Probabilistic Kernels for Improved Text-to-Speech Alignment in Long Audio Tracks

Germán Bordel, *Member, IEEE,* Mikel Penagarikano, *Member, IEEE,*
Luis Javier Rodríguez-Fuentes, *Member, IEEE,* Aitor Álvarez, Amparo Varona, *Member, IEEE*

Abstract

The synchronization of text transcripts with audio tracks is typically solved by forced alignment at the phonetic level. However, when dealing with either very long audio tracks or acoustically inaccurate text transcripts, more complex methods are needed, usually based on heavy and costly ASR systems. In a previous work, we showed that a simple and lightweight method could be effectively applied, based on a free phonetic decoding of the speech signal and the alignment of the free and reference phonetic sequences, allowing the transfer of timestamps from the former to the latter. This method has yielded competitive results on the Hub4-97 dataset and is currently applied to synchronize the videos and minutes of the Basque Parliament plenary sessions.

In this paper, probabilistic kernels (similarity functions) are applied, based on the hypothesis that a confusion matrix computed from a large corpus of speech conveys key information about the behavior of the phonetic decoder, and that the probabilistic interpretation of this information may help design informative kernels leading to improved alignments. The probabilistic kernels proposed in this work outperform our baseline kernels and other alternatives, including a reference ASR-based approach and a knowledge-based kernel, in experiments on the Hub4-97 dataset.

G. Bordel, M. Penagarikano, L.J. Rodríguez-Fuentes and A. Varona are with the Department of Electricity and Electronics, University of the Basque Country, UPV/EHU, 48940, Leioa, Spain (e-mail: {german.bordel, mikel.penagarikano, luisjavier.rodriguez, amparo.varona}@ehu.es)

A. Álvarez is with Human Speech and Language Technologies, Vicomtech-IK4, San Sebastián, Spain (email: aalvarez@vicomtech.org)

This work has been supported by the University of the Basque Country, under grant GIU13/28.

Index Terms

probabilistic kernel, text-to-speech alignment, long audio tracks.

I. Introduction

The need to technically address accessibility issues is making video captioning an increasingly demanding area for speech and language technologies. In 2007, the European Parliament established new rules for the audiovisual industry that posed a significant pressure to the video producers and broadcasters. At that time, there was no economically reasonable solutions to fit the new exigence levels: manual processing would be too intensive and costly, whereas automatic technology was not applicable due to the low quality of automatic speech recognition (ASR) techniques when applied without restrictions. In these situations, a mixed approach can be applied by manually producing a text transcript and then automatically aligning audio and text. In fact, this problem had been already addressed in the late nineties by several authors [1][2]. Shortly afterwards, the method proposed in [3] achieved good quality results, provided that relatively clean speech segments and accurate text transcripts were available. The approach worked by first detecting fairly good matches between the text and the output of an ASR system customized to the transcript, and then recursively processing smaller portions of the problem. More recently, other related works have been reported which address different objectives, such as correcting human transcripts [4], or dealing with highly imperfect transcripts [5]. In [6], an open source toolkit was presented based on a complex strategy similar to [3]. More recently, different approaches aiming to also obtain simple alignment procedures have been proposed [7] [8] [9], mostly related to the generation of resources for training ASR systems.

The method proposed in [3] was really successful but required a large amount of resources and was computationally costly. In [10], the same author presented a different approach, where the segmentation of long sequences was performed at the acoustic level, rejecting noisy regions, and focusing on the nearly-forced alignment of the selected portions. In [11][12], we applied an unconstrained phonetic decoder to a long speech signal and aligned the recognized sequence of phones to the phonetic transcript derived from the reference text (see Figure 1). The accuracy figures obtained in word-level alignment experiments on the widely known Hub4-97 dataset [13] were only slightly lower than those reported in [3]. Being fast and comparatively light in terms of both computational costs and required resources, we have been using this system since 2010 to add subtitles to the videos of plenary sessions of the Basque Parliament. These sessions include speech in both Spanish and Basque, and speakers switch frequently from one to the other. Transcripts come from the official minutes, which are clean versions of the speech actually

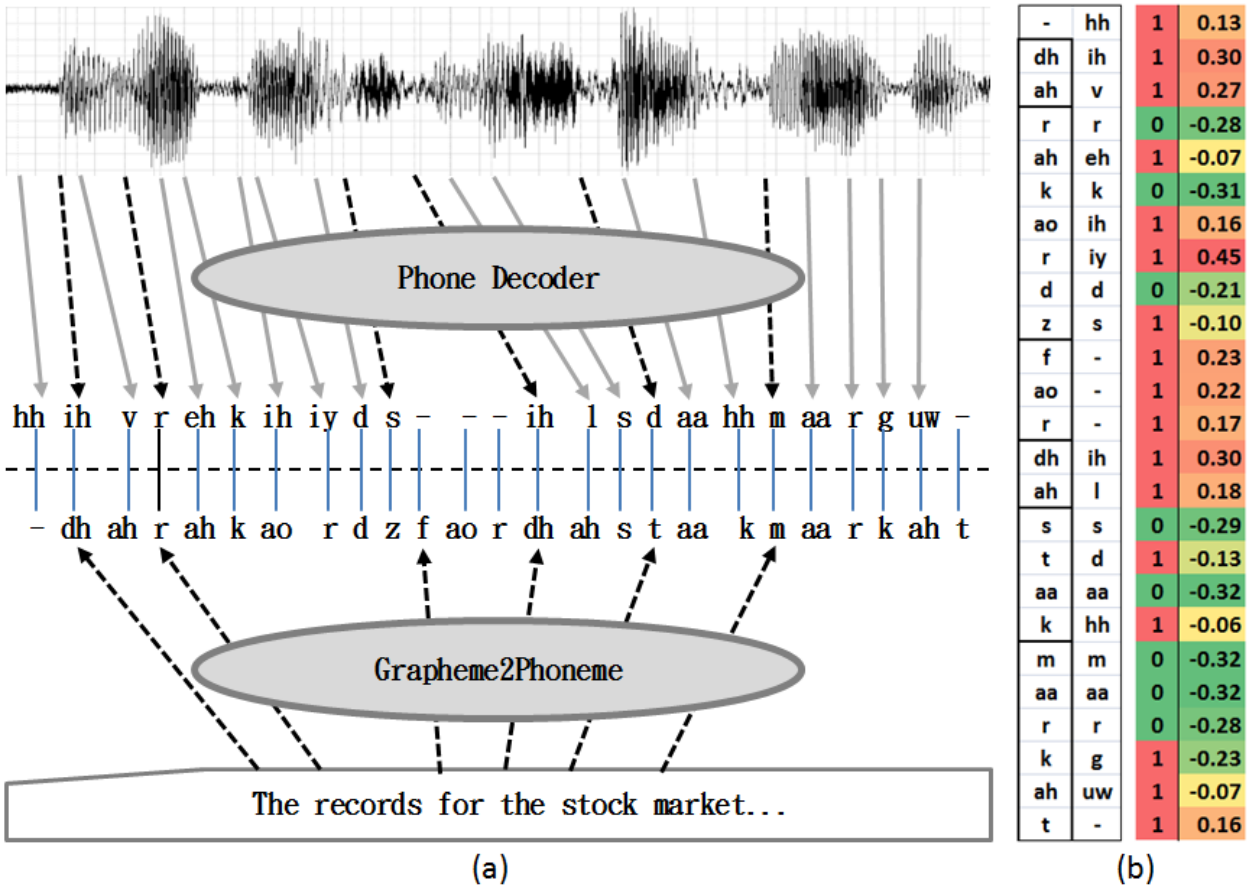

Fig. 1. Speech-to-text alignment is performed in this work by aligning a recognized phone sequence to the phonetic transcript of the reference text (a). A straightforward approach consists in minimizing a distance metric like Levehnstein's, but more informative metrics may lead to better results (b).

uttered (sometimes disfluent or grammatically incorrect). Note that the lack of correspondence between text transcripts and audio recordings increases the difficulty of the alignment task.

The two alignment strategies applied in [11] were: (1) *maximum number of matches* (*MaxMatch*), which is related to the longest common subsequence problem [14]; and (2) the *minimum edit (Levenshtein) distance* (*MinDist*) [15]. Taken as an optimization problem, the first strategy evaluates exact symbol matches as positive events and tries to maximize the number of them, whereas the second strategy evaluates mismatches (substitutions, deletions and insertions) as negative events and tries to minimize the number of them. When comparing both strategies in terms of time deviations with regard to a reference ground-truth, we observed that the *MinDist* approach provided slightly better performance than the *MaxMatch* approach.

As suggested in [11], a potential way for improvement is the use of information about the phone decoder

TABLE I

HUB4-97 DATA DISTRIBUTION PER CATEGORIES, AND PHONE ERROR RATES ON THE SIX MAIN CATEGORIES USING OUR PHONETIC DECODER.

Cat.	Description	Time (Secs.)	Words (count)	PhonERR %
F0	Baseline Broadcast Speech	4405	13040	41.4
F1	Spontaneous Broadcast Speech	1890	6331	48.6
F2	Speech Over Telephone Channels	1495	4672	57.3
F3	Speech in the Presence of Background Music	498	1554	53.3
F4	Speech Under Degraded Acoustic Conditions	1048	3263	48.6
F5	Speech from Non-Native Speakers	232	663	43.5
FX	All other speech (61 transcribed: other languages + 71 non-speech: silence, music, noise...)	1220	2528	—
?	Unclassified (not transcribed: mostly overlapped speech)	140	0	—

confusion probabilities (see Figure 1). In [16], this information is used along with two knowledge-based probability functions.

In this paper, we present an alternative approach, where the information provided by the phone decoder confusions is transformed according to probabilistic criteria. Three baseline systems will be considered for comparison: the *MaxMatch* and *MinDist* systems, and the ASR-based system described in [3].

II. DATABASE AND PHONETIC DECODER

The Hub4-97 database contains about 3 hours of transcribed broadcast audio, classified into six categories according to the acoustic conditions, plus a seventh category (*Other*) and an additional set of *Unclassified* segments (see Table I).

The phonetic decoder used in this work is based on a 40-phone set for English. Acoustic models were initialized on the TIMIT database [17] and then re-estimated on the Wall Street Journal database [18]. Left-to-right monophone continuous Hidden Markov Models, with three states and 64 Gaussian distributions per state, were used. It must be noted that, to keep things simple, the decoder was not particularly suited to the Hub4-97 dataset, that is, no adaptation was performed. Phone error rates attained on the six main Hub4-97 categories are shown in Table I. We would like to emphasize, at this point, that our alignment approach is able to achieve competitive results even though a low performance phone decoder is used.

III. Confusion matrix and kernels

Let us consider a speech corpus for which text transcripts are available. Then, phonetic transcripts can be easily extracted by applying either a pronunciation dictionary or a set of rules (or both). Let Σ be the set of phonetic symbols, $\mathcal{R}$ the reference phonetic transcript of a speech utterance and $\mathcal{H}$ the phonetic sequence hypothesized by a phonetic decoder when processing the speech utterance. Then, by aligning all the pairs of sequences $\mathcal{R}$ and $\mathcal{H}$ in a dataset, we can get the counts c_{rh} representing the number of times that symbols $r, h \in \Sigma$ (r coming from $\mathcal{R}$ and h from $\mathcal{H}$) have been paired together in those alignments. Similarly, we can get the counts c_r representing the number of times that a symbol r from $\mathcal{R}$ has not been paired with any symbol (i.e. *deleted*), and the counts c_h representing the number of times that a symbol h from $\mathcal{H}$ has no counterpart in $\mathcal{R}$ (i.e. it has been *inserted*).

The pairings (r, h) can be *matches* ($r = h$) or *substitutions* ($r \neq h$), but this fact is irrelevant for the rest of the work. We introduce the symbol ϵ in order to represent all these counts under a single structure commonly known as a *confusion matrix* ($c : \Sigma \cup \{\epsilon\} \times \Sigma \cup \{\epsilon\} \rightarrow \mathbb{N}$) considering $c_r \equiv c_{r\epsilon}$ and $c_h \equiv c_{\epsilon h}$.

A perfect decoder processing a perfectly labeled corpus should output a sequence of symbols identical to the reference transcript, so that all counts in the confusion matrix would be zero except for the diagonal elements. Any deviation from this unrealistic situation is somehow *informed* by the non-zero counts outside the diagonal.

To take advantage from the information provided by the confusion matrix in the alignment procedure, we must interpret and transform it into a kernel. Here we use the term "kernel" in the sense it is used in pattern analysis, meaning a similarity function over pairs of data. This kernel is the piece of information used by a sequence alignment algorithm to evaluate different possible subsequence alignments. The simplest one will evaluate different ways of confronting just two symbols, each one coming from one of the aligned sequences. This can be done by considering the primitive editing operations: matches, substitutions, deletions and insertions; or by considering different (particularized) evaluations for each symbol, in a similar manner as that considered in the confusion matrix.

When confronting a symbol r from the reference and a symbol h from the hypothesis, the aligner will need three kernel values: K_{rh}, for the case of pairing them, K_r, for the case of deleting r (not taking into account the presence of h), and K_h, for the case of considering h as an insertion (not taking into account the presence of r).

IV. Probabilistic kernels

The kernels proposed in this work are based on the probabilities of the three alternatives: P_{pair}, the probability of pairing r and h; P_{del}, the probability of deleting r; and P_{ins}, the probability of inserting

h. They can be estimated as the frequency of occurrence of each event in the master alignment:

$$P_{pair} = P(pair|r,h) = c_{rh} / \left(c_{rh} + \frac{c_{r\epsilon}}{N} + \frac{c_{\epsilon h}}{N}\right) \tag{1}$$

$$P_{del} = P(del|r) = c_{r\epsilon} / \left(c_{r\epsilon} + \sum_{\forall h} \left(c_{rh} + \frac{c_{\epsilon h}}{N}\right)\right) \tag{2}$$

$$P_{ins} = P(ins|h) = c_{\epsilon h} / \left(c_{\epsilon h} + \sum_{\forall r} \left(c_{rh} + \frac{c_{r\epsilon}}{N}\right)\right) \tag{3}$$

where $N \equiv |\Sigma|$ is the number of phonetic symbols. N is used to normalize the counts in cases where all possible contexts (represented by ϵ) have been added up. Note that these $P_{(\cdot)}$ values do not sum up to one, that is, we are not calculating the probability to take one of the three possible actions given a confronted pair (r,h), since the deletion and insertion probabilities do not depend on both symbols (*i.e.* $P_r \neq P(del|r,h)$, $P_h \neq P(ins|r,h)$). This fact will result in probabilistic kernels where the deletion and insertion values, K_{del} and K_{ins}, will not depend on their corresponding context symbols, h and r, respectively. Hereinafter, any kernel will be defined as:

$$K = \{K_{pair}, K_{del}, K_{ins}\} \tag{4}$$

The sequence alignment algorithm aims to find the path that minimizes or maximizes the cumulated kernel value, depending on the meaning of their figures, that can be *costs* or *benefits* respectively. Regardless of which of the two representations is considered more natural, we can switch from one to the other by inverting the sign of the values. For example, the two baseline kernels considered in this work can be expressed as:

$$K_{MaxMatch} = \{\delta_{rh}, 0, 0\}_{r,h\in\Sigma} \tag{5}$$

$$K^{\mathtt{cost}}_{MinDist} = \{1 - \delta_{rh}, 1, 1\}_{r,h\in\Sigma} \tag{6}$$

where K is a benefit kernel, $K^{\mathtt{cost}}$ is a cost kernel, and δ_{rh} is the Kronecker delta ($\delta_{rh} = 1$ if $r = h$; otherwise $\delta_{rh} = 0$).

In this work, we adopt the benefit maximization representation, where the minimum distance kernel is given by:

$$K_{MinDist} = \{\delta_{rh} - 1, -1, -1\}_{r,h\in\Sigma} \tag{7}$$

Next we define the three kernels that will be evaluated in this work.

A. Expected Match Kernel

The *MaxMatch* kernel can be directly generalized to a probabilistic kernel just by using the probability P_{pair} (*i.e.* the probability of r and h being paired in an alignment) instead of δ_{rh}:

$$K_{ExpectedMatch} = \{P_{pair}, 0, 0\} \tag{8}$$

B. Expected Edit Distance Kernel

Similarly to the *MaxMatch* generalization, the *MinDist* kernel can be directly converted to a probabilistic one that minimizes the expected number of editing operations (the expected edit distance). Given the probability P_e of an event (a pairing, a deletion or an insertion), the probability of an edit error is $1 - P_e$, and therefore, the benefits-based probabilistic kernel is:

$$K_{ExpectedDist} = \{P_{pair} - 1, P_{del} - 1, P_{ins} - 1\} \tag{9}$$

C. Logit Kernel

The direct translation of probabilities to benefits (or costs) is a way of introducing some information that could improve the alignment results, but a kind of logarithmic transformation converting the multiplicative probabilities into additive benefits could better match the nature of the algorithm. Moreover, given the symmetry of costs and benefits, a symmetric function mapping the $[0, 1]$ domain to $(-\infty, \infty)$ seems a good candidate to be tried. The logit function: $\text{logit}(p) = \log(p/(1-p))$ fulfills these requirements, so we propose a logit kernel:

$$K_{Logit} = \{\text{logit}\,(P_{pair}), \text{logit}\,(P_{del}), \text{logit}\,(P_{ins})\} \tag{10}$$

V. Experimental results

The different approaches presented in Section IV have been tested on the Hub4-97 dataset, using a confusion matrix estimated on the WSJ training corpus. The alignment effectiveness is evaluated in terms of the time deviation of each word hypothesized boundary with respect to the reference timestamps. This reference was built by carefully performing forced alignment in small pieces and checking the results by both hearing and visualizing them. The linguistic knowledge-based *Kondrak* kernel [16][19] was also tested in order to evaluate the amount of phonetic information provided by the probabilistic kernels.

Table II presents the results of the tested kernels for different tolerance intervals. Both the *Expected-Match* and the *ExpectedDist* kernels clearly beat their non-probabilistic versions (*MaxMatch* and *MinDist*), whereas the results of the *Kondrak* kernel are halfway between them. The *Logit* kernel gets the most competitive results, outperforming all the previous ones. Furthermore, at a much lower computational

TABLE II

ALIGNMENT ACCURACY AT THE WORD LEVEL FOR DIFFERENT KERNELS. THE FIGURES REPRESENT THE PERCENTAGE OF WORDS WITH HYPOTHESIZED BOUNDARIES CLOSER THAN A GIVEN DISTANCE $|x|$ (IN SECONDS) TO THE TRUE BOUNDARIES.

$\|x\| <$ seconds	Max Match	Min Dist	Kondrak	ASR[3]	Expected Match	Expected Dist	Logit
$\|x\| < 0.1$	78.31	80.58	85.2		87.76	84.91	**89.02**
$\|x\| < 0.2$	86.75	89.70	92.23		93.64	91.83	**94.4**
$\|x\| < 0.3$	90.65	93.37	95.04		95.85	94.58	**96.39**
$\|x\| < 0.4$	93.50	95.74	96.88		97.39	96.51	**97.79**
$\|x\| < 0.5$	95.24	97.16	97.92	98.5	98.25	97.64	**98.54**
$\|x\| < 1.0$	98.48	99.31	99.63		99.58	99.58	**99.71**
$\|x\| < 1.5$	99.35	99.78	99.91		99.82	99.94	**99.94**
$\|x\| < 2.0$	99.70	99.91	99.97	99.75	99.87	**99.98**	**99.98**

cost and required resources, the *Logit* kernel attains the same performance than the reference method [3] for both 0.5 and 2.0 seconds tolerance intervals.

The improvement provided by probabilistic kernels with respect to their non-probabilistic counterparts can be easily explained by the fact that they include information about the confusability of the phone decoder. If two different symbols r and h present a high confusion rate, they should be considered closer to a match than another pair with a lower confusion rate, so the benefit of accepting a substitution should be higher.

On the other hand, given that the simple *ExpectedMatch* kernel beats the *Kondrak* kernel, we can hypothesize that the information extracted from the confusion matrix is rich enough to avoid the need of any other source of linguistic information.

The high performance of the *Logit* kernel is in accordance with the criteria we adopted to propose it. Given the probability estimate P_e of an event, all the other kernels produce bounded values whereas the *logit* kernel produces values in the range $(-\infty, +\infty)$. This means (for the bounded kernels) that even in the case that $P_e = 0$, this event could be inside the selected path, because the contributions of other pairings can compensate for it. A similar (symmetric) argument could be used for the case $P_e = 1$. On the contrary, the *Logit* kernel value for $P_e = 0$ is $-\infty$ and, therefore, this event will never belong to the best path alignment, whereas for the case where $P_e = 1$, the kernel value is $+\infty$ and, therefore, this event must be part of the best path alignment.

Figure 2 shows the results for the six considered kernels and for the highest precision ($|dev| < 0.1$ seconds), broken down by the acoustic condition categories in Hub4-97. As expected, relative differences among kernels are consistent across acoustic conditions. Moreover, the quality of the alignments does

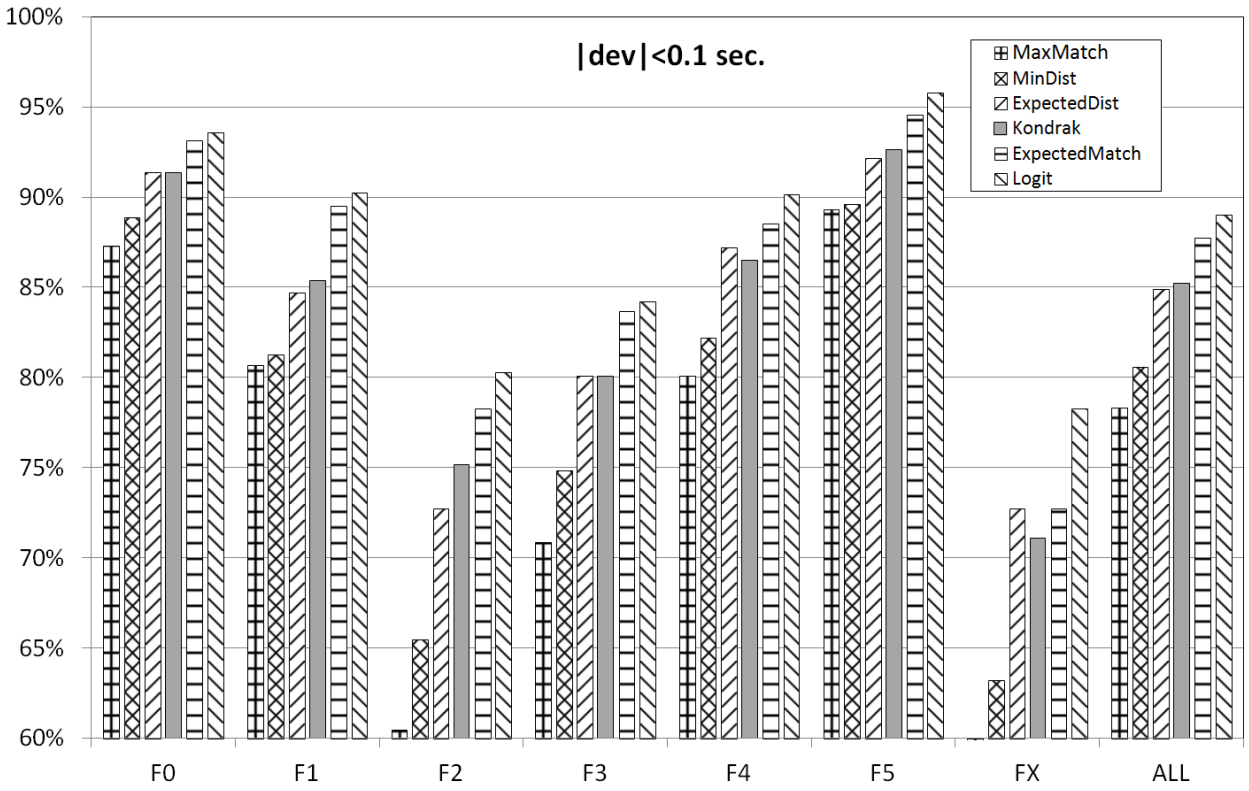

Fig. 2. Alignment accuracy (tolerance interval: 0.1 seconds) for the six kernels considered in this work, broken down by the acoustic conditions of Hub4-97.

not degrade to a higher degree than the quality of the decoded phonetic transcripts. The phone error rate ranges in [41.4, 57.3] whereas the corresponding alignment error ranges in [4.2, 19.7] (for the logit kernel), the size of the interval being approximately 16 in both cases. This shows that the alignment procedure performs reasonable well in all conditions. We should expect, however, that if the acoustic conditions were extremely adverse, the decoded phonetic transcript may become almost random, and the alignment procedure would probably fail.

VI. Conclusion

In this work, we have shown that the characterization of the phonetic decoder by means of a confusion matrix can be effectively used to estimate an informative kernel able to achieve remarkable performance gains in text-to-speech alignment of long audio tracks. The results obtained with a kernel based on phonetic knowledge reveal that the information conveyed by the confusion matrix is more precise and/or more specific to the particular task, leading to better performance.

As a potential line for future research, after using the system to align a speech signal to its text transcript for the first time, a new confusion matrix may be estimated and used to realign the sequences.

In this way, the second matrix would be adapted to the specific problem being treated. The new matrix would carry information about specific phonetic confusability due to the particular acoustic conditions of the signal, and about any specificity of the text sequence, which may eventually lead to a better result in the new realignment. It would be interesting to check whether this iterative procedure would effectively converge to a stable point with maximum alignment performance.

REFERENCES

[1] C. W. Wightman and D. T. Talkin, "The aligner: Text-to-speech alignment using Markov models," in *Progress in Speech Synthesis*. Springer, 1997, pp. 313–323.

[2] J. Robert-Ribes and R. Mukhtar, "Automatic generation of hyperlinks between audio and transcript," in *Fifth European Conference on Speech Communication and Technology*, 1997, pp. 903–906.

[3] P. Moreno, C. Joerg, J. Thong, and O. Glickman, "A recursive algorithm for the forced alignment of very long audio segments," in *Fifth International Conference on Spoken Language Processing*, 1998.

[4] T. Hazen, "Automatic alignment and error correction of human generated transcripts for long speech recordings," in *Proceedings of Interspeech*, 2006, pp. 1606–1609.

[5] A. Haubold and J. Kender, "Alignment of speech to highly imperfect text transcriptions," in *2007 IEEE International Conference on Multimedia and Expo*, 2007, pp. 224–227.

[6] A. Katsamanis, M. Black, P. G. Georgiou, L. Goldstein, and S. Narayanan, "Sailalign: Robust long speech-text alignment," in *Proc. Workshop on New Tools and Methods for Very Large Scale Research in Phonetic Sciences, Philadelphia*, Jan 2011.

[7] S. Hoffmann and B. Pfister, "Text-to-speech alignment of long recordings using universal phone models." in *Proceedings of Interspeech 2013*, Lyon, France, August 2013, pp. 1520–1524.

[8] I. Ahmed and S. K. Kopparapu, "Technique for automatic sentence level alignment of long speech and transcripts." in *Proceedings of Interspeech*, 2013, pp. 1516–1519.

[9] X. Anguera, J. Luque, and C. Gracia, "Audio-to-text alignment for speech recognition with very limited resources," in *Proceedings of Interspeech 2014*, Singapore, September 2014, pp. 1405–1409.

[10] P. Moreno and C. Alberti, "A factor automaton approach for the forced alignment of long speech recordings," in *Proceedings of IEEE ICASSP*, April 2009, pp. 4869–4872.

[11] G. Bordel, M. Penagarikano, L. J. Rodriguez-Fuentes, and A. Varona, "A simple and efficient method to align very long speech signals to acoustically imperfect transcriptions," in *Proceedings of Interspeech 2012*, Portland (OR), USA, September 2012.

[12] G. Bordel, S. Nieto, M. Penagarikano, L. J. Rodriguez-Fuentes, and A. Varona, "Automatic subtitling of the Basque Parliament plenary sessions videos," in *Proceedings of Interspeech 2011*, Florence, Italy, August 2011, pp. 1613–1616.

[13] D. Graff, J. Fiscus, and J. Garofolo, "1997 HUB4 English evaluation speech and transcripts," Linguistic Data Consortium, Philadelphia, 2002.

[14] D. Hirschberg, "A linear space algorithm for computing maximal common subsequences," *Communications of the ACM*, vol. 18, no. 6, pp. 341–343, 1975.

[15] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in *Soviet Physics Doklady*, vol. 10, no. 8, 1966, pp. 707–710.

[16] A. Álvarez, H. Arzelus, and P. Ruiz, "Long audio alignment for automatic subtitling using different phone-relatedness measures," in *Proceedings of IEEE ICASSP*, 2014, pp. 6280–6284.

[17] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT Acoustic-Phonetic Continuous Speech Corpus," Linguistic Data Consortium, Philadelphia, 1993.

[18] J. S. Garofolo, D. Graff, D. Paul, and D. S. Pallett, "CSR-I (WSJ0) Complete," Linguistic Data Consortium, Philadelphia, 2007.

[19] G. Kondrak, "Algorithms for language reconstruction," Ph.D. dissertation, University of Toronto, 2002.