RERANKED ALIGNERS FOR INTERACTIVE TRANSCRIPT CORRECTION

Benoit Favre, Mickael Rouvier, Frederic Bechet

Aix-Marseille Université, CNRS, LIF UMR 7279, 13000, Marseille, France
{firstname.lastname}@lif.univ-mrs.fr

ABSTRACT

Clarification dialogs can help address ASR errors in speech-to-speech translation systems and other interactive applications. We propose to use variants of Levenshtein alignment for merging an errorful utterance with a targeted rephrase of an error segment. ASR errors that might harm the alignment are addressed through phonetic matching, and a word embedding distance is used to account for the use of synonyms outside targeted segments. These features lead to a relative improvement of 30% of word error rate on sentences with ASR errors compared to not performing the clarification. Twice as many utterances are completely corrected compared to using basic word alignment. Furthermore, we generate a set of potential merges and train a neural network on crowd-sourced rephrases in order to select the best merger, leading to 24% more instances completely corrected. The system is deployed in the framework of the BOLT project.

Index Terms— Error correction, Dialog systems, ASR error detection, Reranking, Levenshtein alignment

1. INTRODUCTION

Automatic Speech Recognition systems often generate imperfect transcripts mainly due to challenging acoustic conditions, out-of-vocabulary words or language ambiguities that could only be handled with world knowledge and long term context analysis. Even though there has been a large body of work on robust ASR [1, 2], advances in open-vocabulary speech recognition [3, 4] and long-range language modeling [5, 6, 7], ASR systems still make errors which can impact downstream applications. Interactive systems offer an opportunity for ASR errors to be detected and corrected through clarification dialogs. In closed-domain dialog systems, methods have been developed for explicit and implicit confirmation of user intent [8, 9], but they cannot be applied to open-domain speech because of lack of prior on the message to be understood.

Nevertheless, recent efforts in confidence measure estimation [10, 11], OOV detection [12, 13, 14], error detection [15] and error characterization in term syntactic and semantic classes [16, 17] enable to accurately locate error segments in an ASR transcripts and engage clarification dialogs in order to improve the system's understanding of what the user meant. Such clarification systems can ask the user to disambiguate homophones or word senses, spell out-of-vocabulary words, or rephrase part of her original utterance in order to correct detected errors [18, 19]. These strategies focus on targeted errors instead of asking to rephrase the whole sentence, mimicking how humans correct understanding errors, and therefore leading to more natural and intuitive interactions. In particular, repeated clarification of all errors in a sentence can lead to a perfect transcript in much shorter time.

In this paper, we address the problem of editing the user's original utterance with the answer to a clarification question partially rephrasing the original. The paper is a follow up work to that of [16] which explored error detection and recovery for a speech-to-speech interactive translation system. Our contributions are the following:

- We propose Levenshtein alignment variants for merging an original and clarification utterances given an error segment, with cost functions tailored to the specifics of the task and accounting for ASR errors and paraphrasing through phonetic and word embedding similarity.
- We then rerank merging variants with a Multi-Layer Perceptron trained on crowd-sourced examples, using various features such as probabilities from a Recurrent Neural Network Language Model and agreement between mergers.
- The resulting systems are evaluated on a challenging dataset collected for the BOLT project in which all utterances have at least one ASR error.

The paper is organized as follows: Section 2 lists related work; Section 3 presents task specifics; the mergers are described in Section 4 and the reranker in Section 5. Results are discussed in Section 6 and Section 7 concludes the paper.

2. RELATED WORK

The task of improving automatic transcripts with interactions has mainly been pursued through confirmation in dialog systems and edition commands in multimodal systems ("replace word...").

There is a large body of work on confirming user input in dialog systems [8]. The most simple way to do it is to resynthesize the transcript and ask the user to confirm it or utter a replacement. While generic, this approach is very tedious on longer utterances. Slot-based SLU systems rely on slot confidence in order to only ask confirmation on specific slot values [20]. In addition, systems with strong semantic modeling can assess the coherence of their belief of user inputs in order to target specific values [21]. But the semantic approach cannot be followed in the framework of open-domain dialogs. In that case, task specifics are accounted-for to determine which words to confirm, such as the metrics proposed in [22] for information retrieval. In speech-to-speech translation applications, generic clarification should focus on editing the transcript (or the machine translation output) so that it fits the user's intent.

When multimodal interactions are possible, cross-modal cues have been used to generically improve transcripts. For instance, dictation systems such as Dragon Naturally Speaking allow the user to see mistakes in the displayed transcript and correct them with voice

This work was partially funded by DARPA HR0011-12-C-0016 as an AMU subcontract to SRI International.

commands such as "select", "correct", "spell that". Additionally, in order to make the correction more efficient, a list of alternatives can be extracted from word confusion networks to make the selection easier [23, 24, 25]. In [26], eye gaze was used to select incorrect words in a displayed transcript. In our work, we are interested in situation where speech is the only modality used to interact with the system, and therefore those options do not apply.

In the natural language processing community, there has also been work in sentence fusion, the generation of text from multiple textual sources, which is of interest to our work. Sentence fusion was mainly developed for multidocument summarization in order to obtain shorter versions of sentences with overlapping content. This task is achieved by merging parse trees [27, 28] using an edit distance which can be learned from manual edits [29]. Parsing erroneous ASR output is very challenging, especially when processing partial sentences, but we retain the idea that alignment can be useful for merging sentences.

3. TASK

The task studied in this paper consists in generating a better intended user utterance transcript given an errorful **original** sentence and the **answer** to a clarification question geared towards an **error segment** relative to the original. We call that task **utterance merging**. Words inside the error segment are considered wrong and have to be replaced by words from the clarification answer. Furthermore, there might be errors outside of the clarified segment which could also be repaired by the answer. An example of clarification dialog would run as follows:

Original speech	Where is the **hyperbaric** chamber?
ASR	where is *that* **hyper bar ick** chamber
Detected error	hyper bar ick
Question	Can you rephrase AUDIO(hyperbaric)?
Answer	the high pressure chamber
Edited utterance	where is the **high pressure** chamber

From that example, the input of a merging system would be *original: "where is that* $\langle err \rangle$ *chamber", answer: "the high pressure chamber"* and the reference output would be "where is the high pressure chamber".

We have built systems performing the utterance merging task in the context of live speech-to-speech translation systems in the context of the BOLT project. A study of system logs revealed that users generally adopt the following behavior (see [16]):

- The clarification answer exactly fits the error segment.
- The answer contains additional words to contextualize the editing operation
- The answer is a complete rephrase and should be used in place of the original.
- The answer does not fit the syntactic context of the original (*"the* $\langle err \rangle$ *chamber"* $\Rightarrow$ *"the chamber for oxygen therapy"*)
- Words from the original can be rephrased for conciseness (i.e. use a pronoun in place of a noun phrase)
- Convenience phrases are used to introduce the answer (*"I said that . . . "*)

The following sections present an approach for performing utterance merging tailored to account for the user behaviors listed here.

4. SYSTEM

Generally, given a targeted clarification question, the user utters words which fit the error segment and additional words to contextualize the edit, up to the whole utterance. Therefore, our system is designed around the idea of aligning answer words with the original and replace the error segment with words which would fit at that location. Unfortunately, this strategy is not always successful at generating an acceptable transcript, so we devise a range of variants and train a reranker in order to select the best variant.

Our first two systems are baselines: replace all the words from the original with that of the clarification utterance, or as an alternative, insert all words from the clarification utterance in place of the error segment. Then all other systems are variants of Levenshtein alignment applied in the framework of finite state transducers. In this framework, utterances are represented as linear acceptors which recognize the sequence of uttered words. In the original utterance acceptor, the error segment is replaced by a sub automaton recognizing a sequence of one or more $\langle err \rangle$ tokens which will be matched with repair words (error loop). The alignment is performed by composing these acceptors with an edit transducer which maps words to edit operations: insertions, deletions and substitutions. If A_o and A_c are acceptors which represent the original and clarification utterances, and T_e is the transducer encoding the edit operations, the best path of the composition $A_o \circ T_e \circ A_c$ yields an alignment between the utterances. Figure 1 depicts the structure of those automata. Performing the alignment in the transducer framework makes it easier to implement task-specific constraints and can be extended to word lattices instead of sequences. The final merged utterance is obtained by outputting words from the alignment, using those of the clarification when different from the original.

An affine gap cost function is used to score the edit operations [30]. The cost of a sequence of insertions/deletions is split in two components: α for starting a segment of edits, and β to continue it (see Figure 1). This kind of alignment cost is expected to group misalignments together and result in more compact matches. Moreover, the cost of substituting a word to a $\langle err \rangle$ token is 0, so that clarification words are placed in priority in the error segment. The cost of a substitution, γ, is computed by comparing the two words to be substituted with the following features:

- Wordnet similarity: $\gamma = 0.5$ if the two words to align share a synset in Wordnet, otherwise $\gamma = 1$.
- Word embedding similarity: γ is the cosine similarity between vector representations of the substituted words in a 300-dimensional embedding. This embedding has been trained on words from the TRANSTAC in-domain corpus, taking the hidden layer of a neural network predicting each word given its context [31].
- Phonetic similarity: γ is the minimum edit distance between word phonetization variants from the CMU pronouncing dictionary [32].

The mergers used in our work are devised from various combinations of these costs, resulting in different merging properties and accuracy.

5. RERANKER

The reranker aims at selecting from a set of merge hypotheses the one which will generate the best transcript according to the user intent. The problem is cast as binary classification: given a set of fea-

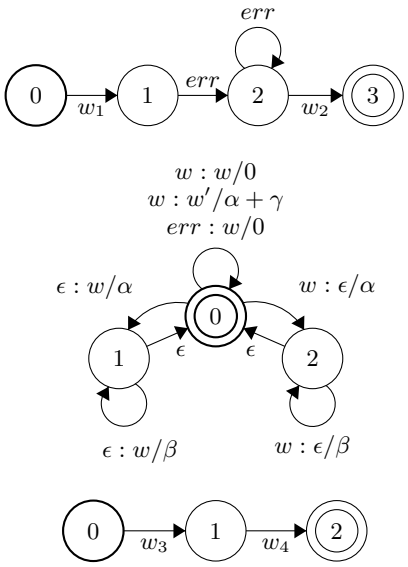

Fig. 1. Affine gap alignment with transducers. The top acceptor A_o is the original sentence with an error loop at the error segment; the bottom acceptor A_c is the clarification answer; the middle transducers T_e is a template for the edit transducer (for instance, each word w can be substituted by another word w'). The edited utterance is obtained from the best path of the composition $A_o \circ T_e \circ A_c$. α is the cost for starting an insertion/deletion segment, β is the cost for continuing it and γ is the substitution cost.

tures characterizing the output of one of the mergers, is this output the human-written reference merge or not?

We extract 14 features to represent the output of a merger. A first set of features is directly related to the merger being evaluated:

- Merger identity
- Cosine similarity with original, with clarification
- Min, max and mean word-level score given by a Recurrent Neural Network language model (RNN)
- Merged sentence length

In addition, another set of features is devoted to modeling the edit operation:

- Number of other mergers agreeing on merged sentence
- Levenshtein distance with original, clarification
- Length difference with original, clarification
- The sentence is included in the original, in the clarification

The reranker is a Multi-Layer Perceptron (MLP) with 6 layers (1 input layer, 4 hidden layers and 1 output layer), trained with the backpropagation algorithm. The input, hidden and output layers are composed respectively of 14, 10 and 1 neurons. In training, the output neuron is set to $+1$ if the merge hypothesis corresponds to the reference and to -1 otherwise. We use a large set of clarification dialogs with reference obtained from crowd sourcing (described in Section 6) as training corpus and the RNN is trained on the TRANSTAC in-domain corpus. Finally, given a set of merger hypotheses, the selected merger is the one that gets the highest score[1] given by output neuron.

[1]For the MLP we use FANN (http://leenissen.dk/fann/wp) and for the RNN we use RNNLM (http://www.fit.vutbr.cz/ imikolov/rnnlm).

6. EXPERIMENTS AND RESULTS

The approach developed in this paper is tested in the framework of BOLT Task b/c, machine-mediated human-human bilingual conversation. The system plays the role of an interpreter who can take the initiative to clarify user input before translating it. An ASR and MT error detector is run upfront to generate error segments about which the dialog module asks for a targeted rephrase. The merger processes the answers to clarification questions to produce a better transcript before translation. In the following, we only look at the English side even though both languages are processed the same way.

In order to assess the quality of our system, we produced two corpora: a text corpus of targeted rephrases collected through Amazon Mechanical Turk (AMT), mainly used for training, and speech recordings reproducing the kind of clarification dialogs in BOLT. The first corpus is a set of 900 in-domain targeted rephrases for which we asked turkers to rephrase a random part of a sentence (3 turkers $\times$ 3 random segments $\times$ 100 input sentences). The dataset is then upsampled by randomly extending the error segments and clarification boundaries to obtain 11,775 unique instances. The speech corpus is a set of 70 dialogs for which the original utterance contains at least one ASR error segment (the targeted error) and might contain additional errors. This corpus is very challenging for ASR because most errors are induced by OOVs. ASR transcripts were obtained by running a DNN-based ASR systems developed by SRI in the course of the BOLT project. Its word error rate is 30.60 on the original utterances and 14.7 on the clarification answers. Note that in all experiments, we consider that targeted error segments have been correctly located by the error detection module. In the following, we call *AMT* the corpus collected though crowd sourcing and *Speech* the corpus of speech recordings.

In the experiments, the following mergers are compared: *replace* and *insert* baselines, basic Levenshtein alignment without an error loop (*Align no-err-loop*), alignment with an error loop (*Err-loop*), alignment with affine-gap costs and an error loop (*Err loop + affine gap*; $\alpha = \beta = \gamma = 1$), the same with phoneme sequences in place of words (*Phonemes only*), affine-gap and error loop with γ the phonetic similarity between words (*Phonetic + words*), the same with γ the minimum between the Wordnet similarity and the phonetic similarity (*Phonetic + Wordnet*), and γ the minimum between the embedding similarity and the phonetic similarity (*Phonetic + embedding*).

Merging performance is evaluated with two metrics: merging accuracy represents the rate of complete recovery compared to the human-written reference, and merging word error rate (WER) is the word error rate of the hypothesis compared to the reference merge that should have been produced (it is not the WER compared to the original reference transcript).

Table 1 shows the results for the different mergers on the *AMT* dataset. All alignment-based mergers perform better than the baselines, leading to a large reduction in WER. The best merging strategy is to perform affine-gap alignment with an error loop and a substitution cost based on both the phonetic edit distance between words and their Wordnet similarity, each feature of the alignment strategy yielding an improvement. The knowledge-poor alignment (without phonetic lexicon nor semantic representation) already performs well with an accuracy of 75.02%. The reranker offers an additional improvement of 10.4% in accuracy and 31.8% in term of WER over the best merger. Note that on this particular corpus, the method proposed in [16] only beats the baselines and basic alignment.

In table 2, we report the same results for the *Speech* corpus for reference text ASR output. The general behavior of mergers is the

Method	Acc.	WER
Replace (baseline)	45.94	20.24
Insert (baseline)	25.41	32.56
Align no err-loop	61.84	13.00
Err loop	69.76	07.30
Err loop + affine gap	75.02	05.37
Phonemes only	51.86	10.51
Phonetic + words	77.60	04.74
Phonetic + Wordnet	80.02	03.99
Phonetic + embedding	78.70	04.42
Reranker	**88.36** $\pm$0.72	**02.72** $\pm$0.23
Oracle	96.18	00.56
[16]	63.70	12.44

Table 1. Merge accuracy and WER results for the mergers and reranker on the *AMT* dataset. Reranker results are averaged on 100 runs. The reranker oracle is computed by systematically selecting the best merger output in term of accuracy. For comparison purposes, we also give results for the method proposed in [16].

same as on the text corpus, except that accuracy (obtaining the correct transcript of the reference merge) is much lower for ASR output. Still, WER is greatly improved compared to using the original without clarification, with a gain of about 10 points absolute. On speech, the Wordnet semantic resource is worse at modeling substitutions than the word embedding, probably because the continuous space is more robust when there are ASR errors. The method proposed in [16] yields good results on reference text but it fails to improve over not clarifying in term of WER on ASR output, a motivational factor for the current work. Concerning the reranker, it obtains an accuracy close to the oracle (selecting the best system), which is favorable in term of perceived accuracy, but the WER improvement is not as high as on *AMT*. This suggests that (1) more merger strategies have to be explored to improve the oracle, and (2) the reranker should be trained to minimize WER in addition to maximizing accuracy.

	Ref.		ASR	
Method	Acc.	WER	Acc.	WER
Replace (baseline)	25.71	49.08	12.86	55.18
Insert (baseline)	37.14	28.09	08.57	46.52
Align no-err-loop	35.71	21.56	10.00	32.34
Err-loop	70.00	08.09	17.14	22.27
Err loop + affine gap	84.29	02.13	21.43	21.56
Phonemes only	74.29	04.68	15.71	21.56
Phonetic + words	85.71	**01.84**	21.43	21.13
Phonetic + Wordnet	82.86	02.70	18.57	21.28
Phonetic + embed.	85.71	**01.84**	21.43	20.99
Reranker	**87.57** $\pm$1.36	01.93 $\pm$0.58	**26.67** $\pm$0.78	**20.97** $\pm$0.58
Oracle	90.00	1.42	28.57	16.17
[16]	84.29	02.27	15.71	30.64
No clarification	0.0	15.14	0.0	30.60

Table 2. Accuracy and WER results on the *Speech* corpus according to merger variants, the reranker, the oracle, the results of [16] and the result if no clarification is performed. The first two columns correspond to the reference transcript while the other show ASR results. The reranker oracle is computed by selecting the best system for each hypothesis.

	AMT	Speech	
	Text	Ref.	ASR
Classifier acc.	90.65 $\pm$1.68	90.83 $\pm$1.43	58.74 $\pm$2.00

Table 3. MLP Accuracy on *AMT* and *Speech* corpora.

In Table 3 we report classification accuracy obtained by the reranker on the *AMT* and *Speech* corpora. *AMT* results are given in a 2-fold setting (results are the average of experiments run with half of the corpus as test set), and *Speech* results are given for a model trained on the whole *AMT* data. Since the weights of the MLP are randomly initialized, we report mean accuracy and standard deviation over 100 runs. It can be observed that on reference text the reranker obtains an accuracy of 90.65% and 90.83% respectively for the *AMT* and *Speech* with reference transcript, while on automatic transcription the accuracy falls to around 58%. This difference can be explained by the fact that the reranker is trained on reference text (*AMT*) and thus it is less robust to ASR errors.

Table 4 shows how often each merger was classified as +1 by the reranker (output neuron > 0) on the *Speech* corpus, and the ratio of correct guesses among them. Note that the number of correct is bound by the oracle, which explains why it is lower on ASR output. Since the reranker is trained on clean text, systems tend to be selected less often on ASR output because of word errors which lower the number of matched words between original utterances and clarifications. Note that even though multiple mergers can be classified as +1 on a given instance, we only use the argmax as output of the reranker.

	Ref.		ASR	
Method	Sel.	Acc.	Sel.	Acc.
Replace (baseline)	27.14	94.74	22.86	50.00
Insert (baseline)	47.14	78.79	45.71	18.75
Align no-err-loop	35.71	100.00	22.86	43.75
Err-loop	74.29	94.23	51.43	33.33
Err-loop + affine-gap	94.29	87.88	80.00	26.79
Phonemes only	80.00	89.29	58.57	24.39
Phonetic + words	92.86	89.23	77.14	27.78
Phonetic + Wordnet	87.14	91.80	70.00	26.53
Phonetic + embedding	92.86	89.23	81.43	26.32

Table 4. Mergers classified as +1 by the reranker on the *Speech* corpus in percentage of instances and rate of correct among them.

7. CONCLUSION

In this paper we propose a system for addressing the problem of merging an answer to a clarification dialog with the errorful original utterance. We propose Levenshtein alignment variants and a reranker to select the best hypothesis. The method results in a relative improvement of about 30% compared to not clarifying the input, the reranker helping to completely remove the error in 26% of the instances, almost reaching the oracle.

For future work, we will investigate the use of non-monotonic alignment with methods elaborated from bitext alignment in machine translation, and make use of word lattices in the merge operation.

8. REFERENCES

[1] Andrew L Maas, Quoc V Le, Tyler M O'Neil, Oriol Vinyals, Patrick Nguyen, and Andrew Y Ng, "Recurrent neural networks for noise reduction in robust asr.," in *Interspeech*, 2012.

[2] Felix Weninger, Martin Wollmer, Jurgen Geiger, Bjorn Schuller, Jort F Gemmeke, Antti Hurmalainen, Tuomas Virtanen, and Gerhard Rigoll, "Non-negative matrix factorization for highly noise-robust asr: To enhance or to recognize?," in *ICASSP*. IEEE, 2012, pp. 4681–4684.

[3] Matteo Gerosa and Marcello Federico, "Coping with out-of-vocabulary words: open versus huge vocabulary asr," in *ICASSP*. IEEE, 2009, pp. 4313–4316.

[4] Carolina Parada, Mark Dredze, Abhinav Sethy, and Ariya Rastrow, "Learning sub-word units for open vocabulary speech recognition.," in *ACL*, 2011, pp. 712–721.

[5] Rebecca Jonson, "Dialogue context-based re-ranking of asr hypotheses," in *SLT*. IEEE, 2006, pp. 174–177.

[6] Jeff Mitchell and Mirella Lapata, "Language models based on semantic composition," in *EMNLP*. Association for Computational Linguistics, 2009, pp. 430–439.

[7] Xunying Liu, Mark JF Gales, and Philip C Woodland, "Use of contexts in language model interpolation and adaptation," *Computer Speech & Language*, 2012.

[8] J. Shin, S. Narayanan, L. Gerber, A. Kazemzadeh, D. Byrd, et al., "Analysis of user behavior under error conditions in spoken dialogs," in *ICSLP*, 2002, vol. 2.

[9] Ramón López-Cózar, David Griol, and José F Quesada, "New technique to enhance the performance of spoken dialogue systems by means of implicit recovery of asr errors," in *Spoken Dialogue Systems for Ambient Environments*, pp. 96–109. Springer, 2010.

[10] Dong Yu and Li Deng, "Semantic confidence calibration for spoken dialog applications," in *ICASSP*. IEEE, 2010, pp. 4450–4453.

[11] Matthew Stephen Seigel, Philip C Woodland, et al., "Combining information sources for confidence estimation with crf models.," in *Interspeech*, 2011, pp. 905–908.

[12] Alex Marin, Tom Kwiatkowski, Mari Ostendorf, and Luke Zettlemoyer, "Using syntactic and confusion network structure for out-of-vocabulary word detection," in *SLT*. IEEE, 2012, pp. 159–164.

[13] Stefan Kombrink, Mirko Hannemann, and Lukáš Burget, "Out-of-vocabulary word detection and beyond," in *Detection and Identification of Rare Audiovisual Cues*, pp. 57–65. Springer, 2012.

[14] C. Parada, M. Dredze, D. Filimonov, and F. Jelinek, "Contextual information improves oov detection in speech," in *NAACL*, 2010.

[15] Thomas Pellegrini and Isabel Trancoso, "Improving asr error detection with non-decoder based features.," in *Interspeech*, 2010, pp. 1950–1953.

[16] Frederic Bechet and Benoit Favre, "Asr Error Segment Localization for Spoken Recovery Strategy," in *ICASSP*, 2013.

[17] Richard Dufour, Géraldine Damnati, and Delphine Charlet, "Automatic error region detection and characterization in lvcsr transcriptions of tv news shows," in *ICASSP*. IEEE, 2012, pp. 4445–4448.

[18] Rohit Prasad, Rohit Kumar, Sankaranarayanan Ananthakrishnan, Wei Chen, Sanjika Hewavitharana, Matthew Roy, Frederick Choi, Aaron Challenner, Enoch Kan, Arvind Neelakantan, et al., "Active error detection and resolution for speech-to-speech translation," *IWSLT*, 2012.

[19] Svetlana Stoyanchev, Alex Liu, and Julia Hirschberg, "Clarification questions with feedback," in *Feedback Behaviors in Dialog*, 2012.

[20] Sangkeun Jung, Cheongjae Lee, and Gary Geunbae Lee, "Using utterance and semantic level confidence for interactive spoken dialog clarification.," *JCSE*, vol. 2, no. 1, pp. 1–25, 2008.

[21] Dan Bohus and Alexander I Rudnicky, "The ravenclaw dialog management framework: Architecture and systems," *Computer Speech & Language*, vol. 23, no. 3, pp. 332–361, 2009.

[22] Teruhisa Misu, Tatsuya Kawahara, and Kazunori Komatani, "Confirmation strategy for document retrieval systems with spoken dialog interface.," in *Interspeech*, 2004.

[23] Jun Ogata and Masataka Goto, "Speech repair: quick error correction just by using selection operation for speech input interfaces.," in *Interspeech*. Citeseer, 2005, pp. 133–136.

[24] D. Huggins-Daines and A.I. Rudnicky, "Interactive asr error correction for touchscreen devices," in *HLT*. Association for Computational Linguistics, 2008, pp. 17–19.

[25] Antoine Laurent, Sylvain Meignier, Teva Merlin, and Paul Deléglise, "Computer-assisted transcription of speech based on confusion network reordering," in *ICASSP*. IEEE, 2011, pp. 4884–4887.

[26] L. Hoste, B. Dumas, and B. Signer, "Speeg: a multimodal speech-and gesture-based text input solution," in *International Working Conference on Advanced Visual Interfaces*. ACM, 2012, pp. 156–163.

[27] K. Filippova and M. Strube, "Sentence fusion via dependency graph compression," in *EMNLP*. Association for Computational Linguistics, 2008, pp. 177–185.

[28] R. Barzilay and K.R. McKeown, "Sentence fusion for multidocument news summarization," *Computational Linguistics*, vol. 31, no. 3, pp. 297–328, 2005.

[29] M. Elsner and D. Santhanam, "Learning to fuse disparate sentences," in *Monolingual Text-To-Text Generation*. Association for Computational Linguistics, 2011, pp. 54–63.

[30] Stephen F Altschul and Bruce W Erickson, "Optimal sequence alignment using affine gap costs," *Bulletin of mathematical biology*, vol. 48, no. 5, pp. 603–616, 1986.

[31] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," in *ICLR*, 2013.

[32] Kevin Lenzo, "The cmu pronouncing dictionary," 2007.