On-line Language Model Biasing for Statistical Machine Translation

Sankaranarayanan Ananthakrishnan, Rohit Prasad and Prem Natarajan
Raytheon BBN Technologies
Cambridge, MA 02138, U.S.A.
{sanantha,rprasad,pnataraj}@bbn.com

Abstract

The language model (LM) is a critical component in most statistical machine translation (SMT) systems, serving to establish a probability distribution over the hypothesis space. Most SMT systems use a static LM, independent of the source language input. While previous work has shown that adapting LMs based on the input improves SMT performance, none of the techniques has thus far been shown to be feasible for on-line systems. In this paper, we develop a novel measure of cross-lingual similarity for biasing the LM based on the test input. We also illustrate an efficient on-line implementation that supports integration with on-line SMT systems by transferring much of the computational load off-line. Our approach yields significant reductions in target perplexity compared to the static LM, as well as consistent improvements in SMT performance across language pairs (English-Dari and English-Pashto).

1 Introduction

While much of the focus in developing a statistical machine translation (SMT) system revolves around the translation model (TM), most systems do not emphasize the role of the language model (LM). The latter generally follows a n-gram structure and is estimated from a large, monolingual corpus of target sentences. In most systems, the LM is independent of the test input, i.e. fixed n-gram probabilities determine the likelihood of all translation hypotheses, regardless of the source input.

Some previous work exists in LM adaptation for SMT. Snover et al. (2008) used a cross-lingual information retrieval (CLIR) system to select a subset of target documents “comparable” to the source document; bias LMs estimated from these subsets were interpolated with a static background LM. Zhao et al. (2004) converted initial SMT hypotheses to queries and retrieved similar sentences from a large monolingual collection. The latter were used to build source-specific LMs that were then interpolated with a background model. A similar approach was proposed by Kim (2005). While feasible in off-line evaluations where the test set is relatively static, the above techniques are computationally expensive and therefore not suitable for low-latency, interactive applications of SMT. Examples include speech-to-speech and web-based interactive translation systems, where test inputs are user-generated and preclude off-line LM adaptation.

In this paper, we present a novel technique for weighting a LM corpus at the sentence level based on the source language input. The weighting scheme relies on a measure of cross-lingual similarity evaluated by projecting sparse vector representations of the target sentences into the space of source sentences using a transformation matrix computed from the bilingual parallel data. The LM estimated from this weighted corpus boosts the probability of relevant target n-grams, while attenuating unrelated target segments. Our formulation, based on simple ideas in linear algebra, alleviates run-time complexity by pre-computing the majority of intermediate products off-line.

The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

Distribution Statement “A” (Approved for Public Release, Distribution Unlimited)

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:shortpapers, pages 445–449,
Portland, Oregon, June 19-24, 2011. ©2011 Association for Computational Linguistics

2 Cross-Lingual Similarity

We propose a novel measure of cross-lingual similarity that evaluates the likeness between an arbitrary pair of source and target language sentences. The proposed approach represents the source and target sentences in sparse vector spaces defined by their corresponding vocabularies, and relies on a bilingual projection matrix to transform vectors in the target language space to the source language space.

Let $S = \{s_1, \ldots, s_M\}$ and $T = \{t_1, \ldots, t_N\}$ represent the source and target language vocabularies. Let $\mathbf{u}$ represent the candidate source sentence in a M-dimensional vector space, whose m^{th} dimension u_m represents the count of vocabulary item s_m in the sentence. Similarly, $\mathbf{v}$ represents the candidate target sentence in a N-dimensional vector space. Thus, $\mathbf{u}$ and $\mathbf{v}$ are sparse term-frequency vectors. Traditionally, the cosine similarity measure is used to evaluate the likeness of two term-frequency representations. However, $\mathbf{u}$ and $\mathbf{v}$ lie in different vector spaces. Thus, it is necessary to find a projection of $\mathbf{v}$ in the source vocabulary vector space before similarity can be evaluated.

Assuming we are able to compute a $M \times N$-dimensional bilingual word co-occurrence matrix $\mathbf{\Sigma}$ from the SMT parallel corpus, the matrix-vector product $\hat{\mathbf{u}} = \mathbf{\Sigma v}$ is a projection of the target sentence in the source vector space. Those source terms of the M-dimensional vector $\hat{\mathbf{u}}$ will be emphasized that most frequently co-occur with the target terms in $\mathbf{v}$. In other words, $\hat{\mathbf{u}}$ can be interpreted as a "bag-of-words" translation of $\mathbf{v}$.

The cross-lingual similarity between the candidate source and target sentences then reduces to the cosine similarity between the source term-frequency vector $\mathbf{u}$ and the projected target term-frequency vector $\hat{\mathbf{u}}$, as shown in Equation 2.1:

$$\begin{aligned}\mathcal{S}(\mathbf{u},\mathbf{v}) &= \frac{1}{\|\mathbf{u}\|\|\hat{\mathbf{u}}\|}\mathbf{u}^T\hat{\mathbf{u}} \\ &= \frac{1}{\|\mathbf{u}\|\|\mathbf{\Sigma v}\|}\mathbf{u}^T\mathbf{\Sigma v} \qquad (2.1)\end{aligned}$$

In the above equation, we ensure that both $\mathbf{u}$ and $\hat{\mathbf{u}}$ are normalized to unit L_2-norm. This prevents over- or under-estimation of cross-lingual similarity due to sentence length mismatch.

We estimate the bilingual word co-occurrence matrix $\mathbf{\Sigma}$ from an unsupervised, automatic word alignment induced over the parallel training corpus $\mathcal{P}$. We use the GIZA++ toolkit (Al-Onaizan et al., 1999) to estimate the parameters of IBM Model 4 (Brown et al., 1993), and combine the forward and backward Viterbi alignments to obtain many-to-many word alignments as described in Koehn et al. (2003). The $(m, n)^{th}$ entry $\Sigma_{m,n}$ of this matrix is the number of times source word s_m aligns to target word t_n in $\mathcal{P}$.

3 Language Model Biasing

In traditional LM training, n-gram counts are evaluated assuming unit weight for each sentence. Our approach to LM biasing involves re-distributing these weights to favor target sentences that are "similar" to the candidate source sentence according to the measure of cross-lingual similarity developed in Section 2. Thus, n-grams that appear in the translation hypothesis for the candidate input will be assigned high probability by the biased LM, and vice-versa.

Let $\mathbf{u}$ be the term-frequency representation of the candidate source sentence for which the LM must be biased. The set of vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_K\}$ similarly represent the K target LM training sentences. We compute the similarity of the source sentence $\mathbf{u}$ to each target sentence $\mathbf{v}_j$ according to Equation 3.1:

$$\begin{aligned}\omega_j &= \mathcal{S}(\mathbf{u},\mathbf{v}_j) \\ &= \frac{1}{\|\mathbf{u}\|\|\mathbf{\Sigma v}_j\|}\mathbf{u}^T\mathbf{\Sigma v}_j \qquad (3.1)\end{aligned}$$

The biased LM is estimated by weighting n-gram counts collected from the j^{th} target sentence with the corresponding cross-lingual similarity ω_j. However, this is computationally intensive because: (a) LM corpora usually consist of hundreds of thousands or millions of sentences; ω_j must be evaluated at run-time for each of them, and (b) the entire LM must be re-estimated at run-time from n-gram counts weighted by sentence-level cross-lingual similarity.

In order to alleviate the run-time complexity of on-line LM biasing, we present an efficient method for obtaining *biased counts* of an arbitrary target

n-gram t. We define $\mathbf{c}_t = \left[c_t^1, \ldots, c_t^K\right]^T$ to be the indicator-count vector where c_t^j is the unbiased count of t in target sentence j. Let $\omega = [\omega_1, \ldots, \omega_K]^T$ be the vector representing cross-lingual similarity between the candidate source sentence and each of the K target sentences. Then, the biased count of this n-gram, denoted by $\mathcal{C}^*(t)$, is given by Equation 3.2:

$$
\begin{aligned}
\mathcal{C}^*(t) &= \mathbf{c}_t^T \omega \\
&= \sum_{j=1}^{K} \frac{1}{\|\mathbf{u}\| \|\boldsymbol{\Sigma}\mathbf{v}_j\|} c_t^j \mathbf{u}^{\mathbf{T}} \boldsymbol{\Sigma} \mathbf{v}_j \\
&= \frac{1}{\|\mathbf{u}\|} \mathbf{u}^T \sum_{j=1}^{K} \frac{1}{\|\boldsymbol{\Sigma}\mathbf{v}_j\|} c_t^j \boldsymbol{\Sigma} \mathbf{v}_j \\
&= \frac{1}{\|\mathbf{u}\|} \mathbf{u}^T \mathbf{b}_t
\end{aligned}
\tag{3.2}
$$

The vector $\mathbf{b}_t$ can be interpreted as the projection of target n-gram t in the source space. Note that $\mathbf{b}_t$ is independent of the source input $\mathbf{u}$, and can therefore be pre-computed off-line. At run-time, the biased count of any n-gram can be obtained via a simple dot product. This adds very little on-line time complexity because $\mathbf{u}$ is a sparse vector. Since $\mathbf{b}_t$ is technically a dense vector, the space complexity of this approach may seem very high. In practice, the mass of $\mathbf{b}_t$ is concentrated around a very small number of source words that frequently co-occur with target n-gram t; thus, it can be "sparsified" with little or no loss of information by simply establishing a cutoff threshold on its elements. Biased counts and probabilities can be computed *on demand* for specific n-grams without re-estimating the entire LM.

4 Experimental Results

We measure the utility of the proposed LM biasing technique in two ways: (a) given a parallel test corpus, by comparing source-conditional target perplexity with biased LMs to target perplexity with the static LM, and (b) by comparing SMT performance with static and biased LMs. We conduct experiments on two resource-poor language pairs commissioned under the DARPA Transtac speech-to-speech translation initiative, viz. English-Dari (E2D) and English-Pashto (E2P), on test sets with single as well as multiple references.

Data set	**E2D**	**E2P**
TM Training	138k pairs	168k pairs
LM Training	179k sentences	302k sentences
Development	3,280 pairs	2,385 pairs
Test (1-ref)	2,819 pairs	1,113 pairs
Test (4-ref)	-	564 samples

Table 1: Data configuration for perplexity/SMT experiments. Multi-reference test set is not available for E2D. LM training data in words: 2.4M (Dari), 3.4M (Pashto)

4.1 Data Configuration

Parallel data were made available under the Transtac program for both language pairs evaluated in this paper. We divided these into training, held-out development, and test sets for building, tuning, and evaluating the SMT system, respectively. These development and test sets provide only one reference translation for each source sentence. For E2P, DARPA has made available to all program participants an additional evaluation set with multiple (four) references for each test input. The Dari and Pashto monolingual corpora for LM training are a superset of target sentences from the parallel training corpus, consisting of additional untranslated sentences, as well as data derived from other sources, such as the web. Table 1 lists the corpora used in our experiments.

4.2 Perplexity Analysis

For both Dari and Pashto, we estimated a static trigram LM with unit sentence level weights that served as a baseline. We tuned this LM by varying the bigram and trigram frequency cutoff thresholds to minimize perplexity on the held-out target sentences. Finally, we evaluated test target perplexity with the optimized baseline LM.

We then applied the proposed technique to estimate trigram LMs biased to source sentences in the held-out and test sets. We evaluated source-conditional target perplexity by computing the total log-probability of all target sentences in a parallel test corpus against the LM biased by the corresponding source sentences. Again, bigram and trigram cutoff thresholds were tuned to minimize source-conditional target perplexity on the held-out set. The tuned biased LMs were used to compute source-conditional target perplexity on the test set.

Eval set	**Static**	**Biased**	**Reduction**
E2D-1ref-dev	159.3	137.7	13.5%
E2D-1ref-tst	178.3	156.3	12.3%
E2P-1ref-dev	147.3	130.6	11.3%
E2P-1ref-tst	122.7	108.8	11.3%

Table 2: Reduction in perplexity using biased LMs.

Test set	**BLEU**		**100-TER**	
	Static	**Biased**	**Static**	**Biased**
E2D-1ref-tst	14.4	14.8	29.6	30.5
E2P-1ref-tst	13.0	13.3	28.3	29.4
E2P-4ref-tst	25.6	26.1	35.0	35.8

Table 3: SMT performance with static and biased LMs.

Witten-Bell discounting was used for smoothing all LMs. Table 2 summarizes the reduction in target perplexity using biased LMs; on the E2D and E2P single-reference test sets, we obtained perplexity reductions of 12.3% and 11.3%, respectively. This indicates that the biased models are significantly better predictors of the corresponding target sentences than the static baseline LM.

4.3 Translation Experiments

Having determined that target sentences of a parallel test corpus better fit biased LMs estimated from the corresponding source-weighted training corpus, we proceeded to conduct SMT experiments on both language pairs to demonstrate the utility of biased LMs in improving translation performance.

We used an internally developed phrase-based SMT system, similar to Moses (Koehn et al., 2007), as a test-bed for our translation experiments. We used GIZA++ to induce automatic word alignments from the parallel training corpus. Phrase translation rules (up to a maximum source span of 5 words) were extracted from a combination of forward and backward word alignments (Koehn et al., 2003). The SMT decoder uses a log-linear model that combines numerous features, including but not limited to phrase translation probability, LM probability, and distortion penalty, to estimate the posterior probability of target hypotheses. We used minimum error rate training (MERT) (Och, 2003) to tune the feature weights for maximum BLEU (Papineni et al., 2001) on the development set. Finally, we evaluated SMT performance on the test set in terms of BLEU and TER (Snover et al., 2006).

The baseline SMT system used the static trigram LM with cutoff frequencies optimized for minimum perplexity on the development set. Biased LMs (with n-gram cutoffs tuned as above) were estimated for all source sentences in the development and test sets, and were used to decode the corresponding inputs. Table 3 summarizes the consistent improvement in BLEU/TER across multiple test sets and language pairs.

5 Discussion and Future Work

Existing methods for target LM biasing for SMT rely on information retrieval to select a comparable subset from the training corpus. A foreground LM estimated from this subset is interpolated with the static background LM. However, given the large size of a typical LM corpus, these methods are unsuitable for on-line, interactive SMT applications.

In this paper, we proposed a novel LM biasing technique based on linear transformations of target sentences in a sparse vector space. We adopted a fine-grained approach, weighting individual target sentences based on the proposed measure of cross-lingual similarity, and by using the entire, weighted corpus to estimate a biased LM. We then sketched an implementation that improves the time and space efficiency of our method by pre-computing and “sparsifying” n-gram projections off-line during the training phase. Thus, our approach can be integrated within on-line, low-latency SMT systems. Finally, we showed that biased LMs yield significant reductions in target perplexity, and consistent improvements in SMT performance.

While we used phrase-based SMT as a test-bed for evaluating translation performance, it should be noted that the proposed LM biasing approach is independent of SMT architecture. We plan to test its effectiveness in hierarchical and syntax-based SMT systems. We also plan to investigate the relative usefulness of LM biasing as we move from low-resource languages to those for which significantly larger parallel corpora and LM training data are available.

References

Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, Dan Melamed, Franz Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation: Final report. Technical report, JHU Summer Workshop.

Peter E. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19:263–311.

Woosung Kim. 2005. *Language Model Adaptation for Automatic Speech Recognition and Statistical Machine Translation*. Ph.D. thesis, The Johns Hopkins University, Baltimore, MD.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 48–54, Morristown, NJ, USA. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167, Morristown, NJ, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. BLEU: A method for automatic evaluation of machine translation. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings AMTA*, pages 223–231, August.

Matthew Snover, Bonnie Dorr, and Richard Schwartz. 2008. Language and translation model adaptation using comparable corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 857–866, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bing Zhao, Matthias Eck, and Stephan Vogel. 2004. Language model adaptation for statistical machine translation with structured query models. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.