A comparison of multi-style DNN-based TTS approaches using small datasets

Siniša Suzić[1, *], *Tijana* Delić[1], *Vladimir* Jovanović[2], *Milan* Sečujski[1], *Darko* Pekar[2] and *Vlado* Delić[1]

[1]Faculty of Technical Sciences, University of Novi Sad, 21000 Novi Sad, Serbia
[2]AlfaNum Speech Technologies, 21000 Novi Sad, Serbia

Abstract. Studies have shown that people already perceive the interaction with computers, robots and media in the same way as they perceive social communication with other people. For that reason it is critical for a high-quality text-to-speech system (TTS) to sound as human-like as possible. However, a major obstacle in creating expressive TTS voices is that the amount of style-specific speech needed for training such a system is often not sufficient. This paper presents a comparison between different approaches to multi-style TTS, with focus on cases when only a small dataset per style is available. The described approaches have been originally proposed for efficient modelling of multiple speakers with a limited amount of data per speaker. Among the suggested approaches the approach based on style codes has emerged as the best, regardless of the target speech style.

1 Introduction

Text-to-speech (TTS), the branch of speech synthesis based on producing artificial speech from an input text, has been a popular research subject for decades, owing to its far-reaching applications in the field of human-computer interaction (HCI). TTS-based screen readers help people with visual and reading impairments in their everyday lives. Driver assistants allow drivers to focus on the road while performing various other tasks. Automated call centers offer quicker response, since waiting time for human operator to become available is eliminated. Speaking robots help children and adults with autism to cope with their conditions [1, 2].

An important goal of speech synthesis is to make synthesized voice as human-like as possible, based on the assumption that such a voice is more pleasant for a human listener [3]. Synthesized speech should not just communicate information, it should do so in a natural manner, by conveying linguistic cues such as prosodic stress as well as paralinguistic information such as emotional state. Furthermore, there is a need to support task-specific speech styles, such as news, commercials, storytelling and warnings [4].

It has been shown that emotion, mood and sentiment affect attention, memory, performance, judgement and decision making in humans. Emotions are contagious, in a sense that a perceived emotional cue in an interlocutor can affect a person's emotions. Excitement for a product shown by an artificial speaker can make the human listener excited. A humorous voice and positive mood can affect the listener's mood [5], which has therapeutic implications.

Studies have shown that people already perceive the interaction with computers, robots and media in the same way as they perceive social communication with other people in real life. People are polite while communicating with computers, and react to male and female synthesized voices differently [6]. The importance of expressing emotions and attitudes in communication is shown in [7] since expressive robots are proven to be preferable over efficient ones. It is more comfortable to perceive an artificial agent as a real person than to think about the complexity and implications of a communicating machine. This innate human tendency to behave naturally in the interaction with computers should only be encouraged by improving the expressiveness of the synthesized voice.

These examples illustrate why speaker style adaptation is an area of very active research. Although earlier TTS techniques such as concatenative synthesis produce synthesized speech of good quality, it is impossible to adapt them to new voices or styles without recording and annotating large amounts of speech data for each new voice/style combination. For that reason, the concatenative approach has given way to parametric approaches, particularly deep neural networks (DNNs), which have been the focus of research of the majority of the scientific community for some time. Deep neural networks have benefited the most from the development of faster hardware and larger storage and bandwidth capabilities, since their performance does not reach a plateau as the amount of data increases, as is the case with other machine learning techniques [8]. They are also the first parametric method which gives results comparable with concatenative synthesis.

[*] Corresponding author: sinisa.suzic@uns.ac.rs

Clearly, an ideal TTS system should not only synthesize speech in a certain style or emotion but should also be able to deduce the target style or emotion based on the analysis of the input text. However, since the latter functionality is outside the scope of this text, the problems associated with it will be abstracted away and it will be assumed that the target style and emotion are known.

As is the case with neutral-style synthesis, emotional concatenative synthesis requires a fairly large amount of speech data for each target emotion [9], although there have been attempts to alleviate this problem by modifying F0 and duration curves using rules learned on small emotional databases [10]. Multiple speech styles and speaker emotions are much better handled by parametric approaches, owing to their flexibility [11]. A detailed analysis of concatenative and parametric approaches to emotional speech synthesis is given in [12]. However, the synthesis methods compared in that paper rely on a large emotional speech database consisting of approximately 100 minutes of speech per emotion. On the other hand, the main idea of this paper is to investigate the performance of different methods applied to multi-style DNN based synthesis in case when only on a small amount of speech data per style is available. The experiments were conducted on a speech database of American English, although it can be expected that the results are largely language independent.

The paper is organised as follows. The introduction is followed by an overview of single-style TTS based on DNN. Section 3 describes the methods used for speaking style modification in detail. Section 4 gives the details concerning the algorithms and models used, and presents the results of our experiments. The concluding section discusses the results and outlines the directions of further research.

2 DNN based TTS

The input text is firstly processed to extract linguistic features relevant for synthesis, which is referred to as *front end*. After that, the speech signal is produced from the phonetic transcription of the text augmented with the extracted linguistic features. The module charged with this task, referred to as *back end*, is based on two neural networks implemented using the *Merlin* toolkit [8].

The first network is used for modelling the durations of phonetic segments. The inputs and outputs of this network are phoneme aligned. The input for each hidden HMM state of the current phoneme is a binary vector containing linguistic information such as the identity of current phone and adjacent phones, number of words in the sentence, phones and syllables in a word, prosodic events indicated by ToBI tags [13], etc. The output is the vector of durations of the corresponding HMM state. State level durations are obtained using forced alignment proposed in *Merlin*. For all experiments in this paper, 5 HMM states per phoneme are used.

The second network is used for acoustic modelling. Its inputs and outputs are frame aligned. The input feature vector is the same as for the first network but extended with a number of frame and phone features such as the duration of the current phone, index of the current state, etc. In the training phase the target output is the vector of acoustic features extracted from wave files by the WORLD vocoder [14]. Acoustic features include MGCs (mel-generalised cepstral coefficients), BAP (band aperiodicities) and log F0, and they are further extended with their first and second derivatives as well as an additional flag indicating whether the current frame is voiced or unvoiced (V/UV). In all experiments in this paper, 40 MGCs, 1 BAP, 1 log F0 and 1 V/UV feature are used, yielding output feature vectors of length 127.

Both networks consist of 4 hidden layers with 1024 neurons each. The first three have tangent hyperbolic as the activation function, while the fourth layer is recursive and uses LSTM neurons. The output layer is linear. Objective function used is mean squared error. The input features are normalized to the interval [0.01, 0.99], while the output features are z-normalized. Networks are separately trained by backpropagation and stochastic gradient descent optimization.

In the synthesis phase, the outputs of the first network are provided to the second one, and the outputs of the second one represent target acoustic features. Smoothness of static features is achieved by using the maximum likelihood parameter generation algorithm [15], taking into account the predicted dynamic features. After the formants are further enhanced by post filtering, the acoustic features are fed to the WORLD vocoder in order to generate speech waveforms.

3 Methods for style conversion

In this chapter, three different approaches for the generation of synthesized speech in different speaking styles will be investigated. All of them have been originally proposed for efficient modelling of multiple speakers with a limited amount of data per speaker. The approach presented in 3.1 belongs to the group of methods for multi-speaker modelling, while the approaches presented in 3.2 and 3.3 can be considered as TTS adaptation methods.

3.1 Style code

The speaker code approach is presented in [16]. It is based on extending the input feature vector with information about speaker identity. Originally, this information is represented using a one-hot vector, and this concept is further extended in [17], by adding information about gender, age, etc. In [18], a similar approach is implemented, but instead of defining speaker codes, i-vectors are used. In the synthesis phase, one speaker voice is chosen by supplying the code of the desired speaker.

In case of multi-style modelling, styles are coded just as one-hot vectors. The input feature vector is extended with additional N binary features, one for each available style (angry, happy, etc.). Only one of the N features is set to 1, while all the others are set to zero (Fig. 1). In case of neutral style, all *N* features are set to zero.

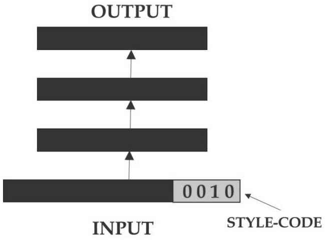

Fig. 1. Multi-style model with style codes.

The choice of a certain style is defined at word level since in natural speech a certain emotion does not need to extend over the entire utterance. A conventional DNN training is performed with input thus extended, both for the duration model and the acoustic model. At synthesis stage, the desired value is set for each word, the input is extended with a corresponding one-hot vector and the synthesized speech will have the characteristics of the desired style.

In multi-speaker modelling it is recommended to use equal amounts of data of each speaker in order to prevent the over-fitting of the final model to a specific speaker's voice. On the other hand, in multi-style modelling it is reasonable to assume the existence of much more speech data corresponding to the neutral style than to any other style. This is not necessarily a drawback and it can even be beneficial, since the speaker is unique for all styles, and many contexts not covered by a certain style can be compensated by the neutral style.

3.2 Separate output layer

Another approach of constructing a multi-speaker model is proposed in [19]. The idea is to have separate output layer(s) for each of the speakers in the database, while the hidden layers are shared between all speakers. This approach is based on the assumption that the shared layers are generally speaker independent and encode linguistic knowledge, while the output layer(s) encode the knowledge related to a particular speaker. This assumption is realistic having in mind the task of the NN in speech synthesis. Namely, NN transforms linguistic features to acoustic features. Linguistic features are extracted from text and are not dependent on any of the voice characteristics. For that reason, the lower layers can be considered as a speaker independent transformation of linguistic features. On the other hand, output acoustic features are speaker dependent and thus the output layer(s) should represent the acoustic space of each individual speaker.

Starting from this hypothesis in [19] is suggested that the shared layers should be trained on all available training data, while speaker dependant parts should be trained only on speaker-specific data. This can be achieved by propagating the error only through appropriate layers. The information about the speaker identity can be provided in different ways and it will be used for the selection of the output layer(s) through which the error will propagate during training, as well as for the activation of desired output layers at synthesis stage. Such an approach still requires a moderate quantity of data per speaker for the corresponding output layer to be trained adequately. Unfortunately, this requirement is rarely met in real-life scenarios.

However, Fan et al. also propose a method for the adaptation on a new speaker when very little data is available. In that case, they consider the speaker independent part of the network to be a global linguistic feature transformation, and one of the speaker dependent parts is used for adaptation to the new speaker. In the training phase only that part is updated using the limited speaker-specific data.

In case of multi-style modelling, the last layers are separated and there is one for each desired style, including neutral (Fig. 2). Since the output of the network is a vector of acoustic features, it is supposed that the last layer is sufficient to differentiate between styles. The network is trained in the same way, by back-propagating the error through the corresponding style-dependent output layer as well as shared layers.

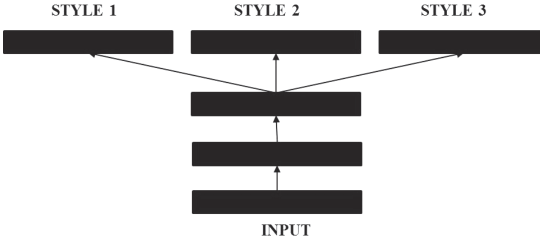

Fig. 2. Multi-style model with separate output layers.

One of the assumptions in the paper is limited amount of data per each style, insufficient to train the output layers from scratch, which would result in synthesized speech of poor quality. For this reason all outputs were firstly trained using neutral-style data. Afterwards only the output layers were trained using style-specific data by back-propagating the error through corresponding output layers only, keeping the weights in shared layers fixed.

3.3 Re-trained model

Although very intuitive and simple, this idea, to the best knowledge of the authors, has not been reported in literature yet, at least not with the aim of changing the speaker identity or speaking style. In [20] we firstly explained the idea in detail, but applied it to the adaptation to different speakers. If starting from model trained on a big database of speaker *A*, it has been shown that much less data of speaker *B* is needed to adapt the model to produce speech with characteristics of speaker *B* than when starting form randomly initialized model. The idea is based on the assumption that the model trained to any speaker has to be closer to the model for any other speaker of the same language than a randomly initialized model. Contrary to the idea presented in 3.2, the adaptation was applied to the entire network, not just the output layer(s).

The case of adapting the model trained on a big neutral-style database to another speaking style seems even easier than adapting it to a new speaker, since the model does not have to capture the characteristics of the speaker. This is in accordance with the fact that the voice of the same person in different emotional states exhibits far less variety than can be found in voices of different persons. The adaptation is completely straightforward. The initial model is trained on a big neutral-style database, and a small style-specific database is used for additional training of the model thus obtained. Both the duration model and the acoustic model are trained using conventional DNN training.

4 Results

For the synthesizers described in sections 3.2 and 3.3, 555 binary linguistic features were used, while for the synthesizer described in section 3.1 there were 3 more features related to the style (angry, happy and sarcastic). All synthesizers had the same architecture, except that in case of the synthesizer from 3.2 the single output layer was replaced with 4 style-dependent output layers. All other parameter values were as specified in Section 2.

For all experiments the same database of American English was used. It consists of 3 h of neutral speech and 5 minutes of speech for each of the 3 styles (3.5 h and ~7 min per style, if silences are included). The speaker is male, a native voice talent. For training the duration model, leading and trailing silences from each utterance were excluded and silences longer than 500 ms were shortened to 500 ms, while any remaining silent phonetic segment was used for training as is. Prior to training the acoustic model, all silences were shortened to 60 ms.

For the purposes of testing and comparing all synthesizers, two different listening tests were conducted among 20 amateur listeners, who are not native English speakers but are proficient in English. For each style, 5 sentences previously not seen during the training were synthesized to be used in the listening test.

4.1 MUSHRA test

The first test was a MUSHRA test [21] aimed at measuring the naturalness of synthesized utterances. It consisted of 15 questions (5 per style). In each question there was one reference utterance – an original recording of natural human speech, clearly indicated to the listener. Besides the reference utterance, there were 4 more utterances – one the same as the reference, and others synthesized by the synthesizers described in sections 3.1, 3.2 and 3.3 respectively – in a random order. Each subject had to rate the naturalness of each utterance, i.e. similarity to the reference utterance on a scale from 0 to 100. In this test the listeners were instructed to focus on the naturalness of style and intonation, and to disregard synthesis artefacts at segmental level. The results of this test are given in Figure 3.

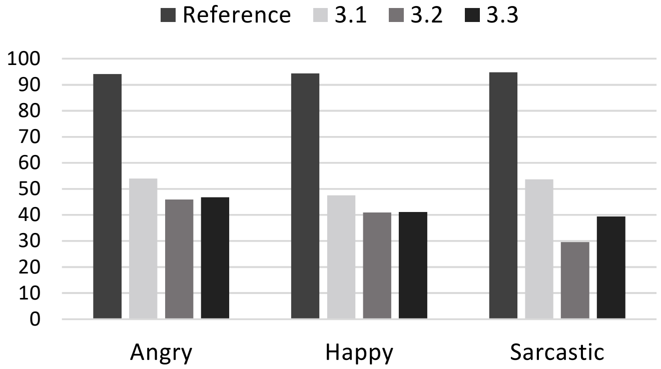

Fig. 3. Grades from MUSHRA test for angry, happy and sarcastic style, respectively.

The average grade for natural speech is 94, while the synthesizers described in sections 3.1, 3.2 and 3.3 were scored 52, 39 and 42, respectively. Based on these scores we can conclude that synthesizer 3.1 is significantly better than the other two, which achieved similar results. The angry style was consistently scored as the most natural-sounding one, regardless of the synthesis method used.

We have also investigated variations across utterances in order to check the consistency of each of the described methods. These results are given in Figure 4. It can be concluded that synthesizer 3.1 is not only the one with the highest grade, but also the most stable one. The range of the average grade is three times wider than the range for natural speech, but around 40% narrower than for other two synthesizers, and its value is 18, while the interquartile range is 10. The least stable synthesizer appears to be the synthesizer 3.2.

4.2 MOS test

The purpose of the second listening test was to measure the quality of synthesized utterances, with particular focus on the intelligibility and absence of synthesis artefacts. The test consisted of 3 groups of sentences. Each group represented one style and it consisted of 4 sets of utterances. One set consisted of original recordings, and other three were utterances synthesized by synthesizers described in 3.1, 3.2 and 3.3. In each of the 4 sets per style there were 5 sentences with identical textual content. The listeners were asked to rate the quality of each set independently on a scale from 1 (extremely bad) to 5 (extremely good). The results are presented in Figure 5.

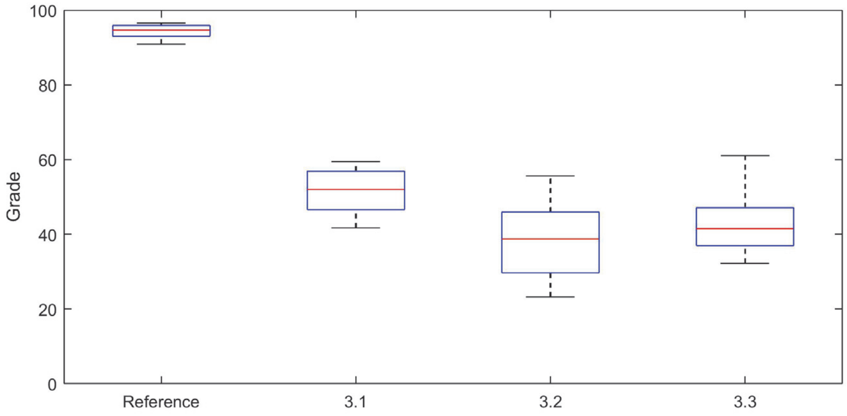

Fig. 4. Variations of grades across sentences.

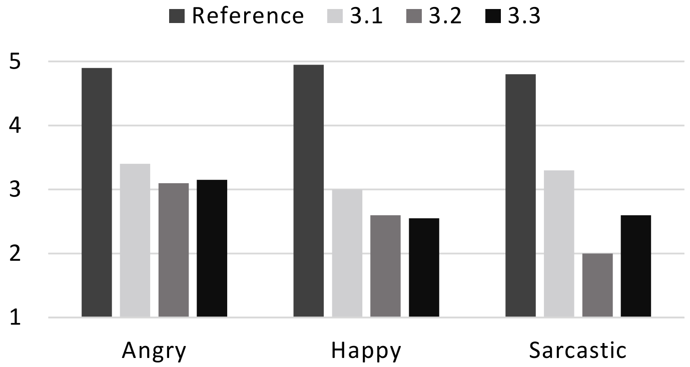

Fig. 5. Grades from MOS test for angry, happy and sarcastic style, respectively.

The average grade for natural speech was 4.9, while the sentences synthesized by the synthesizers 3.1, 3.2 and 3.3 were given grades respectively 3.2, 2.6 and 2.8 respectively. As in the previous test, the 3.1 synthesizer has shown to be the best. Further analysis reveals remarkable correlation with the results of MUSHRA test.

5 Conclusion

The paper presented methods that can make it possible to have machines which will be not only able to communicate, but also to express its virtual emotions and attitudes through speech. This will raise the level in HCI and make it a lot easier to communicate more naturally with call centres, appliances or robots.

Among the investigated methods the one based on style codes produced the speech of best quality. The re-training method and approach with separate output layers are comparable in terms of quality, although the latter is slightly more consistent but suffers from more audible artefacts. Among these three methods style-code approach is the only one in which the input feature vector is extended by additional information and which simultaneously updates all weights and biases regardless of whether expressive or neutral sentence features are used as inputs. In the other two approaches the nodes are updated with only a restricted amount of data for each style – retraining method updates all nodes in the network while in the approach with separate output layers, only the nodes from the output layer are updated. This suggests that it is beneficial to train the entire network with all databases simultaneously (with additional style-related information) than to tune the parameters of an already trained model with a small amount of new data. However, the main drawback of approach using style codes is that in case of adding a new voice, the entire model has to be re-trained from scratch, which can be extremely time-consuming due to the size of the database of neutral style speech. On the other hand, in case of other two approaches, the addition of a new style requires just the additional training on the small database representing the desired style, which is typically much faster, since style-specific databases are very small in comparison to the neutral style database.

As can be seen from Figures 4 and 5, the grades obtained in the two tests are quite similar. This is somewhat surprising, since in the first test the listeners were specifically instructed to disregard synthesis artefacts and focus on naturalness of the style and intonation, while in the second one they were asked to evaluate intelligibility, i.e. the absence of artefacts. This suggests that amateur listeners find it difficult to distinguish between different aspects of speech synthesis quality, and are very sensitive to synthesis artefacts. This conclusion has been confirmed by several TTS experts (with more than 10 years of experience in the field), who agreed that the degree to which the emotional states were conveyed was quite satisfactory, but objected that the synthesis is flawed by artefacts typical of systems trained with very small quantities of data. In order to prevent such situations, future experiments in this field should include a preliminary test intended to emphasize the difference between the intelligibility and the naturalness of synthesized speech to the listeners.

Our future research will investigate the possibility of separating multiple output layers as well as certain combinations of explained approaches, e.g. providing the style code in the training of shared layers within the approach described in Section 3.2 in order to get more stable and natural multi-style speech.

The research was conducted within the project "Development of Dialogue Systems for Serbian and Other South Slavic Languages" (TR32035) and "Interdisciplinary Research into the Quality of Verbal Communication" (OI178027), financed by the Ministry of Education, Science and Technological Development of the Republic of Serbia as well as the EUREKA project DANSPLAT (E!9944).

References

1. L.I. Ismail, S. Shamsuddin, H. Yussof, F.A. Hanapiah, N.I. Zahari, Proc. Eng, Robot-based Intervention Program for Autistic Children with Humanoid Robot NAO: Initial Response in Stereotyped Behavior, **41**, 1441–1447 (2012)
2. A. Csapo et al., Cognitive Infocommunications, IEEE 3rd International Conference, 667–672 (2012)
3. L. Gong, C. Nass, C. Simard, Y. Takhteyev, Usability Evaluation and Interface Design: Cognitive Engineering, Intelligent Agents, and Virtual Reality, **3**, 39–394 (2001)
4. M. Abe, Progress in speech synthesis, 495–510 (1997)
5. S. Brave, N. Clifford, The human-computer interaction handbook, 94–109. (2002)
6. B. Reeves, C. Nass, *The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places*, (Cambridge university press, 1996)
7. A. Hamacher, N. Bianchi-Berthouze, A.G. Pipe, K. Eder, 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 493–500 (2016)
8. Z. Wu, O. Watts, S. King, Proc. 9th ISCA Speech Synthesis Workshop (2016)
9. M. Bulut, S. Narayan, A. Syrdal, Proc. ICSLP 2002, 1265–1268 (2002)
10. J. Pitrelli et al., IEEE Trans. Speech Audio Process, **14**, 1099–1108 (2006)
11. M. Tachibana, J. Yamagishi, T. Masuko, T. Kobayashi, IEICE Trans. Inf. Systems, **89**, 1092–1099 (2006)
12. R. Barra-Chicote, J. Yamagishi, S. King, J.M. Montero, J. Macias-Guarasa, Speech Communication, **52**, 394–404 (2010)
13. M. E. Beckman, J. Hirschberg, S. Shattuck-Hufnagel, Prosodic Typology – The Phonology of Intonation and Phrasing, 9–54 (2005)
14. M. Morise, F. Yokomori, K. Ozawa, IEICE TRANSACTIONS on Information and Systems, **99**, 1877–1884 (2016)
15. K. Tokuda, T. Masuko, T. Yamada, T. Kobayashi, S. Imai, Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, **3**, 1315–1318 (2000)
16. N. Hojo, Y. Ijima, H. Mizuno, INTERSPEECH, 2278–2282 (2016)
17. H.T. Luong, S. Takaki, G.E. Henter, J. Yamagishi, Acoustics, Speech and Signal Processing (ICASSP), 4905–4909 (2017)
18. S. Yang, Z. Wu, L. Xie, Proc. Signal and Information Processing Association Annual Summit and Conference (APSIPA), 1–6 (2016)
19. Y. Fan, Y. Qian, F.K. Soong, L. He, Acoustics, Speech and Signal Processing (ICASSP), 4475–4479 (2015)
20. T. Delić, S. Suzić, M. Sečujski, D. Pekar, 17th International Symposium INFOTEH-JAHORINA (to be published)
21. E. Vincent, M. Jafari, M. Plumbley, Proc. UK ICA Research Network Workshop (2006)