Aligning very long speech signals to bilingual transcriptions of parliamentary sessions*

Germán Bordel, Mikel Penagarikano,
Luis Javier Rodríguez-Fuentes, and Amparo Varona

GTTS (http://gtts.ehu.es), Department of Electricity and Electronics, ZTF/FCT
University of the Basque Country UPV/EHU
Barrio Sarriena, 48940 Leioa, Spain
german.bordel@ehu.es

Abstract. In this paper, we describe and analyse the performance of a simple approach to the alignment of very long speech signals to acoustically inaccurate transcriptions, even when two different languages are employed. The alignment algorithm operates on two phonetic sequences, the first one automatically extracted from the speech signal by means of a phone decoder, and the second one obtained from the reference text by means of a multilingual grapheme-to-phoneme transcriber. The proposed algorithm is compared to a widely known state-of-the-art alignment procedure based on word-level speech recognition. We present alignment accuracy results on two different datasets: (1) the 1997 English Hub4 database; and (2) a set of bilingual (Basque/Spanish) parliamentary sessions. In experiments on the Hub4 dataset, the proposed approach provided only slightly worse alignments than those reported for the state-of-the-art alignment procedure, but at a much lower computational cost and requiring much fewer resources. Moreover, if the resource to be aligned includes speech in two or more languages and speakers conmute between them at any time, applying a speech recognizer becomes unfeasible in practice, whereas our approach can be still applied with very competitive performance at no additional cost.

Index Terms: speech-to-text alignment, multilingual speech, automatic video subtitling.

1 Introduction

The work presented in this paper was motivated by a contract with the Basque Parliament for subtitling videos of bilingual (Basque/Spanish) plenary sessions. The task consisted of aligning very long (around 3 hours long) audio tracks with syntactically correct but acoustically inaccurate transcriptions (since all the silences, noises, disfluencies, mistakes, etc. had been edited).

* This work has been supported by the University of the Basque Country, under grant GIU10/18 and project US11/06, by the Government of the Basque Country, under program SAIOTEK (project S-PE11UN065), and the Spanish MICINN, under *Plan Nacional de I+D+i* (project TIN2009-07446, partially financed by FEDER funds).

The above described task may have been easily solved by means of forced alignment at word level, allowing some mismatch between speech and text to cope with imperfect transcripts or alternative pronunciations [1] [2]. However, forced alignment cannot be directly performed on long audio tracks, due to memory bounds. The algorithm proposed in [3] comes to solve this limitation, by applying a speech recognizer, then looking for anchor phrases (sequences of words matching part of the text), splitting the text at such points and recursively applying the same algorithm on the resulting fragments, until their length is small enough to apply forced alignment.

However, the mix of Basque and Spanish in the parliamentary sessions made language and lexical models needed by the speech recognizer difficult to integrate. Therefore, an alternative procedure was developed, which started from a hybrid set of phonetic units covering Basque and Spanish. Acoustic-phonetic models were estimated and a phone decoder was built based on data from both languages. The alignment algorithm operated on two sequences of phonetic units: the first one produced by the phone decoder and the second one obtained by means of a grapheme-to-phoneme transcriber mapping ortographic transcriptions to sequences of hybrid (Basque/Spanish) phonetic units. Finally, time stamps provided by the phone decoder were mapped to ortographic transcriptions through phonetic alignments. This approach worked pretty well for the intended application, as shown in [4].

With the aim to compare our approach to that presented in [3], we carried out a series of experiments on the 1997 English Hub4 dataset (see [5] for details). Following the evaluation criteria proposed in [3], we found that 96% of the word alignments were within 0.5 seconds the true alignments, and 99.2% within 2 seconds the true alignments. In the reference approach [3], better figures are reported (98.5% and 99.75%, respectively) but at a much higher computational cost, and as we noted above, it could not be easily applied to multilingual speech. In this paper, we summarize the above described efforts and devote more space to analyse and discuss the alignment errors, which may light the way to further improvements.

The rest of the paper is organized as follows. In Section 2, we provide the key features of our simple speech-to-text alignment approach. The experimental setup is briefly described in section 3. Results are presented and commented in Section 4. Finally, conclusions are given in Section 5, along with a discussion on possible ways of improving the method.

2 The speech-to-text alignment method

To synchronize speech and text, we map both streams into a common representation, then align the resulting sequences and relate positions in the original sources by mapping back from the common representation. A suitable candidate for such common representation is the phonetic transcription, which features a small vocabulary size and a small granularity. We assume that phone decoding is performed without any language/phonotactic models, so that the alignment will be language independent, provided that the set of phonetic units covers all the languages appearing in the speech stream.

2.1 Phone inventories

In this paper, we consider two different datasets: (1) Hub4, monolingual (English), for which a set of 40 phonetic units was defined, based on the TIMIT database [6]; and (2) plenary sessions of the Basque Parliament, bilingual (Basque/Spanish), for which a set of 27 phonetic units covering both languages was defined (see Table 1). Note that Basque and Spanish share most of their phones, with few differences. We selected 26 units for Basque and 23 units for Spanish, meaning that just one *foreign* sound was added to Basque (θ in IPA coding) and four *foreign* sounds to Spanish (ʃ, ʦ, ʦ' and s' in IPA coding)[1]. Also, though not specified in Table 1, the sounds corresponding to graphemes 'll' in Spanish and 'il' in Basque are assimilated to IPA dʒ.

Phones are represented so that the original ortographic transcriptions can be fully recovered, which is needed at the end of the process. Internally, articulatory codes related to the physiology of the production of each phone are used. Externally, those codes are mapped to IPA codes. Since Basque/Spanish phonetics is very close to its orthography, we also use a highly readable *single-character* specific coding (GTTS-ASCII, see Table 1).

2.2 From speech to phones

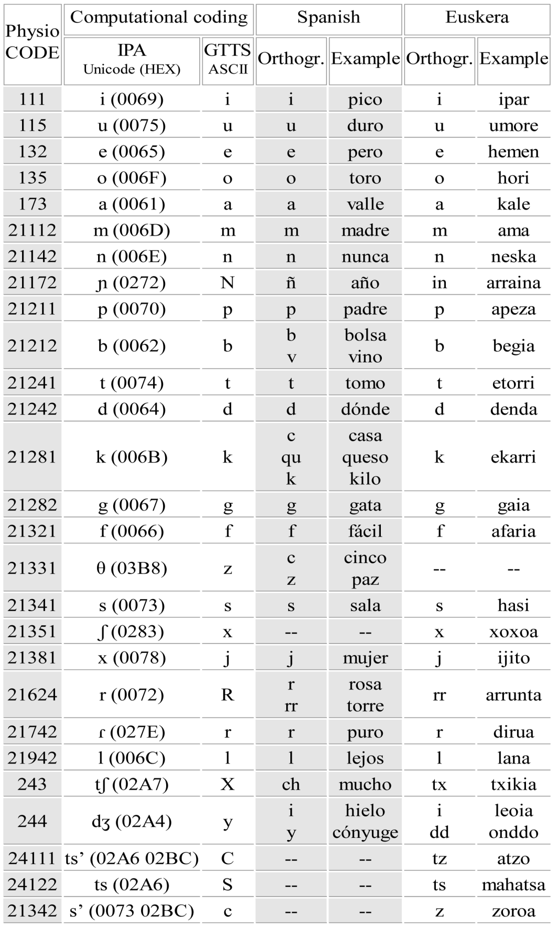
Audio streams were converted to PCM, 16 kHz, 16 bit/sample. The acoustic features consisted of 12 Mel-Frequency Cepstral Coefficients plus the energy and their first and second order deltas (a common parameterization in speech recognition tasks). Left-to-right monophone continuous Hidden Markov Models, with three looped states and 64 Gaussian mixture components per state, were used as acoustic models.

For the Hub4 experiments, a phone decoder was trained on the TIMIT database [6] and then re-trained on the Wall Street Journal database [7]. The phone decoder yielded error rates in the range 40-60%, depending on the acoustic conditions of the Hub4 subset considered for test (see Figure 1).

Defining a common set of phonetic units covering both Basque and Spanish allowed us to train a single phone decoder to cope with the mixed use of Spanish and Basque in the Basque Parliament. The material used to train the phonetic models was the union of the Albayzin [8] and Aditu [9] databases. Albayzin consists of 6800 read sentences in Spanish from 204 speakers and Aditu consists of 8298 sentences in Basque from 233 speakers. The phone decoder trained this way yielded around 80% phone recognition rate in open-set tests on Albayzin and Aditu, and only above 60% on the Basque Parliament sessions, probably due to acoustic mismatch (background noise, speaker variability, etc.) [4].

[1] Note, however, that the sound θ is pronounced by Basque speakers in words imported from Spanish, and that sounds considered foreign in the central Castilian Spanish (such as ʦ) are widely used in other Spanish dialects.

Table 1. Phone inventory for Basque (Euskera) and Spanish, with examples. IPA codes (Unicode) are shown, as well as a highly readable single-character coding (GTTS-ASCII). Internally, the grapheme-to-phoneme transcriber uses the articulatory codes (*physio codes*) shown in the first column.



Physio CODE	Computational coding		Spanish		Euskera	
	IPA Unicode (HEX)	GTTS ASCII	Orthogr.	Example	Orthogr.	Example
111	i (0069)	i	i	pico	i	ipar
115	u (0075)	u	u	duro	u	umore
132	e (0065)	e	e	pero	e	hemen
135	o (006F)	o	o	toro	o	hori
173	a (0061)	a	a	valle	a	kale
21112	m (006D)	m	m	madre	m	ama
21142	n (006E)	n	n	nunca	n	neska
21172	ɲ (0272)	N	ñ	año	in	arraina
21211	p (0070)	p	p	padre	p	apeza
21212	b (0062)	b	b v	bolsa vino	b	begia
21241	t (0074)	t	t	tomo	t	etorri
21242	d (0064)	d	d	dónde	d	denda
21281	k (006B)	k	c qu k	casa queso kilo	k	ekarri
21282	g (0067)	g	g	gata	g	gaia
21321	f (0066)	f	f	fácil	f	afaria
21331	θ (03B8)	z	c z	cinco paz	--	--
21341	s (0073)	s	s	sala	s	hasi
21351	ʃ (0283)	x	--	--	x	xoxoa
21381	x (0078)	j	j	mujer	j	ijito
21624	r (0072)	R	r rr	rosa torre	rr	arrunta
21742	ɾ (027E)	r	r	puro	r	dirua
21942	l (006C)	l	l	lejos	l	lana
243	tʃ (02A7)	X	ch	mucho	tx	txikia
244	dʒ (02A4)	y	i y	hielo cónyuge	i dd	leoia onddo
24111	ts’ (02A6 02BC)	C	--	--	tz	atzo
24122	ts (02A6)	S	--	--	ts	mahatsa
21342	s’ (0073 02BC)	c	--	--	z	zoroa

2.3 From text to phones

In the Hub4 experiments, phonetic transcriptions were extracted from the CMU English pronouncing dictionary [10]. In the case of Basque parliament sessions, a multilingual transcriber architecture was defined, including a specific transcription module for each target language. Each transcription module consists of a dictionary, a set of transcription rules and a *number-and-symbols to text converter* (for numbers, currencies, percentages, degrees, abbreviations, etc). In this work, two modules for Basque and Spanish were used (including their respective dictionaries), and a third auxiliary module was defined, consisting of a dictionary covering all the words falling out of the vocabulary of both languages.

Phonetic transcriptions are generated as follows. First, each input word is searched in the three available dictionaries: Basque, Spanish and out-of-vocabulary words. If the word appears in a single dictionary, the phonetic transcription provided by that dictionary is output. If the word appears in more than one dictionary, the transcriber uses the context to determine the language being used and outputs the phonetic transcription for that language. Finally, if the word doesn't appear in any dictionary, the transcriber outputs a rule based transcription based on the subsystem corresponding to the most likely language. New transcriptions generated by applying rules are added to the corresponding dictionary and reported to be supervised. This mechanism makes dictionaries to grow incrementally and works as a misspelling detector, allowing for the refinement of rules.

2.4 Alignment of very long sequences

A globally optimal solution to the alignment of two symbol sequences is given by the Needleman-Wunsch algorithm [11]. In this work, we apply a variant of this algorithm. Let X and Y be two sequences of n and m symbols, respectively. A $n \times m$ matrix C is filled with the minimum accumulated edition cost, also known as Levenshtein's distance, using an auxiliary $n \times m$ matrix E to store the edition operations that minimize the cost at each step. Four possible edition operations are considered: deletions, insertions, substitutions and matches. The three first operations have cost 1 and matches have cost 0. Note that what we call *best alignment* depends on the set of weights assigned to deletions, insertions, substitutions and matches. We use the Levenshtein distance because, after some experimentation on the set of parliamentary sessions (which is our target application), it gave the best results. Finally, the path generating the minimum cost is tracked back from $E(n, m)$ to $E(1, 1)$, which defines the optimal mapping (i.e. the optimal alignment) between X and Y.

The above described method is prohibitive for very long sequences due to the matrix memory allocation. However, we can still use a *Divide and Conquer* approach known as Hirschberg algorithm [12], where the original problem is optimally split into two sub-problems with half the size, by doing all the matrix calculations but storing only one row that goes half matrix forward from the start, and one row that goes half the matrix backward from the end. This method is recursively applied until the amount of memory needed to apply the non-recursive approach can be allocated. This algorithm reduces dramatically the required memory, increasing less than 2 times the computation time. Besides, since it can be easily parallelized, it can be run even on a desktop computer (e.g. less than 1 minute for a 3-hour signal in an 8-thread Intel i7 2600 processor).

3 Experimental setup

3.1 The 1997 Hub4 dataset

The 1997 Hub4 dataset consists of about 3 hours of transcribed broadcast audio, classified into 6 categories according to acoustic conditions, plus a seventh *Other* category and a set of *Unclassified* segments. The last two subsets (amounting to 8.5% of the phone units) were not considered in this work. The six remaining categories were defined as follows: F0 (clean speech), F1 (spontaneous speech), F2 (telephone-channel speech), F3 (speech with background music), F4 (degraded speech), and F5 (speech from non-native speakers). Their proportions are shown in Figure 1.

3.2 Basque Parliament plenary sessions

We started processing plenary sessions of the Basque Parliament by September 2010. The dataset considered in this work consists of 80 sessions, amounting to 407 hours of video. Due to limitations of video recording media, each video lasts no more than 3 hours, although sessions are 4-8 hours long. Therefore, each session consists of two or three (exceptionally up to four) sections. Videos (originally recorded in high definition using professional media) are converted to RealMedia format for the Basque Parliament web, the audio stream being downsampled to 22050 Hz, 16 bit/sample.

The Session Diary is available almost immediately as a single PDF document, because text transcriptions are produced on the fly by a team of human operators (who are instructed to exclude non-relevant events such as silences, noises, disfluencies, etc.). The session diary is made up of blocks related to operator shifts (approximately 15 minutes per block) with some undefined overlap between them. Also, after each voting procedure in the Basque Parliament, results are not transcribed verbatim as read by the president but just tabulated. All these issues make the synchronization between videos and diaries even more difficult. To address them, we designed specific (in part heuristic, in part supervised) solutions which are out of the scope of this paper.

3.3 Evaluation measure

Following [3], the alignment accuracy is measured in terms of the deviation of the starting point of each word from the available ground truth. In the case of Hub4, the reference positions were obtained by forced alignment (on a sentence by sentence basis) at the phone level, using acoustic models closely adapted to the HUB4 dataset. In the case of parliamentary sessions, we manually annotated the starting point of each word for a continuous fragment containing 876 words. Finally, to evaluate the alignment accuracy, we provide the percentage of words whose starting point is within a tolerance interval with regard to the reference position.

4 Results

4.1 Results on the Hub4 dataset

Results on the Hub4 dataset are summarized in Figure 1. As in [3], tolerance intervals of 0.5 and 2 seconds around the reference positions are considered to measure alignment accuracy. We found that 4.06% of the words deviated more than 0.5 seconds from their reference positions, whereas only 0.82% of the words deviated more than 2 seconds from their reference positions. These figures are slightly worse than those reported in [3] (1.5% and 0.25%, respectively), but the computational savings are quite remarkable, both in terms of time and infrastructure (data, models, etc.).

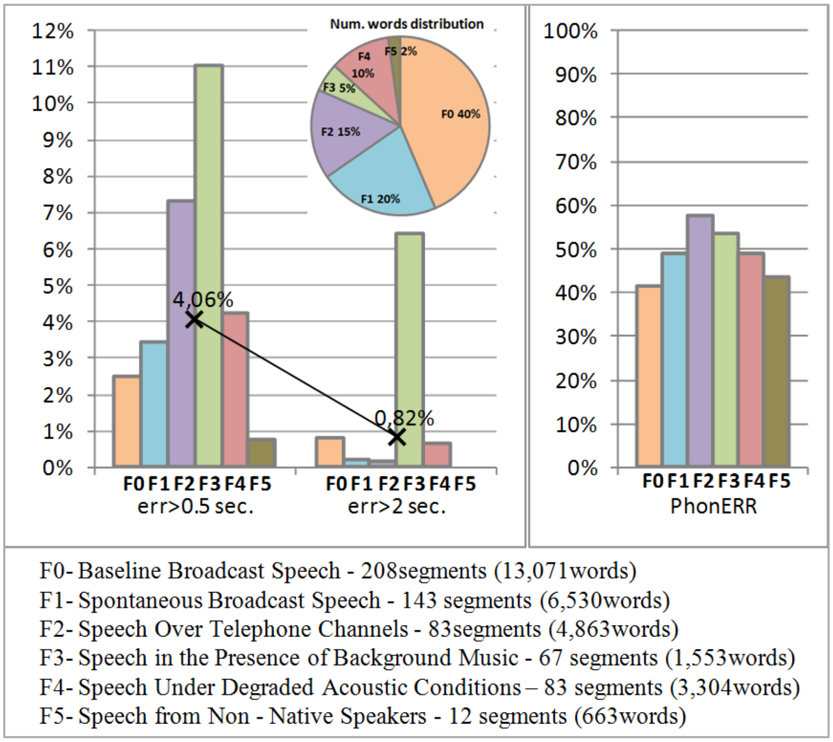


Fig. 1. Results for the 1997 Hub4 dataset: proportions of data in each category, phone decoding error rates and alignment accuracy for tolerance intervals of 0.5 and 2 seconds.

As shown in Figure 1, the alignment accuracy strongly depends on the acoustic condition considered for test. On the other hand, the alignment accuracy for a given condition is not only related to the phone recognition accuracy. For instance, the highest alignment error rate was found for the F3 condition (speech with background music), whereas the highest phone recognition error rate was found for the F2 condition (telephone-channel speech). The large difference between the alignment accuracies for the F2 and F3 conditions when considering a 2-second tolerance interval (despite having very similar phone recognition accuracies) is even more significant.

4.2 Results on the Basque Parliament sessions

As noted above, the alignment accuracy was measured on a continuous fragment of a parliamentary session including 876 words. Table 2 shows the alignment accuracy for different tolerance intervals between 0.1 and 0.5 seconds.

Table 2. Alignment accuracy on a fragment of a session of the Basque Parliament, for different tolerance intervals (in seconds).

Tolerance (seconds)	0.1	0.2	0.3	0.4	0.5
Alignment accuracy	67.69%	88.58%	92.01%	94.41%	95.43%

Note that 95% of the words deviated less than 0.5 seconds from the reference position considered as ground truth, which is enough for the subtitling application that motivated this work. After aligning speech and text, and following a number of rules related to lengths, times and punctuation, the synchronized text stream is split into small segments suitable for captioning. Only the first word of each segment is taken into account to synchronize text and speech, so that errors are perceived by users as these segments being presented with some advance or delay. Since errors involve both advances and delays in random order, having a long run of advances or delays is quite unlikely. In any case, a deviation of 0.5 seconds is not perceived as an error, specially when the caption appears in advance. For instance, after a long silence, when captions blank, users can easily accept a caption appearing in advance but not a delayed caption. Based on this analysis and taking into account that the captioning procedure can be configured to behave in different ways when there are more than one equivalent partial alignment, we tuned the application so that the mismatched segments had a tendency to show captions before the audio.

5 Conclusions and future work

In this paper, we have presented a simple and efficient method to align long speech signals to multilingual transcriptions, taking advantage of a single set of phonetic units covering the sounds of the target languages. We have compared the accuracy of the proposed approach to that of a well-known state-of-the-art alignment procedure, finding a small degradation on the Hub4 dataset, but remarkable savings in both computation time and the required infrastructure. On the other hand, the proposed method can deal with multilingual speech, which is not the case of the state-of-the-art approach used as reference. Alignment results have been also shown for plenary sessions of the Basque Parliament, for which a captioning system has been built based on the proposed algorithm.

Possible ways of increasing the alignment accuracy in future developments include: (1) adapting the acoustic models used in phone decoding to the particular resources to be aligned; and (2) replacing the kernel in the Needleman-Wunsch algorithm (currently representing a Levenshtein distance) with a more

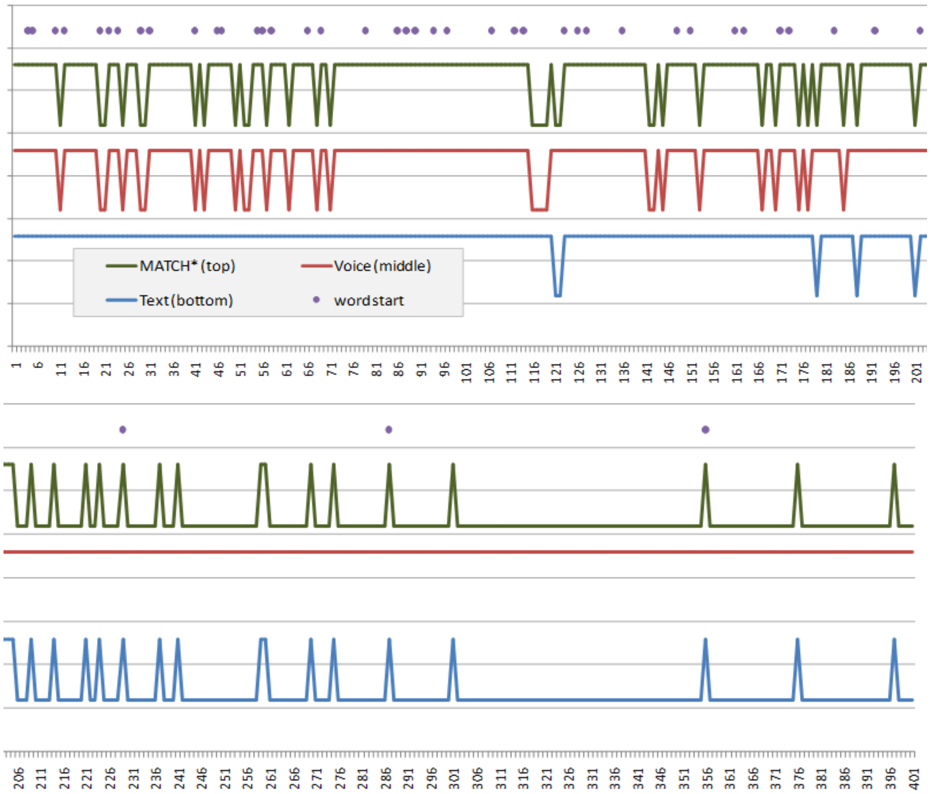


Fig. 2. Patterns found in the alignment path. The bottom line (blue) is up for phones in the text, whereas the middle line (red) is up for phones in the recognized speech. Lines go down at insertions for the bottom line and at deletions for the middle line. The top line (green) represents the AND function of the two other lines, so that it is up for matches and substitutions, and down for insertions and deletions.

informative kernel, e.g. by using continuous weights based on phone confusion probabilities.

Also, by analysing the alignment path, we can search for patterns that may eventually help to automatically reconsider some word synchronizations. Figure 2 represents a section of an alignment path for a Basque parliament session. Two different halves can be identified: the upper half corresponds to a correct alignment, whereas the bottom half corresponds to a wrong alignment due to a missing transcription. In the first half, words are detected at distances that are basically in accordance to phone lengths (upper dots). In the second half, a long run of decoded phones (middle line up) matches to few text phones (bottom line mostly down), meaning that words in the text are sparsely matched to phones from non-transcribed speech. In the first half, we also find that the insertion penalty applied by the phone decoder is too high, since there are much more deletions than insertions in the recognized sequence.

The relation between matches and substitutions and the time span for each word provide key information about the probability of having a perfect match. Places where there is no reference transcription can be detected as long runs of phone insertions, that is, as words spanning in excess through the alignment path. The opposite situation (extra text), which rarely appears in manual transcriptions, would generate long runs of phones in the other axis, that is, a high number of deletions. Both events produce border effects that should be identified and compensated. The *attraction* or *repulsion* that these regions induce on the recognized sequence of phones will depend on the number of deleted or inserted words and will be smoothed by the constraint that both sequences match.

This analysis suggests that, given that most of the alignment is right, we should focus on the problematic areas to isolate the alignment errors and correct the border effects by means of forced alignment. Curiously, this idea is complementary to the algorithm proposed in [3].

References

1. J. Vonwiller, C. Cleirigh, H. Garsden, K. Kumpf, R. Mountstephens, and I. Rogers, “The development and application of an accurate and flexible automatic aligner,” *The International Journal of Speech Technology*, vol. 1, no. 2, pp. 151–160, 1997.
2. P. Moreno and C. Alberti, “A factor automaton approach for the forced alignment of long speech recordings,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2009, pp. 4869 –4872.
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. 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, pp. 1613–1616.
5. 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 *Interspeech 2012*, Portland (OR), USA, September 9-13 2012.
6. 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.
7. J. S. Garofolo, D. Graff, D. Paul, and D. S. Pallett, “CSR-I (WSJ0) Complete,” Linguistic Data Consortium, Philadelphia, 2007.
8. A. Moreno, D. Poch, A. Bonafonte, E. Lleida, J. Llisterri, J. Marino, and C. Nadeu, “Albayzin speech database: design of the phonetic corpus,” in *Proceedings of Eurospeech*, Berlin, Germany, September 22-25 1993, pp. 175–178.
9. Basque Government, “ADITU program,” 2005, Initiative to promote the development of speech technologies for the Basque language.
10. R. Weide, “The Carnegie Mellon pronouncing dictionary [cmudict.0.6],” Carnegie Mellon University, 2005.
11. S. B. Needleman and C. D. Wunsch, “A general method applicable to the search for similarities in the amino acid sequence of two proteins,” *Journal of Molecular Biology*, vol. 48, no. 3, pp. 443 – 453, 1970.
12. D. Hirschberg, “A linear space algorithm for computing maximal common subsequences,” *Communications of the ACM*, vol. 18, no. 6, pp. 341–343, 1975.