FINDING MAXIMUM MARGIN SEGMENTS IN SPEECH

Yago Pereiro Estevan[*] *Vincent Wan*[°] *Odette Scharenborg*[°]

Signal Theory and Communications Department EPS[*],
Universidad Carlos III de Madrid, Spain.
ypereiro@tsc.uc3m.es

Department of Computer Science[°],
University of Sheffield, UK.
{V.Wan,O.Scharenborg}@dcs.shef.ac.uk

ABSTRACT

Maximum margin clustering (MMC) is a relatively new and promising kernel method. In this paper, we apply MMC to the task of unsupervised speech segmentation. We present three automatic speech segmentation methods based on MMC, which are tested on TIMIT and evaluated on the level of phoneme boundary detection. The results show that MMC is highly competitive with existing unsupervised methods for the automatic detection of phoneme boundaries. Furthermore, initial analyses show that MMC is a promising method for the automatic detection of sub-phonetic information in the speech signal.

Index Terms— *speech processing, clustering methods, unsupervised learning.*

1. INTRODUCTION

Kernel methods have become increasingly prominent recently with the development of support vector machines (SVMs) [1] and their successful application in various fields. For example, SVMs have become an integral part of most state-of-the-art speaker recognition systems competing in the annual NIST evaluations [2]. In contrast, the use of kernel methods in other fields of speech processing, such as automatic speech recognition (ASR), is comparatively uncommon.

Maximum margin clustering (MMC) [3] is a relatively new and promising kernel method. It is of interest because of its close relationship to SVMs. MMC is a (semi) unsupervised form of SVM which determines the maximum margin dichotomy when (some or) no labels are specified: the two are related by the maximum margin criterion [1] for finding the optimum solution. Also, kernels developed for SVMs are immediately applicable to MMC. For example, using a sequence kernel developed for speaker verification [4, 5, 6] enables maximum margin speaker clustering and using a temporally discriminant sequence kernel developed for speech recognition [7] enables clustering of variable length speech segments. For the latter it is necessary to provide an initial segmentation of the speech signal. Since MMC and SVMs are closely related, it seems natural to use MMC to segment the speech for later reclassification by SVMs. However, before applying sequence kernels an evaluation of MMC's potential to segment speech is necessary.

In this paper, we examine the use of MMC for frame-level unsupervised speech segmentation using standard kernels. The obtained segmentation will be evaluated on TIMIT [8] by comparing the segment boundaries to the phone boundaries. One has to bear in mind though, that MMC is a speech segmentation method, but the evaluation is on the level of phonemes. Our goal is not strictly phoneme segmentation, but the segmentation of speech into clusters that may be classified later using SVMs.

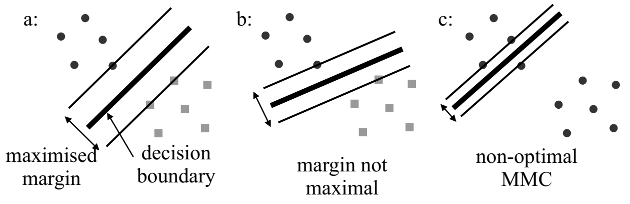

Fig. 1. The maximum margin criterion applied to SVMs and MMC.

The remainder of the paper is as follows. Section 2 describes MMC. Section 3 describes the material used. The segmentation methods and their results are presented in Section 4, and discussed in Section 5. The paper concludes with the most important findings and a brief outlook on future research.

2. MAXIMUM MARGIN CLUSTERING

The principle underlying SVMs is the maximum margin criterion. It states that if a linear decision boundary is to be placed between two separable classes then the optimum position is located exactly mid-way between the two such that the shortest distances from the boundary to the nearest points of each class are equal and maximal. Such an optimal decision boundary is illustrated in figure 1a, whereas figure 1b illustrates a non-optimal decision boundary. The empty region bounded by the two lines running parallel to the decision boundary between the two classes is called the margin and should have maximal width, i.e., it should be as wide as possible while remaining empty. The SVM formulation extends this principle to the non-separable case by penalising incursions into a so-called *soft-margin* and the goal then is to maximise the soft-margin while minimising the penalties.

Maximum margin clustering (MMC) developed by [3] employs the same underlying principle. The difference between SVMs and MMC is as follows. In SVMs the goal is to find the decision boundary that maximises the margin given a set of input vectors and their corresponding cluster labels. This task is illustrated in figures 1a and 1b where the task is to find the optimal decision boundary that maximises the margin between the black dots and the grey squares. Hence SVMs are discriminative classifiers that are trained in a supervised manner. In contrast, the goal of MMC is to find the cluster label assignments given the input vectors such that the margin

This work was undertaken while Yago Pereiro Estevan was on placement in the Speech and Hearing Research Group, Department of Computer Science, University of Sheffield, UK, supported by the University Carlos III of Madrid. Vincent Wan was partly supported by the European Union 6th FWP IST Integrated Project AMI (Augmented Multi-party Interaction, FP6-506811, publication AMI-206). Odette Scharenborg was supported by the Netherlands Organization for Scientific Research (NWO). The authors would like to thank Ascensión Gallardo Antolín for arranging funding for Yago Pereiro Estevan.

1-4244-0728-1/07/$20.00 ©2007 IEEE

between the two resulting classes is maximal. This is illustrated in figure 1c where the task is to find the optimal decision boundary between the two sets of black dots. The proposed boundary in figure 1c is non-optimal, the boundary should be placed as is done in figure 1a. Therefore, MMC is an unsupervised algorithm which can be used to dichotomise a set of feature vectors. The MMC optimisation problem can be formulated as a semidefinite programming problem and our implementation closely follows that outlined in [3]. It is interesting to note that MMC can also be used in a semi-supervised setting in which some of the points are labelled. This leads to a constrained form of MMC which might be useful when handling temporally ordered data such as speech. However, this property is not exploited in this study.

The application of MMC to divide a speech signal into segments separated by maximum margin is relatively straight-forward and is described in section 4. Such MMC segments may be useful on multiple levels. For example, for the analysis of two consecutive phonemes. Alternatively, MMC can be used to analyse a single phoneme such as a long vowel in which case it might be able to detect fine sub-phonetic detail [9]. Thus the method may provide valuable insights into our understanding of speech.

3. MATERIAL

The speech used in this study is taken from the TIMIT corpus [8]. TIMIT consists of reliably hand labelled and segmented data of quasi-phonetically balanced sentences read by native speakers of eight major dialect regions of American English. Of the 630 speakers in the corpus, 438 (70%) were male. We used TIMIT's predefined test set, consisting of 1,344 utterances (the sa sentences are excluded). Note that in our experiments the silence part (i.e. the closure) of the stop consonant is merged with the release part of the stop consonant into a single segment.

The speech was parameterised with 12 MFCC coefficients and log energy, augmented with their first and second derivatives resulting in 39-dimensional MFCC vectors. The MFCC were computed on windows of 15 ms, with a 5 ms frame shift, and cepstral mean and variance normalisation was applied.

4. SEGMENTATION AND RESULTS

To perform a frame-level segmentation of speech, a sliding window, which is N frames wide, is applied to the parameterised speech signal. From initial experiments a value of $N = 18$ was determined to yield the best results. MMC using an RBF kernel (with a width of 200 determined using a small development set) is applied to the frames inside the window and a set of cluster labels is obtained. The window is then shifted by one frame and the process is repeated across a whole utterance. The results of the analysis are shown in figure 2a. The x-axis represents the time of the frame at the centre of the sliding window. Each column of the graph corresponds to a window centred on a different frame so adjacent columns correspond to windows centred on adjacent frames. Elements at the bottom of a column occur earlier than elements at the top. The shading of each element indicates the cluster label assigned to each frame so a change in the shading corresponds to a potential boundary. The TIMIT phoneme label transcription is marked on the x-axis.

Segment boundaries can be seen by comparing the cluster label assignments across the columns. Boundaries that are well defined should shift downwards in subsequent columns leading to diagonal structures in the graph. An example of a phoneme boundary is highlighted in rectangle B of figure 2a: the maximum margin segmentation found by MMC coincides with the hand labelled TIMIT boundary marked on the x-axis.

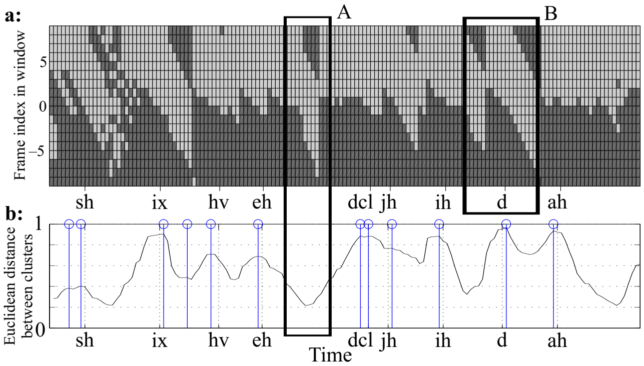

Fig. 2. *a: Sliding window clustering representation; each column shows the cluster label assignments. b: Euclidean distance between cluster means; the detected boundaries (using $\delta = 0.001$) are indicated by the solid vertical lines. The x-axis shows the TIMIT boundaries (dashed vertical lines) of the phrase "She had your dar[k]".*

Section 4.1 describes the metrics used to evaluate the segmentation quantitatively with respect to the TIMIT phoneme labels. A method of finding potential segmentation points by detecting diagonal structures in the graph is described in section 4.2. Section 4.3 describes an alternative method of finding segmentation points based on the Euclidean distance between the clusters. A combination of the two approaches is described in 4.4.

4.1. Evaluation metrics

Firstly, detected boundaries will not generally coincide exactly with manually transcribed phoneme labels. Thus, following [10] a boundary is considered to be correctly detected if the hypothesis and the manual transcription are within 20ms of each other.

Four metrics are used to evaluate the segmentation. The *correct detection rate* (*c.d.r.*) is defined as,

$$c.d.r. = \frac{\text{Total number of correct boundaries detected}}{\text{Total number of true boundaries}} \tag{1}$$

which is a measure of the proportion of the true boundaries detected. A related metric is the miss rate ($m.r.$) which is defined as $m.r. = 1 - c.d.r$ and indicates the proportion of true boundaries that were not detected.

Over-segmentation ($o.s.$) [11] gives an indication of how many segments were hypothesised compared to the actual number of segments.

$$o.s. = \frac{\text{Total number of boundaries found}}{\text{Total number of true boundaries}} - 1 \tag{2}$$

An $o.s. = 0$ indicates that the number of hypothesised segments equals the number of true boundaries. Expressed as a percentage, an $o.s. = 100\%$ means that there are twice as many hypothesised segments as there are true segments. A negative value indicates too few segments were found.

The last metric used is the *false alarm rate* (*f.a.*), which indicates the proportion of boundaries that were incorrectly detected:

$$f.a. = 1 - \frac{\text{Total number of true boundaries found}}{\text{Total number of boundaries found}} \tag{3}$$

mask size	2×1	2×2	4×3
c.d.r. (%)	81.6	59.4	32.4
m.r. (%)	19.4	40.6	67.6
f.a. (%)	67.2	50.3	38.4
o.s. (%)	195.2	39.5	−46.2

Table 1.
Boundary detection performances for different mask sizes.

4.2. Detecting structures (MB)

A mask based (MB) method is used to detect the diagonal structures in figure 2*a*. The mask is an $n \times m$ matrix that is divided along its diagonal into two: each element in the upper right triangle must match the lighter shaded elements of the graph while the elements in the lower left triangle must match the darker shade. One mask is slid across the graph so that the top row of the mask is at the current frame of each column and another (inverted mask) is slid across the top of the graph. The total number of matching elements in the mask is counted each time. When all of the mask's elements are matched then a segment boundary is marked at the time corresponding to the frames along the mask's diagonal.

Table 1 shows the results of phoneme boundary detection on the TIMIT test using different mask sizes. The smallest mask is most sensitive as it looks only at one column whereas larger masks look for consistent boundaries across multiple columns. As is shown by Table 1, larger masks lead to much lower *c.d.r.* as not all structures are as large as those illustrated in figure 2*a*. Consequently, larger masks also led to very low *f.a.* and *o.s.*. Marking boundaries when there was only a partial match was considered. However, this led to an increase in the number of false alarms without a corresponding decrease in the miss rate.

4.3. Segmenting by Euclidean distance (ED)

The structures in figure 2*a* are quite complex and the MB approach is not very robust. An alternative method of determining whether a segment boundary is detected is to calculate the distance between the assigned clusters of a window. When the distance peaks, a boundary is detected. Since this is a maximum margin approach, our first instinct is to use the margin distance. However, this measure was found to be noisy and a relatively poor indicator of a segment boundary: although there may be two distinct clusters the margin separating them can vary significantly depending on the relative positions of the points on the edge of the margin. A more reliable and stable measure is to use the Euclidean distance between the mean vectors of the clusters (referred to as ED). The graph of these Euclidean distances is shown in figure 2*b*. A simple peak detector that registers peaks only if the value has changed by a value greater than a threshold δ is used to find local maxima in the Euclidean distance. These maxima are marked in the graph by the solid vertical bars, the TIMIT phoneme boundaries are marked by the dotted vertical bars. The peaks clearly line up well with the TIMIT phoneme labels for this particular utterance.

Figure 3 shows a graph of miss rate (x-axis) plotted against over-segmentation (y-axis). The various points on the curve are obtained by varying the threshold δ in the peak detection algorithm. Figure 3 shows that at no over-segmentation the miss rate is 25% (equivalently 75% of the phoneme boundaries are correctly detected).

Figure 4 plots miss rate against false alarm (an inverted ROC curve). In fact this method is unable to detect all of the TIMIT phoneme boundaries: missing about 8% with a 60% false alarm. When no false boundaries are detected the method detects approximately 25% of the phone boundaries. Comparing the *f.a.* and *m.r.* in Table 1 for the various mask sizes with the results plotted in Figure 4 clearly shows that the ED method outperforms MB.

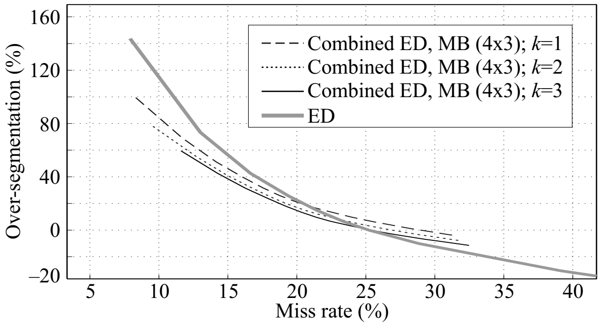

Fig. 3. *Miss rate (x-axis) plotted against over-segmentation (y-axis) for varying δ, for ED and COMB.*

4.4. Combined approach (COMB)

It is interesting to combine the above approaches to determine whether searching for structures gives additional information over the ED method. The detected segments from each method are combined using a "soft" OR operator in which boundaries from the two methods within k frames of each other are combined and replaced by a single boundary located at the mean of the two.

In this approach (referred to as COMB), a 4×3 mask was combined with the Euclidean distance segments. The reason for using a 4×3 mask is that it introduces a low number of false alarms, which makes it well suited to investigate whether MB finds boundaries that ED does not. Figures 3 and 4 show the results of COMB for different distances of *k* (1, 2, or 3 frames) between the boundaries hypothesised by MB and by ED.

The figures show that for lower miss rates, COMB gives fewer false alarms compared to the ED method. Thus MB does indeed find correct boundaries that are not found by ED, without introducing new false alarms.

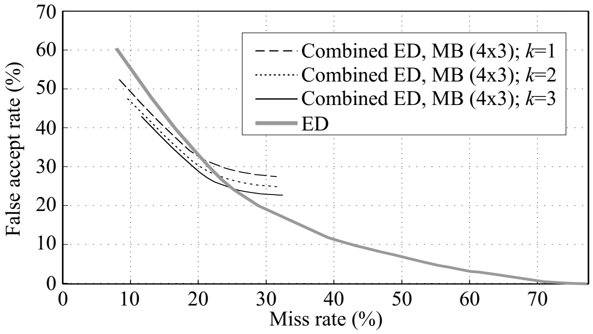

Fig. 4. *Miss rate (x-axis) plotted against false alarm (y-axis) for varying δ, for ED and COMB.*

5. DISCUSSION

The $o.s.$ results in figure 3 and the $f.a.$ results in figure 4 show that combining MB and ED yields the best performance. With an $o.s.$ rate of 0%, 76.0% of the phoneme boundaries are detected correctly, increasing to 90.3% $c.d.r.$ when allowing 75.5% $o.s.$ This compares well with results found in the literature. [10] obtained a $c.d.r.$ of 73.6% with an $o.s.$ of 0%, increasing to 90.0% $c.d.r$ with an $o.s.$ of 63.0% on a subset of 480 utterances from TIMIT. [12] obtained a $c.d.r.$ of 85.9% (they did not report *o.s.*) on the full TIMIT test set while using a supervised method (this in contrast to our and [10]'s method which are unsupervised). Even though the methods proposed here and by [10] are different, the results are strikingly similar. This might suggest that there may be an upper limit on the accuracy of unsupervised automatic detection of phone boundaries.

Figure 4 shows that with ED a 25% $c.d.r.$ can be obtained at no $f.a.$. Combining ED with MB into COMB however leads to an increased $f.a.$ rate. The difference between the $f.a.$ rates for COMB and ED indicates the number of additional boundaries introduced by MB. The Euclidean distance is low for some of the boundaries hypothesised by MB, indicating that the MFCCs on either side of the hypothesised boundary are very similar. This suggests that both sides of the hypothesised boundary belong to the same phoneme. Since MB hypothesises a boundary, this might indicate that there is information in the speech signal on a sub-phonetic level; for an example of such a boundary see structure *A* in figure 2.

We further analysed the *A* structures. Of the 5,827 times such a structure occurred, 35.7% were related to vowels, 17.1% to fricatives, and 14.2% to plosives, the rest were distributed over the other consonant classes and silence. The high percentage for vowels is not surprising considering the coarticulation effects occurring during sound production. During the production of one sound, articulatory features belonging to the preceding or following sound may spread into that sound. Looking more closely at the case where a boundary is hypothesised in the middle of a vowel segment shows that in 36.4% the following TIMIT phoneme label is a plosive, while 25.6% is followed by a nasal. These preliminary results show that the method proposed in this paper is indeed very good at capturing sub-phonetic detail. This is an interesting area for further research.

The automatic detection of sub-phonetic information is also getting increasing attention in the field of ASR. Since 1999, it has been proposed to move away from the standard 'beads-on-a-string' (i.e. phoneme-based) recognition paradigm [13]. One of the proposals of such a new system is based on the modelling of articulatory features (AFs) [14, 15]. Since MMC can extract sub-phonetic information it is an interesting method for the development of a kernel based ASR system that is based on this sub-phonetic AF information. In this paper, the parameter settings were optimised for the automatic segmentation of phonemes. However, different parameter settings will result in the detection of even more detailed information in the speech signal.

6. CONCLUDING REMARKS AND FUTURE WORK

In this paper we have presented a novel application of MMC to the task of unsupervised speech segmentation. It is a first step towards the ultimate goal of building a kernel based ASR system. MMC's potential with respect to the automatic segmentation of speech is evaluated on TIMIT. The results in sections 4 and 5 have shown that MMC is highly competitive with existing unsupervised methods for the automatic detection of phoneme boundaries. Although MMC has been evaluated in terms of *phoneme* boundary detection, it is in fact a *speech* segmentation method. To achieve our ultimate goal, future work will refine then analyse the speech segments. The segements can be reclustered and classified by sequence kernel approaches. The preliminary results in section 5 also show that MMC is a promising method for the automatic detection of sub-phonetic information in the speech signal.

Finally, we should note that this work is based on an earlier version of MMC: there now exists a newer formulation that can handle multiple classes [16].

7. REFERENCES

[1] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2(2), pp. 1–47, 1998.

[2] National Institute of Standards and Technology, "The NIST year 2006 speaker recognition evaluation plan," 2006.

[3] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, "Maximum margin clustering," *Proc NIPS*, 2004.

[4] J. Louradour and K. Daoudi, "Conceiving a new sequence kernel and applying it to SVM speaker verification," *Proc of Interspeech*, pp. 3101–3104, 2005.

[5] V. Wan and S. Renals, "Speaker verification using sequence discriminant sequence kernels," *IEEE Transactions of Speech and Audio Processing*, vol. 13(2), pp. 203–210, 2005.

[6] W. M. Campbell, "Generalised linear discriminant sequence kernels for speaker recognition," *Proc ICASSP*, pp. 161–164, 2002.

[7] V. Wan and J. Carmichael, "Polynomial dynamic time warping kernel support vector machines for dysarthric speech recognition with sparse training data," *Proc of Interspeech*, pp. 3321–3324, 2005.

[8] J. S. Garofolo, "Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database," *National Institute of Standards and Technology (NIST)*, 1998.

[9] S. Hawkins, "Roles and representations of systematic fine phonetic detail in speech understanding," *Journal of Phonetics*, vol. 31, pp. 373–405, 2003.

[10] G. Aversano, A. Esposito, A. Esposito, and M. Marinaro, "A new text-independent method for phoneme segmentation," *Proc the 44th IEEE Midwest Symposium on Circuits and Systems*, vol. 2, pp. 516–519, 2001.

[11] B. Petek, O. Andersen, and P. Dalsgaard, "On the robust automatic segmentation of spontaneous speech," in *Proc. ICSLP '96*, Philadelphia, PA, 1996, vol. 2, pp. 913–916.

[12] B. L. Pellom and J. H. L. Hansen, "Automatic segmentation of speech recorded in unknown noisy channel characteristics," *Speech Communication*, vol. 25 (1-3), pp. 97–116, 1998.

[13] M. Ostendorf, "Moving beyond the beads-on-a-string model of speech," in *Proc IEEE ASRU*, Keystone, CO, 1999, pp. 79–84.

[14] K. Kirchhoff, *Robust speech recognition using articulatory information*, Ph.D. thesis, University of Bielefield, 1999.

[15] S. King and P. Taylor, "Detection of phonological features in continuous speech using neural networks," *Computer Speech and Language*, vol. 14, pp. 333–353, 2000.

[16] L. Xu and D. Schuurmans, "Unsupervised and semi-supervised multi-class support vector machines," *AAAI*, pp. 904–910, 2005.