SailAlign: Robust long speech-text alignment

A. Katsamanis, M. P. Black, P. G. Georgiou, L. Goldstein, S. Narayanan

University of Southern California, Los Angeles, CA, USA
http://sail.usc.edu

Abstract

Long speech-text alignment can facilitate large-scale study of rich spoken language resources that have recently become widely accessible, e.g., collections of audio books, or multimedia documents. For such resources, the conventional Viterbi-based forced alignment may often be proven inadequate mainly due to mismatched audio and text and/or noisy audio. In this paper, we present SailAlign which is an open-source software toolkit for robust long speech-text alignment that circumvents these restrictions. It implements an adaptive, iterative speech recognition and text alignment scheme that allows for the processing of very long (and possibly noisy) audio and is robust to transcription errors. SailAlign is evaluated on artificially created long chunks of the TIMIT database. Audio is artificially contaminated with babble noise, and the corresponding transcriptions are corrupted at various levels. We present the corresponding word boundary detection results. Finally, we demonstrate the potential use of the software for the exploitation of audio books for the study of read speech.

Index Terms: speech-text alignment, open-source, software, imperfect transcriptions, adaptation, audio-books

1. Introduction

Speech-text alignment commonly finds applications in fields such as multimedia indexing and training of large vocabulary speech recognition and synthesis systems [1, 2]. Recently, it has also been shown to be useful in the context of phonetics research for the exploitation of rich spoken language resources such as audio books [3]. Overall, it may be viewed as a mechanism that simultaneously enriches spoken language transcriptions with temporal information and identifies audio segments with their corresponding spoken content. Conventionally, speech-text alignment is performed by application of the standard Viterbi-based forced alignment [4]. However, this process may be proven inadequate in cases when the audio is contaminated with noise or when the transcription is not sufficiently accurate. In this paper, we present our open-source software, SailAlign, that circumvents these restrictions to a significant degree through the implementation of an adaptive, iterative speech recognition - text alignment scheme, sketched out in Fig. 1.

Our work has been particularly motivated by the need to process long, noisy audiovisual data collected for the observational study of marital and family interaction in the domain of psychology [5, 6]. The research and therapeutic paradigm in this area involves the collection and analysis of audiovisual data from the couples or families in focus. At a preprocessing stage, these recordings are oftentimes manually or semi-automatically transcribed to aid in the evaluation of the observed behavior [7]. Fully automatic transcription is usually unreliable for these real-environment, spontaneous recordings. The desired richness of the transcription depends on the evaluation process, but for practical reasons it is only at the word-level. In this context, robust speech-text alignment can facilitate exploitation of spoken language properties of these interaction-related multimodal data [8, 9].

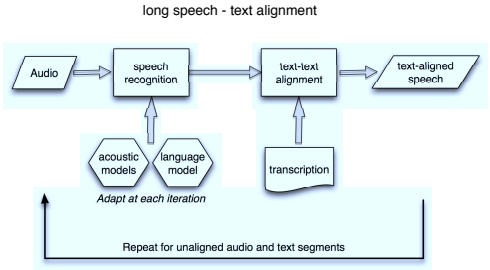

Figure 1: *Adaptive, iterative scheme for robust long speech-text alignment*

The robust long speech-text alignment approach implemented by SailAlign builds upon the iterative segmental application of a large vocabulary continuous speech recognition system, as introduced by Moreno and his colleagues [10]. The main idea in [10] is based on the assumption that with a sufficiently good speech recognition engine, it is possible to pose the speech-text alignment problem as a text-text alignment one. Solution of the latter is normally much less computationally demanding. After selecting regions where alignment is reliable, based on prescribed criteria, the process is iterated on the remaining unaligned regions. Language modeling becomes region specific, resulting in improved recognition and consequently improved alignment. This segmental processing has the additional advantage that it hinders possible local errors from propagating.

Researchers have proposed variations of this system to better cope with imperfect transcriptions [11, 1]. Finite state automata have been used to account for insertions, deletions, and/or substitutions of the transcribed words. SailAlign also implements a finite state grammar approach at the final stage to appropriately constrain the alignment search space if necessary. Regarding noise robustness, in [10] Moreno suggested that processing of noisy audio does not necessarily cause a drop in alignment performance. He even presented successful results of the iterative approach after the addition of white noise to the audio at 15dB to support this claim. SailAlign further improves robustness by label boosting [12], i.e., adaptation of the acoustic models at every iteration to account for mismatched acoustic conditions.

SailAlign has already found real world applications as mentioned above [8, 9] in the alignment of real interaction data with noisy transcripts. However, due to the lack of reference alignments in that setting it is impossible to quantitatively assess the

quality of these alignments. We therefore evaluate SailAlign's performance using the TIMIT database [13] by creating long sequences of audio through the concatenation of TIMIT segments. The resulting corpus provides us with the ground truth information of word alignments. We use this corpus to compare SailAlign with the conventional Viterbi-based forced alignment. To challenge the robustness of the system, we also artificially add noise to the audio and corrupt the transcriptions at various levels. Finally, we present a pilot experiment on the use of SailAlign for processing four audio books, i.e., four versions of Jane Austen's "Emma" as read by four different speakers [14].

2. Adaptive Long Speech-Text Alignment

2.1. Algorithm

SailAlign implements the adaptive, iterative speech-text alignment algorithm described as Algorithm 1 using pseudocode. As mentioned earlier, at the core of the algorithm lies the assumption that the long speech-text alignment problem can be posed as a long text-long text alignment problem given a well-performing speech-text conversion tool, i.e., speech recognition engine. Given that the text-text alignment problem can usually be solved quite efficiently even for long text using dynamic-programming to minimize the Levenshtein distance between the reference and the hypothesized text, the main bottleneck is then at the speech recognition part. The algorithm can be outlined as follows:

Initialization The audio stream has to be segmented into smaller chunks whose duration is constrained by computational limitations of the speech recognition engine used (approximately 10 to 15 seconds in our case). To avoid cutting a word into two, segmentation is guided by a voice activity detection module. For efficiency, segmentation is performed in the acoustic feature domain and not in the audio domain. In this way, acoustic feature extraction is carried through only once and not repeated for every repartitioning of the input stream. Repartitioning will be necessary at subsequent iterations. Acoustic features are extracted from the audio stream. To ensure that the speech recognition output will be as close as possible to the reference transcription, a transcription-specific language model is built at this point.

Speech Recognition, Text-Text Alignment Continuous speech recognition is then applied to identify the lexical content of the individual speech segments. The hypothesized transcripts are concatenated into a single one, which is then aligned with the reference transcript. Reliably aligned regions are selected by applying a minimum-number-of-words criterion, i.e., they should include at least a minimum number of consecutive aligned words. The rest of the audio is considered to be unaligned and it is repartitioned into segments of appropriate length. The transcription is also partitioned appropriately this time to leave the aligned regions out. The recognition-text alignment cycle will be repeated for only the unaligned audio and text segments.

Acoustic and Language Model Adaptation To improve noise robustness, we adapt the acoustic models at each iteration in a supervised manner using the reliably aligned regions. Maximum Likelihood Linear Regression [15] is applied and adaptation is performed in two steps. First, we train a global transformation and then, for groups of phonemes in which we have sufficient adaptation data, we build a class-based transformation. The language models are also updated so that they are trained specifically for each unaligned region. This process, i.e., recognition-alignment-adaptation, is iterated three times. In the subsequent two iterations, the acoustic models are not adapted, and the language model is described by a constrained finite state grammar which only allows the expected sequence of words for the segment (and insertions/deletions for the fourth iteration). This is expected to further increase the number of aligned regions in the case of very noisy audio.

Algorithm 1 The SailAlign speech-text alignment algorithm

Require: Audio file and corresponding transcription (word sequence S)
Ensure: Time-aligned transcription (S, T)
1: Detect speech regions by Voice Activity Detection (VAD)
2: Extract acoustic features A from the audio signal
3: $E_0 \leftarrow$ Generic acoustic models
4: $U_0 \leftarrow (A, S)$ {Unaligned acoustic features and the corresponding word sequence}
5: **for** i=1 to 5 **do**
6: **for all** N segments in U_{i-1} **do**
7: $A_n \leftarrow$acoustic features of the segment
8: $S_n \leftarrow$ corresponding word transcript
9: Segment A_n in K_n subregions $\{A_{nk}\}$ of approximate duration D {Given VAD timestamps, ensure that breaks are not within words}
10: **if** $i < 4$ **then**
11: Build a trigram language model L_n on S_n
12: **else**
13: Build a finite state grammar L_n on S_n
14: **if** $i = 5$ **then**
15: Do not allow insertions or deletions
16: **end if**
17: **end if**
18: **for** k=1 to K_n **do**
19: (R_{nk}, T_{nk}) = SpeechRecognition(A_{nk}, E_{i-1}, L_n)
{R_{nk} is the word sequence, T_{nk} the corresponding set of temporal word boundaries}
20: **end for**
21: **end for**
22: $(R, F) \leftarrow \bigcup_{n,k}(R_{nk}, T_{nk})$
23: Align word sequences S and R using Dynamic Programming to minimize Levenshtein distance
24: $\{(A_{im}, O_{im}, T_{im}), m = 1 \text{ to } M\} \leftarrow$ Subsequences of at least three aligned words and the corresponding acoustic features {Anchors}
25: **if** $i < 4$ **then**
26: $E_i \leftarrow$ Adapted acoustic models on $\{(A_{im}, O_{im}, T_{im})\}$ using regression class tree-based Maximum Likelihood Linear Regression (MLLR)
27: **else**
28: $E_i \leftarrow E_3$
29: **end if**
30: $\{P_j, j = 1 \text{ to } J\} \leftarrow S \setminus \bigcup_m O_{im}$
31: $\{A_j, j = 1 \text{ to } J\} \leftarrow A \setminus \bigcup_m A_{im}$
32: $U_i \leftarrow \{(A_j, P_j), j = 1 \text{ to } J\}$ {Collection of unaligned segments and their untimed transcriptions}
33: **end for**
34: $(S, T) \leftarrow \bigcup_{i,m}(O_{im}, T_{im})$

2.2. Implementation

SailAlign is implemented as a library of Perl modules, appropriate wrapper scripts, configuration files, and a collection of tools; the tools are either packaged with the software (e.g., the voice

activity detection binary), or they can be accessed separately at their corresponding repositories (e.g., the HTK toolkit [16]). Although SailAlign is configurable and has been designed to allow for interchangeable use of various versions or implementations of the separate required tools, the package that we currently release has only been tested with a specific choice of the speech recognition engine, language model building, text-text alignment, and acoustic feature extraction tools.

Voice activity detection (VAD) is currently performed by a separate software that implements the algorithm described in [17]. VAD is not crucial in our setup, and it could even be replaced by a simple energy thresholding algorithm. Segmentation of the feature sequence into smaller segments is performed by an appropriately modified version of the ch_track tool that is provided as part of the Edinburgh Speech Tools Library [18]. For feature extraction and speech recognition, we use the HTK toolkit [16]. The HTK tools that are required cannot be packaged with SailAlign due to license restrictions and have to be downloaded and compiled separately.

The acoustic models currently used are triphone generic models trained on the Wall Street Journal and TIMIT corpora and are available online [19]. The acoustic features extracted from audio after preemphasis with a coefficient of 0.97 are 13 mean-normalized Mel Frequency Cepstral Coefficients using a 26-channel filterbank and their first and second derivatives every 10ms with a 25ms Hamming window. Pronunciations for each word in the transcription are generated using the CMU pronunciation dictionary [20], while additional dictionaries can also be used. Trigram language modeling is done by means of the SRILM toolkit [21], and we used Witten-Bell smoothing which is appropriate for language models built on limited datasets [22]. Text-text alignment is performed by the NIST sclite Scoring Package. SailAlign is released as open-source software and is available at: `http://sail.usc.edu/software.php`.

3. Experiments

The performance of SailAlign was evaluated in a word boundary detection task in the TIMIT database. We present the corresponding findings. Further, we present a pilot experiment using SailAlign to extract reading style information from audio books.

3.1. Word Boundary Detection in the TIMIT Database

Comparison of SailAlign with the standard Viterbi-based forced alignment was performed on an artificially created 1-hour audio chunk of the TIMIT database and its transcription. This was generated by randomly concatenating TIMIT audio recordings. By properly offseting the corresponding segmental transcriptions we were able to also generate the 1-hour audio chunk transcription and ground-truth alignment. The chunk duration was chosen to be similar to the duration of an audio book chapter. Alignment results for both algorithms are given in Fig. 2 for various tolerance levels. A word is considered to be aligned if each of its aligned boundaries differ from the corresponding ground truth start and end times less than the specified tolerance. Performance of both algorithms is similar. Also note that SailAlign does offer the possibility of post-processing resulting alignments with a Viterbi alignment step, although for clarity in the comparisons we did not use that functionality in this paper.

The real advantage of SailAlign becomes apparent when the audio is noisy or transcriptions are imperfect. To evaluate the robustness of the algorithm in such cases, we contaminated the

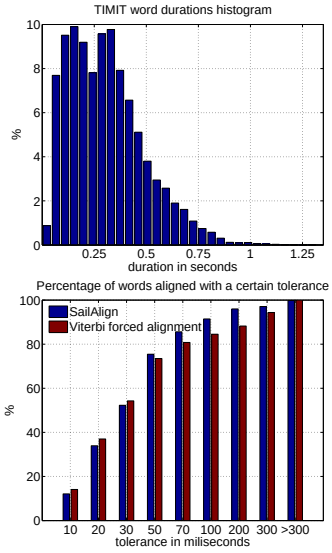

Figure 2: Top: Histogram of word durations for the 1-hour segment of the TIMIT database. Bottom: Percentage of aligned words using SailAlign or the Viterbi-based forced alignment.

TIMIT audio chunk with babble noise at various levels, and we corrupted the transcription by randomly introducing insertion, deletion, and substitution errors. Corresponding word alignment results are shown in Fig. 3 for a tolerance of 50 ms. Given that most of the words in the chunk have duration at least three times longer (Fig. 2, left), the choice of this tolerance level is reasonable. It appears that even in the cases of very low signal-to-noise ration (SNR), i.e., 10 and 5 dBs, SailAlign still provides acceptable results while the Viterbi-based forced alignment fails. Similarly, for imperfect transcriptions, even when 10% of the transcribed words are corrupted, SailAlign is robust enough to provide accurate alignment while Viterbi fails after the 3% corruption point.

3.2. Processing "Emma"

Using SailAlign, we processed four read versions of the book "Emma" by Jane Austen by four different speakers. On average, 150.090 words are aligned for each speaker and the corresponding average speaking time is approximately 11h and 17mins. We estimated the durations of all the spoken words and quantized them in 100 bins. We were then able to measure the frequency of appearance of each quantized duration value in each speaker's version. We show the log-log plot of these frequencies sorted in descending order versus the corresponding ordering rank in Fig. 4. Differences between the different contours are assumingly related with reading style differences among the readers. For example, the reader SC appears to use a significantly greater range of durations when reading.

4. Conclusions

We presented SailAlign, an open-source software that implements an adaptive, iterative long speech-text alignment algorithm. Alignment experiments with the TIMIT database demonstrate the increased robustness of the algorithm compared with the standard Viterbi-based forced alignment algorithm. Even when the transcription is imperfect or the audio is

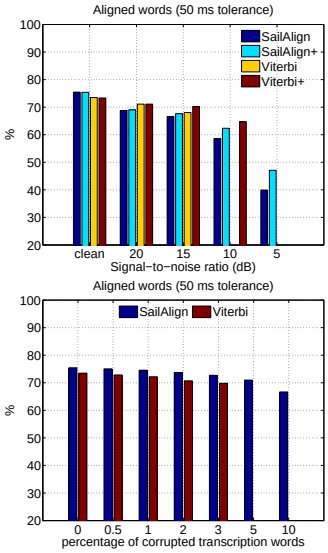

Figure 3: Top: Aligned words for various audio noise levels. Results for four cases are shown, i.e., SailAlign, SailAlign with acoustic model adaptation activated (SailAlign+), Viterbi forced alignment and alignment with adapted models (Viterbi+). Bottom: Aligned words for various levels of transcription corruption.

noisy, SailAlign manages to provide accurate alignment results while the standard forced alignment may fail. Results of a pilot experiment, run on four spoken versions of "Emma," further show the potential of the use of SailAlign for the exploitation of rich spoken language resources such as collections of audio books.

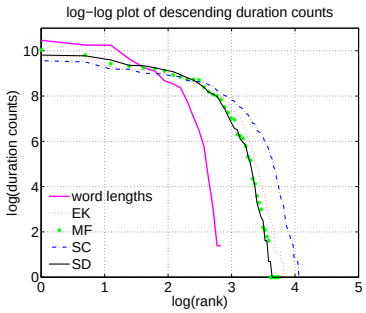

Figure 4: Log-log plot of the ordered duration counts versus their rank in this ordering. Durations are of the aligned words for the four readers of "Emma", i.e., EK, MF, SC, SD. The log-log plot of the corresponding word length counts is also given.

5. References

[1] P. J. Moreno and C. Alberti, "A factor automaton approach for the forced alignment of long speech recordings," in *Proc. IEEE Int'l Conf. Acous., Speech, and Signal Processing*, 2009.

[2] D. Caseiro, H. Meinedo, A. Serralheiro, I. Trancoso, and J. Neto, "Spoken book alignment using WFSTs," in *HLT'02 Proceedings of the second international conference on Human Language Technology Research*, 2002.

[3] J. Yuan and M. Liberman, "Vowel acoustic space in continuous speech: An example of using audio books for research." in *CatCod*, 2008.

[4] A. Ljolje and M. Riley, "Automatic segmentation and labeling of speech," in *Proc. IEEE Int'l Conf. Acous., Speech, and Signal Processing*, 1991.

[5] G. Margolin, P. H. Oliver, E. B. Gordis, H. G. O'Hearn, A. M. Medina, C. M. Ghosh, and L. Morland, "The nuts and bolts of behavioral observation of marital and family interaction," *Clinical Child and Family Psychology Review*, vol. 1, no. 4, pp. 195–213, 1998.

[6] J. Gottman, H. Markman, and C. Notarius, "The topography of marital conflict: A sequential analysis of verbal and nonverbal behavior," *Journal of Marriage and the Family*, vol. 39, no. 3, pp. 461–477, 1977.

[7] J. Jones and A. Christensen, *Couples interaction study: Social support interaction rating system*, University of California, Los Angeles, 1998. [Online]. Available: http://christensenresearch.psych.ucla.edu/

[8] M. P. Black, A. Katsamanis, C.-C. Lee, A. C. Lammert, B. R. Baucom, A. Christensen, P. G. Georgiou, and S. Narayanan, "Automatic classification of married couples' behavior using audio features," in *Proc. Int'l Conf. on Speech Communication and Technology*, 2010.

[9] C.-C. Lee, M. P. Black, A. Katsamanis, A. C. Lammert, B. R. Baucom, A. Christensen, P. G. Georgiou, and S. Narayanan, "Quantification of prosodic entrainment in affective spontaneous spoken interactions of married couples," in *Proc. Int'l Conf. on Speech Communication and Technology*, 2010.

[10] P. Moreno, C. Joerg, J.-M. van Thong, and O. Glickman, "A recursive algorithm for the forced alignment of very long audio segments," in *Proc. Int'l Conf. on Spoken Language Processing*, 1998.

[11] T. J. Hazen, "Visual model structures and synchrony constrants for audio-visual speech recognition," *IEEE Trans. Speech and Audio Process.*, vol. 14, pp. 1082–1089, 2006.

[12] M. Finke and A. Waibel, "Flexible transcription alignment," in *Proc. IEEE Workshop Automatic Speech Recognition and Understanding*, 1997.

[13] V. Zue, S. Seneff, and J. Glass, "Speech database development at MIT: Timit and beyond," *Speech Communication*, vol. 9, pp. 351–356, 1990.

[14] J. Austen. Emma. Audio book. [Online]. Available: http://librivox.org

[15] C. J. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 9, pp. 171–185, 1995.

[16] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, "The HTK book (for HTK version 3.2)," Cambridge University Engineering Department, Tech. Rep., Dec. 2002.

[17] P. K. Ghosh, A. Tsiartas, and S. S. Narayanan, "Robust voice activity detection using long-term signal variability," *IEEE Trans. Audio, Speech, and Language Processing*, 2010, accepted.

[18] S. King, R. Clark, A. W. Black, K. Richmond, and V. Strom. The edinburgh speech tools library. [Online]. Available: http://www.cstr.ed.ac.uk/projects/speech_tools/

[19] K. Vertanen, "Baseline WSJ acoustic models for HTK and Sphinx: Training recipes and recognition experiments," Cavendish Laboratory, Tech. Rep., 2006.

[20] R. Weide, "CMU pronouncing dictionary," Carnegie Mellon University, 1994. [Online]. Available: http://www.speech.cs.cmu.edu/cgi-bin/cmudict/

[21] A. Stolcke, "SRILM-an extensible language modeling toolkit," in *Proc. Int'l Conf. on Spoken Language Processing*, 2002.

[22] I. H. Witten and T. Bell, "The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression," *IEEE Trans. Inf. Theory*, vol. 37, pp. 1085–1094, 1991.