Semi-supervised learning for acoustic model retraining: Handling speech data with noisy transcript

Abhijith Madan, Ayush Khopkar, Shreekantha Nadig, K. M. Srinivasa Raghavan
Dhanya Eledath, V. Ramasubramanian
International Institute of Information Technology - Bangalore (IIIT-B), Bangalore, India
{abhijith.m, ayush.khopkar, shreekantha.nadig, srinivasaraghavan.km}@iiitb.org
dhanya.eledath@iiitb.org, v.ramasubramanian@iiitb.ac.in

***Abstract*—We address the problem of retraining a seed acoustic model from a large corpus which is associated with noisy labeling. We propose a forced-alignment likelihood and fuzzy string matching score based iterative selection of the corpus data to retrain the acoustic model in an order of increasing degree of noise in the transcript, yielding a succession of enhanced acoustic-models, offering progressively lower error rates on an held-out test data. We show results in terms of PER (phoneme-error-rate) on a large broadcast news data from a national broadcast network containing multiple languages of transcribed-speech, demonstrating the strong utility of such an approach for training of acoustic models from noisy-transcript.**

I. Introduction

We address the problem of efficient acoustic model training and retraining for the case when a large speech corpus is associated with noisy labeling. The proposed solution is adapted from a semi-supervised learning (also 'self-training') framework, typically applicable when the data is unlabeled. Here, in our scenario, a large 'labeled' speech corpus is considered to have noisy labels, i.e. the speech labels are considered as 'noisy transcripts'. Such a noisy transcript corpus is defined as having discrepancies between the speech and its orthographic labeling arising due to various reasons, such as for example, in broadcast news and human annotation errors.

To build an efficient speech recognition system using such 'noisy' transcription calls for techniques to select 'clean' utterances with relatively accurate alignments with the corresponding speech and using such well 'aligned' utterances to refine the acoustic model to yield lowering error-rates with the use of increased corpus from such noisy transcript data.

The broad framework of semi-supervised learning (and the closely associated active learning which has a complementary framework) has a long history in both machine learning in general [1] and particularly in speech recognition [2]. With respect to speech recognition, the early variants of semi-supervised learning were in the form of lightly-supervised acoustic model training and more recently has attracted renewed attention with the requirements arising from voice-search type of applications [3], [4] and low resource setting [5], [6], [7], [8], [9]. More particularly, in the context of handling noisy transcript, there has been some attention to generating forced alignments for long audios and to train acoustic models with alignment and correction techniques from human generated erroneous transcripts [10], [11], [12], [13], [14], [15].

We propose to adapt an approach of efficient acoustic-model refinement using a semi-supervised framework. We first present this framework to motivate the approach we adapt from this framework for the noisy-transcript handling. The corpus used is associated with noisy labeling instead of unlabeled data and is used in semi-supervised learning scenario in such a way that the large corpus can be treated similar to semi-supervised learning decoded output from a large unlabeled corpus with associated decoding errors.

II. Semi-supervised learning framework

Here, we outline a non-iterative semi-supervised framework which we adapt for the handling of noisy-transcript in this paper. This semi-supervised learning scenario essentially involves starting with a seed acoustic model trained from a small seed data with labeling (assumed available in a low resource setting), and be able to use a significantly larger data set without labeling (as is typical in a low resource setting) and establish means of using the unlabeled data in the larger data set to refine (retrain) the seed acoustic model in such a way that the resulting refined acoustic model performs on a held out test data with performances close to what would be obtained if the acoustic model were trained with the larger data labeled with ground-truth, i.e., on all of the available data, including the seed data and the larger data, with the larger data now being labeled - for purposes of establishing the best performance realizable from the entire available data as in a high-resource setting. This belongs to the class of 'self-training' approaches. The key idea used is to order the large unlabeled data in decreasing order of decoding accuracy (i.e. higher 'confidence measure' corresponding to lower errors), where the confidence measure (or confidence level) of the decoded utterance is defined as the posterior of each sentence

978-1-7281-8895-9/20/$31.00 ©2020 IEEE

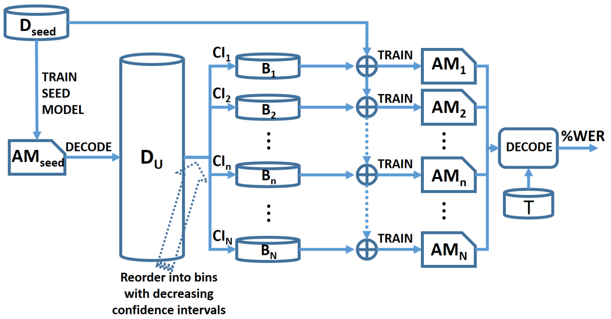

Fig. 1. Framework for semi-supervised learning

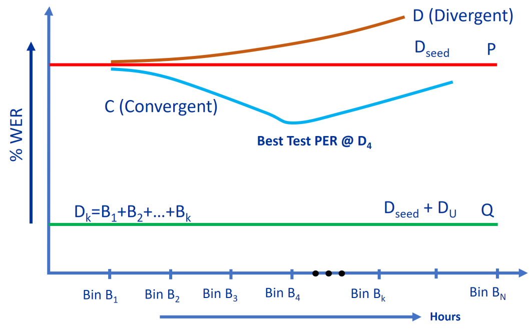

Fig. 2. WER profiles for the semi-supervised framework

(or word segment) with respect to the sentence-level (or word-level) label(s) it is aligned to in the decoding.

The semi-supervised learning framework is as in Fig. 1. Here, D_{seed} is the 'labeled' seed data set from which the seed acoustic model AM_{seed} is trained, D_U the unlabeled larger data set which is to be used by semi-supervised learning, and T the held out test data on which to perform the test decoding to derive the WER to characterize the efficacy of the acoustic model retrained by the semi-supervised learning on D_U.

The seed acoustic model AM_{seed} is used to decode D_U to derive word label sequences, with an inherent WER distributed across the utterances of the data set. Utterances with lower WER can be treated as close to ground truth labels and used for retraining AM_{seed}, thereby making available more data from D_U to improve AM_{seed}. Since the WER is not available (as the ground truth of D_U is by definition not available), other metrics by which the accuracy of the decoding of the utterances in D_U is measured are needed. One of the readily available measures is the confidence level of a decoded utterance, derived from the posterior of each word segment with respect to the word-level label it is aligned to in the decoding.

With the availability of the utterance level confidence level as a metric correlated to WER, in a 'non-iterative' procedure, as in Fig. 1, D_U is split into bins $B_n, n = 1, \ldots, N$ with confidence intervals $CI_n, n = 1, \ldots, N$. The bins, in the order of decreasing confidence levels, correspond to increasing WERs and can be used to derive acoustic models $AM_n, n = 1, \ldots, N$, with AM_1 derived from available training data, i.e., $D_{seed} + B_1$, with B_1 having utterances with the highest confidence levels, or lowest WERs, and likewise, AM_n, from $D_{seed} + B_1 + \ldots + B_n$ in a cumulative manner. By this overall 'non-iterative' procedure, we can derive progressively refined acoustic models $AM_n, n = 1, \ldots, N$, which have better decoding performance on the test data set T.

Fig. 2 shows a schematized WER profile (on T) for the non-iterative procedure of Fig. 1, for a typical split of the data into D_{seed}: D_U: T sets. This shows WER (on T) for different acoustic models:

1) AM_{seed} (trained with D_{seed} and with a WER marked as line 'P' on T).
2) $AM_{seed}+D_U$ which is the acoustic model derived from the combined data set 'D_{seed} and D_U with ground-truth labels' - this sets the performance limit (WER line marked $D_{seed} + D_U$ as 'Q') reachable by any semi-supervised protocol on D_U via decoding.
3) The semi-supervised models $AM_1, \ldots, AM_n \ldots, AM_N$ - whose WER profiles can of the types marked 'C' (for Converging) or 'D' (for Diverging).

The two WER profiles 'C' (Converging) and 'D' (Diverging) arise due to the following underlying conditions:

- **WER Profile 'C':** This profile is explained and understood as follows: AM_{seed}, trained with D_{seed}, induces a distribution of confidence levels on D_U, where the bins with a higher confidence levels are more populated, showing a good decoding for the higher confidence levels, progressively reducing for the lower confidence levels. With this bin distribution, the corresponding progressive WER-profile 'C' in Fig. 2, for AM_{seed}, AM_1, AM_2, $\ldots$, AM_N shows a 'convergent' behavior, i.e., the WER with addition of B_1 to D_{seed} results in an acoustic model AM_1 which is 'better' than AM_{seed} and correspondingly lowers the WER from 'P', which progressively decreases with increasing bins, until an intermediate bin (here for example, bin B_4), after which the WER increases, due to the addition of the lower bin B_5 with lower confidence levels and hence the latter stage re-training being affected by noisy, erroneous decoded labels, causing the acoustic models to train poorly.
- **WER Profile 'D':** Alternately, when the seed data D_{seed} is very small, the lower bins (corresponding to relatively higher confidence level decoding) suffer from noisy decoding (due to poorly trained seed acoustic model AM_{seed}). In this case, the WER profile 'D' has a divergent behavior in the sense that the progressive WERs of the acoustic models AM_{seed}, AM_1, AM_2, $\ldots$, AM_N gets worse than the baseline 'P' of the seed-model, due to the fact that even the first bin in the ordered set has highly erroneous decoding and label errors and makes the successive acoustic-model poorer i.e., 'diverge' away from the seed model.

III. Handling noisy-transcripts

We now propose two main adaptations of this semi-supervised learning framework for handling noisy-transcripts:

1) An acoustic model re-training protocol within this broader framework, but using forced alignment likelihoods in the place of Viterbi decoding likelihoods.

2) An ‘iterative’ variant of the above procedure, wherein the best acoustic-model (corresponding to the lowest PER (phoneme-error-rate) obtained in the non-iterative bin-wise retraining) to iteratively force-align the speech data with the noisy transcripts to realize improved representations of the likelihoods now obtained due to an improved acoustic model, and with the resulting re-ordered (data D, transcript T) pairs - carrying out the bin-wise re-training steps of acoustic model retraining until convergence over iterations.

The key to the advantage to be derived by adapting the above semi-supervised framework to the problem of handling noisy-transcript is as follows. Note that in Fig. 2, the best performance (WER on held-out test data) is realized at Bin 4, representing the size of the unlabeled data (sorted from Bin 1 in the order of decreasing confidence levels) that is optimal for re-training the acoustic-model. This is optimal in the sense that use of data beyond this Bin 4 is detrimental to the overall performance as the latter bins (i.e., bins beyond Bin 4) have ‘unreliable’ decoding, and have decoded transcript labels that are far too noisy to be relied on for acoustic-model retraining. This is the point of ‘convergent’ behavior alluded above.

Noting this point of convergent behavior or optimality, we now consider Fig. 3 to illustrate the main result we achieve (as in Fig. 6) in adapting the above semi-supervised learning framework to the problem of noisy-transcript handling. Here, the Bins are shown along the x-axis and the PER resulting from using $D_{seed} + D_k$ on the y-axis, where D_k corresponds to the cumulative data up to Bin B_k starting from Bin B_1 added to D_{seed} in retraining the acoustic model - which yields a particular PER (as in the y-axis). It can be seen that while D_{seed} gives a poor PER (19.1%), the use of $D_{seed} + D_{bulk}$, i.e, the entire ’speech data’ with its noisy transcript does indeed lower the PER (to 14.7%). However, going by the above treatment of semi-supervised learning, we expect the performance profile (as shown in the wavy curve) to ‘dip’ at some D_k (with a 8.4% PER) that is significantly ‘lower’ than the 14.7% simply by taking congizance of the fact that a partial ordered data from B_1 to B_k is advantageous for acoustic model training - as this ‘partial’ data has more ‘reliable’ paired labeled data (D, T) - thanks to the likelihood from forced-alignment identifying such more reliable data among the top-part of the sorted data. Going beyond this ‘optimal’ point of D_k allows less reliable data to be used in the retraining process, causing the PER to increase further - eventually reaching the 14.7% limit of using the entire ‘noisy transcript’ data - naively, without suspecting it is indeed noisy. Thus, any criterion to sort the data via forced alignment (or otherwise) can identify and select the top D_k data which is conducive for a good acoustic-model retraining. Specifically, a point of interest is the location of this optimal converging point D_k i.e., the absolute duration of data when Bin k occurs, e.g. in our case, 23 hours of sorted data according to forced alignment likelihood offers the best performance as against the full 28 hours of data. This is a pleasing result, as we have gained a 6.3% (absolute) improvement in PER (on test data) by identifying this optimal converging point - via the semi-supervised learning approach, without which the use of the entire data would tantamount to training a poor acoustic model with a 6.3% higher PER. The figure Fig. 3 shows the PERs as obtained in the actual experiments reported here (in Sec. V-C) and is shown here upfront to convey the nature of the main result and the reasoning behind it.

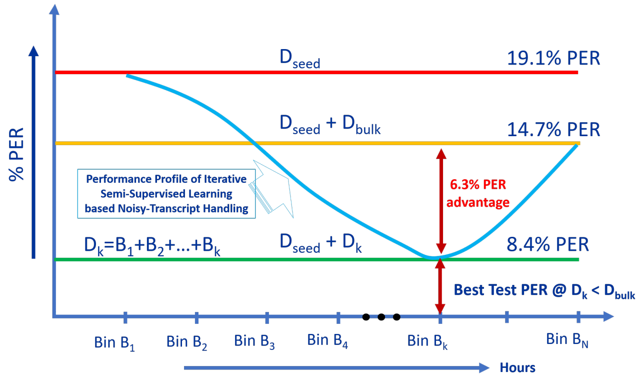

Fig. 3. The PER profile and advantage derived by the proposed semi-supervised learning framework for handling noisy-transcript

IV. Noisy-transcript data-set

A large archive of national broadcast news articles was obtained whose speech as well as the transcript corresponding to the speech were available online. In this work, around 30 hours of speech-transcript data for the regional language Kannada (an Indian language) was chosen to work with to show effectiveness of the approach.

Single channel speech signals were used for training. Each speech file was approximately 12 minutes in length and contained around 800-1000 words in the transcript file. The speech and transcript were split into length of 20 seconds each along with the corresponding transcripts so that it aligns with the speech (as outlined in Sec. V-B)

A. Speech-Transcript Dissimilarities

There were multiple errors present in the transcript with respect to the speech. Music was present in the beginning and ending of the speech files. The speech signal would contain advertisements along with background music whose utterances aren’t transcribed in the text. The news segments contain small recorded statements of other speakers such as politicians, actors or even phone calls to other people - such conversations aren’t transcribed in the transcript. Since the speech signal is a news segment, it is subjected to last minute changes from time to time hence the speaker sometimes skips an entire paragraph from the transcript. The speaker sometimes deviates from what is present in the transcript by mistake and may utter something which isn’t completely present in the transcript. The transcript contained numbers which needed to be converted to the way it is supposed to be uttered. A number-to-words converter fixed these kind of errors. Some of the errors mentioned above were removed automatically by using the methods described in Sec. V-B. The remaining errors were retained since they could not be automatically identified. It is due to this reason that there was a need to utilize the approach to select high accuracy transcripts from a bulk selection of noisy transcripts.

V. Algorithm for noisy-transcript handling

The proposed iterative procedure for handling noisy-transcripts to retrain the acoustic-model set in a semi-supervised learning framework outlined above is illustrated in Fig. 4. Here, a small seed data D_{seed} is used to train a seed acoustic model AM_{seed}. A metric is needed to reorder the bulk data D_{bulk} so that sentences with high confidence measure can be used for retraining AM_{seed}. We have used Viterbi forced alignment (FA) likelihood score (FA score) as a metric for measuring the confidence level of the sentence. AM_{seed} is used for forced alignment of the D_{bulk} so as to reorder according to the Viterbi forced alignment likelihood value. After the reordering of D_{bulk} is done, it is split into bins $B_n; n = 1, ..., N$ where N is the number of bins.

The bins are arranged in the order of decreasing FA likelihoods which implies increasing order of noise and are further used to derive acoustic models $AM_n; n = 1, ..., N$ such that AM_1 is trained using $D_{seed} + B_1$ where B_1 contains utterances having highest likelihood values or lowest noise, and likewise, AM_n is derived from $D_{seed} + B_1 + ... + B_n$ in a cumulative manner. On completion of training of all acoustic models i.e. $AM_n; n = 1, ..., N$, the test data D_{test} is decoded using the newly trained AMs to compute PER. Acoustic model corresponding to the lowest PER is selected to again perform Viterbi forced alignment on the D_{bulk} and reorder all utterances.

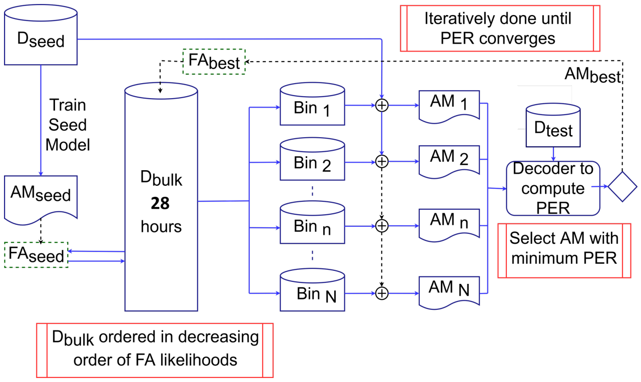

Fig. 4. Proposed iterative retraining procedure

Training of all AM_i corresponding to each B_i and then selecting the AM_i with minimum PER constitutes one complete iteration. This process of splitting D_{bulk} into bins, training the AM for each bin and then force aligning D_{bulk} using AM having minimum PER is repeated iteratively until the PER converges. PER convergence is achieved when the PER does not reduce further from what was obtained in the previous iteration - as illustrated in Fig. 3 - and whose actual results will be shown in the further sections.

A. Acoustic model framework

In this work, we have used DNN-HMM framework trained using Kaldi [16]. The scripts from the Kaldi 'TIMIT recipe' were modified to suit the requirements. A phone level tri-gram language model is trained on the training corpus using the Kaldi-LM language model available in Kaldi.

To build a seed ASR model and test set, around 4.5 hours (D_{clean}) of data was manually corrected by listening and reading through the transcripts. The speech file was used as the ground truth and corrections were made on the transcript when discrepancies in the speech and transcript files was found. From D_{clean}, 1 hour of data (D_{seed}) was used as the seed data for the seed model training, the remaining 3.5 hours (D_{test}) of data was used in the test set to calculate the efficiency of the model by computing the PER. D_{bulk} is 28 hours of speech data with noisy transcript, further processed in the following section to create 20 sec segments of speech - (noisy) transcription pairs on which the above algorithm is applied.

B. Noisy data alignment and scoring

Here, we outline a method to automatically align the noisy-transcript with the speech in small segments of 20secs each, to further compute such paired-data's forced alignment scores. Sec. IV showed that the speech files were approximately 12 minutes long. These were split into lengths of 20 seconds by identifying points of silence. This was done so that the speech file is not split during the pronunciation of a word. Each news segment now consisted of approximately 30-35 smaller utterances of length 20 seconds each. The seed acoustic model is used to obtain decoding transcripts for the split speech files by performing speech-to-phoneme decoding. After this process, there exists two sets of transcripts: i) the original full-text transcript ($T_{original}$) and ii) the smaller transcript (T_{decode_i}) generated by the decoding of the smaller speech files where i is the number of utterances per news segment.

Each of the T_{decode_i} is then searched in the entire $T_{original}$ file with an approximate string search. This algorithm compares two text files which give a score out of 100 that indicates how close the two strings are with respect to one another. The approach of finding the similarity index score is done by calculating the ”Levenshtein Distance” [17] of the two strings. To use Levenshtein distance to compare two strings works well when the strings are of comparable lengths, but since the entire $T_{original}$ is much larger than any of the T_{decode_i}, we cannot directly use it, Python's difflib library is used which contains a class, to be specific, SequenceMatcher which looks at sets of arrangements of any kind, and is appropriate to use as long as the grouping components are hashable. The idea is to locate the longest contiguous matching subsequence present in the larger string by searching with the smaller one.

SequenceMatcher is quadratic time for the worst case and has an average-case behavior dependent on how many components the sequences share in common; the best case time is linear. Once the matching blocks are found, the best match is selected by using the ”Levenshtein Distance” and is saved to a file. This way the $T_{original}$ is split into smaller chunks that are aligned to that of the split speech files.

Fuzzy string matching with respect to edit distance is the use of edit distance as a measurement and finding the

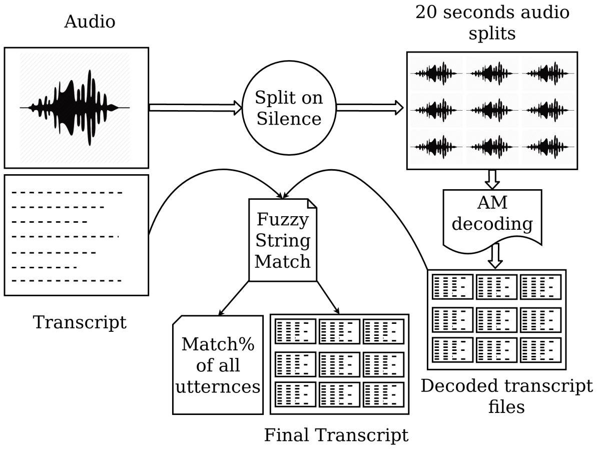

Fig. 5. Generation of aligned data segments by approximate string search using Fuzzy String Matching

minimum edit distance required to match two distinct strings together. This work uses fuzzywuzzy [18] to calculate the match percentage of two strings which internally uses the Levenshtein distance and SequenceMatcher. The best match is selected from the list of matching subsequences and is stored as the transcript for the particular speech.

C. Experiments and Results

Using the corpus generated as above in terms of 'paired and aligned' speech and noisy-transcript segments, the algorithm described in the proposed architecture (Fig. 4) was applied starting from training the AM_{seed} and then repeating the steps of splitting into bins and training them for two iterations named as 'A Loop' and 'B Loop'. Fig. 6 shows the results obtained for the experiment performed.

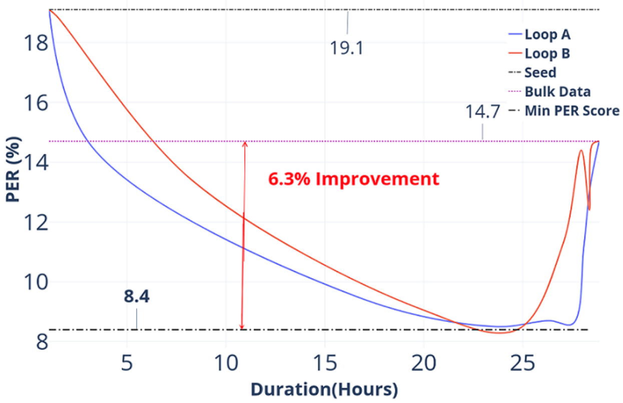

Fig. 6. Performance profiles of proposed approach

From Fig. 6, two major observations are drawn. First, the best model (AM_{best}) gave of a PER of 8.4% which is an improvement of 6.3% PER when compared with the AM_{bulk}'s PER of the complete bulk data which was 14.7%. Second, with this significant improvement in PER, the minimum PER from AM_{best} is achieved by using approximately 23 hours of D_{bulk} for training which constitutes around 80% of D_{bulk} - with the overall performance as presented in Sec. III and Fig. 3.

VI. Conclusion

We have addressed the problem of retraining a seed acoustic model from a large corpus associated with highly noisy transcripts by adapting an iterative bootstrap semi-supervised learning (SSL) framework. The SSL approach realizes a 6.3% PER improvement using the forced-alignment score as a metric to order the utterances. This semi-supervised process can be used for efficient acoustic-model training from large data with noisy transcriptions.

VII. Acknowledgement

This work was supported by Karnataka Innovation & Technology Society, Dept. of IT, BT and S&T, Govt. of Karnataka vide GO No. ITD 76 ADM 2017, Bengaluru; Dated 28.02.2018.

References

[1] J. Zhu, "Semi-supervised learning literature survey", Computer Sciences, Univ. of Wisconsin-Madison, Tech. Rep., 2006.

[2] Li Deng and Xiao Li, "Machine Learning Paradigms for Speech Recognition: An Overview", IEEE Trans. on Audio, Speech and Language Processing, vol., 21, no. 5, pp. 1060-1089, May 2013.

[3] Francoise Beaufays, Vincent Vanhoucke and Brian Strope "Unsupervised discovery and training of maximally dissimilar cluster models", Proc. Interspeech 2010.

[4] Olga Kapralova, John Alex, Eugene Weinstein, Pedro Moreno and Olivier Siohan, "A big data approach to acoustic model training corpus selection", Proc. Interspeech 2014

[5] S. Novotney, R. M. Schwartz, and J. Z. Ma, "Unsupervised acoustic and language model training with small amounts of labelled data", Proc. ICASSP, pp. 42974300, 2009.

[6] S. Novotney and R. M. Schwartz, "Analysis of low resource acoustic model self-training", Proc. Interspeech, pp. 244247, 2009.

[7] Marelie H. Davel, Charl van Heerden, Neil Kleynhans and Etienne Barnard, "Efficient harvesting of Internet audio for resource-scarce ASR", Proc. Interspeech 11, pp. 3153-3156, Florence, Italy, 2011.

[8] Neil Kleynhans, Febe de Wet and Etienne Barnard, "Unsupervised acoustic model training: comparing South African English and isiZulu", Proc. Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), pp. 136-141, Port Elizabeth, South Africa, Nov. 2015.

[9] M. Chellapriyadharshini et al., "Semi-supervised and active-learning scenarios: Efficient acoustic model refinement for a low resource Indian language," *Proc. Interspeech '18, Hyderabad, Sep 2018.*, 2018.

[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. of ICSLP, Sydney, Australia, December 1998.

[11] T. Hazen. Automatic alignment and error correction of human generated transcripts for long speech recordings. Proc. ICSLP, 2006.

[12] S. Novotney and C. Callison-Burch. Cheap, fast and good enough: automatic speech recognition with non-expert transcription. Proc. ACL, 2010, Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL, pages 207215, Los Angeles, California, June 2010.

[13] K. Yu, M. Gales, L. Wang and P. C. Woodland. Unsupervised training and directed manual transcription for LVCSR. Speech Communication, 2010.

[14] A. Katsamanis, M. Black, P. Georgiou, L. Goldstein and S. Narayanan. SailAlign: Robust Long Speech-Text Alignment. In Proc. of Workshop on New Tools and Methods or Very-Large Scale Phonetics Research, Jan. 2011.

[15] S. Tanamala, J.J. Prakash and Hema Murthy. A Semi-Automatic Method for Transcription Error Correction for Indian Language TTS Systems. In Proc. NCC, Chennai, India, 2017.

[16] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The Kaldi speech recognition toolkit," Tech. Rep., IEEE Signal Processing Society, 2011.

[17] Li Yujian and Liu Bo, "A normalized Levenshtein distance metric," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 1091–1095, 2007.

[18] Fuzzywuzzy usage documentation https://pypi.org/project/fuzzywuzzy/, Last Accessed: 2019-04-25.