Experiments with One–Class Classifier as a Predictor of Spectral Discontinuities in Unit Concatenation

Daniel Tihelka[1(✉)], Martin Grůber[1], and Markéta Jůzová[2]

[1] NTIS – New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Czech Republic
{dtihelka,gruber}@ntis.zcu.cz

[2] Department of Cybernetics, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Czech Republic
juzova@kky.zcu.cz

Abstract. We present a sequence of experiments with one–class classification, aimed at examining the ability of such a classifier to detect spectral smoothness of units, as an alternative to heuristics–based measures used within unit selection speech synthesizers. A set of spectral feature distances was computed between neighbouring frames in natural speech recordings, i.e. those representing natural joins, from which the per–vowel classifier was trained. In total, three types of classifiers were examined for distances computed from several different signal parametrizations. For the evaluation, the trained classifiers were tested against smooth or discontinuous joins as they were perceived by human listeners in the ad–hoc listening test designed for this purpose.

Keywords: Speech synthesis · Unit selection · One–class classification · Concatenation cost · Speech parametrization · Spectral distance

1 Introduction

Although unit selection speech synthesis systems are still often preferred in the commercial sphere, according to [5] and our own experience, it is clear that heuristics–based approaches of unit selection features tuning basically fail. For example, papers such as [1,6,7,15,17,19,20,23–25] examined various concatenation cost features, but the results are rather inconsistent and sometimes even in contradiction. Therefore, instead of manual features tuning, we have started to examine machine–learning techniques for a data–driven automatic per–voice unit selection tuning.

One of the interesting ideas was introduced in [4], where the one–class classification (OCC) technique was used as a replacement for a classic spectral–related smoothness measure in concatenation cost computation. In [22], we tried to validate the results of the original research on our own speech database. In this paper, we present extended results, primarily focusing on parametrizations computed

A. Ronzhin et al. (Eds.): SPECOM 2016, LNAI 9811, pp. 296–303, 2016.
DOI: 10.1007/978-3-319-43958-7_35

from various speech signal framings and their impact on the ability of OCC to detect the joins of speech units where unnatural artefacts are perceived by humans.

2 One–Class Classification in Unit Selection

One–class classification [10,21], also known as *anomaly* or *novelty detection*, is used to address the problem of finding such occurrences in data that do not conform to expected behaviour. This is very advantageous and not yet widely used for unit selection speech synthesis, where usually large speech databases with natural recordings are available. However, it is common in this synthesis technique that unnatural disturbing artefacts may occur when incompatible units are concatenated. The reason is that the target and concatenation costs are generally designed to prefer units minimizing the trade-off of features evaluating *similarity to the requirements*, instead of reflecting whether the units will sound natural in the sequence they are used in. These artefacts, obviously not occurring in the source speech corpus, can thus be viewed as "anomalies" or "outliers". However, the occurrence of the artefacts can be considered as a random process (if they could be predicted, they can be avoided), which makes their collection and the reliable analysis of their causes rather difficult. Therefore, the existence of natural sequences and the unavailability of unnatural anomalies lead to the idea of exploring the abilities of OCC to detect, and thus to avoid, those anomalies.

2.1 Distances to Train the Classifiers on

For the initial experiment [22], we focus only on spectral continuity classification (following [4]) but using our Czech male speech corpus [3] containing approximately 15 h of speech, designed as described in [11,14].

To capture natural spectral transitions, for every two consecutive speech frames, with signal pre–emphasized by value 0.95 and Hamming–windowed, we computed *Euclidean* and *Mahalanobis* distances between MFCC vectors, *Itakura–Saito* distance between LPC coefficients and *symmetrical Kullback–Leibler* distance between spectral envelopes obtained from the LPC and between power FFT spectrum (referred to as "targets" or "references"); each distance vector thus consists of 5 values. Contrary to [4,22], however, we examined several different framings of the signals:

async 20/20 is the original scheme from the initial experiment, where the signal frames are 20 ms long without overlap (20 ms shift). Since we compute feature distances on rather stable vowel parts (see Fig. 1), it is supposed that the spectrum does not change very much within a particular phone. Thus, the natural transition of neighbouring frames should lead to rather small features distance, contrary to a spectral change perceived as an artefact.

async 04/25 is a scheme with frames 25 ms long, shifted by 4 ms. This scheme was chosen as it provides the most accurate automatic phone segmentation for this voice. The significant signal overlap, and thus accented spectral similarity of the consecutive frames, was assumed to emphasize the effect of natural and smooth signal transition pattern which the OCC is required to train.

async 12/25 scheme, having 25 ms long frames with 12 ms shift, was chosen as a compromise between large overlap (4 ms shift) and no overlap at all, while there is still slight preference towards frame overlapping.

psync pm/25 is a pitch–synchronous framing, where 25 ms long frames are centred around pitch–marks [8,9]. In this way, the MFCC, energy and F_0 are computed for the "classic" concatenation cost computation in our TTS system. Contrary to the previous schemes, the shift is always one pitch period long and the overlap varies dynamically as pitch changes. In unvoiced regions, the distances were not computed.

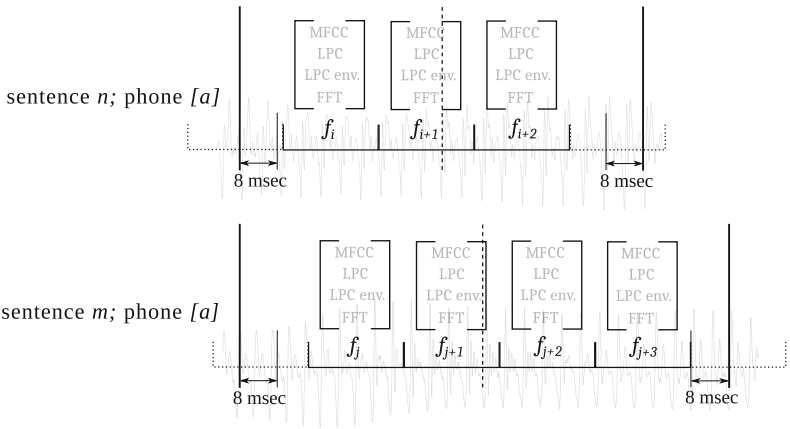

Fig. 1. The example of non-overlapped framing for two illustrative variants of phone [*a*] with phone boundaries and centre marked by bold and dashed, respectively, vertical lines. Feature vectors are outlined for each frame.

As already mentioned, we limit the experiment to vowels only, as unnatural artefacts are perceived more strongly due to their larger amplitude. Nevertheless, the extension to other voiced phones is planned as soon as reliable results are obtained.

For all the various signal framings, the target (natural) distance vectors used to train OCC were collected per–vowel from:

– all the consecutive frames covering the signal of the vowel, except frames spanning 8 ms at the vowel's beginning and end, i.e. for $(f_i, f_{i+1}), (f_{i+1}, f_{i+2})$ and $(f_j, f_{j+1}), (f_{j+1}, f_{j+2}), (f_{j+2}, f_{j+3})$ pairs from Fig. 1. By using of diphones in our TTS system, with boundaries approximately in the middle of the underlying phone, this exclusion allows us to avoid distances near phone (vowel) transitions in the training/testing set.
– the two consecutive frames nearest to the middle of each vowel, i.e. for (f_{i+1}, f_{i+2}) and (f_{j+1}, f_{j+2}) pairs from Fig. 1 — we will mark it as *mid.only* in Table 1. This might seem to be a natural choice reflecting the fact that

only signal around phone centre is examined for smoothness during diphones concatenation.

2.2 Evaluation of Real Concatenations

When using only (smooth) distances computed on the corpus data, we do not know much about how well a trained classifier is able to detect real non–continuous spectral transitions. Therefore, we created artificial join in the middle vowel of several words by concatenating two halves of the words from different parts of the corpus. Around the join, the distance was computed in the way that when $[a]$ from sentence m is to be concatenated with $[a]$ from sentence n (see Fig. 1), (f_{i+1}, f_{j+2}) vectors are used — f_{i+1} is nearest to the middle of the initial vowel half and f_{j+2} is the one after the vector nearest to the middle of the final vowel half. Each such distance was coupled with the listeners evaluation whenever a concatenation discontinuity is perceived in the word (further referred to as outlier distances) or not. Since details can be found in [22], we just summarize here that only examples where at least two of three listeners agreed were taken for further processing.

2.3 Classifiers Examined

Having obtained positive experience with OCC [12,13], we examined 3 classifier types, all implemented in *Scikit-learn toolkit* [16]. The first one is *Multivariate Gaussian distribution* (MGD), with all the distances modelled together in one go, tied through covariance matrix. The second one is *One-class SVM* (OCSVM), mapping distances into a high dimensional feature space via a kernel function, and iteratively finding the maximal margin hyperplane which best separates the training data from the outliers [18]. And the last one is *Grubbs' test* [2] modified as described in [12] to detect multidimensional distance vector as outlier when any of the individual features is detected outlying (GRT).

Prior the training, the whole per–vowel set of target distances was reduced to 4,000 randomly selected vectors, mostly due to speeding up the training process, but also to prevent potential OCC overfitting (see [22] for the total number of distances in *async 20/20*, which is the lowest of all used here). This reduced set was then further randomly split into 80 % for training distances targets and 20 % distances being held out for the final evaluation (see Sect. 3). From the training targets, 20 % were randomly chosen for 10–fold cross–validation. All the classifiers were trained to minimize F1 score, the details about parameters setup can be found in [22].

To further increase the robustness of the training, we added 50 % of the outlier distances (with discontinuity perceived, see Sect. 2.2) to the cross–validation process, if these were available for the corresponding vowel.

3 Results

Once the classifiers are trained, the 20 % of target corpus distance vectors and all the distance vectors for smooth joins evaluated by listeners (i.e. those without an

Table 1. The classification performance when the given number of target distances (for all the words without artefacts perceived) and the remaining 50 % of outlier distances (those not used for cross–validation), obtained by evaluations described in Sect. 2.2 and computed for the given framing, were passed to the classifier trained on the corresponding data. All the values are in %.

phones		a	e	i	o	a:	a	e	i	o	a:
		No. of examples to classify									
targets		60	45	30	50	17					
outliers		9	18	10	21	52					
		async 20/20					**async 20/20, mid.only**				
OCSVM	TPR	48.3	82.2	93.3	72.0	100.0	56.7	77.8	80.0	86.0	94.1
	TNR	55.6	33.3	50.0	23.8	0.0	44.4	33.3	80.0	0.0	96.2
	F1	62.4	78.7	88.9	70.6	73.9	68.7	76.1	85.7	75.4	91.4
MGD	TPR	100.0	95.6	96.7	98.0	100.0	100.0	75.6	96.7	100.0	88.2
	TNR	0.0	0.0	0.0	4.8	78.8	0.0	22.2	10.0	0.0	94.2
	F1	93.0	81.1	84.1	82.4	75.6	93.0	73.1	85.3	82.6	85.7
GRT	TPR	55.0	86.7	83.3	96.0	100.0	63.3	82.2	86.7	94.0	100.0
	TNR	44.4	33.3	40.0	4.8	53.8	33.3	27.8	70.0	0.0	82.7
	F1	67.3	81.2	82.0	81.4	58.6	73.1	77.9	88.1	79.7	79.1
		async 04/25					**async 04/25, mid.only**				
OCSVM	TPR	8.3	0.0	0.0	8.0	5.9	6.7	0.0	0.0	2.0	5.9
	TNR	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	F1	15.4	n/a	n/a	14.8	11.1	12.5	n/a	n/a	3.9	11.1
MGD	TPR	41.7	44.4	26.7	36.0	58.8	38.3	42.2	30.0	34.0	29.4
	TNR	100.0	88.9	90.0	95.2	100.0	100.0	88.9	90.0	95.2	100.0
	F1	58.8	59.7	41.0	52.2	74.1	55.4	57.6	45.0	50.0	45.5
GRT	TPR	53.3	24.4	63.3	44.0	29.4	46.7	17.8	23.3	38.0	17.6
	TNR	88.9	100.0	100.0	100.0	100.0	88.9	100.0	100.0	100.0	100.0
	F1	68.8	39.3	77.6	61.1	45.5	62.9	30.2	37.8	55.1	30.0
		async 12/25					**async 12/25, mid.only**				
OCSVM	TPR	56.7	44.4	63.3	58.0	100.0	45.0	37.8	50.0	50.0	64.7
	TNR	100.0	72.2	90.0	85.7	100.0	100.0	94.4	100.0	0.5	100.0
	F1	72.3	57.1	76.0	70.7	100.0	62.1	54.0	66.7	64.9	78.6
MGD	TPR	46.7	51.1	70.0	54.0	100.0	55.0	44.4	66.7	44.0	58.8
	TNR	100.0	72.2	50.0	66.7	84.6	55.6	88.9	80.0	90.5	92.3
	F1	63.6	63.0	75.0	64.3	81.0	68.0	59.7	76.9	59.5	64.5
GRT	TPR	65.0	51.1	73.3	66.0	100.0	68.3	46.7	56.7	72.0	88.2
	TNR	88.9	61.1	70.0	66.7	90.4	44.4	77.8	100.0	57.1	100.0
	F1	78.0	61.3	80.0	73.3	87.2	77.4	60.0	72.3	75.8	93.8
		psync pm/25					**psync pm/25, mid.only**				
OCSVM	TPR	38.3	37.8	43.3	46.0	58.8	30.0	22.2	40.0	30.0	11.8
	TNR	100.0	88.9	100.0	90.5	100.0	100.0	88.9	90.0	100.0	100.0
	F1	55.4	53.1	60.5	61.3	74.1	46.2	35.1	55.8	46.2	21.1
MGD	TPR	65.0	26.7	26.7	50.0	76.5	45.0	33.3	46.7	48.0	47.1
	TNR	55.6	88.9	80.0	95.2	92.3	100.0	88.9	60.0	90.5	96.2
	F1	75.7	40.7	40.0	65.8	76.5	62.1	48.4	58.3	63.2	59.3
GRT	TPR	53.3	48.9	70.0	52.0	100.0	63.3	35.6	63.3	56.0	100.0
	TNR	100.0	83.3	90.0	81.0	100.0	100.0	88.9	90	85.7	100.0
	F1	69.6	62.9	80.8	65.0	100.0	77.6	50.8	76.0	69.1	100.0

artefact perceived) were used to evaluate the ability of classifiers to recognize target distances never seen. Also, the remaining 50 % of outlier distances not used in cross–validation were used to enumerate the reliability of probable artefacts detection.

In Table 1 we present results for all the framings mentioned in Sect. 2.1 and all the classifiers described in Sect. 2.3. In the table, the abbreviation TP describes *true positives* (targets detected as targets) and TPR is then percentage of TP from all the targets to be classified (also called *recall*). Similarly, TN stands for *true negatives* (correct outliers classification) and TNR is its percentage (*specificity*). Due to space limitation, we exclude here vowels with a smaller number of examples to evaluate (both due to less joined words evaluated and lower mutual agreement of listeners on artefact absence/presence, see Sect. 2.2). Also, we do not present here the classification of the 20 % target distances being held out.

It can clearly be seen that the results are rather shuffled, with no significant preference for a framing and/or classifier type. In general, the *mid.only* variant behaves worse than when distances taken through the whole vowels are taken into account. Another surprising fact is that the larger overlap leads to worse results – although the distances to train are computed from very similar signals, the classifiers are not able to recognise outlier distances. It can be said that distances between non–overlapping frames are better in recognising targets, while distances between frames with large overlap recognise outliers instead. The best compromise seems to be *async 12/25*, for which *OCSVM* can reliably classify phone [*a:*] and rather successfully detect outliers for other phones as well.

Looking at raw F1 scores, most of the best results are for *async 20/20* framing, spread through various classifiers. However, taking for example phone [*a*] with F1 = 93 % (*MGD*), none of the 9 outliers was detected successfully. Similar situation is for [*i*] (F1 = 88.9 %, *OCSVM*), where only 5 out of 10 outliers were detected. From the point of view of unit selection, where the classifiers should finally be used, we would prefer reliable detection of outliers at the expense of higher FN (continuous joins classified as outliers). This would ensure that no audible artefact (or minimum of them) will appear in the synthesized speech. On the other hand, however, discarding wrongly classified smooth joins can easily lead to the inability of following the required target specifications (those with better match were discarded), which is not a desirable situation either.

4 Conclusion

Hopefully, we have shown that this alternative approach to feature hand–tuning may have its potential despite the fact that there is no ultimate answer to the question of what features/classifiers to use to avoid unnatural artefacts sometimes occurring in unit selection generated speech (neither did in [4]).

To address further research directions, it is important to start with an error analysis, i.e. to examine the causes of the classification failures. Our hypothesis for them is that the cause of the artefacts perceived is either due to a mismatch of non–spectral related features, or due to a spectral mismatch not covered well by the features and distance scheme computations chosen. Therefore, we need

to search for another set of features, not necessarily entirely spectral–related, which has a better capability of capturing the causes of artefacts perception — this may affect both concatenation and target cost features. And since the vowel joins evaluated by listeners (described in Sect. 2.2 and in details in [22]) were intentionally not limited with respect to spectral features anyway, they can be gradually extended and reused when searching for and experimenting with some other mismatch–describing features.

To make our results verifiable as well as to provide a solid springboard for prospective followers, we put all the data required to repeat the experiment on github under `ARTIC-TTS-experiments/2016_SPECOM/` repository. Also, more detailed results can be found there. Do not hesitate to contact us in case of any questions.

Acknowledgments. This work was supported by the Grant Agency of the Czech Republic, project No. GA16-04420S and by the grant of the University of West Bohemia, project No. SGS-2016-039. Computational resources were provided by the CESNET LM2015042 under the program "Projects of Large Research, Development, and Innovations Infrastructures".

References

1. Bellegarda, J.R.: A novel discontinuity metric for unit selection text-to-speech synthesis. In: Proceedings of the 5th Speech Synthesis Workshop (SSW5), pp. 133–138. Pittsburgh, PA, USA (2004)
2. Grubbs, F.E.: Procedures for detecting outlying observations in samples. Technometrics **11**, 1–21 (1969)
3. Hanzlíček, Z., Matoušek, J., Tihelka, D.: Experiments on reducing footprint of unit selection TTS system. In: Habernal, I. (ed.) TSD 2013. LNCS, vol. 8082, pp. 249–256. Springer, Heidelberg (2013)
4. Karabetsos, S., Tsiakoulis, P., Chalamandaris, A., Raptis, S.: One-class classification for spectral join cost calculation in unit selection speech synthesis. IEEE Signal Process. Lett. **17**(8), 746–749 (2010)
5. King, S.: Measuring a decade of progress in text-to-speech. Loquens **1**(1), e006 (2014)
6. Klabbers, E., Veldhuis, R.N.J.: Reducing audible spectral discontinuities. IEEE Trans. Speech Audio Process. **9**(1), 39–51 (2001)
7. Legát, M., Matoušek, J.: Analysis of data collected in listening tests for the purpose of evaluation of concatenation cost functions. In: Habernal, I., Matoušek, V. (eds.) TSD 2011. LNCS, vol. 6836, pp. 33–40. Springer, Heidelberg (2011)
8. Legát, M., Matoušek, J., Tihelka, D.: On the detection of pitch marks using a robust multi-phase algorithm. Speech Commun. **53**(4), 552–566 (2011)
9. Legát, M., Tihelka, D., Matoušek, J.: Pitch marks at peaks or valleys? In: Matoušek, V., Mautner, P. (eds.) TSD 2007. LNCS (LNAI), vol. 4629, pp. 502–507. Springer, Heidelberg (2007)
10. Markou, M., Singh, S.: Novelty detection: a review-part 1: statistical approaches. Signal Process. **83**(12), 2481–2497 (2003)

11. Matoušek, J., Romportl, J.: On building phonetically and prosodically rich speech corpus for text-to-speech synthesis. In: Proceeding of the 2nd IASTED International Conference on Computational Intelligence, pp. 442–447. ACTA Press, San Francisco (2006)
12. Matoušek, J., Tihelka, D.: Voting detector: A combination of anomaly detectors to reveal annotation errors in TTS corpora. Submitted to the Interspeech (2016)
13. Matoušek, J., Tihelka, D.: Anomaly-based annotation errors detection in TTS corpora. In: Proceedings of the 16th Annual Conference of the International Speech Communication Association (Interspeech 2015), pp. 314–318. Dresden, Germany (2015)
14. Matoušek, J., Tihelka, D., Romportl, J.: Building of a speech corpus optimised for unit selection TTS synthesis. In: Proceedings of 6th International Conference on Language Resources and Evaluation, LREC 2008. ELRA (2008)
15. Pantazis, Y., Stylianou, Y.: On the detection of discontinuities in concatenative speech synthesis. In: Stylianou, Y., Faundez-Zanuy, M., Esposito, A. (eds.) COST 277. LNCS, vol. 4391, pp. 89–100. Springer, Heidelberg (2007)
16. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
17. Přibil, J., Přibilová, A.: Evaluation of influence of spectral and prosodic features on GMM classification of Czech and Slovak emotional speech. EURASIP J. Audio Speech Music Process. **33**(3), 1–22 (2013)
18. Schölkopf, B., Platt, J.C., Shawe-Taylor, J.C., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Comput. **13**(7), 1443–1471 (2001)
19. Stylianou, Y., Syrdal, A.K.: Perceptual and objective detection of discontinuities in concatenative speech synthesis. In: Proceedings of the IEEE Acoustics, Speech, and Signal Processing (ICASSP), pp. 837–840 (2001)
20. Syrdal, A.K., Conkie, A.D.: Data-driven perceptually based join costs. In: Proceedings of the 5th Speech Synthesis Workshop (SSW5), pp. 49–54. Pittsburgh, PA, USA (2004)
21. Tax, D.M.J.: One-class classification: concept learning in the absence of counter-examples. Ph.D. thesis, Technische Universiteit Delft (2001)
22. Tihelka, D., Grůber, M., Matoušek, J., Jůzová, M.: Examining the ability of one-class classifier to ensure the spectral smoothness of concatenated units. Submitted to the 13th IEEE International Conference on Signal Processing (ICSP) 2016. If not accepted, the paper will be placed to github, under `ARTIC-TTS-experiments/2016_SPECOM/` repository where the experiment data are
23. Vepa, J.: Join cost for unit selection speech synthesis. Ph.D. thesis, The University of Edinburgh, College of Science and Engineering, School of Informatics (2004)
24. Vepa, J., King, S.: Kalman-filter based join cost for unit-selection speech synthesis. In: Proceedings of the EUROSPEECH 2003 - INTERSPEECH 2003. Proceedings of 8th European Conference on Speech Communication and Technology, pp. 293–296. ISCA (2003)
25. Vít, J., Matoušek, J.: Concatenation artifact detection trained from listeners evaluations. In: Habernal, I. (ed.) TSD 2013. LNCS, vol. 8082, pp. 169–176. Springer, Heidelberg (2013)