Abstract—Amyotrophic Lateral Sclerosis (ALS) is a group of rare neurological diseases that mainly affects the nerve cells responsible for voluntary muscle movement. The disease is progressive, meaning that the symptoms get worse over time, leaving the patient in a highly incapacitating state. Due to the progressive muscle atrophy that affects patients, dependence on caregivers grows, and the role of assistive technologies comes into play. Most patients experience a dramatic loss in speech intelligibility caused by an increasing muscle weakness, and hence both written and spoken communication are distressed. By developing a way for patients to preserve their voice in a speech synthesizer, we are helping them retaining a vital part of their identity, instead of using off-the-shelf synthesizers with generic voices. During the course of this work we will develop a Portuguese speech synthesizer that is able to perform speaker adaptation on small voice banks, which is most useful in the case of ALS, where the task of recording extensive voice banks is both physically and mentally exhausting.

Index Terms—Amyotrophic Lateral Sclerosis, Speech Synthesis, Deep Learning, Transfer Learning, Assistive Technologies.

I. INTRODUCTION

Myotrophic Lateral Sclerosis (ALS) is an incurable, rapidly progressive degenerative disease involving damage to both upper and lower motor neurons. The neuromuscular systems affected progressively loose motor function to end in total movement impairment. As a consequence, locomotion, limbic movements, breathing and speech are progressively lost [1].

Speech is not only the most direct form of communication, it is also a signal that contains relevant non-verbal information that can be used to understand a vast array of factors like context, emotion and health states. People unable to speak are in a significantly inferior position when it comes to socialization and everyday communication, regardless of whether their condition is congenital or acquired. There are many ways to help people with impaired speech communicate, some methods are more archaic like text boards or pre-recorded sentences, and some others more technological like eye-tracking devices. An efficient alternative to augment communication is text-to-speech (TTS) software, which allows patients to convert typed text into audible speech using a smart phone or a similar portable device.

Despite the significant offer of commercial available speech synthesizers, the majority of them sounds impersonal as patients are using a generic voice instead of their own. Also, not all the available systems are adapted to the Portuguese language. Preserving a patient’s voice helps the person to retain a large part of his/her identity. This turns out to be a positive aspect in the face of such a devastating disease [2].

The main goal of this work is to generate a state-of-the-art speech synthesizer for the Portuguese language that can be adapted to ALS patient’s voice. Although state of the art speech synthesizers that use deep learning technologies usually require a large amount of data to obtain high quality models, the fact that ALS is a highly incapacitating disease is a very strong motivation to decrease the amount of required data as much as possible.

During the course of this work we will also review the particular challenges that European Portuguese may pose to TTS systems, and the way that the developed system tackles these issues. This is a relevant analysis since self-learning approaches like Deep Learning may tend to learn difficult tasks (e.g. pronunciation of certain words) inherently, while previous TTS systems typically required a rule based approach.

We will approach the building of a Portuguese speech synthesizer that can be fine-tuned to different speakers with reduced amounts of data, in order to cater to the needs of ALS patients. All the available recordings, however, corresponded to healthy subjects.

II. BACKGROUND

Our work used as a baseline model an online repository that contains a Deep Learning Speech Synthesizer called Tacotron-2. This model is the result of the combination of Tacotron and WaveNet, and yields state of the art speech synthesis. The repository was developed and regularly reviewed by Rayhane Mamah and had a large community engagement [3].

A. Tacotron

Tacotron is an end-to-end generative text-to-speech model that synthesizes speech directly from characters. Given text and audio pairs, one can train a model completely from scratch with random initialization. It does not require phoneme-level alignment, so it can easily scale to using large amounts of acoustic data accompanied with orthographic transcripts.

Tacotron is a sequence-to-sequence model optimized for TTS that maps a sequence of letters to a sequence of features that encode audio. Sequence-to-sequence models, also called encoder/decoder models, are deep learning models that aim to map a fixed length input with a fixed length output, where the length of the input and output may differ. These models are particularly useful in machine translation, video/image captioning and also automatic speech recognition [4]. The
features (80-dimensional audio spectrogram with frames computed every 12.5 milliseconds) capture both pronunciation of words and various subtleties of human speech, e.g. volume, speed and intonation. Finally the features are converted to a waveform using a Griffin-Lim reconstruction. [5].

Tacotron is organized in 4 different building blocks: encoder, CBHG module (composed of convolutional bank, highway network, and a bidirectional Gated Recurrent Unit), decoder and post-processing network. The goal of the encoder is to extract robust sequential representations of text. The input is a character sequence where each character is embedded into a continuous vector, a set of non-linear transformations are then applied (collectively called “pre-net”) to each embedding. The CBHG module transforms the pre-net outputs into the final encoder representation, or called intermediate vector. A content-based tanh attention decoder is used, where a stateful recurrent layer produces the attention query at each decoder time step. Attention mechanisms [6] are very important in machine learning since they provide the ability to capture context and in our case, learn relationships between different phonemes and characters. Finally, the post processing network converts the decoder’s output to a target that can be converted to a waveform by a Griffin-Lim reconstructor.

Therefore, each audio sample $x_t$ is conditioned on the previous timesteps. Similarly to Neural Networks used for pixel image recognition, Wavenet’s conditional probability distribution is modelled by a stack of convolutional layers. There are no pooling layers, and the output has the same time dimensionality as the input. The main aspect that differentiates wavenet from other Deep Learning synthesizers is the usage of causal convolutions. By using causal convolutions the model cannot make predictions dependent on future timesteps, and since it does not have any recurrent connections it is usually faster to train than Recurrent Neural Network models.

The problem that rises when using causal convolutions is the large layer requirement in order to increase the receptive field (defined region in the input space), which increases the overall computational cost. WaveNet architecture introduces dilated convolutions in order to greatly increase the receptive field without compromising the computational cost. Firstly used for image classification tasks [8], a dilated convolution is a convolution where the filter is applied over an area larger than its length, by skipping input values with a certain step. It is equivalent to a convolution with a larger filter derived from the original filter by dilating it with zeros, but is significantly more efficient. A dilated convolution effectively allows the network to operate on a coarser scale than a normal convolution without compromising computational cost.

The global architecture of WaveNet is present in Figure 3, the stacked blocks that output skip and residual connections are called residual blocks. The residual blocks can be seen as the building blocks of WaveNet. There are as many as dilation layers, and for each residual block, the output of the previous one is fed directly. By doing this, local features in low dilation layers are accumulated sequentially to capture dependency between the data points at distance. The block produces two outputs: a feature map that is fed directly into the next block and a skip-connection which will be used to calculate loss function for the batch after the aggregation. Both residual and skip-connections
are used throughout WaveNet to speed up convergence and enable deeper training.

C. Tacotron-2

Despite WaveNet being a state of the art raw waveform generator, it does not generate mel-spectrograms or some other features to be conditioned with by itself. In order to have a fully functioning TTS system, WaveNet needs to be combined with a feature generator system. The combination of both Tacotron and WaveNet is called Tacotron-2 [9]. Tacotron serves as a sequence-to-sequence feature predictor with attention that predicts mel-spectrograms, while WaveNet generates time-domain waveform samples conditioned on the generated mel-spectrograms.

Tacotron-2 shares very similar traits with Tacotron, but instead of having an appended Griffin-Lim vocoder it has a WaveNet. Figure 4 displays an overview of the Tacotron-2 architecture, in a general way.

III. PORTUGUESE SPEECH SYNTHESIS - BASE MODEL

We chose Tacotron-2 to be the TTS system used during this work. Despite being produced by Google and Google’s DeepMind, the companies did not release any official software implementation. Regardless, there are several open-source implementations available online to be used and developed. After an extensive research and experimentation, we chose to base our work on Rayhane Mamah’s Tacotron-2 implementation [10], as it was previously mentioned.

The voice banks used throughout this work are displayed on table I, followed by the sentence type distribution in each case (Interrogative, Exclamatory and Declarative). All voice banks were recorded in European Portuguese and have a sampling rate of 16kHz.

<table>
<thead>
<tr>
<th>Name</th>
<th>Nr. Utterances</th>
<th>Dur.</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSF1</td>
<td>7303</td>
<td>14h 16min 28sec</td>
<td>F</td>
</tr>
<tr>
<td>SSF2</td>
<td>728</td>
<td>1h 3min 30sec</td>
<td>F</td>
</tr>
<tr>
<td>SSM2</td>
<td>728</td>
<td>1h 0min 27sec</td>
<td>M</td>
</tr>
</tbody>
</table>

**TABLE I**

VOICE BANKS USED, FOLLOWED BY SENTENCE TYPE DISTRIBUTION.

In order to create our base model, we will train Tacotron-2 with the SSF1 voice bank, since it is the one that presents larger input data. This was a voice bank recorded and primarily used for concatenative speech synthesis, therefore all sentences were recorded in a slow and articulate way.

A. Audio Pre-processing

Since Tacotron serves as a mel-spectrogram generator, we need to convert the input data to mel-spectrograms, so they can serve as target for the training procedure. In order to do so, it is required to perform a pre-processing of the input audio before feeding it to the system. All the audio we used during the course of this work had a sampling rate of 16kHz. As a window size, we used 50ms and as hop size 12.5ms. These two values were the ones suggested in Tacotron-2 release paper [9]. The \( n_{fft} \) parameter stands for Non-Uniform Fast Fourier Transform, and it is used to give the length of the signal that we want to compute the FFT of. The \( n_{fft} \) algorithm is much more efficient if set to powers of 2. Due to this, we set this value to 1024, which is the first power of two above window size (50ms, 800 samples). Trimming and frequency parameters were also adapted accordingly.

B. Tacotron Training

The first step in building our system was training a Tacotron model with the SSF1 voice bank, which is sufficiently larger to learn proper linguistic features. Since Tacotron-2 is a pipeline comprised of a WaveNet attached to a Tacotron, we decided to train these two systems independently for easier tuning and evaluation. Tacotron training is performed in mini batches that are randomly sampled from the training set. We chose a batch...
size of 32. A training step corresponds to one iteration of a batch through all the network. Regarding learning rate, we chose an exponential decay, from step 20k until step 300k. We used a dropout rate of 0.4 and constant teacher forcing.

There are three main ways that we can access how good a Tacotron model is training. We can first do a loss analysis on both the training and validation loss, we can listen to the output synthesis for a perceptual analysis and we can also check the alignment graph. The alignment graph is a graphic representation of the alignment between the encoder and the decoder, and it is a very important tool to assess how both the encoder/decoder and attention mechanism are learning. This graph is generated at validation phase using the timesteps of a random utterance.

In short, the encoder reads input characters step-by-step and outputs status vectors. The decoder reads all status vectors and generates audio frames step-by-step. A good alignment simply means that, for example, a $\beta$ sound generated by the decoder should be the result of focusing on the vector generated by the encoder from reading the respective “v” character. Not having a proper alignment means that the system is outputting vocal sounds that do not correspond to the written characters at the same timestep. A diagonal line is the result when audio frames are created by focusing on the correct input characters. At each decoding step, the whole y-axis is a weighted sum. The focused vector is actually the weighted average of all encoder’s status vectors [11]. Usually a clearer diagonal line stands for a better alignment between the encoder and the decoder. We can find the alignment plot of our SSF1 Tacotron model on figure 5.

![Alignment plot of the Tacotron model trained with SSF1.](image)

C. WaveNet Training

WaveNet Training is also performed in mini-batches, similarly to Tacotron. A training step corresponds to an iteration of a batch through all the WaveNet network. We used mini-batches of size 8 and kept one batch as a test batch. We also set an exponential learning rate decay over 500k steps decaying from 1e-3 with a decay rate of 0.5.

Rayhane’s WaveNet accepts 3 different input types regarding quantization: raw, mulaw or mulaw-quantize. These types differ in the number of quantization channels, which in the raw case is 65536 (16-bit), and in the mulaw/mulaw-quantize is 256 (8-bit). In our first experiments using the raw input type, we faced exploding gradient problems and the resulting synthesis had prominent artifacts. These problems were also found across different voice banks and languages. We ended up choosing mulaw quantization over raw, that despite introducing a slight decrease in quality, did not present any exploding gradients and noisy outputs.

We chose to train our WaveNet models with Ground Truth Alignment (GTA). This parameter serves as a Tacotron-2 integration parameter. It essentially allows WaveNet to be trained with mel-spectrograms previously aligned by Tacotron, instead of the ground truth mel-spectrograms. This parameter increases the synergy of both Tacotron and WaveNet and was a staple in all our experiments.

We set our WaveNet to train for 500k steps, with the alignment provided by the previously developed the Tacotron model. Heuristically, we discovered that the quality of the output synthesis was directly correlated with the value of WaveNet’s validation loss. Due to this, we chose as our final WaveNet model the checkpoint we obtained during training at step 410k, which presented the minimum validation loss achieved, since the training loss had already reached a plateau at this point.

D. Evaluation

The evaluation of a Text-To-Speech system is an intricate task. There is no universal protocol on how to assess a TTS system, since there are several ways that synthesized speech can be evaluated, such as assessing speech intelligibility, naturalness or even efficiency in a given task. To evaluate the results of the model developed, we used two methods suggested by the Special Interest Group of ISCA (International Speech Communication Association) on Synthesis (SynSIG) [12]: A Mean Opinion Score (MOS) to assess both naturalness & quality, and also a Word Error Rate to evaluate intelligibility.

1) Mean Opinion Score: To evaluate the quality and naturalness of our model, we conducted a MOS test, where the subjects were asked to rate the naturalness of the stimuli in a 5-point Likert scale score (1-bad, 2-poor, 3-fair, 4-good, 5-excellent). 100 unseen sentences were used for the tests and each utterance received at least 8 ratings. The results can be found on table II compared to Natural Speech.

<table>
<thead>
<tr>
<th>System</th>
<th>MOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Natural Speech</td>
<td>4.42 ± 0.60</td>
</tr>
<tr>
<td>Tacotron-2 w/ SSF1</td>
<td>3.82 ± 0.69</td>
</tr>
</tbody>
</table>

2) Semantically Unpredictable Sentences: In order to evaluate the intelligibility of our TTS system developed we performed a word-error-rate test in the form of semantically
unpredictable sentences. 10 unpredictable sentences were synthesized and received at least 8 different ratings. The results were computed using the following expression:

\[ WER(\%) = \frac{Sw + Iw + Dw}{Tw} \times 100 \]

Where \( Sw \) stands for substitutions, \( Iw \) for insertions, \( Dw \) for deleted words and \( Tw \) for total words. The system achieved a WER of 1.42\%, therefore it is able to produce reliable intelligible speech.

3) Preference Test: Since we had access to synthesized speech obtained by a parametric speech synthesizer enhanced by Deep Learning (Merlin, [13]) that used the SSF1 Voice Bank, we also conducted a preference test.

The test was performed in a pair comparison way, where after listening to each pair of samples, the subjects were asked to choose which one they preferred in terms of naturalness, though they could choose “neutral” if they did not have any clear preference. A total of 10 pairs was subject to evaluation by at least 8 listeners. The results can be found on Table III, showing a very clear preference for Tacotron-2.

<table>
<thead>
<tr>
<th>System</th>
<th>Preference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tacotron-2</td>
<td>70.0%</td>
</tr>
<tr>
<td>Merlin</td>
<td>17.5%</td>
</tr>
<tr>
<td>Neutral</td>
<td>12.5%</td>
</tr>
</tbody>
</table>

TABLE III
PREFERENCE TEST RESULTS.

4) Concerning Interrogative Sentences: It is known that Deep Learning has large data requirements in order to learn specific features. Given the limited representation of interrogative sentences in the SSF1 Voice Bank, we investigated if the model had learned the specific intonation that characterizes this type of sentences.

One can divide interrogative intonation in 3 distinct parts: total, partial and alternative [14]. Total intonation type is mostly present in yes/no questions, and presents an ascending pitch contour towards the end of the sentence. Partial intonation type is regularly present in wh-questions (what, where, how, who, etc). These questions expect more information in the answer than a simple yes/no, and usually present a descending pitch contour. Finally, the alternate question intonation presents an alternative to the listener, usually with the presence of the conjunction “or”. This intonation pattern can be a bit more complex, displaying several pitch contour patterns like an ascending-descending-ascending or ascending-descending [15].

Figure 6 displays an example of the spectrogram and pitch contour achieved by a yes/no question synthesized with our SSF1 TTS model.

E. Grapheme-to-Phone Errors

It is known that self-learning methods like Deep Learning tend to perform worse when addressing the synthesis of words whose pronunciation does not follow typical orthographic rules. This has been addressed for several G2P (Grapheme-to-Phone) modules in European Portuguese, including rule-based and neural network approaches [16].

In this section we want to perform an analysis on how our develop system addresses these issues, stating the common problems that Deep Learning approaches face regarding the limitations in training material. For this approach we will focus on 3 relevant problems that TTS systems face for European Portuguese: the specific pronunciation of \( e \), \( o \) and \( x \), acronyms and finally homograph words, that deal on the distinction between verb and name. These are the cases where self-learning approaches tend to fail the most.

1) The Specific Case of \( e \), \( o \) and \( x \): In Grapheme-to-Phone tasks for European Portuguese, the most common type of errors concerns the transcription of the graphemes \( e \) and \( o \), to which two different phonological representations can be associated in stressed syllables, and also the transcription of \( x \) whose contextual variation is difficult to predict, since it depends, among other facts, on the time the word entered the lexicon [16].

In order to evaluate this aspect, we synthesized a list of 51 words that had the presence of these graphemes in difficult instances, and also some extra words that usually present
difficulties to TTS systems. Some of these words can be found on table IV.

<table>
<thead>
<tr>
<th>Words</th>
<th>Achieved</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>aquarela</td>
<td>[6kEs ər]</td>
<td>[6kEs ər]</td>
</tr>
<tr>
<td>telefone</td>
<td>[t@l@f ən@]</td>
<td>[t@l@f ən@]</td>
</tr>
<tr>
<td>acorda</td>
<td>[t6s ɔ rds]</td>
<td>[t6s ɔ rds]</td>
</tr>
<tr>
<td>hexagonal</td>
<td>[f36ggun ɔ l]</td>
<td>[f36ggun ɔ l]</td>
</tr>
</tbody>
</table>

### TABLE IV
ACHIEVED PRONUNCIATION OF WORDS THAT GENERALLY PRESENT GRAPHEME-TO-PHONIC ERRORS.

From all the specific words synthesized in this test, the system achieved 78.4% of correctness regarding pronunciation. This can be seen as a promising result due to the difficult nature of these words and also due to the nature of the voice bank, that did not have any examples specially designed for this evaluation’s purpose.

2) **Acronyms**: Acronyms usually have a pronunciation that follows rules significantly different than the ones observed in common lexicon, specially in the case of siglae in which each character is spelled individually. As a result, the correct pronunciation of these acronyms is highly correlated to their representation in the voice bank. We found that the majority of the acronyms present in the voice bank were synthesized correctly, surprisingly, even the siglae PSD was spelled correctly, however it had 15 instances present in the voice bank.

3) **Homographs**: A homograph is a group (usually a pair) of words that are spelled the same way, but do not have the same pronunciation. Homographs present complications to the general performance of TTS systems, since the pronunciation of these words is mostly given by context instead of orthographic rules. The most common type of errors these word-pairs face concerns the transcription of graphemes e and o, as seen before.

In order to evaluate the way our model is conveying the different phonological representations associated to homographs, we synthesized 10 examples that contemplated these word pairs. Table V displays the sentences used, where red corresponds to mispronunciations.

<table>
<thead>
<tr>
<th>Nr.</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Eu acordo no dia do acordo.</td>
</tr>
<tr>
<td>2</td>
<td>Usei a colher para colher as sementes.</td>
</tr>
<tr>
<td>3</td>
<td>No começo do dia começo por dançar.</td>
</tr>
<tr>
<td>4</td>
<td>Eu jogo o jogo que ele me deu.</td>
</tr>
<tr>
<td>5</td>
<td>Estou com sede depois de sair da sede.</td>
</tr>
<tr>
<td>6</td>
<td>Eu olho para o olho da Maria.</td>
</tr>
<tr>
<td>7</td>
<td>O Rei teve de fazer um corte na corte.</td>
</tr>
<tr>
<td>8</td>
<td>Não quero que sobre nada sobre a mesa.</td>
</tr>
<tr>
<td>9</td>
<td>Eu molho a taça com o molho.</td>
</tr>
<tr>
<td>10</td>
<td>Eu acerto no dia de fazer o acerto.</td>
</tr>
</tbody>
</table>

### TABLE V
HOMOGRAPH TEST PERFORMED. RED MARKED WORDS DISPLAY MISPRONUNCIATIONS.

From the results achieved, 3 out of 10 of these sentences did pronounce the word-pairs correctly in the two instances they appeared, however did not generalize to the same word pairs in different contexts. These aspects may display some traits of model learning regarding word context in a sentence, dealing with distinction between verb and noun, which could be an interesting Deep Learning feature to investigate. It is important to state that the words used did not have a strong representation in the voice bank, either as a verb or a noun, therefore a deeper assessment on this topic should include a larger representation of these word pairs in order to better study this phonetic distinction between name and verb.

### IV. SPEECH SYNTHESIS - SPEAKER ADAPTATION

This section describes the transfer learning process we went through in order to fine-tune smaller amounts of data from another speaker to the baseline model previously developed. Fine-tuning can be seen as the process of taking a network model that has already been trained on a given task, and make it perform a second similar task, assuming that the second task shares similar traits. In our task we will take advantage of the TTS system we developed using the SSF1 voice bank, and fine-tune its Tacotron and WaveNet to new, smaller voice banks.

#### A. Tacotron

In order to fine-tune our Tacotron model to new voice banks, the most relevant hyperparameter we changed was the encoder freezing. This hyperparameter is suggested to be used while fine-tuning small amounts of data and essentially freezes the encoder, only training the decoder on the new data [17]. Tacotron’s initial layers have been seen to capture generic features, while the later ones focus more on the specific task at hand. Since we want to fine-tune a speech synthesizer, we can safely assume that the first few layers of Tacotron will share features with the fine-tuned one. Using this insight, we can freeze the encoder while fine-tuning the decoder. This presented a faster fine-tuning and also better model convergence.

The Tacotron fine-tuning of the SSF2 voice bank was performed during 30,000 steps, while the fine-tuning of the SSM2 voice bank was performed on only 7,500 steps, both using the SSF1 model as basis. The checkpoints achieved at these particular steps were chosen since they were the ones obtained with minimal validation loss, since the learning loss at this point had already reached a plateau. The alignment plot of both models at these steps also displayed signs of a proper alignment, therefore no further training was required.

#### B. WaveNet

Regarding WaveNet, the main change performed while fine-tuning was the learning rate scheduling, that was changed from an exponential decay to a scheduled learning rate with warmup. This is a new approach in machine learning that intends to decrease computation time by increasing model convergence in fewer steps [18]. Heuristically, we discovered that in this particular case, this approach promoted a better and faster learning in our speaker adaptation tests, therefore we chose it over an exponential decay.

We performed the fine-tuning process over the previously WaveNet trained on the SSF1 voice bank during 90,000 steps.
for the SSF2 voice bank, and for 117,500 for the SSM2 counterpart. These checkpoints promoted the minimal validation loss. The validation loss achieved was slightly larger in the male model, which translated into noisier synthesis.

C. Evaluation

Similarly to III-D, we also performed the evaluation of our Speaker Adaptation models using a MOS and WER test. The results obtained can be found next.

1) Mean Opinion Score: We also asked 8 different listeners to rate the naturalness and quality of 100 unseen synthesized sentences. The results obtained can be found in table VI, compared to the ratings obtained by natural speech and also the previous model.

<table>
<thead>
<tr>
<th>System</th>
<th>MOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Natural Speech</td>
<td>4.42 ± 0.60</td>
</tr>
<tr>
<td>SSF1 Basis TTS</td>
<td>3.82 ± 0.60</td>
</tr>
<tr>
<td>SSF2 Speaker Adaptation</td>
<td>3.62 ± 0.68</td>
</tr>
<tr>
<td>SSM2 Speaker Adaptation</td>
<td>3.37 ± 0.84</td>
</tr>
</tbody>
</table>

**TABLE VI**
**MEAN OPINION SCORE ACHIEVED BY OUR SPEAKER ADAPTATION MODELS.**

2) Semantically Unpredictable Sentences: In order to assess model intelligibility, we also conducted a SUS test. We created 10 semantically unpredictable sentences for each model being assessed (SSF2 and SSM2) and each sentence was transcribed 8 times. The generated sentences were different for each system assessed with the SUS test. The total Word Error Rate achieved can be found on table VII.

<table>
<thead>
<tr>
<th>System</th>
<th>WER</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSF1 Basis TTS</td>
<td>1.42%</td>
</tr>
<tr>
<td>SSF2 Speaker Adaptation</td>
<td>0.82%</td>
</tr>
<tr>
<td>SSM2 Speaker Adaptation</td>
<td>7.46%</td>
</tr>
</tbody>
</table>

**TABLE VII**
**WORD ERROR RATE ACHIEVED BY THE SUS TEST.**

From the WER test performed, one can notice the difference between the SSF2 model score and the SSM2 counterpart. As it was stated, the SSM2 adapted model presented noisier synthesis than the SSF2 model. Due to this fact, overall intelligibility score was affected, specially in plosives, also known as consonants obtained from the stop of airflow using either teeth, lips or palate and followed by a sudden release of air (/p/, /t/, /b/, /k/, /g/ and /d/).

3) Concerning Interrogative Sentences: As we have seen before, interrogative intonation can be divided in 3 distinct parts [14]. We also performed a similar assessment to III-D4 in order to evaluate the interrogative ability of our fine-tuned models, both male and female.

Contrary to the previous assessment, we faced some issues regarding interrogative intonation on our speaker adaptation models. The main problems found were essentially either lack of specific intonation, noisy endings or word mispronunciation. This aspect was expected since the representation of interrogative sentences on the Speaker Adaptation voice banks (SSF2, SSM2) was fairly low (only around 10%), which translated to only 77 utterances. Nevertheless, we achieved some reliable examples regarding wh-question, an example can be seen on Figure 7, where the characteristic descending pitch pattern of a wh-question is evident. On the alternate question type we were also able to output utterances with a proper intonation, however in many cases we faced issues regarding word mispronunciation.

4) Regarding Vowel Reduction and Resyllabification: An interesting occurrence we found while evaluating our models was the presence of linguistic phenomena such as resyllabification and vowel reduction.

Resyllabification is a phonological process in which consonants are attached to syllables other than those from which they originally came [19]. The term “vowel reduction” is often applied to a number of linguistic phenomena. It can either be used to refer to the deletion of unstressed vowels or it can be used to refer to non-neutralizing changes in pronunciation of stressed and unstressed vowels [20].

These two phenomena are very recurrent in Portuguese, and are usually speaker dependent. In the development of our TTS system trained with the SSF1 voice bank, we did not find a large presence of resyllabification or vowel reduction, which was expected since the voice bank was originally developed for a concatenative speech synthesis approach. The majority of the SSF1 recordings presented speech recorded in a slow rate, with every word articulated correctly. Due to this fact, the resulting TTS system also displayed these characteristics, which despite being intelligible, sounded a bit unnatural or artificial since these phenomena are very prevalent in Portuguese. On the other hand, in the case of our Speaker Adaptation models we did find several occurrences of these phenomena, which contributed to a more natural feel of the synthesized results.

With this analysis, we can conclude that a big part of the naturalness that Tacotron-2 displays is very dependent on the Voice Bank that is being trained with. Several other linguistic phenomena can also be learned using this system, however they can decrease the overall quality of the model since they may prevent the system from learning the correct pronunciation of certain words.

V. Conclusions

The present work addressed the creation of a Portuguese speech synthesizer in order to be fine-tuned to Amyotrophic Lateral Sclerosis patients. Since we did not have access to a specific ALS patient voice bank, we performed the speaker adaptation on regular Portuguese voice banks. Despite that, we showed that we can use transfer learning techniques in order to fine-tune a Tacotron-2 model to a specific voice.

Even tough the two voice banks used for speaker adaptation are comprised of healthy speech, the techniques developed and explored during this work can easily be adapted to ALS patients that still do not have a severe degree of speech impairment.

A. Limitations

Since we only had access to a sufficiently larger Portuguese female voice bank, we were only able to develop a baseline
artificial voice for the female counterpart. Due to this, the male speaker adaptation experiments displayed slightly worse results when compared to the female adaptation. A minor limitation we faced as well was the statistical distribution of sentence type (declarative, interrogative and exclamatory) in the voice banks. As we have seen before on table I, the presence of interrogative and exclamatory sentences is very low when compared to the rest. This prevented the speaker adaptation models from learning specific cases of interrogative intonation.

B. Contributions

During the course of this work we were able to develop a functional Speech Synthesizer that was able to perform speaker adaptation in a reliable way with as much as 1 hour of recorded data. We also performed further reduction tests on the speaker adaptation voice banks, but found that the quality decreased in a gradual fashion.

The present work was also relevant to assess how a Deep Learning speech synthesizer addresses common errors TTS systems face when synthesizing for European Portuguese, showing promising results regarding words whose pronunciation does not follow typical orthographic rules.

This work also shared similar traits with Unbabel’s Voice Morphing Project, where the objective is to achieve vocal machine translation preserving the speakers voice in the target language. The first stage of the project is related to Deep Learning speech synthesis, and the development of our Portuguese model was shared and updated with the collaboration of Junior Researcher Luís Bernardo, from Unbabel. This collaboration resulted in the following paper: Unbabel Talk - Human Verified Translations for Voice Instant Messaging, accepted as a Show and Tell for Interspeech 2019 [21].

C. Future Work

Regarding the creation of an ALS specific voice bank, we would suggest a scheduled and extensive voice collection, specially in the first stages post-diagnosis. An interesting idea to explore is using WhatsApp as a voice collection tool. By exploring this idea, the process of voice banking would be made significantly easier for patients, since no dislocations or expensive hardware is required. Parallel WaveNet [22] is an improvement over the traditional WaveNet that significantly cuts down the synthesis time. Since WaveNet relies on a sequential generation of one audio frame at the time, it is still poorly suited for the fast demand in generation time that some applications require. The usage of a fast, high fidelity WaveNet will decrease the synthesis time significantly. Despite the paper being released, there are still no reliable open-source implementations of the system yet.

We consider the results obtained a promising starting point to develop a Portuguese Assistive Artificial voice, that mimics the patients original voice. We can conclude that Tacotron-2 is a solid and reliable choice regarding state-of-the-art speech synthesis that can be fine-tuned to smaller voice banks, while preserving a good quality and intelligibility standard.

REFERENCES