bijscit.2023.31
www.bohrpub.com

METHODS

Text to song synchronization using deep neural network

Ayub Shaikh, Jay Dedhia, Mayank Gohil, Shreyas Tejam and Pratik Kanani

Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India

*__Correspondence:__
Ayub Shaikh,
ayubshaikh0989@gmail.com
Jay Dedhia,
2501.jay@gmail.com
Mayank Gohil,
mayankgohil999@gmail.com
Shreyas Tejam,
123shreyastejam@gmail.com
Pratik Kanani,
Pratikkanani123@gmail.com

Received: 12 January 2023; **Accepted:** 28 January 2023; **Published:** 15 February 2023

The most crucial aspect of the learning process is memory. There is no knowledge without memory. However, most people lack memory because they memorize things through rote memorization. So in this paper, we implemented a deep neural network (DNN) model that converts plain text into a song, which is done by synchronization of a Musical Instrument Digital Surface (MIDI) file or the music with the text. MIDI files are symbolic representations of the music score. The melodic line of the composition is normally on one of them, while the background music is on the others. Analyzing the audio to identify voice portions where a human voice is present, as well as non-voice segments in proximity to the voice segments, is part of a method for syncing text with audio. The text segments of segmented text related to the audio can be recognized and synchronized to the speech segments.

Keywords: synchronization, DNN, musical memory, NLP, ML, Alzheimer, MIDI files

Introduction

This bestows a solution and executes programming for the following obstacle: consequently synchronizing text to a fitting tune. The customized AI model synchronizes the text with the verses of the tune. The beats and recurrence are matched to the original form of the song. Thus, singing the text with this ambient sound makes it simple for the learner.

Music has its own separate space in our memory, hence an average person tends to remember musical lyrics faster than any textbook texts. Music is mostly in patterns; they have beats. Our product fills the data with beats that are melodic examples. Large lumps of drilling data are broken into melodic beats and given to the client so he appreciates learning. Learning is made fun using Sur-Smaran.

Music is also found to be helpful to people performing their best in high-pressure situations. We always wonder why it is so simple to remember song lyrics, unlike academic content that our brains tend to forget so easily. The reason is that our brain searches for patterns to understand the lines and recall and process the information. We have experienced that sometimes a part of the song gets stuck in our head for an extended period of time and we cannot get it out of our mind. It just so happens, it is just a way of improving the brain's memory, which is called "Earworm." So, we now understand that music and memory have a very strong connection in our brain and that music can be very beneficial to study.

Sur-Smaran will analyze the lyrics and the digital score to produce a singing voice. You can keep and use the tune as if it was your own if you like it. If you do not like it, try again as many times as necessary until you find your favorite produced track. All these functionalities make our application your partner for remembering (Smaran) musically (Sur).

The most crucial aspect of the learning process is memory. There is no knowledge without memory. However, most people lack memory because they memorize things through rote memorization. Most students memorize by repeating a fact or a quote until it is temporarily burned in. Although people tend to forget this information as soon as they no longer require it and that this type of memory rarely leads to mastery of the content memorized, here comes our Sur-Smaran. With the help of it, you can memorize quickly and barely forget anything. People tend to remember music for decades, and if you add information to music, you can create musical memories of your information. In today's world, it will be used by:

- Students who need to remember lots of facts and data.
- For the preparation of speech.
- Can be used by Alzheimer's patients because the parts of the brain that stores musical memory (anterior cingulate and ventral pre-supplementary motor areas) are unaffected in the case of Alzheimer's.
- Any individual who wants to remember a series of names or a list. Many more applications can be listed here. This product is extremely useful for society today dealing with memory issues.

This project is an attempt to make memorizing easy and fun. All you have to do is take the piece of information you want to remember and give it to our application, and then you will be able to generate a melody similar to the song you would love to sing with the paragraph. Attempts have been made in many applications to synchronize users' voices to the pitch and amplitude of songs or karaoke. But, the motive of our project is to create musical memories of anything a person wants to remember.

The project deals with several types of data inputs. Input can be a scanned image of text or a typed word. Each type of input file will be processed. We tried to translate the text into lyrics that the machine could sing. In addition, we must overcome a few obstacles, such as data inefficiency, because it is difficult to find or create a perfect dataset that can be directly converted into a song. It is also difficult for the machine to follow the performance pitch and rhythm synthesis and add emotions to the song using singing techniques, which has a direct impact on perceived quality (1). We attempted to improve the naturalness and quality of the singing voice synthesis (SVS) based on the Hidden Markov Model (HMM), which was recently proposed by implementing a singing voices synthesis system based on deep neural network (DNN), which is similar to text-to-speech based on DNN. Pitch normalization, trajectory models, and durational models are among the techniques introduced to help improve the efficiency of the SVS system (2). As we know, the SVS system is used to convert text into a song that will be sung by the machine. A rule-based model can be used to focus on duration modeling. We can also use system designs like the length regulator, acoustic model, and audio synthesizer to take advantage of the phoneme duration, and with this model, we:

Research gap

The difficulties which can be faced by the model are data inefficiency because it is difficult to find or create a perfect dataset that can be directly converted into a song. It is also difficult for the machine to follow the performance pitch and rhythm synthesis and add emotions to the song using singing techniques, which has a direct impact on perceived quality. Text-to-speech based on DNN, Pitch normalization, trajectory models, and durational models can also be introduced to help improve the efficiency of the system. System designs like the Length regulator, Acoustic model, and Audio synthesizer can also be implemented to take advantage of the phoneme duration to achieve a well utilized and efficient system even with a small dataset.

Objectives

To remember anything, be it a paragraph or a whole chapter, we have to revise it a lot of times, which sometimes becomes boring, Sur-Smaran can make it much easier and more interesting.

Differently. For a scanned image the first step would be to recognize the texts from the image and then allow the system to read it. Typed text input does not need any scanning process. Generating the memory appropriate to the user 's text input is the main job here. Depending on the letters or words, the melody will be generated. This will make it sound more like the user is required to tune and hence the user will better remember the text.

Literature review

Can achieve a well-utilized and efficient SVS system even with a small dataset (3).

As we all know, the text is converted into lyrics and sung by the computer from musical scores using the SVS system based on HMM, but because the voice is easily distinguishable from the natural singing voice, we can also implement the SVS using DNN, and DNN-based systems provide more utilized and efficient results when compared to HMM-based systems. It also outperforms the HMM-based system in subjective listening (4). The PeriodNet system, which converts/models periodic and aperiodic waveforms into speech waveforms, can also be implemented. It can be used to generate natural sounds and reproduce an accurate pitch in general. It can also be used in SVS systems' Vocoder for text-to-song conversion. The generator in the PeriodNet system has two subgenerators, and the output waveform is essentially the sum of the subgenerators' outputs (5). We can also implement a system that extracts training data from a social media app called "nana," which can then be used on many-to-one singing VC (voice conversion) (6).

- Some people find it difficult to remember things on their own, and a few of them have also been through memory loss and confusion. This is very important at times, like in an exam hall or right before giving a speech. So at a time like this, Sur-Smaran can be very useful.
- Sur-Smaran can be very useful for artists also. If an artist does not like the song or a small part of the lyrics, then he/she can just change the lyrics and the new lyrics will be attached to the Musical Instrument Digital Surface (MIDI) file and a whole new song can be generated.
- Sur-Smaran could be used for entertainment purposes too, like to make songs personalized like people could insert their names in the song lyrics.
- A lyric could be tried and tested by attaching it with various MIDI files to get the best-combined output of the song.
- The genre of the song can be changed without changing the lyrics by just replacing the MIDI file.
- The Sur-Smaran output may be recorded and listened to numerous times to help you enjoyably recall information because it is a song with beats and tunes.

Methodology

In this paper, we have referred to the papers based on SVS, PeriodNet, and DNN. Our purpose was to design a model which can convert any text into a song, which is sung by the machine and which is to be non-distinguishable from the natural singing voice.

The main purpose of this project is to make memorizing easy and fun. It just takes the piece of information the user wants to remember and then generates the melody similar to the song any user would love to sing with the paragraph. Attempts have been made in many applications to synchronize users' voices to the pitch and amplitude of songs or karaoke. But, the motive of our project is to create musical memories of anything a person wants to remember.

In the process, the user gives a text file and our database will give a MIDI file (the user can also provide their desired MIDI file). After the input, an audio is generated using a Python library via SVS. The music separates the background music. When all this process is done, the synchronization will happen, which will give the final output of the song where the user can hear the text file in the form of the song. **Figure 1** shows the proposed design for the same.

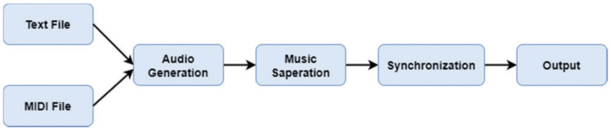

FIGURE 1 | Proposed design.

Implementation

The Sur-Smaran project is implemented in three steps. The first step requires the user to give input that he wants to be converted into music and an audio file is generated of that input. Second, the selection of music genre or the song to which he would like his data to sound. Finally, the audio is mapped to the song with the help of a MIDI file and the final output is given to the user. Given the steps separated, each step is conquered with different pieces of machine learning technology.

When we give the text input, we use ESPnet (7) for text-to-speech generation.

ESPnet is a speech processing toolkit. It has been implemented based on a Conformer model, which is a deep learning engine. It has the capability of generating audio that will be suitable for further machine learning algorithms. For mapping the audio to the specified music genre or the specified song, we use a non-autoregressive neural network with a MIDI file. Using this MIDI file, the phoneme durations are decided. Rhymes may not match, but the frequency of words will make it musical. Training of this model requires the original songs to be separated from their music and their musical scores. So, we will just take the vocals without instrumental music. We then generate audio from the lyrics of the song and feed this generated audio as input and the vocals from the songs plus the MIDI file (shown in **Figure 2**) to further processing. This will map machine-generated audio to an actual human-like voice with all the pauses in between. For the separation of vocals and music from songs, we would use Spleeter (8). Spleeter works on Deezer (9), which is a singing voice separation model.

After training the model, we can give any text input to the model and select the desired song and our model will make it musical.

Input

Text File:

Standing like a mountain,
This is more than the limit,
56 inch chest,
He has come to tear the earth,
If you speak loudly, he will rip you apart, This is sultan
Don't believe it, don't believe in fire,

Let it go,
Look, God has sent,
He has come to take revenge,

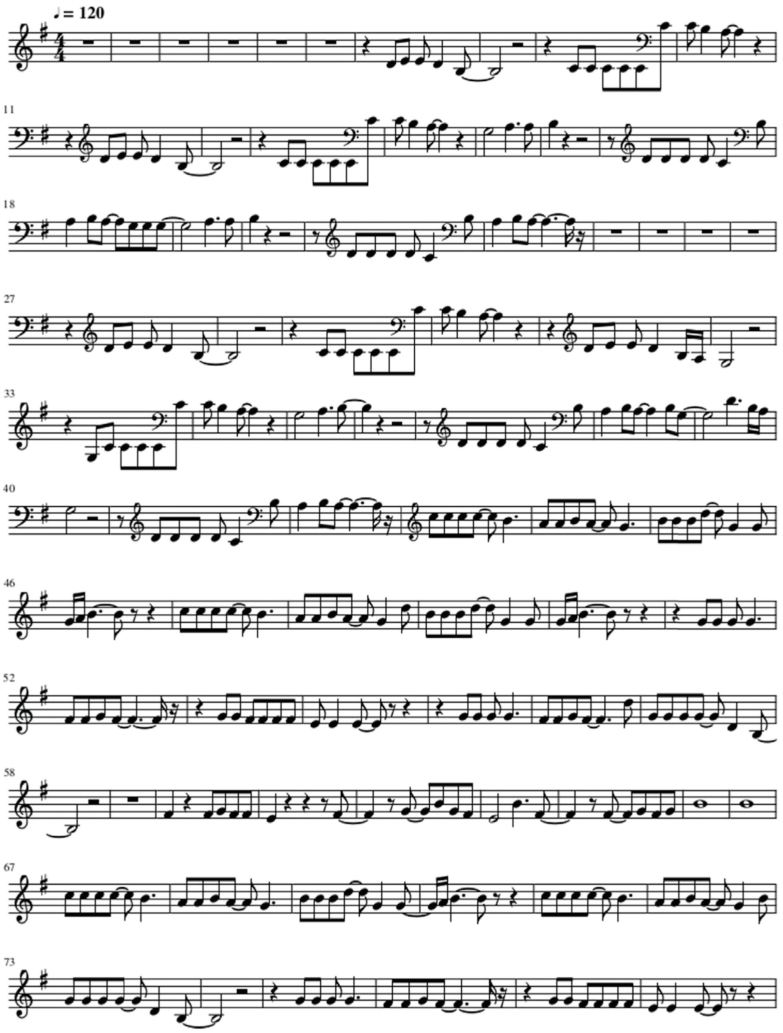

FIGURE 2 | MIDI file from the collection (10).

Stay away and be careful
This is sultan

Don't forget to do this
Give me anything, still don't be afraid
Do it or not, the world doesn't care
My prayer will always be with you

Look look look, the sultan is strong
Look look look, the sultan is so fast
Look look look, the sultan is mighty
Look look look sultan is fantastic

Output

First, the model will take user input in the form of text, as it has data chords in the MIDI file. The input text is then attached to the MIDI file, and the output is generated in audio format.

For the time being, it will choose a random MIDI file from the collection. Every section of the text is transformed into an audio file, which the user (student) can listen to as many times as he or she wishes. https://drive.google.com/file/d/1xJcPd4gFseEHeE5CWGF0JIrJXfLNwPcW/view?usp=sharing.

Result

We conducted a survey to see the need for our application. We asked the users to answer a few questions from which we can analyze our solution. Based on the questions, we can conclude that the major audience of our project is students. We can also notice that music is a very common interest for most people who face problems in remembering large texts. Since we got a good response for converting plain text into musical form, we are sure our project will help the masses. **Figure 3** shows the survey results.

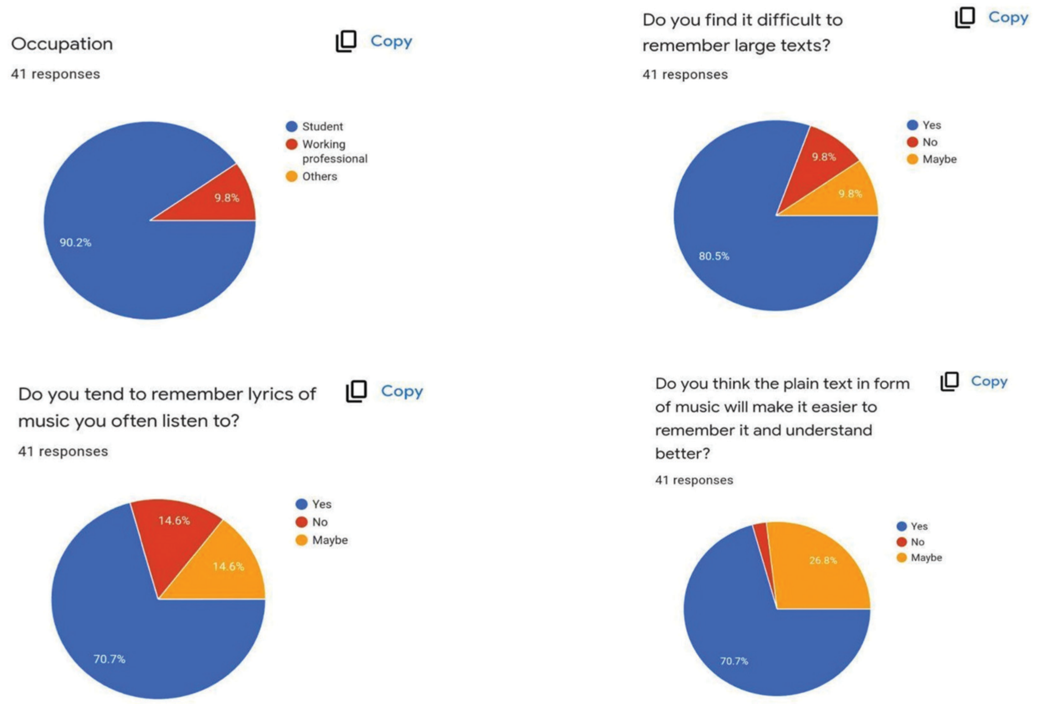

FIGURE 3 | Survey charts.

Conclusion

We have designed a solution for the difficulty in memorization by automatically generating melody tunes. In our solution, the user will have to log in first, then the user will have to upload the text file, and our application will give them the final generated melody. At the backend, our solution will use the midi2voice module to convert the given input text file into a processable audio file; it synchronizes the input with the MIDI file and a melody generated in this operation. The rhythms and beats of the output file are according to the MIDI file. Currently, we can generate songs based on the melody of the MIDI file. After successful training and testing of this, we will scale our solution to all major input types. On completion, Sur- Smaran will help people follow up with interest properly.

Future scope

We have successfully implemented the synchronization of MIDI files with the text. In the future, we can try to visualize things to make the learning and understanding of the text interesting and more efficient. This can be achieved by using the concept of an open CV in a project. As of now, a user can utilize only one MIDI file and one text entry once. But in the future, we can give the user the liberty to customize (add) more MIDI files to make the output interesting in a way they want.

References

1. Xue H, Yang S, Lei Y, Xie L, Li X. "Learn2Sing: target speaker singing voice synthesis by learning from a singing teacher," in *Proceedings of the 2021 IEEE spoken language technology workshop (SLT)*. Shenzhen: (2021). p. 522–529. doi: 10.1109/SLT48900.2021.9383585
2. Hono Y, Mase A, Yamada T, Muto S, Nankaku Y, Tokuda K. "Recent development of the DNN-based singing voice synthesis system–sinsy," in *Proceedings of the 2018 Asia-Pacific signal and information processing association annual summit and conference (APSIPA ASC)*. Honolulu, HI: (2018). p. 1003–1009. doi: 10.23919/APSIPA.2018.8659797
3. Briot JP, Pachet F. Deep learning for music generation: challenges and directions. *Neural Comput Appl.* (2020) 32:981–993. doi: 10.1007/s00521-018-3813-6
4. Hono Y, Hashimoto K, Oura K, Nankaku Y, Tokuda K. Sinsy: a deep neural network-based singing voice synthesis system. *IEEE/ACM Trans Audio Speech Lang Proc.* (2021) 29:2803–2815.
5. Hono Y, Takaki S, Hashimoto K, Oura K, Nankaku Y, Tokuda K. PeriodNet: a non-autoregressive raw waveform generative model with a structure separating periodic and aperiodic components. *IEEE Access.* (2021) 9, 137599–137612. doi: 10.1109/ACCESS.2021.3118033
6. Sendaetal K. "Singing voice conversion using posted waveform data on music social media," in *Proceedings of the 2018 Asia- Pacific signal and information processing association annual summit and conference (APSIPAASC)*. Honolulu, HI: (2018). p. 1913–1917.
7. Guoetal P. "Recent developments on espnet toolkit boosted by conformer," in *Proceedings of the ICASSP 2021-2021 IEEE international conference on acoustics, speech, and signal processing (ICASSP)*. Toronto, ON: (2021). p. 5874–5878.
8. Hennequin R, Khlif A, Voituret F, Moussallam M. Spleeter: a fast and efficient music source separation tool with pre-trained models. *J Open Source Softw.* (2020) 5:2154. doi: 10.21105/joss.02154
9. Pretet L, Hennequin R, Royo-Letelier J, Vaglio A. "Singing voice separation: a study on training data," in *Proceedings of the ICASSP 2019-2019 IEEE international conference on acoustics, speech, and signal processing (ICASSP)*. Brighton: (2019). p. 506–510.

10. Available online at: https://musescore.com/dashboard
11. Watanabe S, Hori T, Karita S, Hayashi T, Nishitoba J, Unno Y, et al. Espnet: end-to-end speech processing toolkit. *arXiv* [preprint]. (2018): arXiv:1804.00015 doi: 10.21437/Interspeech.2018-1456
12. Rizo D, Ponce de León P, Pérez-Sancho C, Pertusa A, Iñesta J. "A pattern recognition approach for melody track selection in MIDI files," in *Proceedings of the 7th international society for music information retrieval conference*. Victoria: (2006). p. 61–66.
13. Kenmochi H, Ohshita H. VOCALOID- commercial singing synthesizer based on sample concatenation. *Interspeech.* (2007) 2007:4009–4010.
14. Yang F-R, Cho Y-P, Yang Y-H, Wu D-Y, Wu S-H, Liu Y-W. "Mandarin singing voice synthesis with a phonology-based duration model," in *Proceedings of the 2021 Asia-Pacific signal and information processing association annual summit and conference (APSIPA ASC)*. Tokyo: (2021). p. 1975–1981.