Learning to Determine Who is the Better Speaker

Timo Baumann

Language Technology Institute, Carnegie Mellon University, Pittsburgh, USA
tbaumann@cs.cmu.edu

Abstract

Speech can be more or less likable in various ways and comparing speakers by likability has important applications such as speaker selection or matching. Determining the likability of a speaker is a difficult task which can be simplified by breaking it down into pairwise preference decisions. Using a corpus of 5440 pairwise preference ratings collected previously through crowd-sourcing, we train classifiers to determine which of two speakers is “better”. We find that modeling the speech feature sequences using LSTMs outperforms conventional methods that pre-aggregate feature averages by a large margin, indicating that the prosodic structure should be taken into account when determining speech quality. Our classifier reaches an accuracy of 97 % for coarse-grained decisions, where differences between speech quality in both stimuli is relatively large.

Index Terms: speech quality, likability ratings, sequence modeling

1. Introduction

Speaker traits (such as age or gender), emotional coloring (such as anger or distress), socio-cultural aspects (such as accent or dialects), conscious or subconscious coloring towards the addressee (such as friendliness or clarity), and other paralinguistic aspects are expressed through various prosodic, suprasegmental, segmental and non-segmental aspects of one’s speech and voice, where the combination of features is far from trivial. Together, they form the ‘quality’ of speech. It is important to note that no one ‘best’ combination of all features exists that would constitute ‘ideal’ speech, but that voice is a highly personal and subjective matter and that a multitude of combinations of these features result in a ‘good’ voice, which often makes likability comparisons hard. As a countermeasure, we limit our likability judgements to one specific reading genre: the reading of encyclopaedic entries in Wikipedia. Also, the combination of features is non-linear in that intermittent deficiencies (e. g. a lisp) or deviations limited to a few features (e. g. nasalisation) can have strong influences on the perceived quality.

We are interested in modeling speaker likability based on the aforementioned aspects of speech, using human annotations of pairwise *likability preference* ratings, i. e. the preference of one speaker over another given a particular speaking domain (in our case: the reading of encyclopaedic entries in Wikipedia) by multiple raters.

While likability is inherently subjective, *intersubjective* agreement on the abovementioned criteria can often by found by-and-large. We have shown in previous work [1] that consensus rankings can be created from inconsistent ratings with a high degree of stability. Such intersubjectively generalized likability rankings can be useful, for example to cast news speakers, readers or other speaking roles, or to select among publicly available material for training purposes (e. g. for speech synthesis). Another purpose of automated and explainable likability ratings would be the assessment and training of speakers to learn how to speak in a ‘likable’ way and improve one’s ability to communicate more effectively.

In previous work [2], speaker likability has been modeled using OpenSmile [3] features based on linear and non-linear aggregation functions (such as means and medians) to aggregate over the duration of the stimulus. Features were used to train classifiers such as SVMs which resulted in moderately high (but better than chance) performance in classifying speakers as above or below median likability [2]. The abovementioned aggregation functions cannot take into account the context of feature characteristics in the stimulus, and are unlikely to accurately express more fine-grained details relevant for speech quality (such as where and how a pitch accent is realized, beyond mean pitch). In the present paper, we use neural sequence-learning methods (in particular: LSTMs [4]) to encode the complex speech quality into a latent feature space and use the difference in these features for pairs of speech stimuli to train our classifier. To the best of our knowledge, we are the first to apply neural sequence learning to the task of speech quality estimation.

We use recordings from the Spoken Wikipedia[1] as a broad sample of read *speech in the wild*. The Spoken Wikipedia project unites volunteer readers who devote significant amounts of time and effort into producing read versions of Wikipedia articles as an alternate form of access to encyclopaedic content, read by a broad speaker population. It can thus be considered a valid source of speech produced by ambitious but not always perfect readers. The data has been prepared as a corpus [5] and the German subset of the corpus, which we use here, contains ~300 hours of speech read by ~300 speakers.

To simplify the human effort involved in creating a ranking, we have participants take pairwise decisions on which of two stimuli is better. We then create a ranking from the pairwise comparisons. The number of possible pairs grows quadratically with the number of the stimuli compared. Thus, while full comparisons for each rater are possible for small speaker groups (10 speakers → 45 rating pairs), these are infeasible for large speaker groups (225 speakers → 25000 rating pairs), in particular when relying on volunteer raters. Thus, we need a method that is able to build rankings from incomplete comparisons. Note, however, that many of the ratings (with one strong and one weak speaker) will have predictable outcomes and human input on speakers of similar quality is most informative.

The remainder of this paper is structured as follows: in the next section, we describe in detail the corpus that we use in our experiment as well as the evaluation sets and metrics we used. We describe the features that we extract for the stimuli in Section 3 and present the neural model architecture for speech quality preference ratings in Section 4. We present our experiments and discuss results in Section 5 and draw conclusions and sketch out avenues for future work in Section 6.

[1] https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Spoken_Wikipedia

2. Data and Evaluation Method

We use a speaker preference rating corpus[2] that has been collected previously with the goal of combining the ratings into one global speaker preference ranking [1]. The corpus contains 5440 pairwise preference ratings for 227 speakers. Speech stimuli are extracted from the Spoken Wikipedia Corpus [5] to best represent read *speech in the wild*; all stimuli contain (almost) the same read sentence[3] making them more easily comparable. Pairwise preference ratings have been collected with a web interface [6] through crowd-sourcing (voluntary/unpaid and hence not prone to vandalistic ratings); a total of 168 different raters participated in the rating experiment, coming from both genders, and diverse dialect and age groups of German speakers.

The original purpose of the rating collection was to create a ranking and effort was put into maximizing the efficiency of human annotation by focusing the human effort on 'difficult' pairs using *active sampling*: from initial ratings, a ranking was produced using Microsoft TrueSkill™ [7] and stimulus pairs for further ratings were stochastically sampled from the preliminary ranking in order to 'tease apart' most efficiently 'good' from 'bad' speakers and to focus human annotation effort on 'similarly good' speakers. As a result, the stimulus pairs that were rated by participants are much more 'difficult' than average stimulus pairs would be.

Given the focus of the data collection on acquiring ratings for 'difficult' pairs of stimuli, the data is more difficult to model than 'average' pairs of stimuli. As a result, inconsistency in the data set is high, as are pairs of stimuli that have been rated multiple times.

Previously, we have computed the minimum feedback arc set, i.e., the subset of ratings that lead to a fully consistent ranking [8]. We found the proportion of conflicting arcs to be 29 %, which can act as an indicator of the proportion of ratings that are inconsistent (where potentially different raters have different preferences, or simply cannot reliably tell the difference). In addition, we here compute an oracle correctness for all pairs that have been rated more than once, by checking for each rating, if it is the majority rating for this pair (deciding randomly to resolve draws). We find that such an *oracle classifier* reaches a correctness of only 65 % for those pairs that have been rated more than once. Pairs that were rated just once *potentially* are easier to classify, which makes it possible to beat this performance.

For evaluations, we report multiple settings below. The settings are meant to counter-balance the difficulties introduced by the data elicitation technique:

naïve we sample randomly among the evaluation instances from the corpus of human-rated pairs; as outlined above, the corpus focuses on difficult pairs, hence we cannot expect a spectacular performance;

easy based on the ranking in [1], we sample instances with 'large' ranking differences (distance on the ranking scale > 0.25 or > 0.5), in order to test if our classifiers fare better with stronger preference differences (and hence easier to identify differences in speech quality).

Given that stimuli were presented in random order, the data set is balanced in terms of which stimulus outperforms the other. Thus, we focus on accuracy as the only evaluation metric.

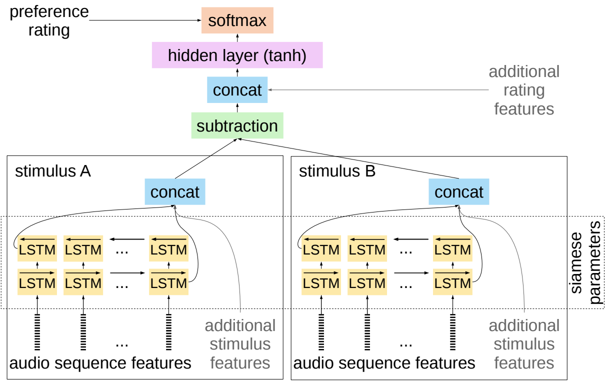

Figure 1: *Diagram of the neural architecture for speech likability preference. The task is symmetric (whether a stimulus is A or B is irrelevant) and hence the parameters for the LSTMs can be shared (siamese network). Additional features about stimuli or the rating can be conveniently concatenated in.*

3. Features and Conditions

Using a sliding window, we derive a multitude of local features from the audio stream that might capture aspects of speech quality. All features use a frame shift of 10 ms. In particular, we measure Mel-frequency cepstral coefficients (MFCCs, 12+1 energy) to capture voice and recording characteristics, f_0 (measured using `Snack`'s `esps` implementation) as a first measure of speech melody, and Fundamental Frequency Variation (FFV) features [9] as these are more robust (and might contain more valuable information) than single f_0. Using Praat [10], we compute jitter (PPQ5) shimmer (APQ5), and harmonics-to-noise ratio [11].

The Spoken Wikipedia Corpus also contains phonetic alignments that were computed using the MAUS tool [12]. The alignments allow us to assign phone annotations to every frame and using this information, the model is able to learn that phonetic characteristics (such as MFCCs) are conditioned on the phones spoken, making our input into the model much more expressive.

One frame of features for every 10 ms may overwhelm the model with very large amounts of parameters, reducing training efficiency as well as effectiveness. In order to keep training tractable, we subsample the feature frames with various values (see *seq. step size* in Table 1). When we do so, we use mean aggregation for numeric values (ignoring missing values for pitch).

We have not performed z-scale normalization on any of the parameters, although this would likely improve performance.

4. Model Architecture

The task of preference ranking is asymmetric in the sense that if the two stimuli to be compared are swapped, then the comparison result is the opposite. This has two consequences: (a) parameters for sequence analysis of both stimuli can be shared which is called a *siamese model* [13] and reduces the degrees of freedom of the model, making learning more efficient, and (b) the outputs from sequence analysis of each stimulus can simply be subtracted and the difference be subjected to a final decision layer as in logistic regression.

In our model and as shown in Figure 1, we use two layers

[2] Available as part of: `http://islrn.org/resources/684-927-624-257-3/`

[3] Expected reading: "Sie hören den Artikel *article lemma* aus Wikipedia, der freien Enzyklopädie." (You are listening to the article *article lemma* from Wikipedia, the free encyclopedia.); the article lemma itself is replaced with noise.

Table 1: *Meta parameters considered in grid search. Best values are shown in boldface.*

meta parameter	values
seq. step size	**5**, 10, 15
pho. embed size	8, **16**, 24
seq. state size	24, 32, **48**, 64
hidden layer size	2, **3**, 4 × seq. state size

of bidirectional LSTM to model the feature sequence of each stimulus and concatenate the outputs of the forward and backward pass. In the future, we could also concatenate additional stimulus-level features into the representation at this time, e. g. measures of signal quality such as ITU-T P.563 [14], or meta information about the speaker or the audio recording.

We subtract both stimuli's vectors as our final decision is based on the quality *difference* alone, not the overall quality. We then pass the difference to one hidden layer and a final binary softmax layer that models the preference decision. While we opted to not include additional meta features of the rating (such as identity, age or dialectal region of the rater), these could easily be concatenated in before the hidden layer, in order to model the relative preferences of individual raters or rater groups. For example, we previously found [1] that preferences differ by gender for both the rater as well as the rated speakers, making this a worthwhile endeavour for future work.

5. Experiments and Results

We separate out 400 of the 5440 ratings as the **naïve** test set and we sample among those ratings with 'large' differences 100 ratings each for the > 0.25 and > 0.5 **easy** test sets.

We implemented our network in dynet [15]. In the experiments reported below, we train for 50 epochs using AdamTrainer and report the results achieved after the final epoch. We concatenate the various audio features that are computed for every frame. We use embeddings to characterize the phonetic labels.

In preliminary experiments we found that subtraction of per-stimulus feature vectors performs much better than concatenation and that two-layer LSTMs outperform single-layer LSTMs (presumably because they can store context for longer).

5.1. Meta parameter optimization

We first performed an optimization to find good sizes for the various meta parameters of the model:

- To reduce the length of the sequence that need to be learned by the LSTMs (and to avoid the problem of vanishing gradients through long sequences), we subsample the audio features by mean-aggregating values over a number of frames (5, 10, or 15).
- To represent the discrete phonetic labels, we use embeddings of varying sizes (8, 16, or 24), in order to allow the model to cluster similar phones.
- The sequential LSTM state size determines how many dimensions can be considered during the sequence analysis and we experiment with various sizes (24, 32, 48, or 64).
- The output from concatenation of both forward and backward LSTMs doubles the size of the next layer's input. For the hidden layer size, we hence consider scaling factors (2, 3, or 4) over the size of the sequential state size.

We performed a grid search over the possible meta parameter values as summarized in Table 1 and focusing on the naïve data set. We found an optimum for sequence step size of 5 (i.e., one feature frame for every 50 ms of speech), phone embedding with 16 dimensions, sequence state size of 48, and hidden layer size of $3 \times 48 = 144$ (sequence state size of 32 and $4 \times 32 = 128$ was a close contender).

Table 2: *Accuracy (in percent) of full and reduced feature sets.*

setting	accuracy		
	naïve	easy-0.25	easy-0.5
full	67.25	93	97
w/o phones	58.75	73	80

At these settings, our model yields an accuracy of 67.25 % on the naïve test set, 93 % on the easy-0.25 test set and 97 % on the easy-0.5 test set. The accuracy on the naïve test set is close to what we estimated as the upper limit for the harder part of our training data.

5.2. Reduced Input Feature Set

We hypothesized above that our performance gain over previous work may be largely due to the model being able to relate prosodic parameters to the phones spoken. To test this hypothesis, we remove the phoneme embeddings from the input features. As shown in Table 2, we find performance to drop when phone identity is unavailable to help make sense of features.

6. Discussion and Future Work

We have presented a neural architecture for determining which of two speech stimuli is rated as the better of the two in noisy human annotations. Our model is able to make detailed use of sequential information, in particular to relate parameters to the phones spoken rather than more coarse-grained aggregation functions as have been used before.

In [2], the authors train classifiers to differentiate whether a stimulus is better/worse than average and reach a classification accuracy of 67.6 %. Their setup is comparable to our decisions for stimuli that are relatively far apart on the rating scale for which we achieve a classification accuracy of 93–97 %. We believe this to be caused by the better temporal modeling of our approach including the phonetic identities during aggregation.

Where our setup uses a 'fixed' speech stimulus for all comparisons, the spoken text in the compared stimuli differs in [2], which potentially makes their task harder. Our next steps will involve testing our approach on stimulus pairs that contain different texts. We can do this easily, as much more speech material for every rated speaker is available as part of the Spoken Wikipedia Corpus. For comparison, we also intend to use our likability encoding method for the corpus used in [2].

Despite the relatively good results, our method is still basic in terms of the neural architecture employed. In particular, our method does not yet employ an attention mechanism that could help to better aggregate the speech quality encoding. Given that both speakers in our corpus speak (more or less) the same content, we envision that our model would profit greatly if the comparison between both stimuli could attend to particular differences rather than only the comparison of the final BiLSTM output vectors. An attention model would also help the analysis of *why* a speaker is rated as better than another, as it would indicate the relative importance of parts of the stimulus in the comparison. Another venue, at least for comparisons on shared text would be connectionist temporal classification to temporally relate the feature streams before comparison for a better notion of timing differences between the stimuli. Finally, it might be worthwhile to pre-train the intermediate representations of the model.

In the end, our model could weigh slight mis-pronunciations against voice quality or prosodic phrasing, and we intend to use analysis techniques to ultimately understand the relative weights of these aspects in comparisons.

7. References

[1] T. Baumann, “Large-scale speaker ranking from crowdsourced pairwise listener ratings,” in *Proceedings of Interspeech*, 2017.

[2] F. Burkhardt, B. Schuller, B. Weiss, and F. Weninger, ““would you buy a car from me?”-on the likability of telephone voices,” in *Proceedings of Interspeech*. ISCA, 2011.

[3] F. Eyben, M. Wöllmer, and B. Schuller, “Opensmile: the munich versatile and fast open-source audio feature extractor,” in *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 2010, pp. 1459–1462.

[4] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[5] A. Köhn, F. Stegen, and T. Baumann, “Mining the Spoken Wikipedia for speech data and beyond,” in *Proceedings of the Language Resource and Evaluation Conference*, 2016.

[6] S. Kraft and U. Zölzer, “BeaqleJS: HTML5 and JavaScript based framework for the subjective evaluation of audio quality,” in *Linux Audio Conference*, 2014.

[7] R. Herbrich, T. Minka, and T. Graepel, “Trueskill™: A Bayesian skill rating system,” in *Advances in Neural Information Processing Systems 20*. MIT Press, January 2007, pp. 569–576.

[8] P. Eades, X. Lin, and W. F. Smyth, “A fast and effective heuristic for the feedback arc set problem,” *Information Processing Letters*, vol. 47, no. 6, pp. 319–323, 1993.

[9] K. Laskowski, M. Heldner, and J. Edlund, “The fundamental frequency variation spectrum,” in *Proceedings of FONETIK 2008*, 2008.

[10] P. Boersma, “Praat, a system for doing phonetics by computer,” *Glot international*, vol. 5, no. 9/10, pp. 341–345, 2002.

[11] ——, “Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound,” *Proceedings of the Institute of Phonetic Sciences*, vol. 17, pp. 97–110, 1993.

[12] F. Schiel, “MAUS goes iterative,” in *Proceedings of the LREC*, 2004.

[13] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, “Signature verification using a" siamese" time delay neural network,” in *Advances in Neural Information Processing Systems*, 1994, pp. 737–744.

[14] L. Malfait, J. Berger, and M. Kastner, “P.563 — The ITU-T standard for single-ended speech quality assessment,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 6, pp. 1924–1934, Nov 2006.

[15] G. Neubig, C. Dyer, Y. Goldberg, A. Matthews, W. Ammar, A. Anastasopoulos, M. Ballesteros, D. Chiang, D. Clothiaux, T. Cohn, K. Duh, M. Faruqui, C. Gan, D. Garrette, Y. Ji, L. Kong, A. Kuncoro, G. Kumar, C. Malaviya, P. Michel, Y. Oda, M. Richardson, N. Saphra, S. Swayamdipta, and P. Yin, “Dynet: The dynamic neural network toolkit,” *arXiv preprint arXiv:1701.03980*, 2017.