Edinburgh's Phrase-based Machine Translation Systems for WMT-14

Nadir Durrani **Barry Haddow** **Philipp Koehn**
School of Informatics
University of Edinburgh
{dnadir,bhaddow,pkoehn}@inf.ed.ac.uk

Kenneth Heafield
Computer Science Department
Stanford University
heafield@cs.stanford.edu

Abstract

This paper describes the University of Edinburgh's (UEDIN) phrase-based submissions to the translation and medical translation shared tasks of the 2014 Workshop on Statistical Machine Translation (WMT). We participated in all language pairs. We have improved upon our 2013 system by i) using generalized representations, specifically automatic word clusters for translations out of English, ii) using unsupervised character-based models to translate unknown words in Russian-English and Hindi-English pairs, iii) synthesizing Hindi data from closely-related Urdu data, and iv) building huge language on the common crawl corpus.

1 Translation Task

Our baseline systems are based on the setup described in (Durrani et al., 2013b) that we used for the Eighth Workshop on Statistical Machine Translation (Bojar et al., 2013). The notable features of these systems are described in the following section. The experiments that we carried out for this year's translation task are described in the following sections.

1.1 Baseline

We trained our systems with the following settings: a maximum sentence length of 80, grow-diag-final-and symmetrization of GIZA++ alignments, an interpolated Kneser-Ney smoothed 5-gram language model with KenLM (Heafield, 2011) used at runtime, hierarchical lexicalized reordering (Galley and Manning, 2008), a lexically-driven 5-gram operation sequence model (OSM) (Durrani et al., 2013a) with 4 count-based supportive features, sparse domain indicator, phrase length, and count bin features (Blunsom and Osborne, 2008; Chiang et al., 2009), a distortion limit of 6, maximum phrase-length of 5, 100-best translation options, Minimum Bayes Risk decoding (Kumar and Byrne, 2004), Cube Pruning (Huang and Chiang, 2007), with a stack-size of 1000 during tuning and 5000 during test and the no-reordering-over-punctuation heuristic (Koehn and Haddow, 2009). We used POS and morphological tags as additional factors in phrase translation models (Koehn and Hoang, 2007) for German-English language pairs. We also trained target sequence models on the in-domain subset of the parallel corpus using Kneser-Ney smoothed 7-gram models. We used syntactic-preordering (Collins et al., 2005) and compound splitting (Koehn and Knight, 2003) for German-to-English systems. We used trivia tokenizer for tokenizing Hindi.

The systems were tuned on a very large tuning set consisting of the test sets from 2008-2012, with a total of 13,071 sentences. We used news-test 2013 for the dev experiments. For Russian-English pairs news-test 2012 was used for tuning and for Hindi-English pairs, we divided the news-dev 2014 into two halves, used the first half for tuning and second for dev experiments.

1.2 Using Generalized Word Representations

We explored the use of automatic word clusters in phrase-based models (Durrani et al., 2014a). We computed the clusters with GIZA++'s `mkcls` (Och, 1999) on the source and target side of the parallel training corpus. Clusters are word classes that are optimized to reduce n-gram perplexity. By generating a cluster identifier for each output word, we are able to add an n-gram model

Proceedings of the Ninth Workshop on Statistical Machine Translation, pages 97–104,
Baltimore, Maryland USA, June 26–27, 2014.

over these identifiers as an additional scoring function. The inclusion of such an additional factor is trivial given the factored model implementation (Koehn and Hoang, 2007) of Moses (Koehn et al., 2007). The n-gram model is trained in the similar way as the regular language model. We trained domain-specific language models separately and then linearly interpolated them using SRILM with weights optimized on the tuning set (Schwenk and Koehn, 2008).

We also trained OSM models over cluster-ids (Durrani et al., 2014a). The lexically driven OSM model falls back to very small context sizes of two to three operations due to data sparsity. Learning operation sequences over cluster-ids enables us to learn richer translation and reordering patterns that can generalize better in sparse data conditions. Table 1 shows gains from adding target LM and OSM models over cluster-ids. Using word clusters was found more useful translating from English-to-*.

	from English			into English		
Lang	$\mathbf{B_0}$	+Cid	Δ	$\mathbf{B_0}$	+Cid	Δ
de	20.60	20.85	+0.25	27.44	27.34	-0.10
cs	18.84	19.39	+0.55	26.42	26.42	±0.00
fr	30.73	30.82	+0.09	31.64	31.76	+0.12
ru	18.78	19.67	+0.89	24.45	24.63	+0.18
hi	10.39	10.52	+0.13	15.48	15.26	-0.22

Table 1: Using Word Clusters in Phrase-based and OSM models – $\mathbf{B_0}$ = System without Clusters, **+Cid** = with Cluster

We also trained OSM models over POS and morph tags. For the English-to-German system we added an OSM model over [pos, morph] (source:pos, target:morph) and for the German-to-English system we added an OSM model over [morph,pos] (source:morph, target:pos), a configuration that was found to work best in our previous experiments (Birch et al., 2013). Table 2 shows gains from additionally using OSM models over POS/morph tags.

Lang	$\mathbf{B_0}$	$+\mathbf{OSM_{p,m}}$	Δ
en-de	20.44	20.60	+0.16
de-en	27.24	27.44	+0.20

Table 2: Using POS and Morph Tags in OSM models – $\mathbf{B_0}$ = Baseline, $+\mathbf{OSM_{p,m}}$ = POS/Morph-based OSM

1.3 Unsupervised Transliteration Model

Last year, our Russian-English systems performed badly on the human evaluation. In comparison other participants that used transliteration did well. We could not train a transliteration system due to unavailability of a transliteration training data. This year we used an EM-based method to induce unsupervised transliteration models (Durrani et al., 2014b). We extracted transliteration pairs automatically from the word-aligned parallel data and used it to learn a transliteration system. We then built transliteration phrase-tables for translating OOV words and used the post-decoding method (Method 2 as described in the paper) to translate these.

Pair	Training	OOV	$\mathbf{B_0}$	$+\mathbf{T_r}$	Δ
ru-en	232K	1356	24.63	25.06	+0.41
en-ru	232K	681	19.67	19.91	+0.24
hi-en	38K	503	14.67	15.48	+0.81
en-hi	38K	394	11.76	12.83	+1.07

Table 3: Using Unsupervised Transliteration Model – Training = Extracted Transliteration Corpus (types), OOV = Out-of-vocabulary words (tokens) $\mathbf{B_0}$ = System without Transliteration, $+\mathbf{T_r}$ = Transliterating OOVs

Table 3 shows the number (types) of transliteration pairs extracted using unsupervised mining, number of OOV words (tokens) in each pair and the gains achieved by transliterating unknown words.

1.4 Synthesizing Hindi Data from Urdu

Hindi and Urdu are closely related language pairs that share grammatical structure and have a large overlap in vocabulary. This provides a strong motivation to transform any Urdu-English parallel data into Hindi-English by translating the Urdu part into Hindi. We made use of the Urdu-English segment of the Indic multi-parallel corpus (Post et al., 2012) which contains roughly 87K sentence pairs. The Hindi-English segment of this corpus is a subset of parallel data made available for the translation task but is completely disjoint from the Urdu-English segment.

We initially trained a Urdu-to-Hindi SMT system using a very tiny EMILLE[1] corpus (Baker

[1]EMILLE corpus contains roughly 12000 sentences of Hindi and Urdu comparable data. From these we were able to sentence align 7000 sentences to build an Urdu-to-Hindi system.

et al., 2002). But we found this system to be useless for translating the Urdu part of Indic data due to domain mismatch and huge number of OOV words (approximately 310K tokens). To reduce sparsity we synthesized additional phrase-tables using interpolation and transliteration.

Interpolation: We trained two phrase translation tables $p(\bar{u_i}|\bar{e_i})$ and $p(\bar{e_i}|\bar{h_i})$, from Urdu-English (Indic corpus) and Hindi-English (HindEnCorp (Bojar et al., 2014)) bilingual corpora. Given the phrase-table for Urdu-English $p(\bar{u_i}|\bar{e_i})$ and the phrase-table for English-Hindi $p(\bar{e_i}|\bar{h_i})$, we estimated a Urdu-Hindi phrase-table $p(\bar{u_i}|\bar{h_i})$ using the well-known convolution model (Utiyama and Isahara, 2007; Wu and Wang, 2007):

$$p(\bar{u_i}|\bar{h_i}) = \sum_{\bar{e_i}} p(\bar{u_i}|\bar{e_i})p(\bar{e_i}|\bar{h_i})$$

The number of entries in the baseline Urdu-to-Hindi phrase-table were approximately 254K. Using interpolation we were able to build a phrase-table containing roughly 10M phrases. This reduced the number of OOV tokens from 310K to approximately 50K.

Transliteration: Urdu and Hindi are written in different scripts (Arabic and Devanagri respectively). We added a transliteration component to our Urdu-to-Hindi system. An unsupervised transliteration model is learned from the word-alignments of Urdu-Hindi parallel data. We were able to extract around 2800 transliteration pairs. To learn a richer transliteration model, we additionally fed the interpolated phrase-table, as described above, to the transliteration miner. We were able to mine additional 21000 transliteration pairs and built a Urdu-Hindi character-based model from it. The transliteration module can be used to translate the 50K OOV words but previous research (Durrani et al., 2010; Nakov and Tiedemann, 2012) has shown that transliteration is useful for more than just translating OOV words when translating closely related language pairs. To fully capitalize on the large overlap in Hindi–Urdu vocabulary, we transliterated each word in the Urdu test-data into Hindi and produced a phrase-table with 100-best transliterations. The two synthesized (triangulated and transliterated) phrase-tables are then used along with the baseline Urdu-to-Hindi phrase-table in a log-linear model. Detailed results on Urdu-to-Hindi baseline and improvements obtained from using transliteration and triangulated phrase-tables are presented in Durrani and Koehn (2014). Using our best Urdu-to-Hindi system, we translated the Urdu part of the multi-indic corpus to form Hindi-English parallel data. Table 4 shows results from using the synthesized Hindi-English corpus in isolation (**Syn**) and on top of the baseline system (**$\mathbf{B_0}$ + Syn**).

Pair	$\mathbf{B_0}$	**Syn**	Δ	$\mathbf{B_0}$ + **Syn**	Δ
hi-en	14.28	10.49	-3.79	14.72	+0.44
en-hi	10.59	9.01	-1.58	11.76	+1.17

Table 4: Evaluating Synthesized (Syn) Hindi-English Parallel Data, $\mathbf{B_0}$ = System without Synthesized Data

1.5 Huge Language Models

Our unconstrained submissions use an additional language model trained on web pages from the 2012, 2013, and winter 2013 CommonCrawl.[2] The additional language model is the only difference between the constrained and unconstrained submissions; we did not use additional parallel data. These language models were trained on text provided by the CommonCrawl foundation, which they converted to UTF-8 after stripping HTML. Languages were detected using the Compact Language Detection 2[3] and, except for Hindi where we lack tools, sentences were split with the Europarl sentence splitter (Koehn, 2005). All text was then deduplicated, minimizing the impact of boilerplate, such as social media sharing buttons. We then tokenized and truecased the text as usual. Statistics are shown in Table 5. A full description of the pipeline, including a public data release, appears in Buck et al. (2014).

Lang	Lines (B)	Tokens (B)	Bytes
en	59.13	975.63	5.14 TiB
de	3.87	51.93	317.46 GiB
fr	3.04	49.31	273.96 GiB
ru	1.79	21.41	220.62 GiB
cs	0.47	5.79	34.67 GiB
hi	0.01	0.28	3.39 GiB

Table 5: Size of huge language model training data

We built unpruned modified Kneser-Ney language models using lmplz (Heafield et al., 2013).

[2] http://commoncrawl.org

[3] https://code.google.com/p/cld2/

Pair	$\mathbf{B_0}$		+L	
newstest	2013	2014	2013	2014
en-de	20.85	20.10	–	20.61 +0.51
en-cs	19.39	21.00	20.03 +0.64	21.60 +0.60
en-ru	19.90	28.70	20.80 +0.90	29.90 +1.20
en-hi	11.43	11.10	12.83 +1.40	12.50 +1.40
hi-en	15.48	13.90	–	14.80 +0.90

Table 6: Gains obtained by using huge language models – $\mathbf{B_0}$ = Baseline, $+\mathbf{L}$ = Adding Huge LM

While the Hindi and Czech models are small enough to run directly, models for other languages are quite large.We therefore created a filter that operates directly on files in KenLM trie binary format, preserving only n-grams whose words all appear in the target side vocabulary of at least one source sentence. For example, an English language model trained on just the 2012 and 2013 crawls takes 3.5 TB without any quantization. After filtering to the Hindi-English tuning set, the model fit in 908 GB, again without quantization. We were then able to tune the system on a machine with 1 TB RAM. Results are shown in Table 6; we did not submit to English-French because the system takes too long to tune.

1.6 Miscellaneous

Hindi-English: 1) A large number of Hindi sentences in the Hindi-English parallel corpus were ending with a full-stop ".", although the end-of-the-sentence marker in Hindi is "Danda" (|). Replacing full-stops with Danda gave improvement of +0.20 for hi-en and +0.40 in en-hi. 2) Using Wiki subtitles did not give any improvement in BLEU and were in fact harmful for the en-hi direction.

Russian-English: We tried to improve word-alignments by integrating a transliteration submodel into GIZA++ word aligner. The probability of a word pair is calculated as an interpolation of the transliteration probability and translation probability stored in the t-table of the different alignment models used by the GIZA++ aligner. This interpolation is done for all iterations of all alignment models (See Sajjad et al. (2013) for details). Due to shortage of time we could only run it for Russian-to-English. The improved alignments gave a gain of +0.21 on news-test 2013 and +0.40 on news-test 2014.

Pair	GIZA++	Fast Align	Δ
de-en	24.02	23.89	–.13
fr-en	30.78	30.66	–.12
es-en	34.07	34.24	+.17
cs-en	22.63	22.44	–.19
ru-en	31.68	32.03	+.35
en-de	18.04	17.88	–.16
en-fr	28.96	28.83	–.13
en-es	34.15	34.32	+.17
en-cs	15.70	16.02	+.32
avg			+.03

Table 7: Comparison of fast word alignment method (Dyer et al., 2013) against GIZA++ (WMT 2013 data condition, test on newstest2012). The method was not used in the official submission.

Pair	Baseline MSD	Hier. MSD	Hier. MSLR
de-en	27.04	27.10 +.06	27.17 +.13
fr-en	31.63	-	31.65 +.02
es-en	31.20	31.14 –.06	31.25 +.05
cs-en	26.11	26.32 +.21	26.26 +.15
ru-en	24.09	24.01 –.08	24.19 +.11
en-de	20.43	20.34 –.09	20.32 -.11
en-fr	30.54	-	30.52 –.02
en-es	30.36	30.44 +.08	30.51 +.15
en-cs	18.53	18.59 +.06	18.66 +.13
en-ru	18.37	18.47 +.10	18.19 –.18
avg		+ .035	+.045

Table 8: Hierarchical lexicalized reordering model (Galley and Manning, 2008).

Fast align: In preliminary experiments, we compared the fast word alignment method by Dyer et al. (2013) against our traditional use of GIZA++. Results are quite mixed (Table 7), ranging from a gain of +.35 for Russian-English to a loss of –.19 for Czech-English. We stayed with GIZA++ for all of our other experiments.

Hierarchical lexicalized reordering model: We explored the use of the hierarchical lexicalized reordering model (Galley and Manning, 2008) in two variants: using the same orientations as our traditional model (*monotone, discontinuous, swap*), and one that distinguishes the *discontinuous* orientations to the *left* and *right*. Table 8 shows slight improvements with these models, so we used them in our baseline.

Threshold filtering of phrase table: We experimented with discarding some phrase table entry due to their low probability. We found that phrase translations with the phrase translation probability

$\phi(f|e)<10^{-4}$ can be safely discarded with almost no change in translations. However, discarding phrase translations with the inverse phrase translation probability $\phi(e|f)<10^{-4}$ is more risky, especially with morphologically rich target languages, so we kept those.

1.7 Summary

Table 9 shows cumulative gains obtained from using word classes, transliteration and big language models[4] over the baseline system. Our German-English constrained systems were used for EU-Bridge system combination, a collaborative effort to improve the state-of-the-art in machine translation (See Freitag et al. (2014) for details).

	from English			into English		
Lang	$\mathbf{B_0}$	$\mathbf{B_1}$	Δ	$\mathbf{B_0}$	$\mathbf{B_1}$	Δ
de	20.44	20.85	+0.41	27.24	27.44	+0.20
cs	18.84	20.03	+1.19	26.42	26.42	±0.00
fr	30.73	30.82	+0.09	31.64	31.76	+0.12
ru	18.78	20.81	+2.03	24.45	25.21	+0.76
hi	9.27	12.83	+3.56	14.08	15.48	+1.40

Table 9: Cumulative gains obtained for each language – $\mathbf{B_0}$ = Baseline, $\mathbf{B_1}$ = Best System

2 Medical Translation Task

For the medical translation task, the organisers supplied several medical domain corpora (detailed on the task website), as well some out-of-domain patent data, and also all the data available for the constrained track of the news translation task was permitted. In general, we attempted to use all of this data, except for the LDC Gigaword language model data (for reasons of time) and we divided the data into "in-domain" and "out-of-domain" corpora. The data sets are summarised in Tables 10 and 11.

In order to create systems for the medical translation tasks, we used phrase-based Moses with exactly the same settings as for the news translation task, including the OSM (Durrani et al., 2011), and compound splitting Koehn and Knight (2003) for German source. We did not use word clusters (Section 1.2), as they did not give good results on this task, but we have yet to find a reason for this. For language model training, we decided not to build separate models on each corpus as there was a large variation in corpus sizes. Instead we concatenated the in-domain target sides with the in-domain extra monolingual data to create training data for an in-domain language model, and similarly for the out-of-domain data. The two language models were interpolated using SRILM, minimising perplexity on the Khresmoi summary development data.

During system development, we only had 500 sentences of development data (SUMMARY-DEV) from the Khresmoi project, so we decided to select further development and devtest data from the EMEA corpus, reasoning that it was fairly close in domain to SUMMARY-DEV. We selected a tuning set (5000 sentence pairs, which were added to SUMMARY-DEV) and a devtest set (3000 sentence pairs) from EMEA after first de-duplicating it, and ignoring sentence pairs which were too short, or

[4]Cumulative gains do not include gains obtain from big language models for hi-en and en-de.

Data Set	cs-en	de-en	fr-en
coppa-in	n	n	y
PatTR-in-claims	n	y	y
PatTR-in-abstract	n	y	y
PatTR-in-titles	n	y	y
UMLS	y	y	y
MuchMore	n	y	n
EMEA	y	y	y
WikiTitles	y	y	y
PatTR-out	n	y	y
coppa-out	n	n	y
MultiUN	n	n	y
czeng	y	n	n
europarl	y	y	y
news-comm	y	y	y
commoncrawl	y	y	y
FrEnGiga	n	n	y

Table 10: Parallel data sets used in the medical translation task. The sets above the line were classified as "in-domain" and those below as "out-of-domain".

Data Set	cs	de	en	fr
PIL	n	n	y	n
DrugBank	n	n	y	n
WikiArticles	y	y	y	y
PatTR-in-description	n	y	y	y
GENIA	n	n	y	n
FMA	n	n	y	n
AACT	n	n	y	n
PatTR-out-description	n	y	y	y

Table 11: Additional monolingual data used in the medical translation task. Those above the line were classified as "in-domain" and the one below as "out-of-domain". We also used the target sides of all the parallel corpora for language modelling.

contained too many capital letters or numbers. The EMEA contains many duplicated sentences, and we removed all sentence pairs where either side was a duplicate, reducing the size of the corpus to about 25% of the original. We also removed EMEA from Czeng, since otherwise it would overlap with our selected development sets.

We also experimented with modified Moore-Lewis (Moore and Lewis, 2010; Axelrod et al., 2011) data selection, using the EMEA corpus as the in-domain corpus (for the language model required in MML) and selecting from all the out-of-domain data.

When running on the final test set (SUMMARY-TEST) we found that it was better to tune just on SUMMARY-DEV, even though it was much smaller than the EMEA dev set we had selected. All but two (cs-en, de-en) of our submitted systems used the MML selection, because it worked better on our EMEA devtest set. However, as can be seen from Table 12, systems built with all the data generally perform better. We concluded that EMEA was not a good representative of the Khresmoi data, perhaps because of domain differences, or perhaps just because of the alignment noise that appears (from informal inspection) to be present in EMEA.

	from English			**into English**		
	in	in+20	in+out	in	in+20	in+out
de	18.59	*20.88*	–	36.17	–	38.57
cs	18.78	*23.45*	23.77	30.12	–	36.32
fr	35.24	*40.74*	41.04	45.15	*46.44*	46.58

Table 12: Results (cased BLEU) on the khresmoi summary test set. The "in" systems include all in-domain data, the "in+20" systems also include 20% of the out-of-domain data and the "out" systems include all data. The submitted systems are shown in italics, except for de-en and cs-en where we submitted a "in+out" systems. For de-en, this was tuned on SUMMARY-DEV plus the EMEA dev set and scored 37.31, whilst for cs-en we included LDC Giga in the LM, and scored 36.65.

For translating the Khresmoi queries, we used the same systems as for the summaries, except that generally we did not retune on the SUMMARY-DEV data. We added a post-processing script to strip out extraneous stop words, which improved BLEU, but we would not expect it to matter in a real CLIR system as it would do its own stop-word removal.

Acknowledgments

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreements n° 287658. (EU-BRIDGE) and n° 288769 (ACCEPT). Huge language model experiments made use of the Stampede supercomputer provided by the Texas Advanced Computing Center (TACC) at The University of Texas at Austin under NSF XSEDE allocation TG-CCR140009. We also acknowledge the support of the Defense Advanced Research Projects Agency (DARPA) Broad Operational Language Translation (BOLT) program through IBM. This publication only reflects the authors' views.

References

Axelrod, A., He, X., and Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 355–362, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Baker, P., Hardie, A., McEnery, T., Cunningham, H., and Gaizauskas, R. J. (2002). EMILLE, a 67-million word corpus of indic languages: Data collection, mark-up and harmonisation. In *LREC*.

Birch, A., Durrani, N., and Koehn, P. (2013). Edinburgh SLT and MT system description for the IWSLT 2013 evaluation. In *Proceedings of the 10th International Workshop on Spoken Language Translation*, pages 40–48, Heidelberg, Germany.

Blunsom, P. and Osborne, M. (2008). Probabilistic inference for machine translation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 215–223, Honolulu, Hawaii. Association for Computational Linguistics.

Bojar, O., Buck, C., Callison-Burch, C., Federmann, C., Haddow, B., Koehn, P., Monz, C., Post, M., Soricut, R., and Specia, L. (2013). Findings of the 2013 workshop on statistical machine translation. In *Eighth Workshop on Statistical Machine Translation*, WMT-2013, pages 1–44, Sofia, Bulgaria.

Bojar, O., Diatka, V., Rychlý, P., Straňák, P., Tamchyna, A., and Zeman, D. (2014). Hindi-English and Hindi-only Corpus for Machine

Translation. In *Proceedings of the Ninth International Language Resources and Evaluation Conference (LREC'14)*, Reykjavik, Iceland. ELRA, European Language Resources Association. in prep.

Buck, C., Heafield, K., and van Ooyen, B. (2014). N-gram counts and language models from the common crawl. In *Proceedings of the Language Resources and Evaluation Conference*, Reykjavík, Iceland.

Chiang, D., Knight, K., and Wang, W. (2009). 11,001 New Features for Statistical Machine Translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 218–226, Boulder, Colorado. Association for Computational Linguistics.

Collins, M., Koehn, P., and Kucerova, I. (2005). Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540, Ann Arbor, Michigan. Association for Computational Linguistics.

Durrani, N., Fraser, A., Schmid, H., Hoang, H., and Koehn, P. (2013a). Can markov models over minimal translation units help phrase-based SMT? In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, Sofia, Bulgaria. Association for Computational Linguistics.

Durrani, N., Haddow, B., Heafield, K., and Koehn, P. (2013b). Edinburgh's machine translation systems for european language pairs. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, Sofia, Bulgaria. Association for Computational Linguistics.

Durrani, N. and Koehn, P. (2014). Improving machine translation via triangulation and transliteration. In *Proceedings of the 17th Annual Conference of the European Association for Machine Translation*, Dubrovnik, Croatia. To Appear.

Durrani, N., Koehn, P., Schmid, H., and Fraser, A. (2014a). Investigating the usefulness of generalized word representations in SMT. In *Proceedings of the 17th Annual Conference of the European Association for Machine Translation*, Dubrovnik, Croatia. To Appear.

Durrani, N., Sajjad, H., Fraser, A., and Schmid, H. (2010). Hindi-to-urdu machine translation through transliteration. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 465–474, Uppsala, Sweden. Association for Computational Linguistics.

Durrani, N., Sajjad, H., Hoang, H., and Koehn, P. (2014b). Integrating an unsupervised transliteration model into statistical machine translation. In *Proceedings of the 15th Conference of the European Chapter of the ACL (EACL 2014)*, Gothenburg, Sweden. Association for Computational Linguistics.

Durrani, N., Schmid, H., and Fraser, A. (2011). A joint sequence translation model with integrated reordering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1045–1054, Portland, Oregon, USA.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Freitag, M., Peitz, S., Wuebker, J., Ney, H., Huck, M., Sennrich, R., Durrani, N., Nadejde, M., Williams, P., Koehn, P., Herrmann, T., Cho, E., and Waibel, A. (2014). EU-BRIDGE MT: combined machine translation. In *Proceedings of the ACL 2014 Ninth Workshop on Statistical Machine Translation*, Baltimore, MD, USA.

Galley, M. and Manning, C. D. (2008). A simple and effective hierarchical phrase reordering model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 848–856, Honolulu, Hawaii.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland, United Kingdom.

Heafield, K., Pouzyrevsky, I., Clark, J. H., and Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. In *Proceedings*

of the 51st Annual Meeting of the Association for Computational Linguistics, Sofia, Bulgaria.

Huang, L. and Chiang, D. (2007). Forest rescoring: Faster decoding with integrated language models. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 144–151, Prague, Czech Republic. Association for Computational Linguistics.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT Summit*.

Koehn, P. and Haddow, B. (2009). Edinburgh's Submission to all Tracks of the WMT 2009 Shared Task with Reordering and Speed Improvements to Moses. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 160–164, Athens, Greece. Association for Computational Linguistics.

Koehn, P. and Hoang, H. (2007). Factored translation models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 868–876, Prague, Czech Republic. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *ACL 2007 Demonstrations*, Prague, Czech Republic.

Koehn, P. and Knight, K. (2003). Empirical methods for compound splitting. In *Proceedings of Meeting of the European Chapter of the Association of Computational Linguistics (EACL)*.

Kumar, S. and Byrne, W. J. (2004). Minimum bayes-risk decoding for statistical machine translation. In *HLT-NAACL*, pages 169–176.

Moore, R. C. and Lewis, W. (2010). Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 220–224, Uppsala, Sweden. Association for Computational Linguistics.

Nakov, P. and Tiedemann, J. (2012). Combining word-level and character-level models for machine translation between closely-related languages. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 301–305, Jeju Island, Korea. Association for Computational Linguistics.

Och, F. J. (1999). An efficient method for determining bilingual word classes. In *Ninth Conference the European Chapter of the Association for Computational Linguistics (EACL)*, pages 71–76.

Post, M., Callison-Burch, C., and Osborne, M. (2012). Constructing parallel corpora for six indian languages via crowdsourcing. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 401–409, Montréal, Canada. Association for Computational Linguistics.

Sajjad, H., Smekalova, S., Durrani, N., Fraser, A., and Schmid, H. (2013). QCRI-MES submission at wmt13: Using transliteration mining to improve statistical machine translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, Sofia, Bulgaria. Association for Computational Linguistics.

Schwenk, H. and Koehn, P. (2008). Large and diverse language models for statistical machine translation. In *International Joint Conference on Natural Language Processing*, pages 661–666.

Utiyama, M. and Isahara, H. (2007). A comparison of pivot methods for phrase-based statistical machine translation. In *2007 Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 484–491.

Wu, H. and Wang, H. (2007). Pivot language approach for phrase-based statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 856–863, Prague, Czech Republic. Association for Computational Linguistics.