Adapting Machine Translation Models toward Misrecognized Speech with Text-to-Speech Pronunciation Rules and Acoustic Confusability

Nicholas Ruiz[1], *Qin Gao*[2], *William Lewis*[2], *Marcello Federico*[1]

[1]Fondazione Bruno Kessler, Trento, Italy
[2]Microsoft Research, Redmond, WA, USA
{nicruiz,federico}@fbk.eu {qigao,wilewis}@microsoft.com

10.21437/Interspeech.2015-494

Abstract

In the spoken language translation pipeline, machine translation systems that are trained solely on written bitexts are often unable to recover from speech recognition errors due to the mismatch in training data. We propose a novel technique to simulate the errors generated by an ASR system, using the ASR system's pronunciation dictionary and language model. Lexical entries in the pronunciation dictionary are converted into phoneme sequences using a text-to-speech (TTS) analyzer and stored in a phoneme-to-word translation model. The translation model and ASR language model are combined into a phoneme-to-word MT system that "damages" clean texts to look like ASR outputs based on acoustic confusions. Training texts are TTS-converted and damaged into synthetic ASR data for use as adaptation data for training a speech translation system. Our proposed technique yields consistent improvements in translation quality on English-French lectures.

Index Terms: spoken language translation, machine translation, pronunciation modeling, error modeling

1. Introduction

A spoken language translation (SLT) system minimally consists of two main components: an automatic speech recognition (ASR) system that transcribes source language utterances into a transcript and a machine translation (MT) system which translates the transcripts. While there have been a number of efforts to construct tightly-coupled ASR and MT systems that are jointly trained and/or optimized [1, 2], the majority of SLT systems employ a cascading approach in which speech recognition is first performed and the results are translated by an MT system [3, 4, 5]. The major disadvantage of using the cascading approach is the mismatch between MT training data and ASR output. Most statistical MT (SMT) systems are trained on written bitexts which have different artifacts from ASR output. The ASR system may contain recognition errors and cannot output out-of-vocabulary (OOV) words. Ideally, this could be overcome by training the SMT system on speech corpora that have been both transcribed and translated; however, there are few corpora of this type available and they are expensive to construct. Thus, SMT training is usually limited to using small amounts of translated speech transcripts as adaptation data.

To overcome the paucity of bilingual speech training data, an ideal solution is to convert the source side of a bitext to ASR-like outputs. Considering the ASR system as a noisy channel that converts the actual transcripts of the speech input to error-prone outputs, we can model the production of ASR errors and apply it on a large amount of bitexts to introduce possible ASR errors into the training data of SMT system. A straightforward method is to actually pronounce every source language sentence in the corpus into a microphone and pass the audio signal through the actual ASR system that will be used in the pipeline. However, this method is too costly. Instead of mapping the text to a signal representation and back to text, we can stop at the phoneme level and model (1) the phonetic confusion between phonemes; and (2) the ambiguity of phoneme sequences, using actual ASR output on a small amount of speech data.

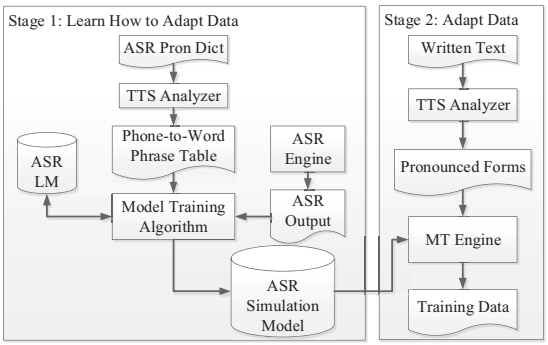

Figure 1: Damaging channel pipeline. Source language texts are transformed into phoneme sequences and translated back into words, corresponding by a phoneme-to-word SMT system that models errors performed during ASR decoding.

This text→phoneme→text pipeline requires (a) a reliable conversion from written text to phonemes, and (b) a modeling technology that can optimize towards a small development set of actual ASR output. For (a), we employ the text analysis component of a TTS engine, which dictates written text based on a pronunciation dictionary (PD), letter-to-sound (LTS) rules, and context-dependent pronunciation rules for numbers, ordinals and acronyms. For (b), we use phrase-based MT with MERT on actual ASR outputs. In a nutshell, we build a MT system that translates TTS-generated phoneme sequences into ASR-like output, and apply it on the source side of MT training data to improve ASR error tolerance.

2. Damaging Channel

Our SLT system is a standard cascading ASR-MT pipeline, where the MT system accepts as input a single-best hypothesis from an ASR system, which is recased, punctuated, and tokenized prior to translation. Our goal is to build a phrase-based SMT system [6] that translates TTS-generated phoneme sequences to ASR-like output and to apply it on the larger SMT training data. We divide the pipeline into two stages, as shown in Fig. 1. First, the damaging channel learns how to transform clean source language texts into output that con-

Copyright © 2015 ISCA

tains synthetic ASR errors. Each word in an ASR system's PD is converted into a sequence of phonemes using the LTS rules provided by a TTS analyzer. The mapping between phoneme sequences and their lexical forms are stored in a phoneme-to-word phrase table with uniform forward and backward probabilities. This phrase table is combined with the language model used by the original ASR system. Since the conversion of phonemes into words is monotonic, no reordering table is required. The phoneme-to-word SMT system is tuned using Minimum Error Rate Training (MERT) [7], using a small supervised set of source language speech transcripts and the corresponding single-best ASR hypotheses. Due to homophones and other pronunciation anomalies, the phrase table may have multiple entries for a single phoneme sequence. For example, the phoneme sequence /T UW/ may be mapped to *two*, *to* and *too*.

In the second stage, the source side of the training bitexts are again transformed into phoneme sequences by the TTS analyzer, which are subsequently translated by the phoneme-to-word SMT system to generate synthetic ASR outputs for training the MT component. All training bitexts are duplicated prior to "damaging" to allow MT training simultaneously on clean source language bitexts and synthetic ASR outputs.

2.1. TTS-based pronunciation generation

There are several problems in using the phoneme sequences in an ASR PD to build a phoneme-to-word phrase table:

1. No coverage for OOV words. SMT vocabularies contain many OOV words w.r.t. ASR pronunciation dictionaries; usually they will be output as phonetically similar words or phrases in ASR output. In order to simulate such phenomena, we must employ LTS rules on OOV words.
2. No rules for some acronyms (e.g. *ADHD*, *MTV*) and numeric sequences (e.g. *1998* or *$275,000*). We need to apply rules to correctly "pronounce" these tokens.
3. Context dependency. Words may contain different pronunciations given their context (i.e. *record* in *to record music* vs. *a music record*).

Instead, we use a TTS system's text analysis module, which can provide a pronunciation hypothesis for any word. TTS analyzers may use different phoneme sets from the ASR PD or they may have been trained on different dialects; thus, we replace the pronunciations from the ASR PD with the hypotheses from the text analysis module before building the phone-to-word phrase table. To account for multiple pronunciations of words in the ASR PD, we may also augment the phone-to-word phrase table with alternative pronunciations of words from the written text.

2.2. Phoneme-level confusion

Thus far, we have assumed that the PD contains only valid transcriptions. As such, the decoding process undergone by the phoneme-to-word SMT system defines segmentation boundaries on a sequence of phonemes to reconstruct words. However, during ASR decoding, phonemes may be missing or distorted in the input signal, rendering the decoder likely to misrecognize parts of the actual utterance. In response, we introduce an additional step in the damaging channel pipeline to model phonetic confusability by introducing distortions into a sequence of phonemes, based on the observed decoding behavior of an ASR system. This process is a phoneme-to-phoneme SMT pipeline, similar to that of [8]. A phrase table is estimated on phoneme-transcribed bitexts of source language transcripts and their single-best ASR hypotheses. A language model is estimated on the phoneme sequences of the ASR hypotheses. The weights of the models are optimized using MERT on a held-out development set. The trained phoneme-to-phoneme SMT system can perform the following operations: (1) delete one or more potentially silent or unrecognizable phonemes; (2) insert one or more adjacent phonemes; and (3) exchange phonemes that have similar context. The resulting system is applied to each lexical entry in the ASR PD to generate n distorted pronunciation alternatives which are used to expand the dictionary.

Phone Trans	Pron Dict	ASR BLEU	ASR TER	Transcript BLEU	Transcript TER
lex	lex	74.68	16.20	98.37	0.81
	tts*	20.39	79.53	25.79	72.05
	lex+tts	74.67	16.22	98.27	0.87
tts	lex*	27.97	58.13	33.57	52.00
	tts	47.84	40.37	57.26	32.23
	lex+tts	51.73	35.82	61.94	27.15

*Mismatch between pronunciations.

Table 2: Damaging channel models, converting English transcripts into ASR-like outputs, evaluated on dev2010. Phoneme conversion uses either the ASR PD (*lex*) or TTS (*tts*). Evaluated on the target ASR texts and the original transcripts.

3. Experiments

We perform experiments on the English-French TED speech translation task from the IWSLT 2014 evaluation campaign [9]. Our baseline SLT system is a cascaded ASR-MT pipeline. The ASR system is described in [10]; as a brief summary, the acoustic model is trained on TED talk videos released before December 31, 2010, corresponding to 820 talks and 144 hours of speech after filtering. It uses a deep neural network (DNN) that is trained using the Karel setup of the open-source Kaldi ASR toolkit [11]. It is trained over acoustic features generated in the second pass after having applied LDA-MLLT-fMLLR transformations with SAT HMMs. An eleven-frame context window of LDA-MLLT-fMLLR features (5 frames at each side) is used as input to form a 440-dimensional feature vector. The DNN has 6 hidden layers each with 2048 neurons and is pre-trained with Restricted Boltzmann Machines (RBM), followed by mini-batch Stochastic Gradient Descent training, and sequence-discriminative training such as Minimum Phone Error (MPE) and state-level Minimum Bayes Risk (sMBR). The single-best ASR hypotheses are punctuated, recased, and tokenized prior to being translated by the MT system. Our ASR system yields a word error rate (WER) of 11.7% on tst2012.

The baseline MT component of our SLT system is a phrase-based Moses system [6, 12], trained on the TED talk training set permitted in the IWSLT 2014 evaluation. Our baseline system features a statistical log-linear model including a phrase-based translation model (TM) and a lexicalized phrase-based reordering model (RM), both trained on TED data, a 5-gram language model (LM) trained with IRSTLM [13] and converted into KenLM's binary format [14] on the French side of the TED training corpus, and distortion, word, and phrase penalties. MT quality is evaluated using the BLEU [15] and TER [16] metrics.

3.1. Damaging channel

The monotonic phoneme-to-word SMT system is trained on one of three PD configurations: (1) the ASR pronunciations (*lex*); (2) a TTS-generated set of pronunciations for each word (*tts*); or (3) a union of the two (*lex+tts*). In *tts* configurations, each word in the original PD is converted into phonemes using the Festival TTS system with the CMU PD [17].

The ASR system's language model is included in the phoneme-to-word SMT system and all model weights are tuned

Transcript	**Their** hunters could smell animal urine at **40 paces and** tell you what **left it behind**
LEX PHONEMES	**dh axr** hh ah n **t axr z** k uh d s m eh l ae n ax **m el y uh r ih n** ae t **40** p ey s ax z ...
LEX-DAMAGE	**they're** hunters could smell animal urine at **40 paces and** tell you what species **left it behind**
TTS PHONEMES	**dh eh r** hh ah n **t er z** k uh d s m eh l ae n ax **m ax l y er ax n** ae t **f ao r t iy** p ey s ax z ...
TTS-DAMAGE	**their** hunters could smell animal urine at **forty paisa Zand** tell you what species left **Iturbe a hind**
TTS-DAMAGE-P2P	**their** hunters could smell animal urine at **forty paces as and** tell you what species **left it behind**

Table 1: Example damaging channel output on dev2010, using the original ASR pronunciation dictionary and TTS.

via MERT and evaluated on bitexts that map ASR references to our ASR system's single-best hypotheses. The clean transcripts are transcribed into phonemes prior to translation, either using the ASR PD or by running Festival's TTS analysis component. We additionally augment the PD described above with phoneme confusions using a phoneme-to-phoneme SMT system, which is trained on phoneme sequencies corresponding to English bitexts from tst2010. A 4-gram language model is estimated on the ASR phoneme sequences using IRSTLM and is binarized in KenLM format. The model weights are tuned on dev2010. Five or 10-best lists of phoneme sequences are generated for each word in the PD by translating each TTS-generated phoneme sequence into damaged phonemes. The resulting damaging channel configurations are used to generate SMT adaptation data from the TED training bitexts, where the source-side transcripts are processed through the damaging channel to generate synthetic ASR output. The synthetic outputs are tokenized, recased, and punctuated prior to inclusion.

3.2. Synthetic ASR outputs

No phoneme confusions. We first measure how well the damaging channel converts reference transcripts into ASR hypotheses, compared to how much it diverges from itself. Table 2 evaluates the effects of phoneme-to-word translation, without factoring in phonetic confusability, both on the ASR hypotheses and the original, unpunctuated transcripts.[1] While damaging channel models trained on the original ASR PD (*lex*) yield TER scores around 16% against the ASR hypotheses, the damaged texts are virtually the same as the originals; thus, it does not model acoustic confusability well enough. On the other hand, TTS-generated pronunciations yield TER scores around 40% on ASR hypotheses and a similar amount on the original transcripts. We similarly observe a 5% absolute TER improvement when combining the *tts* and *lex* pronunciations. Mismatches between phoneme converters (e.g. transcribing transcripts with *lex* and damaging with *tts*) yield abysmal results.

Phoneme confusions. Fig. 2 shows the effects of phoneme transduction on the damaging channel. In nearly every damaging channel configuration, adding up to 10 distorted phoneme sequences to each PD before training the damaging channel yields nearly a 10% absolute improvement in TER, both against the ASR outputs and the original transcripts. The effects of merging *tts* and *lex* dictionaries become insignificant when phoneme confusions are introduced, since the valid pronunciation variants are covered in the n-best lists.

Table 1 provides an example of synthetic ASR outputs on dev2010. The PD-driven damaging channel (LEX-DAMAGE) treats some numbers in digital form as OOV words (e.g. "40"). SMT phrase pairs containing these numbers will never be used in the SLT pipeline. The TTS-driven damaging channel (TTS-DAMAGE) successfully converts them to phoneme sequences and reconstructs their lexical form. At the same time, there are cases where the TTS-driven damaging channel's TM may

[1] While we report both BLEU and TER scores, the TER metric better measures this divergence and it is closely correlated with conventional WER metrics in ASR evaluation.

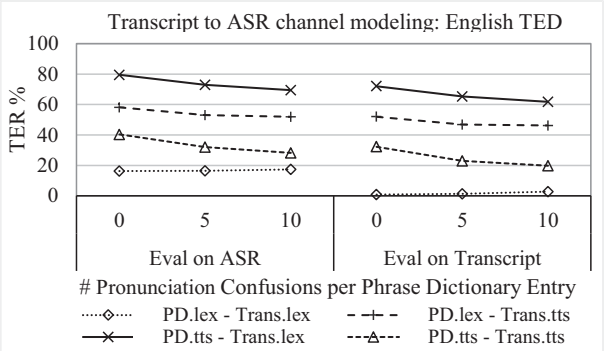

Figure 2: Effects of augmenting the PD with phoneme confusions on dev2010 (in TER).

give higher scores to low frequency words than common words (e.g. *paisa Zand*, instead of *Paces and*), Our TM assigns uniform probabilities to phoneme-to-word and word-to-phoneme entries. Since the PD was generated in a data-driven fashion, junk entries appear that usually are never encountered during ASR decoding. By introducing phoneme confusions through the phoneme-to-phoneme SMT system (TTS-DMG-P2P), the TM scores are smoothed with the addition of 5 pronunciations per lexical entry. TTS-DAMAGE-P2P assigns *paces* a pronunciation with a dropped "s" (/P EY S AX/) and duplicates /AX/, rendering the damaged output as *paces as and* (/P EY S AX AX Z AE N D/). We discuss this issue in more detail in Section 3.4.

3.3. SLT evaluation

We conduct two sets of TED-only experiments to simulate two domain adaptation scenarios: (1) the damaged TED transcripts and their translations are concatenated with the clean TED training data to estimate the translation model and reordering model (CONCAT); and (2) a separate phrase table is estimated on the damaged bitexts, where previously unseen phrases are appended using the FILL-UP technique [18, 19] with a provenance feature that marks the phrase as synthetic. In Table 3 we show the evaluation results. To control for optimizer instability [20], we run MERT three times on each experiment and evaluate the each system's performance on the tst2012 data set using

	Phoneme Confusion n-best					
System	0	5	10	0	5	10
Baseline	28.44	–	–	28.44	–	–
lex	29.19	29.04	28.92	29.06	29.02	28.83
tts	29.08	29.24	29.06	28.90	28.54	28.94
lex+tts	28.91	29.13	29.20	28.90	28.84	28.77
	CONCAT			FILL-UP		

Table 3: Evaluation results on tst2012 (in BLEU). Damaged TED transcripts are either CONCATenated with clean transcripts or used to generate new FILL-UP phrase table entries on the baseline TED phrase-table.

	Example 1	Example 2
English ref	Since **it's** digital, we can do reverse dissection.	...I've studied **technologies of mobile communication**...
ASR output	Since **its** digital we can do reverse dissection .	...I've studied **technologies , of mobile , communication**...
Baseline MT	**Depuis que** nous pouvons faire **son numérique** inverser sentinelles.	...j'ai étudié **technologies , de téléphones , la communication**...
LEX-DAMAGE	**Puisque c'est que** nous pouvons faire renverser dissection **du numérique**.	...j'ai étudié **les technologies de communication , de technologie mobile**...
TTS-DAMAGE	**Depuis ses digital** , nous pouvons faire régresser axillaire.	...j' ai étudié **les technologies, de technologie mobile, la communication**...
TTS-DMG-P2P	**Depuis ses numérique** , nous pouvons faire renverser axillaire.	...j' ai étudié **les technologies de communication , de portable**...
French Ref	**Puisque c' est numérique** , nous pouvons faire une dissection à l' envers .	...j' ai étudié **les technologies de communication mobile**...

Table 4: Example SLT outputs from tst2012, using damaging channel output as concatentated training data.

MultEval[2]. We observe statistically significant improvements in BLEU, ranging from 0.6-0.8 for our CONCAT and 0.4-0.6 for FILL-UP ($p < 0.01$), with the exception of the TTS-trained damaging channel. The fill-up results are weaker due to the lack of training data to estimate count statistics for each phrase table. However, concatenating corpora causes the larger pool of out-of-domain corpora to dominate the TM as the amount of training data increases [21].

Table 4 provides examples of end-to-end SLT English-French translations on tst2012, generated by the baseline SMT system and SMT systems trained with LEX-DAMAGE, TTS-DAMAGE, and TTS-DAMAGE-P2P. In the first example, the contraction *it's* is misrecognized as the possessive pronoun *its*. While all damaging channel systems permit the error-tolerant mapping of *its* to *c'est*, only LEX-DAMAGE applies it successfully. However, it comes at the cost of splitting the source phrase *it's digital* into two separate phrases and *digital* is reordered incorrectly to the end of the sentence. The second example demonstrates punctuation errors that change a segment's meaning. *Technologies of mobile communication* becomes a list of three items. The baseline and TTS-DAMAGE-P2P systems translate *mobile* either as a physical telephone device or a portable object. LEX-DAMAGE and TTS-DAMAGE-P2P generate translations related to *communication technologies*, which captures part of the original meaning. TTS-DAMAGE, on the other hand, generates a translation for *mobile technology*. While imperfect, each damaging channel-trained system manages to reorder phrase pairs in order to cross the erroneous punctuation boundaries, thereby improving the translation quality.

3.4. Analysis

Our damaging channel's phoneme-to-word TM suffers from forward probability dilution when multiple pronunciations for a word exist. For instance, LEX-DAMAGE has 12 pronunciations for *intercontinental*, each with a forward score of 0.077. The problem is exacerbated when introducing phoneme confusions. The 12 original pronunciations inflate to 34 and 69 when adding the 5- and 10-best phoneme confusions, respectively, while a word with a single pronunciation gains a quantity proportional to n. This behavior may result in junk word sequences like *in ter continent tall* to be favored, in spite of the word penalty feature and the low LM probabilities. This impact of this issue may be reduced by weighting the probability distribution by corpus frequencies, or pruning infrequent words.

Using a single TTS pronunciation for each word proves to be detrimental to the damaging channel. Gerund words such as *doing* and *creating* in the PD are transcribed with a /IH NG/ suffix in isolation by Festival, but in context they are commonly transcribed as /AX NG/.[3] No valid pronunciations exist in the phrase table, causing the damager to back off to nonsense constructions like *due a ng* and *create ng*. Phoneme-to-phoneme pronunciation expansions minimize this effect, at the cost of diluting phrase table scores. Instead, the TTS analyzer should generate additional word pronunciations by leveraging the pronunciation contexts in a corpus.

[2]https://github.com/jhclark/multeval

[3]This issue may occur anytime there is a mismatch between TTS and the entries in the PD.

4. Related work

Techniques to generate synthetic ASR errors have been used for discriminative language modeling [22, 23, 24], ASR error prediction [25], and speech translation [26, 27].

In [22, 28], a weighted finite state transducer (WFST) is compiled from an ASR PD to convert phoneme sequence back into words. The ASR system's acoustic model is used to measure confusability between phonemes. [24] propose a variant to phoneme transduction by estimating phoneme substitution probabilities using maximum likelihood estimates on Levenshtein alignments between the reference transcript and a n-best list of ASR hypotheses. In both methods N-Best outputs were generated and utilized in discriminative LM training.

[8] implement a similar phoneme-to-phoneme transducer, modeled as a SMT system and propose its use in conjunction with a FST-based phoneme-to-word transducer to damage texts. However, they assume that no OOVs are present in the texts to damage and did not apply their work on MT training data. Our method uses a TTS analyzer to bridge the crucial gap between ASR PD and MT data. An extension of [8] is proposed by [27], which uses a phone confusion transducer that allows substitutions based on phone clusters, consonant deletions, vowel duplications, and suffix insertions. The transducer is composed with the ASR PD and LM and applied on each entry in the SMT phrase table, generating alternative source phrases.

Our approach is an extension and deeper analysis of the text normalization approach of [26], which uses a text-to-speech engine to introduce phonetic confusability by generating alternative pronunciations for existing words in an ASR lexicon and using phoneme-to-word SMT to reconstruct word sequences constrained in the lexicon.

5. Conclusion

We have constructed several variants of a damaging channel that utilize principles of acoustic and phonetic confusability to convert sequences of phonemes to synthetic ASR outputs containing potential errors. Clean texts are converted to phoneme sequences by a TTS analyzer and "translated" back into words based on the observed behavior of an ASR system. Our TTS-driven approach successfully converts OOV words, acronyms, and numeric sequences into words belonging to a ASR PD and can be used to generate synthetic speech data to adapt a MT system to the SLT task. Our experiments show that MT systems adapted with damaged texts are better suited to receive ASR outputs as input than systems trained only on bitexts. While the TTS-driven damaging channel performs similarly to baselines which use the ASR PD, the TTS-driven approach is capable of generating synthetic texts that diverge further from the original transcripts in such a way that utilizing multiple damaged hypotheses could improve error coverage during MT training.

6. References

[1] X. He and L. Deng, "Speech-Centric Information Processing: An Optimization-Oriented Approach," *Proceedings of the IEEE*, May 2013.

[2] B. Zhou, "Statistical machine translation for speech: A perspective on structures, learning, and decoding," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1180–1202, 2013. [Online]. Available: http://dx.doi.org/10.1109/JPROC.2013.2249491

[3] E. Matusov, S. Kanthak, and H. Ney, "Integrating speech recognition and machine translation: Where do we stand?" in *Proceedings of ICASSP*, Toulouse, France, 2006, pp. 1217–1220.

[4] N. Bertoldi, R. Zens, and M. Federico, "Speech translation by confusion network decoding," in *Proceedings of ICASSP*, Honolulu, HA, 2007, pp. 1297–1300.

[5] F. Casacuberta, M. Federico, H. Ney, and E. Vidal, "Recent efforts in spoken language processing," *IEEE Signal Processing Magazine*, vol. 25, no. 3, pp. 80–88, May 2008.

[6] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proceedings of HLT-NAACL 2003*, Edmonton, Canada, 2003, pp. 127–133. [Online]. Available: http://aclweb.org/anthology-new/N/N03/N03-1017.pdf

[7] F. J. Och, "Minimum Error Rate Training in Statistical Machine Translation," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, E. Hinrichs and D. Roth, Eds., 2003, pp. 160–167. [Online]. Available: http://www.aclweb.org/anthology/P03-1021.pdf

[8] Q. F. Tan, K. Audhkhasi, P. G. Georgiou, E. Ettelaie, and S. S. Narayanan, "Automatic speech recognition system channel modeling," in *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, T. Kobayashi, K. Hirose, and S. Nakamura, Eds. ISCA, 2010, pp. 2442–2445.

[9] M. Cettolo, J. Niehues, S. Stüker, L. Bentivogli, and M. Federico, "Report on the 11th IWSLT Evaluation Campaign," in *Proceedings of the International Workshop on Spoken Language Trnaslation (IWSLT)*, Lake Tahoe, USA, December 2014.

[10] B. BabaAli, R. Serizel, S. Jalalvand, D. Falavigna, R. Gretter, and D. Giuliani, "FBK @ IWSLT 2014 - ASR track," in *Proceedings of the International Workshop on Spoken Language Trnaslation (IWSLT)*, Lake Tahoe, USA, December 2014.

[11] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011.

[12] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open Source Toolkit for Statistical Machine Translation," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, Prague, Czech Republic, 2007, pp. 177–180. [Online]. Available: http://aclweb.org/anthology-new/P/P07/P07-2045.pdf

[13] M. Federico, N. Bertoldi, and M. Cettolo, "IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models," in *Proceedings of Interspeech*, Brisbane, Australia, 2008, pp. 1618–1621.

[14] K. Heafield, "KenLM: Faster and Smaller Language Model Queries," in *Proceedings of the Sixth Workshop on Statistical Machine Translation*, Edinburgh, Scotland, United Kingdom, July 2011, pp. 187–197. [Online]. Available: http://kheafield.com/professional/avenue/kenlm.pdf

[15] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics (ACL)*, Philadelphia, PA, 2002, pp. 311–318. [Online]. Available: http://aclweb.org/anthology-new/P/P02/P02-1040.pdf

[16] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *5th Conference of the Association for Machine Translation in the Americas (AMTA)*, Boston, Massachusetts, August 2006.

[17] A. W. Black and P. A. Taylor, "The Festival Speech Synthesis System: System documentation," Human Communciation Research Centre, University of Edinburgh, Scotland, UK, Tech. Rep. HCRC/TR-83, 1997, avaliable at http://www.cstr.ed.ac.uk/projects/festival.html.

[18] P. Nakov, "Improving English-Spanish Statistical Machine Translation: Experiments in Domain Adaptation, Sentence Paraphrasing, Tokenization, and Recasing," in *Workshop on Statistical Machine Translation, Association for Computational Linguistics*, 2008.

[19] A. Bisazza, N. Ruiz, and M. Federico, "Fill-up versus Interpolation Methods for Phrase-based SMT Adaptation," in *International Workshop on Spoken Language Translation (IWSLT)*, San Francisco, CA, 2011, pp. 136–143.

[20] J. Clark, C. Dyer, A. Lavie, and N. Smith, "Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability," in *Proceedings of the Association for Computational Lingustics*, ser. ACL 2011. Portland, Oregon, USA: Association for Computational Linguistics, 2011.

[21] P. Koehn and J. Schroeder, "Experiments in Domain Adaptation for Statistical Machine Translation," in *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 224–227. [Online]. Available: http://www.aclweb.org/anthology/W/W07/W07-0233

[22] G. Kurata, N. Itoh, and M. Nishimura, "Acoustically discriminative training for language models," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, April 2009, pp. 4717–4720.

[23] P. Jyothi and E. Fosler-Lussier, "Discriminative language modeling using simulated asr errors." in *INTERSPEECH*, T. Kobayashi, K. Hirose, and S. Nakamura, Eds. ISCA, 2010, pp. 1049–1052.

[24] K. Sagae, M. Lehr, E. T. Prud'hommeaux, P. Xu, N. Glenn, D. Karakos, S. Khudanpur, B. Roark, M. Saraclar, I. Shafran, D. M. Bikel, C. Callison-Burch, Y. Cao, K. Hall, E. Hasler, P. Koehn, A. Lopez, M. Post, and D. Riley, "Hallucinated n-best lists for discriminative language modeling," in *ICASSP*. IEEE, 2012, pp. 5001–5004.

[25] P. Jyothi and E. Fosler-Lussier, "A comparison of audio-free speech recognition error prediction methods." in *INTERSPEECH*. ISCA, 2009, pp. 1211–1214.

[26] A. Aue, Q. Gao, H. Hassan, X. He, G. Li, N. Ruiz, and F. Seide, "MSR-FBK IWSLT 2013 SLT System Description," in *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, Heidelberg, Germany, December 2013. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=205206

[27] Y. Tsvetkov, F. Metze, and C. Dyer, "Augmenting translation models with simulated acoustic confusions for improved spoken language translation." in *EACL*, 2014, pp. 616–625.

[28] G. Kurata, N. Itoh, and M. Nishimura, "Training of error-corrective model for asr without using audio data." in *ICASSP*. IEEE, 2011, pp. 5576–5579.