RECOVERY OF ACRONYMS, OUT-OF-LATTICE WORDS AND PRONUNCIATIONS FROM PARALLEL MULTILINGUAL SPEECH

João Miranda[1,2], *João Paulo Neto*[1] *and Alan W Black*[2]

[1]INESC-ID / Instituto Superior Técnico, Lisboa, Portugal
[2]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
jrsm@l2f.inesc-id.pt, Joao.Neto@inesc-id.pt, awb@cs.cmu.edu

ABSTRACT

In this work we present a set of techniques which explore information from multiple, different language versions of the same speech, to improve Automatic Speech Recognition (ASR) performance. Using this redundant information we are able to recover acronyms, words that cannot be found in the multiple hypotheses produced by the ASR systems, and pronunciations absent from their pronunciation dictionaries. When used together, the three techniques yield a relative improvement of 5.0% over the WER of our baseline system, and 24.8% relative when compared with standard speech recognition, in an Europarl Committee dataset with three different languages (Portuguese, Spanish and English). One full iteration of the system has a parallel Real Time Factor (RTF) of 3.08 and a sequential RTF of 6.44.

Index Terms— speech recognition, machine translation, pronunciation , out-of-lattice, acronyms

1. INTRODUCTION

Several language technologies can benefit from their mutual integration, in the sense that the outputs produced by one of these technologies can be used as inputs to another or to enhance their performance. Among those technologies best suited to such an integration are Automatic Speech Recognition (ASR) and Machine Translation (MT). We investigate how to combine ASR and MT models in parallel: the case where multiple speech streams (or a combination of speech and text streams) are available is of particular interest. These streams should represent direct or approximate translations of each other, so that our algorithms can exploit such redundant information in order to enhance the performance of the ASR and MT modules. In other words, since the errors that occur in different speech streams will be relatively independent from each other, we expect to recover from many of them by resorting to the information in the remaining streams. To a certain extent, this is analogous to the ROVER method [1] for combining speech recognizers, although the latter uses a single speech stream.

Applications of parallel combination of ASR and MT streams include the automatic multilingual transcription of simultaneously interpreted speeches, both at the United Nations and the European Parliament, as well as in other multilingual institutions or countries, since interpreted speech is challenging for ASR systems to recognize due to the disfluencies and fast speech it often contains. Other applications of this method include TV shows and series, as well as major sport events broadcast in multiple languages.

Recently, there has been considerable interest in the integration of ASR and MT models. This has mostly been done in a sequential manner, for speech-to-speech or speech-to-text translation, where the outputs of the ASR module are passed on to the SMT system. This is often achieved by generating multiple hypotheses as the output of the ASR system, in the form of lattices or confusion networks, and passing this probabilistic description of the output downstream, rather than simply the 1-best hypothesis generated by the recognizer. Despite this, several authors have tried to combine ASR and MT in a parallel fashion. Some of these methods are used to combine speech with a text stream, usually for an application such as machine-aided human translation [2, 3], although a few works have considered combining multiple speech streams [4, 5].

In previous work [6], we combined the outputs of recognizers of original and interpreted speeches in different languages, in the form of lattices, to yield improved recognition results. In order to link the language pairs together, we used phrase tables trained for a Statistical Machine Translation (SMT) system. A sequence of words in the lattice of a given language is mapped to a corresponding sequence of words in the lattice of a different language through such a phrase table. From this mapping, we build an alignment between two languages, consisting of correspondences between phrase pairs, and eventually alignments over an unrestricted number of languages. These alignments will allow us to uncover word sequences which originally had low posterior probabilities in their lattices, but which are likely to have occurred in the speech stream, given the fact that the various speech streams represent translations of each other. The process is explained in more detail in Section 2 .

Even though the lattices we use to compactly represent recognition hypotheses cover many different possibilities, not all the word sequences can be found in them, since the decoder, to enable a tractable search for the best word sequence, will prune away the vast majority of the alternative hypotheses. As a result, certain words that could, in principle, be recovered through multilingual information will be lost. In this work, we therefore intend to recover these words that cannot be found in their respective lattices, which we call *out-of-lattice words*, as well as acronyms (which may, additionaly, be out-of-vocabulary words). We also correct pronunciations of words in the dictionary that mismatch with their observed acoustic realizations, one of the main reasons for a word to be out-of-lattice, using the information in the generated alignments to select those words with a high confidence of having been said.

The rest of this paper is organized as follows. Section 2 summarizes the system that this work builds upon, our baseline system. Section 3 discusses in detail the improvements to the baseline system, which are the recovery of out-of-lattice words, acronyms and word pronunciations. Section 4 describes the experiments that assess these improvements, both in terms of word error rate and computational complexity. Finally, Section 5 concludes and suggests ideas for future work.

2. BASELINE SYSTEM

The overall baseline system architecture is described in Figure 1. An iteration of the baseline method [6] consists of the following steps:

- Generate phrase tables for each of the language pairs that we wish to combine.
- Using an ASR system, transcribe the speech in both the original and interpreted languages. This generates a set of lattices that encode a posterior probability distribution over word sequences. In particular, we compute posterior probabilities for all n-grams with $n <= 3$.
- For each language pair, intersect the lattices with the respective phrase table, obtaining a set of bilingual phrase pairs that appear in both the lattices and the phrase table.
- Rescore the phrase pairs from the previous step, estimating their likelihood of actually having appeared in the speech. The highest-scoring among these pairs are used to construct a phrase pair alignment.
- Finally, the phrase pairs contained in the alignment are used to rescore the lattices and produce new transcriptions.

2.1. ASR and SMT systems description

Three languages, Portuguese, English, and Spanish, were used in the development of our ASR and SMT systems. Audimus [7], a hybrid ANN-MLP WFST-based recognizer, is the ASR engine we used in this work. We trained 4-gram language models for each of the languages using the Europarl Parallel Corpus [8], and used our existing acoustic models and lexica for these three languages [9]. We created phrase tables for the 3 possible language combinations (Portuguese-Spanish, Portuguese-English, and Spanish-English), with the Moses toolkit [10], also using the Europarl Parallel Corpus as parallel training data.

2.2. Intersection between lattices and phrase tables

The intersection step selects the phrase pairs $source \,|||\, target$ that simultaneously are in the phrase table and for which both $source$ and $target$ can be found in the source and target lattices, respectively. The source and target phrases must be occur sufficiently close in terms of time. The maximum allowable time separation between the phrases is controlled by a parameter $\delta = 10s$. The efficient computation of this intersection uses a specialized algorithm [6].

2.3. Phrase pair scoring and selection

Not all phrase pairs in the intersection are added to the output. Instead, a number of features of each phrase pair are considered in scoring and selecting these to build an alignment, such as the posterior probabilities of each of the phrases in the phrase pair, its phrase table features, language model scores, and the time distances between both phrases of the pair. The output of this step is an alignment between phrase pairs.

2.4. Lattice rescoring

The rescoring step is an A* search of the lattices, producing new recognition hypotheses, where the language model is modified so that it assigns higher probability to word sequences that can be found in the generated alignments, at the correct times (i.e. whose timestamps match with the current time of the decoder).

3. PROPOSED IMPROVEMENTS

3.1. Acronym detection

Abbreviations or acronyms are very common in parliamentary speech or in technical talks. However, their correct recognition presents a number of challenges, since many of them are not present in the dictionary, or have incorrect pronunciations rather than a pronunciation that spells all of their letters out.

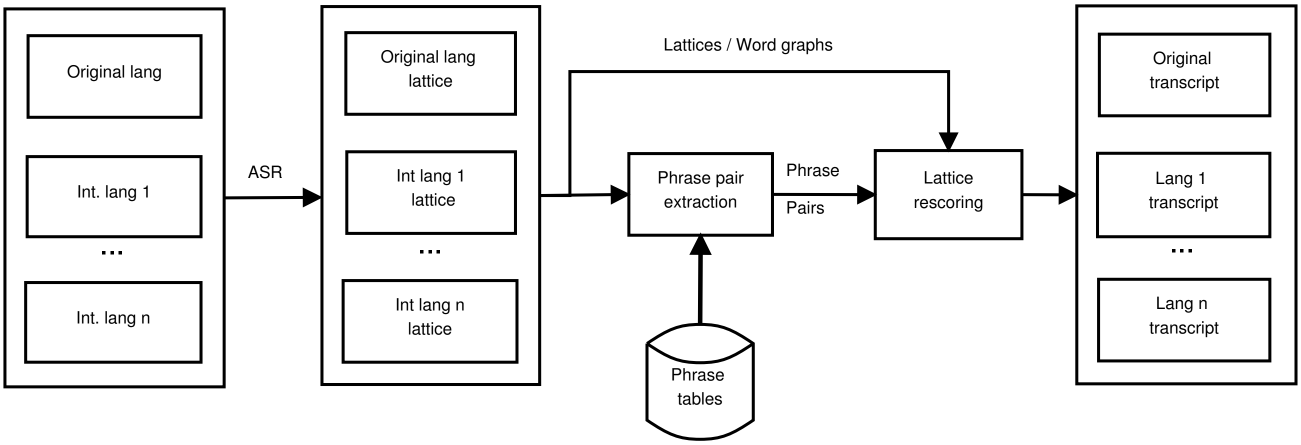

Fig. 1. The proposed system architecture

To recover abbreviations, we first locate candidates 3 to 5 letters in length, such as IPOA (Istanbul Programme of Action) or EBA (European Banking Authority). We also group together plural versions of the candidate abbreviations, for instance, LDC (Least Developed Country) and LDCS (Least Developed Countries). To search for the abbreviations, we build a finite state automaton which encodes all possible abbreviation sequences, and run it through the acoustic data, selecting the best abbreviation at every time step, together with its acoustic score. Among the candidates, only a subset is selected for further processing. To rank the candidates according to the likelihood that they are actually present in the speech, we extract a number of features, namely:

- The number of different languages in which the acronym appears and its total number of occurrences in all languages.
- The number of letters in the acronym.
- Whether the acronym had been spelled out before its occurrences. It is often the case, when introducing an acronym or abbreviation, that the speaker will explain the acronym's meaning to the audience by spelling it out. It is possible to try to guess whether that happened by comparing the acronym to word sequences in the transcriptions.
- The average score of the acronym occurrence and the acronym's language model score.

The language model used for acronym recovery is a trigram language model trained from a list containing 500 abbreviations using SRILM [11].

The feature values are linearly combined to form a score, and the top k (where we fixed $k = 10$) acronyms are then searched for the occurrence of close pairs (those having a temporal separation of less than $\delta = 10s$). These are then added to the phrase table / lattice intersection result before step 2.3 is re-run.

Recovered acronyms that are present in the final alignment but were not present in the original lattice are added, during the final decoding step described in Section 2.4, to the lattices being redecoded. This is accomplished by adding arcs to the lattice on-the-fly. These arcs have as endpoints pairs of states whose timestamps are the same as those of the acronym that we wish to recover, and the acoustic score which was determined when generating the abbreviation candidates.

3.2. Out-of-lattice word recovery

Although our method considers multiple alternatives in the form of lattices, these are limited in size and cannot contain all the possibile word sequences. In several cases it may be desirable to recover these words that do not appear in the output lattice, since this can improve transcription accuracy. We generate a list of locations where these out-of-lattice words could potentially appear according to the following criteria:

- If a certain word appears in two or more languages, its translation is predicted to appear in the third and subsequent languages as well. For example, the word “dossier” may appear close in time both in Spanish and Portuguese, but not in the English version. Still, it would be reasonable to expect the probability that it has been said in the English version to be high.
- If two words are aligned in a given language pair, then the surrounding words are also predicted to be translations of each other. For instance, if “European”, in English, is aligned with “Europeia” in Portuguese, and the word “Commission” follows the word “European” in the English transcription, then it is natural to assume that the word “Comissão” will precede the word “Europeia” in Portuguese, even if if does not appear in any

of the lattices. We therefore hypothesize the occurrence of an out-of-lattice word ending near the beginning of "Europeia".

When selecting the potential translations of a given word to a target language, we use the single-word translations in the appropriate phrase table that score above a manually preselected threshold. We then find the optimal starting and ending times, along with the optimal score, in an interval which is centered around the predicted occurrence time, by performing a sliding forced alignment between the word's pronunciation and the posterior probabilities generated by the recognizer's acoustic model. The optimal score is subsequently compared to a fixed threshold; if it is below, this potential out-of-lattice word occurrence is discarded. Otherwise, it is carried on to the next step where it is added to the phrase pairs that originate from the phrase-table-lattice intersection process. The phrase pair scoring method described in Section 2.3 is augmented to use an extra feature, which indicates whether the current pair is an out-of-lattice pair. In this way, we can have out-of-lattice phrase pairs compete in a balanced way with regularly extracted pairs.

At this point, step 2.3 is run to generate a new alignment. Finally, and in a way comparable with what is done for the recovery of abbreviations, recovered words are added to the lattices as new edges at the appropriate locations, during the execution of the lattice rescoring step.

3.3. Pronunciation recovery

Pronunciation lexica are key components of ASR systems - if the correct pronunciation is not in the lexicon, recognition performance degrades substantially - and can be notoriously hard to build. Manually creating these dictionaries is a very laborious and expensive task, and most languages have an open vocabulary, which means that a combination of manual and automatic methods is often used to generate these pronunciations. However, in many languages, such as English, the pronunciation of a word is very hard to predict from its ortography alone. Words imported from foreign languages, as well as names of people and locations, also pose important challenges.

In this work, we capitalize on the multi-stream alignment that we generated, described in Section 2.3. The idea is that the words that we were able to recover by way of increasing their language model scores are more likely to have been pronounced in a manner which differs from their dictionary pronunciation. From the alignment, we select high-confidence words - those that match with a high score. Then, for each of the word occurrences, we perform a Viterbi decoding with a finite state machine encoding all possible phone sequences. This leads to the most likely pronunciation on purely acoustic grounds.

At this point we calculate, using a string edit-distance algorithm, the number of insertions, deletions and substitutions required to transform the obtained pronunciation to the closest among the reference (dictionary) pronunciations. The alignment between the two strings defines a set of operations that one would have to apply to transform the reference pronunciation into the obtained pronunciation. We insert all the pronunciations that can be obtained by performing at most two such operations into a candidate pronunciation set $p_1..p_k$. Each of $p_1..p_k$ is rescored using the following expression, where p_0 is the original pronunciation:

$$SC(p_i) = \alpha L(p_{i_1}..p_{i_n}) + \beta \frac{\sum_{j=1}^{v} A_i(o_j)}{v} + \gamma \delta(p_0, p_i) \quad (1)$$

In Equation 1, v stands for the number of occurrences of the word, $L(p_{i_1}..p_{i_n})$ denotes a 5-gram language model over phone sequences, whereas $A_i(o_j)$ indicates the acoustic score of the j^{th} occurrence of the word assuming pronunciation i, and $\delta(p_0, p_i)$ is the edit distance between the dictionary pronunciation and pronunciation p_i.

We then select $p^* = \arg\max_i SC(p_i)$ as the pronunciation to be recovered and add it to the lexicon if it is not one of the existing dictionary pronunciations.

The language models, one for each language, were trained with the SRILM toolkit [11] with modified Kneser-Ney smoothing. The automatically generated lexica for each of the languages were used as training data.

4. RESULTS

Our evaluation and testing data consist of two data sets that were collected from the ENVI, DEVE, IMCO and LEGAL committees of the European Parliament. The first of these two data sets, with two speeches, is a held-out set used for tuning the various parameters for phrase pair selection and pronunciation recovery. The second set consists of four English speeches by both native and non-native speakers, drawn from each of the four aforementioned committees. Besides the original speeches, we also collected the Portuguese and Spanish interpreted versions. However, we only have manual reference transcriptions for the English version of the speeches, so we only present WER values for the English version.

Speech	EN	+PT	+ES	+PT+ES
DEVE	24.54%	22.33%	21.82%	20.40%
ENVI	20.60%	17.83%	18.84%	16.28%
IMCO	35.12%	31.03%	33.00%	29.97%
LEGAL	33.76%	31.43%	32.45%	28.42%
Average	28.50%	25.65%	26.52%	23.77%

Table 1. WER for the 4 speeches. The 1^{st} column is the error of the baseline system , the 2^{nd} and 3^{rd} the WER of the English original speech after combining with the Portuguese and Spanish interpretations, respectively, and the 4^{th} the error after combining with both interpretations.

Speech	EN	1^{st} iter	2^{nd} iter
DEVE	24.54%	20.40%	18.35%
ENVI	20.60%	16.28%	14.70%
IMCO	35.12%	29.97%	29.79%
LEGAL	33.76%	28.42%	27.38%
Average	28.50%	23.77%	22.56%

Table 2. WER for two iterations of the system (3^{rd} column) compared with the baseline system (1^{st} column) and the first iteration (2^{nd} column)

	No pron. recovery			Pron. recovery		
Speech	Base	+ar	+ar+ool	Base	+ar	+ar+ool
DEVE	18.35%	17.11%	16.92%	18.04%	16.81%	16.71%
ENVI	14.70%	14.35%	13.88%	14.36%	14.04%	13.69%
IMCO	29.79%	29.62%	29.30%	29.34%	29.18%	29.03%
LEGAL	27.38%	27.02%	26.85%	26.77%	26.44%	26.31%
Average	22.56%	22.03%	21.74%	22.13%	21.61%	21.43%

Table 3. WER for the system improvements. The factor that differs between the left and the right half of the table is whether pronunciation recovery is applied. The leftmost column of each half represents the WER with no acronym or out-of-lattice word recovery; the middle column indicates the WER with acronym recovery; and the last column presents the WER with both acronym and out-of-lattice word recovery.

Table 1 summarizes the results of one iteration of the baseline method. Using two interpreted languages (Spanish and Portuguese, 16.6%) is superior to using only one language (Portuguese, 10.0%) or (Spanish, 6.9%).

The next step is to perform unsupervised speaker adaptation of the English acoustic model, with the output of the first iteration as a reference. We then executed a second iteration of the baseline method, and collected the results in Table 2 (in this case the English version is combined with both the Portuguese and Spanish interpretations). This led to an additional 4.2% relative improvement, which is a cumulative 20.8% better than the original system (the plain English ASR system).

At this point we integrate the improvements of Section 3. We performed both out-of-lattice word recovery and abbreviation recovery on the results of the second iteration presented in Table 2. In order to take the impact of pronunciation recovery into account, we applied it after the 1^{st} iteration had been completed, by adding the recovered pronunciations into the recognition lexicon. Table 3 shows the results of our experiments for these two cases (where we apply, and do not apply the pronunciation recovery algorithm).

In Table 3, we see that the recovery of acronyms only seems to significantly improve results in the talk from the DEVE committee. In fact, of the four tested talks this is the one with the largest proportion of acronyms and abbreviations. In other talks, results are also slightly improved, which suggests that we aren't recovering many spurious abbreviations. On average, acronym recovery improves 2.4% relative. Also, the results of Table 3 demonstrate improvements in performance with out-of-lattice word recovery , in a more uniform way across all of the different speeches. The average relative improvement from out-of-lattice word recovery is 1.3% relative. Finally, the presence of pronunciation recovery appears to affect results in a additive manner relative to the other factors, and has a positive impact of 1.5% relative. Combining all the methods, we achieved an overall relative WER improvement of 5.0%, when compared to the second iteration of the baseline system. This translates to a cumulative 24.8% when compared with speech recognition-only (without running the baseline system).

Operation	Parallel RTF	Sequential RTF
PT - lattice intersections	0.69	1.96
Init. align. generation	0.18	0.18
Abbreviation recovery	0.23	0.23
OOL word recovery	0.42	0.42
Pronunciation recovery	0.51	1.34
Alignment generation	0.19	0.19
Final decoding	0.86	2.12
Total	3.08	6.44

Table 4. Average real time factor (RTF), over the four testing talks, of each of the main operations of the algorithm (for a single iteration). The first column indicates the parallel RTF whereas the second column indicates the sequential RTF.

4.1. Running time analysis

In this section we empirically analyze the computational overhead, in terms of running time, incurred by the algorithms described in the paper, for the case of $N = 3$ languages (one original language and two interpreted languages).

We measured the overhead of each of the developed components. Table 4 summarizes the real time factors of the algorithm, averaged over the four testing speeches, for one full iteration. We distinguish the sequential RTF from the parallel RTF. The latter assumes that a number of operations can be executed in parallel, since they are independent of each other and non-overlapping, and considers only the running time of the longest among these operations. The operations that can be executed in parallel are intersecting multiple phrase table - lattice pairs, the final decoding steps - obtaining improved transcriptions is parallelizable for the various languages since there aren't any dependencies between the instances of the search algorithm - and pronunciation recovery, which is done separately for each of the languages. Table 4 shows that the methods presented in this work are responsible for 1.99 xRT sequential, whereas a full iteration of the algorithm takes 6.44 xRT sequentially, but only 3.08 xRT if it can be executed in parallel.

The total time complexity grows quadratically with the number of languages N, which would be intractable for large N. However, the most time-consuming part of the algorithm - running a series of phrase table-lattice intersections - can be paralellized, and so with sufficient computational power this would not slow the system down. Furthermore, the number of phrase table-lattice intersections can be kept to a minimum by selecting a subset of the $\binom{N}{2}$ possible intersections, in such a way as to minimize the impact in result quality.

5. CONCLUSIONS

In this work we have described a number of techniques - the recovery of acronyms, pronunciations and out-of-lattice words, that were designed to explore multilingual information in order to enhance the performance of our baseline system. We applied these three methods to four speeches, each drawn from a different European Parliament Committee, and considered three different languages : English, Spanish, and Portuguese, English being the original language of the speeches. Combined, the three techniques yield a relative improvement of 5.0% over the WER of our baseline system and 24.8% over speech recognition only. Running a full iteration of the algorithm takes an average of 3.08 xRT if done in parallel, and 6.44 xRT if done sequentially.

The topic of pronunciation recovery from multiple speech streams shows significant promise for future work in that it can, in principle, be run in a fully unsupervised manner to collect pronunciations, in large scale, for different speaker groups, languages, foreign words or names. We also intend to experiment with larger numbers of languages to verify if the improvements in recognition performance continue to increase, since we expect new out-of-lattice words and pronunciations to be recovered. Finally, we expect to be able to further reduce the running time overhead of our algorithm to ensure it remains practical for larger numbers of languages.

6. ACKNOWLEDGEMENTS

Support for this research was provided by the Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) through the Carnegie Mellon Portugal Program under Grant SFRH/BD/33767/2009, and through projects CMU-PT/HuMach/0039/2008, CMU-PT/0005/2007, and PEst-OE/EEI/LA0021/2011.

7. REFERENCES

[1] J.G. Fiscus, “A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER),” in *Proceedings of the ASRU*, Santa Barbara, USA, 1997.

[2] S. Khadivi and H. Ney, “Integration of Speech Recognition and Machine Translation in Computer-Assisted Translation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1551–1564, 2008.

[3] A. Reddy and R.C Rose, “Integration of statistical models for dictation of document translations in a machine-aided human translation task,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 2015–2027, 2010.

[4] M. Paulik, S. Stüker, C. Fügen, T. Schultz, T. Schaaf, and A. Waibel, “Speech Translation Enhanced Automatic Speech Recognition,” in *Proceedings of the ASRU*, San Juan, Puerto Rico, 2005.

[5] M. Paulik and A. Waibel, “Extracting Clues from Human Interpreter Speech for Spoken Language Translation,” in *Proceedings of ICASSP*, Las Vegas, USA, 2008.

[6] J. Miranda, J. P. Neto, and A. W Black, “Parallel combination of speech streams for improved ASR,” in *Proceedings of the Interspeech*, Portland, USA, 2012.

[7] H. Meinedo, D. A. Caseiro, J. P. Neto, and I. Trancoso, “AUDIMUS.media: a Broadcast News speech recognition system for the European Portuguese language,” in *Proceedings of PROPOR*, Faro, Portugal, 2003.

[8] P. Koehn, “Europarl: A Parallel Corpus for Statistical Machine Translation,” in *Proceedings of the tenth Machine Translation Summit*, Phuket, Thailand, 2005.

[9] H. Meinedo, A. Abad, T. Pellegrini, I. Trancoso, and J. P. Neto, “The L2F Broadcast News Speech Recognition System,” in *Proceedings of Fala2010*, Vigo, Spain, 2010.

[10] 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 ACL demo session*, Prague, Czech Republic, 2007.

[11] A. Stolcke, “SRILM-an extensible language modeling toolkit,” in *Proceedings of the International Conference on Spoken Language Processing*, November 2002, pp. 257–286.