Text-to-Speech Synthesis in European Portuguese using Deep Learning

Ana Catarina Gonçalves

Instituto Superior Técnico, Lisboa, Portugal
INESC-ID Lisboa, Portugal
a.catarina.goncalves@tecnico.ulisboa.pt

November 2018

Abstract
This thesis has one main goal: to synthesise speech from text for European Portuguese, using recent deep learning techniques. The motivation was to use this work on one hand as a first step for building a much needed child's voice, and on the other hand as a framework to synthesise expressive speech. These are two limitations that are faced by synthesizers in many areas of application, in particular, in the interaction with robots and serious games. This topic involves many areas of study such as machine learning, speech synthesis, linguistics and speech acquisition. This article reports the work done, the framework (Merlin) used to synthesise speech and the final conclusions of this project. The choice of Merlin was guided by the target application, the previous experience of the team, and the available information about each method, within the range of speech synthesis methods involving deep learning. The work done with Merlin will enable the synthesis of a child's voice depending mainly on the recordings of a child.
Keywords: Speech Synthesis, Deep Learning, European Portuguese Language

1. Introduction

Text to Speech systems have been in development for a long time with many different applications. Usually, the voices used or synthesised are from adults or when a child's voice is synthesised, it is usually very robotic, inexpressive and does not sound genuine because of the difficulties faced when recording a child and when developing a system to synthesise voice that sounds natural.

There are many applications for TTS systems with a child's voice, the most pressing one as Augmentative and Alternative Communications (AAC) device. The AssistiveWare project [1] is an example of this type of application. Serious games are also an application area for TTS systems with a child's voice. The ECIRCUS[17, 16] project was an example of this type of application. The project had a goal the development of a serious game to help children aged 9 to 12 with bullying problems.

The third type of application can be found in human-robot interaction. Currently, in Portugal, there are several systems under development or already in use, that involve the interaction of robots with children. Monarch[2] is a friendly robot used at IPO to interact with children that are there for treatment. Another national project is INSIDE[3] that involves the interaction with children with Autism Spectrum Disorder.

The original purpose of this thesis was then to develop a child's voice that could be applied for instance in the robots used for the Monarch and the INSIDE projects to have interaction with the children with a child's voice. This may increase the help given to these children reducing the apprehension felt by them, since the communications could be done through a voice from an equal.

When talking about speech synthesis, two different characteristics have to be evaluated, intelligibility and naturalness. The first one is the ability of being understood and describes the clarity of the synthesized speech. The second one assesses information that is not included in intelligibility, such as the listen easiness, stylistic consistency and nuances level. [15]

With today's TTS systems, intelligibility is no longer a major issue. However, in order to evaluate naturalness, subjective methods are typically required. Mean Opinion Score (MOS) is commonly used for audio, video or audiovisual quality and is expressed as a number between 1 to 5, 1 being the lowest quality and 5 the highest. See Table 1

[1] http://www.assistiveware.com

[2] http://monarch-fp7.eu/

[3] http://www.project-inside.pt/

for the labels used in this subjective test.

Table 1: MOS rating scale

Rating	Label
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

The overall MOS score is calculated by the following expression:

$$MOS = \frac{\sum_{n=0}^{N} R_n}{N}. \quad (1)$$

where N is the number of subjects that participate in the evaluation and R_n the rating given by each of them.

2. Background

The first generation of Text-to-Speech (TTS) systems was characterised by very robotic voices produced by rule-based systems. Until a few years ago, most commercial TTS systems were based on concatenative speech synthesis, which can be considered the second generation. The main idea behind this method, as the name implies, was to concatenate pieces of recorded speech of variable length to have the desired output. Statistical parametric speech synthesis was proposed as an alternative to concatenative synthesis in order to facilitate the manipulation of speech parameters for greater expressiveness. This type of method is based in Hidden Markov Models (HMMs) [8]. A parametric representation such as spectral info and excitation is extracted and then modelled with the HMMs. The most recent trend in TTS systems is based on Deep Neural Networks (DNNs). A DNN is an Artificial Neural Network composed of multiple hidden layers that can model complex non-linear relationships and generate compositional models. This architecture can have many variants. For speech synthesis applications they are usually feedforward networks with Recurrent Neural Networks (RNNs). The use of DNNs in TTS systems started in the waveform generation module, but has pervasively replaced most modules, originating end-to-end systems.

The main goal of this thesis is to develop a 4th generation TTS system for European Portuguese, replacing the in-house system known as DIXI, based on concatenative synthesis.

2.1. Speech Synthesis

After reviewing all the existing methods for speech synthesis, we chose for this project the Merlin framework. The main reasons for this choice were the fact that it was the only public domain[4] model at the time that the choice was made, and it could be trained for multiple languages. Moreover, the team had previous experience with the public domain Festival framework on which it is based.[14]

2.2. European Portuguese Phonology

European Portuguese is language composed phonetically by 37 phonemes, including the 14 vowels, 9 diphthongs and 23 consonants. We used the SAMPA (Speech Assessment Methods of Alphabet) script to understand the articulation classifiers pre-established.

This phonemes can be classified from the speech articulation system by manner which can be labial, coronal, alveolar, dental, velar and palatal/dorsal. Also, by place that can be nasal, lateral and trill that can also be considered liquid, plosive and fricative that can be voiced and unvoiced. Also by place, we can have semi-vowels that are defined as *phonetically similar to a vowel sound but with function as the syllable boundary*. It can be observed in the figure 1 the several organs that form the speech system and some of the correspondent manner and place classifiers for consonants.

Figure 1: Speech Organs

As consonants, vowels can as well be classified from the tongue position. The classifiers are then, regarding the place of the mouth, front, central and back, the opening of the mouth, close, close-mid, mid, open-mid, and open, and regarding the produced sound, oral and nasal. It can be seen in figure 2 we have a vowel quadrangle and then in figure 3 this quadrangle placed in the mouth.[5]

[4] https://github.com/CSTR-Edinburgh/merlin

[5] http://www.aston.ac.uk/lss/research/lss-research/ccisc/discourse-and-culture/west-midlands-english-speech-and-society/sounds-of-english/sound-production/

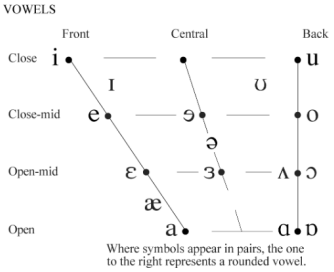

Figure 2: Vowel Quadrangle placed in the Mouthfootnote

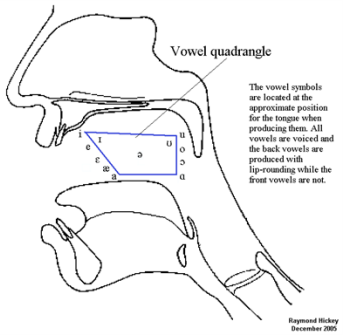

Figure 3: Vowel Quadrangle placed in the Mouth

3. Merlin

Merlin [18] is a neural network-based speech synthesis system based in statistical parametric speech synthesis. It must have a front-end in order to process text, and a vocoder that creates the final audio wave files. The front-ends that are currently used are Festival and Ossian, since their outputs are HTS-style labels with state-level alignment. These labels are converted into vectors of binary and continuous features for the input of the neural network.

For vocoder, the system supports currently STRAIGHT [6, 12] and WORLD [13]. STRAIGHT uses F0 extraction [7], while WORLD uses the DIO algorithm in order to estimate F0 and CheapTrick [11, 10] for spectral envelop estimation. For synthesis, WORLD uses excitation signal and spectrograms in order to calculate vocal cord vibration using PLATINUM, an aperiodic parameter extraction algorithm. Figure 4 (left) shows the standard Merlin architecture with two neural networks.

The right side of the figure shows a modified architecture, designed to make the synthesised voice more expressive [9]. This solution aims at improving the realisation of prosody in TTS by advancing and implementing the model of prominence.

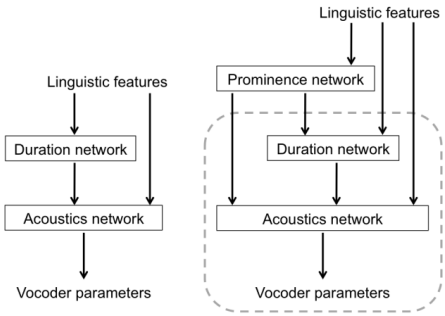

Figure 4: Left: standard Merlin DNN synthesis architecture. Right: modified architecture with explicit prominence modelling.

The modified method includes a third network to provide prominence features and to enable the control of prominence and emphasis. The main concerns in this project was related to the fact that prominence has a large number of acoustic features, such as duration, F0, energy, and spectrum. The results for this were considered to be very satisfying since they were able to control word prominence and the ratings for speech naturalness were better than for the baseline voice.

Although this method seems to be a first start to synthesise a more expressive speech, in this master thesis I did not work on this method.

3.1. Build_your_own_voice in English

In order to synthesise a different voice from the demo voices in Merlin, we started to work with the Build_your_own_voice in Merlin toolkit, first with the dataset from CMU ARCTIC, and then with the LJSpeech [6] [4]. The first dataset, slt_arctic, from CMU ARCTIC was specially designed for research in speech synthesis and is publicly available. The utterances are 16 bit, mono waveforms, sampled at 16kHz. The dataset comprises 1,032 utterances. The LJSpeech dataset is also public domain, and includes 13,100 short audio files, amounting to a total of approximately 24 hours of speech. The audio files have a sampling rate of 22,050Hz. Merlin accepts sampling frequencies 16kHz or 48kHz, (this last one is very uncommon). In order to use this second dataset, we had to convert the sampling frequency from each file to a frequency accepted from Merlin.

Although the LJSpeech has 13,100 audio files we used a maximum of 6,119 files as we can see in table 2. These files were chosen according to the most relevant features and utterances to have in the dataset. This choice was made mainly because of the performance of the machine where this system was tested. Another reason for this choice is that 6,119 audio files is already a very big dataset when we compare it to the CMU ARCTIC. Besides, with the results from this dataset size

[6] https://keithito.com/LJ-Speech-Dataset/

Table 2: Number of files in global_setings_cfg for 6,119 utterances

Total number of files	6,119
Training Files	5,519
Validation Files	300
Test Files	300

Table 3: Number of files in global_setings_cfg for Vitalina

Total number of files	7,308
Training Files	6,578
Validation Files	365
Test Files	365

and comparing them with the results obtained with trainings made to debug with smaller datasets from the same database were very consistent, allowing us to draw conclusions about the results with this database.

Build_your_own_voice method requires speech tools[7], festival[8], festvoz[9] and HTK[10] to be installed in this order. Some libraries are needed to install each of these packages. One should also take into account that the HTK toolkit was developed to work in a system with 32 bits.

In order to train Merlin with these datasets, one must follow seven steps. In the first one, one must make some changes, namely to include the correct path for the required tools, festival, festvox and HTK. We also change or include the sentences that we want to synthesise. And we also include the question_file that we want to use. This file contains the phonological characteristics from the language to use. For the English synthesis, this parameter was not changed because we had already a question_file for English in Merlin. Besides that we have to choose the number of files to train, validate and test Merlin. This is chosen with a percentage from the the total number of files, for the training we use 90% of the files and for the validation and test we use 5% each. As we can see in table 2 these were the values used in order to train Merlin with the English Voice LJSpeech.[11] This first step creates the config_file that has all the information to do the other steps from Merlin. The other 6 steps do not require any special action, we just have to run them in the correct order and with the correct inputs.

Some of the synthesised phrases are presented below and then for the classification of obtained results, we just listened to the audio and observed the waveforms of the files in wavesurfer.

- "Hello, I am Catarina."
- "Hi, this is a demo voice from Merlin."
- "Printing, in the only sense with which we are at present concerned, differs from most if not from all the arts and crafts represented in the Exhibition in being comparatively modern."

[7] https://github.com/festvox/speech_tools
[8] https://github.com/festvox/festival
[9] https://github.com/festvox/festvox
[10] https://github.com/ibillxia/htk_3_4_1
[11] https://github.com/CSTR-Edinburgh/merlin/issues/203

Some of the conclusions taken from these results were relative to the LJSpeech dataset. This one contains a more expressive speech in terms of energy of speech, but Merlin works with probabilistic methods that began to calculate the means of the energies of each syllable with the purpose to define the pitch of the speech. So, when we have a dataset with energy changes, we can detect some noise in the audio produced and because of that we can also conclude that some of the speech created with the slt_arctic_full has a better quality than with the LJSpeech, even if the last dataset is bigger.

With these observations and knowing how to create an expressive speech synthesizer with Merlin, the next steps were directed to the synthesis of a Portuguese voice where the dataset used had to be most neutral in terms of energy .

3.2. Build_your_own_voice in Portuguese

In order to synthesise the Portuguese voice it was necessary to make changes in Festival and to create a proper question_file for the Portuguese phonology. This file was done according to the information from section 2.2 and according with some information defined in Festival for the European Portuguese Speech. For Festival, the changes were mainly done by changing the English voice for the Portuguese one l2f_sgpAlign_diphone provided by INESC. That change was made by adding to festival the directories from the Portuguese voice and by changing the used voice in the file init.scm which is the file responsible for the initialisation of Festival.

After changing the question_file and the necessary Festival files, we chose the first 60 utterances from the Vitalina dataset in order to test and debug Merlin. After debugging the system, the final Merlin synthesizer was trained with the 7,303 audio files from Vitalina. This dataset from VoiceInteraction was chosen because it has recordings from a female voice with higher pitch, thus closer to a child's voice than any other possible choice of adult voices.

Such as we did for the English voice and explained in section 3.1, the choice of the number of training, validation and testing files is presented in table 3.

In order to understand the duration of the training for Merlin, we did several time measures in some of the performed trainings and related them with

the duration of the dataset used. These results are presented in table 4.

We also measured the necessary time to synthesise different amount of Voice in Merlin with 1,100 and 7,303 training utterances. The obtained results are presented in table 5. From this we can conclude that, as expected, the amount of synthesised phrases and the size of the training dataset does not have a big influence in the duration of synthesis. The conclusion that can be taken from technical experience regarding the time of synthesis is that this also depends on the computer load at the moment of synthesis. If is using RAM for another process it can take a little bit more time to synthesise the phrases. However, for me, it never took more than 2 minutes to synthesise phrases in Merlin.

Some of the synthesised sentences can be found below. We decided to have four main groups of synthesised phrases. The first one relates to common words and sentences and to this group belong the first two phrases presented. The next three phrases belong to Portuguese news and to the second group of phrases. Then we present two phrases that belong to the third group and relate to meteorology. Finally, we present some phrases that belong to the fourth and final group which are phrases from children books.

- ”Estou a trabalhar com o Merlim para sintetizar voz em Português para a minha dissertação de mestrado.”
- ”Eu almoço um bom almoço.”
- ”O memorando sobre a recuperação do material de guerra roubado em Tancos não deixou qualquer rasto no Ministério da Defesa. Ou seja, pura e simplesmente não existe.”
- ”Não é uma escolha, eu sou assim!”
- ”A selecção portuguesa defronta a sua congénere polaca para a Liga das Nações. Um triunfo deixa as portas abertas para a final a quatro da competição.”
- ”A tempestade tropical Michael está prestes a transformar-se em furacão, ameaçando com fortes chuvas o oeste de Cuba.”[12]
- Céu muito nublado, diminuindo de nebulosidade a partir do início da tarde.[13]
- ”Só depois se punha de pé sempre o último da fila lá ia pelo corredor arrastando-se penosamente e nem sequer tinha o cuidado de disfarçar longos bocejos.”
- Estamos no Inverno e lá fora está muito frio.[5]
- Vamos começar por fazer uma experiência, para ver se tens bom ouvido.[2]
- ”Como é que ele pode saber quem eu sou se nunca me tinha visto? Aproxima-te, para eu te ver melhor. Proíbo-te que bocejes! Posso-me sentar? E quando é isso? Mas para que é que isso te interessa?”[1]
- Com as suas grandes patas, o Simão começa a saltar. E os gafanhotos verdes também o querem imitar.[3]

4. Results

In order to evaluate the results obtained, we created a form based in MOS comparing our results with the in house DIXI system.[14]

Contrarily to the initial expectations, the DIXI voice corresponding to Vitalina was not available, hence we chose *Violeta*, another female European Portuguese voice in DIXI. This enabled a first comparison between Merlin using Vitalina, with different dataset sizes.

A second comparison was made at a later stage by training Merlin with a dataset of 1,000 phrases from Violeta, thus enabling a more fair comparison with the same voice in both systems.

In the first test, the participants were asked for a subjective MOS score (see 1 more specifically in table 1). The results are presented in the figure 5 and in table 6 where we compare the results with the dataset duration. We also chose to compare the MOS obtained inside each group of phrases. We can divide our results in four groups, the first with common phrases, the second with news, the third with phrases from a limited domain (meteorology) and finally the fourth with child's book phrases with dialogues. These results are presented in figure 6.

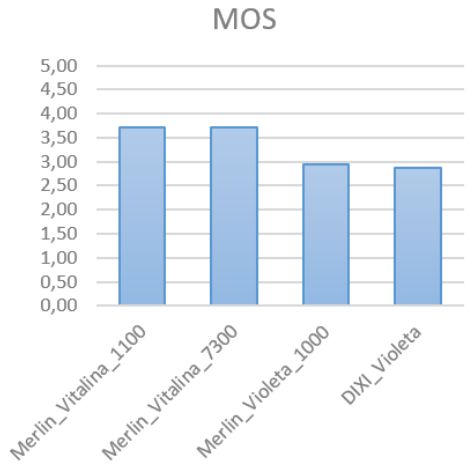

Figure 5: MOS

[12] https://www.tempo.pt/noticias/previsao/outono-tarda-em-chegar-tempo-esta-semana.html

[13] http://www.ipma.pt/pt/index.html

[14] https://www.l2f.inesc-id.pt/demos/voices/

Table 4: Datasets used and training duration

Dataset	Utterances	Dataset Duration [hh:mm:ss]	Training Time [hh:mm:ss]
Vitalina [4]	60	00:20:45	00:50:32
	1,100	01:18:36	09:17:28
	7,303	14:16:12	72:04:08

Table 5: Number of files in global_setings_cfg for Vitalina

Number of training files	Number of synthesised files	Synthesis duration [mm:ss]
1,100	1	01:40
	32	02:57
7,303	1	01:56
	32	02:42

Table 6: MOS obtained for each method

Method	Approximate Duration	MOS
Merlin_Vitalina_1,100	1 hour	3,73
Merlin_Vitalina_7,300	14 hours	3,72
Merlin_Violeta_1,000	1 hour	2,95
DIXI_Violeta	6hours	2,87

As we can observe, the higher MOS has been obtained by the Merlin synthesizer, trained with the Vitalina dataset with 1,100 phrases which consists in x hour in the dataset. The results obtained by training in Merlin with 1,100 and 7,300 phrases are very close in terms of MOS scores. In general, Merlin with 1,100 phrases is slightly better, but if we take into account the MOS distribution per group of synthesised phrases in figure 6 there are some situations where Merlin trained with 7,300 phrases from Vitalina is a better method.

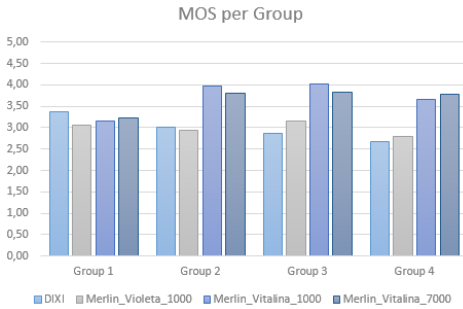

Figure 6: MOS

In the second part of the tests, we synthesised 30 phrases for each of the described synthesis methods and voices. So we made groups of four phrases, one from each method and we asked for the comparison between the phrases within the same group. For each phrase compared with each one the other three in the group we asked in the form to compare each pair of phrases as much better, better, equal, worse or much worse. We repeated this processes making all the possible combination between each group of synthesised audio. In total we had 30 groups. The results of this form are presented in figures 7, 8, 9.

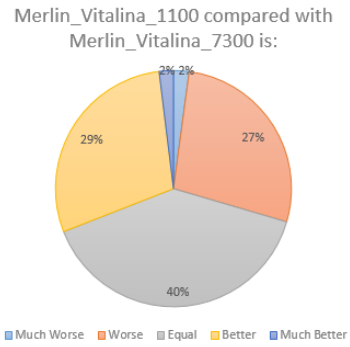

Figure 7: Comparison between the results of Merlin trained with 1,100 and 7,300 phrases from the Vitalina dataset.

As can be observed in figure 7, the two different synthesis used in Merlin are very similar. The results show that 40% of the inquired people found both synthesis equal, 29% found Merlin with 1,100 phrases to be better than when trained with 7,300 phrases and 27% of the inquired said that is worse.

In order to have better conclusions about these results, as we explained above while explaining the MOS evaluation, we have 4 main group of phrases. These groups are presented in figure 8, so we can understand where each training in Merlin has a better result comparing the different synthesis that were made with Vitalina. As it can be seen, in group 3, Merlin with 1,100 phrases performed better than Merlin with 7,300 phrases. However in the other 3 groups the results are not very conclusive as we realised above with the MOS test in figures 5 and 6, and with the figure 7 where we saw the general results for this comparison.

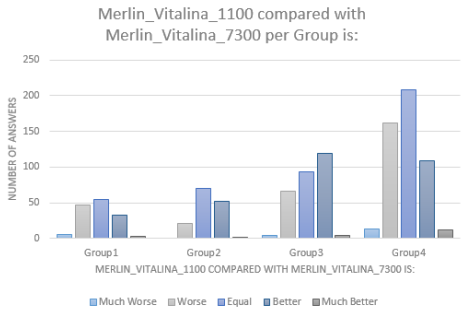

Figure 8: Comparison between the results of Merlin trained with 1,100 and 7,300 phrases from the Vitalina dataset.

Another important comparison we have to make is with DIXI, the concatenive speech synthesis system. Figure 9 shows the results corresponding to the comparison of the other methods with DIXI. From this, we can conclude that clearly Merlin is a better method to synthesise speech when trained with *Vitalina* than DIXI or even Merlin trained with the small *Violeta* dataset.

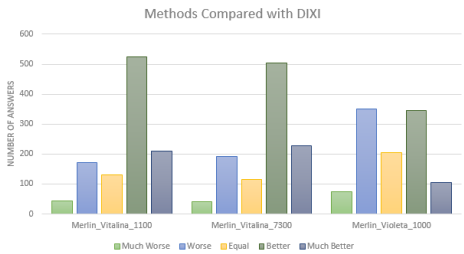

Figure 9: Comparison between the Vitalina in Merlin trained with 1,100 and 7,300 and Violeta with Violeta in DIXI

From figure 10, the results show that 49% of the inquired people found Merlin with 1,100 phrases to be better than DIXI and 19% of the inquired said that is much better. From these values we could conclude that Merlin with 1,100 phrases in training is a better method since we have a total of 68% of the inquired people giving a higher than better classifications and that 12% of the people found them to be equal.

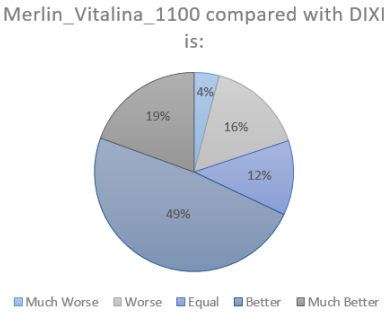

Figure 10: Comparison between the results of DIXI with the Violeta voice, and Merlin trained with 1,100 from the Vitalina dataset.

5. Conclusions

This paper summarises the work developed for the Master Thesis in Electrical and Computer Engineering. We present an overview of the main theme, speech synthesis, and the motivation that led to the proposed work. Since the main task of this thesis is to synthesise an European Portuguese voice, I made some research about the phonological characteristics of the Portuguese speech. Also, the chosen method to synthesise Portuguese speech was Merlin, mainly because it has an open source code and uses Festival as frontend which was already used for DIXI, a concatenative speech synthesis system.

As it was described in this paper I managed to synthesise an European Portuguese voice with two different datasets. For one of those datasets, I did several synthesis using a different number of phrases from them.

The main technical problem that I encountered during this project relates to the machine characteristics necessary to run Merlin without problems. Therefore, with a computer with 2Gb of RAM memory and an Intel Core 2 Duo CPU, the results were MemoryError in the acoustic model training and with more tests we realised that 4Gb of RAM were not enough as well. Finally, with a computer with 6Gb of RAM memory and an Intel Core i7 CPU and system Lubuntu 17.04, 64 bits version we got to run Merlin without any major issue, but with very large times of training.

Given the results presented in section 4, we can conclude that the main purpose was achieved, since the results comparing Merlin with DIXI showed that the majority of the inquired people preferred Merlin trained with *Vitalina* over DIXI. We could also observe that when comparing Merlin synthesis with *Violeta* and DIXI the results were inconclusive. These results can be justified with the amount of phrases from *Violeta* used to trained Merlin. However, since the dataset is just 100 phrases smaller than the one from *Vitalina* used, the results can be explained if the corpus of these two voices is different, since *Violeta* can be lacking some important phoneme sequences.

Thus, we believe that the main goal of this Thesis was achieved and it can also be recognised the importance of these work for future applications including maybe the replacement of DIXI.

For future work there are some possible directions this research can take. The first one is to record a Child in order to synthesise a child's voice that can be applied in the robots used for the Monarch and the INSIDE projects in order to have interaction with the children with a child's voice. This may increase the help given to these children reducing the apprehension felt by them since the

communications are done through a voice from an equal.

A more challenging future possible application is to develop a method to have a more expressive voice in Merlin. This could be achieved for example using the method of adding a prominence model. [9]

References

[1] A. de Sant-Exupéry. *O Principezinho*. Relógi D'Água.

[2] G. Delahaye. *Anita descobre a música*. Verbo Infantil.

[3] B. Doumerc. *O Urso Simão*. Porto Editora.

[4] K. Ito. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/, 2017.

[5] J. Ivens. *Vitor conhece o Pai Natal*. Edições ASA II.

[6] H. Kawahara. Straight, exploitation of the other aspect of vocoder: Perceptually isomorphic decomposition of speech sounds. *Acoustical Science and Technology*, 27(6):349–353, 2006.

[7] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné. Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based {F0} extraction: Possible role of a repetitive structure in sounds1. *Speech Communication*, 27(3–4):187 – 207, 1999.

[8] S. King. A beginners' guide to statistical parametric speech synthesis. *The Centre for Speech Technology Research University of Edinburgh*, June 2010.

[9] Z. Malisz, H. Berthelsen, J. Beskow, and J. Gustafson. Controlling prominence realisation in parametric dnn-based speech synthesis. In *INTERSPEECH*, 2017.

[10] M. MORISE. Error evaluation of an f0-adaptive spectral envelope estimator in robustness against the additive noise and f0 error. *IEICE Transactions on Information and Systems*, E98.D(7):1405–1408.

[11] M. Morise. Cheaptrick, a spectral envelope estimator for high-quality speech synthesis. *Speech Communication*, 67:1 – 7, 2015.

[12] M. Morise. D4c, a band-aperiodicity estimator for high-quality speech synthesis. *Speech Communication*, 84:57–65, 2016.

[13] M. Morise, F. Yokomori, and K. Ozawa. WORLD: A vocoder-based high-quality speech synthesis system for real-time applications. *IEICE Transactions on Information and Systems*, E99-D(7):1877–1884, 2016.

[14] S. Ronanki, Z. Wu, O. Watts, and S. King. A Demonstration of the Merlin Open Source Neural Network Speech Synthesis System, September 2016.

[15] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastne, A. Courville, and Y. Bengio. Char2Wav: End-to-End Speech Synthesis. https://openreview.net/forum?id=B1VWyySKx, February 2017.

[16] P. Wagner, J. Abresch, S. Breuer, and W. Hess, editors. *Sixth ISCA Workshop on Speech Synthesis, Bonn, Germany, August 22-24, 2007*. ISCA, 2007.

[17] C. Weiss, L. C. Oliveira, S. Paulo, C. Mendes, L. Figueira, M. Vala, P. Sequeira, A. Paiva, T. Vogt, and E. André. ecircus: building voices for autonomous speaking agents. In Wagner et al. [16], pages 300–303.

[18] Z. Wu, O. Watts, and S. King. *Merlin: An Open Source Neural Network Speech Synthesis System*, pages 218–223. Sunnyvale, CA, USA, 9 2016.