Chapter 3
Machine Translation System Combination with Enhanced Alignments Using Word Embeddings

Ch Ram Anirudh and Kavi Narayana Murthy

Abstract Machine Translation (MT) is a challenging problem and various techniques proposed for MT have their own strengths and weaknesses. Combining various MT systems has shown promising results. *Confusion network decoding* is one such approach. In this work, we propose using word embeddings for aligning words from different hypotheses during confusion network generation. Our experiments, on English-Hindi language pair, have shown statistically significant improvement in BLEU scores, when compared to the baseline system combination. Four data-driven MT systems are combined, namely, a phrase based MT, hierarchical-phrase based MT, bi-directional recurrent neural network MT and transformer based MT. All of these have been trained on IIT Bombay English-Hindi parallel corpus.

3.1 Introduction

Machine Translation (MT) is a challenging problem and various techniques proposed for MT have their own strengths and weaknesses. For example, Neural MT systems are good at fluency, whereas they suffer with the problems of unknown words, amount of training data, length of sentences, word alignment and domain mismatch [17]. Combining various MT systems could capitalize on these differences to obtain improved translation quality.

Combining MT systems has shown improvement in performance [3, 8, 10, 28, 30]. Systems may be combined in two ways: (1) by intervention at the level of architectures and (2) by combining only the outputs of various MT systems. The focus of the current work is of the latter kind: *confusion network decoding* [3, 8, 28]. In this method, the outputs (*hypotheses*) from various MT systems are combined in a directed acyclic graph structure called *confusion network*.

C. R. Anirudh (✉) · K. N. Murthy
School of Computer and Information Sciences,
University of Hyderabad, Hyderabad 500046, India

S. C. Satapathy et al. (eds.), *Intelligent Data Engineering and Analytics*,
Smart Innovation, Systems and Technologies 266,
https://doi.org/10.1007/978-981-16-6624-7_3

System combination using confusion network involves three steps: *confusion network generation, scoring and decoding*. *Confusion network generation* proceeds by choosing a hypothesis called *primary hypothesis* from different MT hypotheses and aligning semantically similar words from remaining hypotheses to the primary hypothesis. *Scoring* involves assigning scores using ideas such as majority voting score (number of hypotheses a word occurred in), language model probability and word penalty. *Decoding* proceeds by beam search through the hypotheses space generated by traversing the network.

Alignment of words from different hypotheses is a crucial step in confusion network generation. Various methods have been proposed for alignment: Levenshtein Distance [3], Translation Edit Rate (TER) [28], IBM model alignments [22] and alignments in Meteor [8]. An open source Statistical MT system *Jane* [8], is used for system combination in our work. Jane uses Meteor [2] matcher for alignments. Meteor aligns words based on *exact* string matching, *stem* matching and *synonyms*. Meteor is flexible and modules can be further added subject to the availability of resources. In this work, we add a fourth module: *word embeddings* matcher to Meteor. Word embeddings represent words as vectors in an n-dimensional space, capturing a significant amount of semantic information. *word2vec* [23], *fastText* [5] and *GloVe* [26] are some of the well known word embedding schemes. Semantically similar words tend to form clusters in this n-dimensional space [23]. Therefore, distance between the vectors (word embeddings) can be used to align semantically similar words from different MT systems. To the best of our knowledge, this is the first attempt to use word embeddings for alignment in confusion network decoding.

In this work, outputs of four English to Hindi MT systems, namely, *Phrase Based statistical MT (PBMT)* [18], *Hierarchical Phrase Based MT (HPBMT)* [7], *Bi-directional Recurrent Neural Network based neural MT (BRNN)* [21] *and Transformer based neural MT (TMT)* [33] are combined. A system combination baseline is built with default settings (exact, stem and synonymy) in Meteor. Four different system combinations are built using four different word embeddings for alignment, namely word2vec [23] (two variants: *skip-gram and continuous bag of words (cbow)*), *fastText* [5] and *GloVe* [26]. We compare our proposed approach with the individual MT systems and the baseline system combination. Statistically significant improvements in BLEU [25] scores are observed in three cases: *skip-gram, cbow* and *fastText*. IIT Bombay English-Hindi parallel corpus [20] is used for training the individual MT systems. PBMT and HPBMT systems are trained using *Moses* [16]. BRNN and TMT systems are trained using *OpenNMT* [14].

3.2 Methodology

Ensembling outputs of various MT systems is done by boosting. In boosting for classification, various classifiers are trained and the class label that is assigned by a majority of classifiers (majority voting) is chosen as the final label. Unlike a classification task, output of an MT system is a sequence of words. Majority voting may not work since each MT system may generate a different sequence. To handle

1. उन्होंने कहा कि यह मनोरंजन पर केंद्रित होगा
   unhone kahā ki yah manoranjan par kendrit hogā
2. उसने कहा कि निवासियों के मनोरंजन पर ध्यान केंद्रित किया जाएगा
   usne kahā ki nivāsiyon ke manoranjan par kendrit kiyājāyegā
3. उन्होंने कहा कि पर्यटकों के मनोरंजन पर फोकस होगा
   unhone kahā ki paryatakon ke manoranjan par fokas hogā
4. उन्होंने कहा कि आवासियों के मनोरंजन पर ध्यान केंद्रित होगा
   unhone kahā ki āvāsiyon ke manoranjan par dhyān kendrit hogā

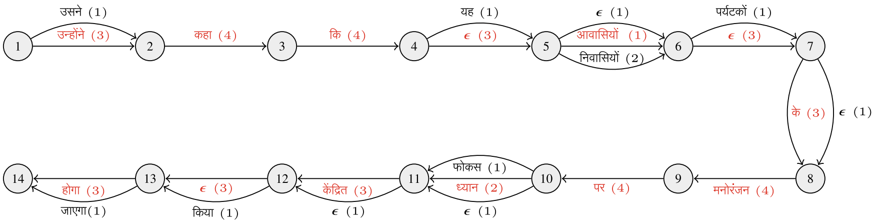

Fig. 3.1 An example of confusion network (primary hypothesis is in red color)

this, the hypotheses are arranged in a graph data structure called *confusion network*. *Confusion network* is a directed acyclic graph with the following property: any path from start to end node passes through all the nodes. Each arc label consists of a word and a confidence score. Epsilon (NULL) arcs are allowed and scored 1. Ideally, arcs from a node i to $i + 1$ should consist of words that are semantically related. In a naive model, confidence score of a word could be the number of systems that generated the word in the output. An example (Hindi) of various MT hypotheses and the corresponding confusion network generated is given in Fig. 3.1.

3.2.1 Confusion Network Generation

The method of generating a confusion network used in *Jane* [8] is briefly described here. Given the outputs of m MT systems, a primary hypothesis is chosen, and the remaining hypotheses are aligned word-to-word with the primary hypothesis using Meteor matcher. These alignments are used to generate a confusion network. Jane uses a systematic way of generating confusion network to accommodate relations for words from secondary hypotheses that are not aligned to primary hypothesis but could be potential matches with words from the other hypotheses. First, a confusion network skeleton is created with words in the primary hypothesis as arcs. Secondary hypotheses are ranked based on a language model score built on the hypotheses. Words from the best ranked hypothesis that are aligned with the words in the primary hypothesis are added to the skeleton by inserting a new arc between the corresponding nodes. Words that are not aligned are inserted by adding a new node to the previously inserted arc. The new node is joined with the next node via an epsilon arc and an arc with the unaligned word. This procedure continues until all secondary hypotheses are added to the confusion network.

m confusion networks are built with each one of m hypotheses as a primary hypothesis. These m networks are combined by linking the start nodes of all the networks to a single start node and the end nodes of all the networks to a single end node, resulting in a lattice. The final output is generated by a beam search over all possible hypotheses obtained by traversing the network. The objective function for the search is a weighted log-linear combination of various parameters like: weight for each member system, language model trained on input hypotheses, word-penalty score and score for epsilon arcs. These weights are obtained by optimizing on a held-out set, using Minimum Error Rate Training (MERT) [24].

3.2.2 Alignment of Hypotheses Using Word Embeddings

The major contribution of our work is alignment of hypotheses in confusion network using semantic similarity based on word embeddings. For this, we modify the implementation of Indic meteor provided by IIT-Bombay,[1] which uses Indo-WordNet [31] for *synonymy* module. Words that are left out after *exact*, *stem* and *synonymy* matching, are matched using word embeddings. This is done using cosine similarity. Words that have cosine similarity score above a threshold α are considered semantically similar. α is set by maximizing the correlation of the Meteor scores, with a data-set of human evaluations for *post-editability* [1]. The evaluators score from 1 to 4 based on the effort required to post-edit the MT outputs: 4 means no post-editing required, 3 means minimal post-editing, 2 means post-editing required but better than translating from scratch, 1 means it is better to translate from scratch. The data-set consists of 100 sentences from 3 MT systems: PBMT, RNN based NMT and Google translate, evaluated by professional translators. α for each word embedding is shown in Table 3.1. Pre-trained word embeddings for Indian languages provided by Kumar et al. [19] are used for all the experiments. Size of the embedding vector considered is 50. Adding word embeddings to meteor matcher (Table 3.2) has resulted in increased number of word-matches, tested on transformer MT output and reference translations.[2]

Table 3.1 Thresholds of semantic similarity, obtained by maximizing the correlation with a human evaluated data-set

Word embedding	Semantic similarity threshold (α)
Skip-gram	0.82
Cbow	0.73
FastText	0.80
GloVe	0.94

[1] https://www.cfilt.iitb.ac.in/~moses/download/meteor_indic/register.html.

[2] details of the data-set used are given in Sect. 3.4.

Table 3.2 Number of word matches by each module in Meteor when test data is aligned with reference data

Module	Matches
Exact	25,876
Stem	1278
Synonym	1602
Embeddings	5240
Total	33,996

There are 48622 tokens in test data and 51019 tokens in reference data

Table 3.3 Examples of words matched using word embeddings in Meteor, that are missed out by other modules

Reference	MT output
ऑइल	तेल
युनाइटेड	संयुक्त
चौकी	आउटपोस्ट
कनेक्टिविटी	संपर्क
स्टाफ	कर्मचारी
दुष्कर	कठिन
रक्षा	संरक्षण
विराम	गतिरोध
अथवा	या
कराने	करने
किया	किए
मैने	मैंने

Words given in Table 3.3 give an idea of potential word alignments that *exact, stem and synonym* modules fail to match, but are matched using word embeddings. These examples are taken from Meteor (word2vec) matches between TMT output and reference translations. There are Hindi equivalents of loan words/transliterated words, synonyms of words, inflected forms of same root, spelling variants, etc.

3.3 Related Literature

Ensembling using confusion network for machine translation was first proposed by Bangalore et al. [3]. The authors used Multiple String Alignment (MSA) algorithm based on Levenshtein distance between a pair of strings. Matusov et al. [22] proposed using alignment models from IBM models of SMT [6]. The set of hypotheses generated from different MT systems are used as the training data for learning alignments. If the size of the hypothesis set is m sentences and if there are n MT systems, there will be $m * (n * (n - 1)/2)$ pairs of strings for learning the alignments. The authors reported some improvements in BLEU scores for Chinese-English, Spanish-English, Japanese-English language pairs. Sim et al. [30] and Rosti et al. [27, 28] used a relatively simpler alignment method that uses edit operations from Translation Edit Rate (TER) [32] computation and obtained improved results. Rosti et al. [28] further improved consensus decoding by adding features like language model scores, number of epsilon arcs, number of words in hypothesis in a log-linear model. System combination using Jane [8] has outperformed the best single systems as well as best system combination task of WMT 2011.

Jayaraman and Lavie [12] aligned words in hypotheses by matching explicitly. The aligned hypotheses are used to generate a new set of synthetic hypotheses and ranked using confidence scores. Confusion network is not used in this method of system combination. Heafield et al. [10] enhanced this system further by introducing an alignment sensitive method for synchronizing available hypothesis extensions across the systems. They also packed similar partial hypothesis, allowing greater diversity in beam search. Banik et al. [4] follow a similar approach and score the hypotheses using various features like language model score, BLEU score, word movers distance and cosine similarity between hypotheses using word2vec. It may be noted that word2vec has been used here for scoring and not alignment. To the best of our knowledge, there have been no attempts in literature to align the hypotheses in confusion networks using word embeddings.

3.4 Set up of the Experiments

3.4.1 Data

English-Hindi parallel corpus developed and provided by Center for Indian Language Technologies (CFILT) at IIT Bombay,[3] is used for training all MT systems. Version 3.0 [20] consists of 1,609,682 sentence pairs from various domains. The development set consists of 520 sentence pairs and test set consists of 2507 sentence pairs. Hindi monolingual corpus, which is also shared by the same group, consists of 45 Million sentences and 844 Million tokens approximately.

3.4.2 Data Pre-processing

We use Moses [16] toolkit for tokenization, cleaning and true-casing for English language data. Hindi language data is tokenized using Indic NLP library.[4] Length of the sentences is limited to 80. For NMT systems, we use *byte pair encoding* (BPE)

[3] http://www.cfilt.iitb.ac.in/iitb_parallel/.

[4] https://anoopkunchukuttan.github.io/indicnlplibrary/.

[29] word segmentation with 32K merge operations. This segments the tokens into sub-words and reduces the vocabulary size by a large degree. Using BPE has been found to alleviate out-of-vocabulary problem, which was a major bottleneck in NMT.

3.4.3 Training of MT Systems

We use Moses [16] for building PBMT and HPBMT systems. Word alignments are trained using `mgiza`. After training, alignments are symmetrized with `-grow-diag-final-and` heuristic. Reordering model is built with `msd-bidirectional` option. A 5-gram language model with Kneser-Ney smoothing is built with `lmplz` (kenlm) which comes along with Moses. Tuning is performed using MERT. HPBMT is built with default options in Moses.

We use OpenNMT [14] for building BRNN and TMT systems. BRNN system is trained using LSTM based bi-directional RNNs with global attention, with 4 encoding and 4 decoding layers. TMT system is trained using transformers with 6 encoding layers and 6 decoding layers, with 8 attention heads. Adam [13] optimizer is used in both NMT systems. The choice of hyper-parameters is based on configurations of various state-of-the-art NMT systems implemented in Workshop on Asian Translation (WAT) [9] and our own experimental observations. Both the NMT systems are trained on NVIDIA GeForce RTX 2080 GPUs with 8GB memory.

3.5 Experiments and Results

We train four MT systems namely PBMT, HPBMT, BRNN and TMT. Five system combinations are performed in total: baseline system with default Meteor alignments (*baseline*), alignment with word2vec skip-gram (*sg*), alignment with word2vec CBOW (*cbow*), alignment with fastText (*fastText*) and alignment with GloVe (*GloVe*). BLEU and RIBES [11] evaluation scores are reported in Table 3.4. Bold entries indicate improvement in BLEU and Ribes scores over best individual system (TMT).

System combination using word embeddings has shown improvement in performance in all cases except *GloVe*. System combination *baseline* has shown marginally poor performance in comparison with best individual system (TMT). To check whether the difference between the BLEU scores is statistically significant, we perform *students t-test* by *bootstrap resampling* [15] for each pair of MT systems. The null hypothesis is that the outputs are from the same system. We reject the null hypothesis at $p < 0.05$. The p-values are reported in Table 3.5.

From Table 3.5, it is evident that out of the four system combinations, three models: *sg*, *cbow* and *fastText* show a significant improvement in BLEU scores, when compared with the best individual system (TMT) and the *baseline* combination system. Improvement in performance of *GloVe* against *baseline* is marginal but not statistically significant. The decrease in performance with respect to *TMT* is also not

Table 3.4 Performance of individual MT systems and their combinations

MT system	BLEU	RIBES
PBMT	12.06	0.652
HPBMT	13.28	0.655
BRNN	13.31	0.715
TMT	18.66	0.735
baseline	18.52	0.724
sg	**18.96**	0.731
cbow	**19.00**	**0.735**
fastText	**18.99**	0.731
GloVe	18.53	0.729

Table 3.5 $p-$values for difference between BLEU scores for each pair of MT systems; $p < 0.05$ are shown in blue color (reject null hypothesis) and others are shown in red

-	PBMT	HPBMT	BRNN	TMT	baseline	sg	cbow	fastText	GloVe
PBMT	1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
HPBMT		1.000	0.361	0.000	0.000	0.000	0.000	0.000	0.000
BRNN			1.000	0.000	0.000	0.000	0.000	0.000	0.000
TMT				1.000	0.183	0.013	0.006	0.010	0.155
baseline					1.000	0.000	0.000	0.000	0.419
sg						1.000	0.237	0.287	0.000
cbow							1.000	0.332	0.000
fastText								1.000	0.000
GloVe									1.000

statistically significant. The difference between *GloVe* and other combinations based on word embeddings, is statistically significant. There is no statistically significant difference between the following pairs: TMT-*baseline*, *sg-cbow, sg-fastText, cbow-fastText* and HPBMT-BRNN.

3.6 Conclusion

In this work, we have used word embeddings for aligning hypotheses in confusion-network based system combination. An open source statistical MT toolkit *Jane*, which uses *Meteor* matcher for confusion network generation is used for our experiments. Meteor is appended with a module for matching words using word embeddings. Cosine similarity between the vectors is used for aligning semantically similar words. Four English-Hindi MT systems PBMT, HPBMT, BRNN and TMT are trained on IIT-Bombay English-Hindi parallel corpus. Outputs of these four systems are combined in four different settings using word-embeddings: *word2vec skip-gram (sg)*, *word2vec cbow*, *fastText* and *GloVe*. A system combination *baseline* is built with default Meteor setting without word embeddings. Three systems (*sg*, *cbow* and *fastText*) out of four combination systems have shown statistically significant improvement in BLEU scores, compared to *baseline* and the best performing individual system (TMT). *GloVe* shows a marginal improvement in BLEU score compared to the baseline, but it is not statistically significant. Thus, we see that using word embeddings for alignment in confusion network decoding holds promise.

References

1. Anirudh, C.R., Murthy, K.N.: On post-editability of machine translated texts. Transl. Today (2021). (in press)
2. Banerjee, S., Lavie, A.: Meteor: an automatic metric for mt evaluation with improved correlation with human judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. vol. 29, pp. 65–72. Association for Computational Linguistics, University of Michigan, Ann Arbor (29 June 2005)
3. Bangalore, S., Bordel, G., Riccardi, G.: Computing consensus translation from multiple machine translation systems. In: IEEE Workshop on Automatic Speech Recognition and Understanding, 2001. ASRU'01. pp. 351–354. IEEE (2001)
4. Banik, D., Ekbal, A., Bhattacharyya, P., Bhattacharyya, S., Platos, J.: Statistical-based system combination approach to gain advantages over different machine translation systems. Heliyon 5(9), e02504 (2019). https://doi.org/10.1016/j.heliyon.2019.e02504
5. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606 (2016)
6. Brown, P.F., Pietra, V.J.D., Pietra, S.A.D., Mercer, R.L.: The mathematics of statistical machine translation: parameter estimation. Comput. Linguist. **19**(2), 263–311 (1993)
7. Chiang, D.: Hierarchical phrase-based translation. Comput. Linguist. **33**(2), 201–228 (2007). https://doi.org/10.1162/coli.2007.33.2.201
8. Freitag, M., Huck, M., Ney, H.: Jane: Open source machine translation system combination. In: Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics. pp. 29–32. Association for Computational Linguistics, Gothenburg, Sweden (Apr 2014). https://doi.org/10.3115/v1/E14-2008
9. Goyal, V., Sharma, D.M.: LTRC-MT simple & effective Hindi-English neural machine translation systems at WAT 2019. In: Proceedings of the 6th Workshop on Asian Translation. pp. 137–140. Association for Computational Linguistics, Hong Kong, China (Nov 2019). https://doi.org/10.18653/v1/D19-5216
10. Heafield, K., Hanneman, G., Lavie, A.: Machine translation system combination with flexible word ordering. In: Proceedings of the Fourth Workshop on Statistical Machine Translation, pp. 56–60 (2009)
11. Isozaki, H., Hirao, T., Duh, K., Sudoh, K., Tsukada, H.: Automatic evaluation of translation quality for distant language pairs. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 944–952 (2010)
12. Jayaraman, S., Lavie, A.: Multi-engine machine translation guided by explicit word matching. In: Proceedings of the ACL Interactive Poster and Demonstration Sessions, pp. 101–104 (2005)
13. Kingma, D. P., Ba, J.: Adam: A method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)

14. Klein, G., Kim, Y., Deng, Y., Nguyen, V., Senellart, J., Rush, A.: OpenNMT: Neural machine translation toolkit. In: Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers). pp. 177–184. Association for Machine Translation in the Americas, Boston, MA (Mar 2018)
15. Koehn, P.: Statistical significance tests for machine translation evaluation. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pp. 388–395 (2004)
16. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al.: Moses: open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, pp. 177–180. Association for Computational Linguistics, Prague, Czech Republic (June 2007)
17. Koehn, P., Knowles, R.: Six challenges for neural machine translation. In: Proceedings of the First Workshop on Neural Machine Translation. pp. 28–39. Association for Computational Linguistics, Vancouver (Aug 2017). https://doi.org/10.18653/v1/W17-3204
18. Koehn, P., Och, F.J., Marcu, D.: Statistical phrase-based translation. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, vol. 1. pp. 48–54. Association for Computational Linguistics, Edmonton (2003)
19. Kumar, S., Kumar, S., Kanojia, D., Bhattacharyya, P.: "A passage to India": Pre-trained word embeddings for Indian languages. In: Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL), pp. 352–357. European Language Resources association, Marseille, France (May 2020)
20. Kunchukuttan, A., Mehta, P., Bhattacharyya, P.: The iit bombay english-hindi parallel corpus. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018), pp. 3473–3476. European Language Resources Association (ELRA), Miyazaki, Japan (May 2018)
21. Luong, T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421. Association for Computational Linguistics, Lisbon, Portugal (Sep 2015). https://doi.org/10.18653/v1/D15-1166
22. Matusov, E., Ueffing, N., Ney, H.: Computing consensus translation for multiple machine translation systems using enhanced hypothesis alignment. In: 11th Conference of the European Chapter of the Association for Computational Linguistics (2006)
23. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 26, pp. 3111–3119. Curran Associates, Inc. (2013)
24. Och, F.J.: Minimum error rate training in statistical machine translation. In: Proceedings of the 41st annual meeting of the Association for Computational Linguistics, pp. 160–167 (2003)
25. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 311–318. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA (Jul 2002). https://doi.org/10.3115/1073083.1073135
26. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
27. Rosti, A.V., Ayan, N.F., Xiang, B., Matsoukas, S., Schwartz, R., Dorr, B.: Combining outputs from multiple machine translation systems. In: Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, pp. 228–235 (2007)
28. Rosti, A.V., Matsoukas, S., Schwartz, R.: Improved word-level system combination for machine translation. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 312–319 (2007)

29. Sennrich, R., Haddow, B., Birch, A.: Improving neural machine translation models with monolingual data. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 86–96. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/P16-1009
30. Sim, K.C., Byrne, W.J., Gales, M.J., Sahbi, H., Woodland, P.C.: Consensus network decoding for statistical machine translation system combination. In: 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07. vol. 4, pp. IV–105, IEEE (2007)
31. Sinha, M., Reddy, M., Bhattacharyya, P.: An approach towards construction and application of multilingual indo-wordnet. In: 3rd Global Wordnet Conference (GWC 06). Jeju Island, Korea (2006)
32. Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J.: A study of translation edit rate with targeted human annotation. In: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, pp. 223–231. The Association for Machine Translation in the Americas, Cambridge (2006)
33. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: NIPS. pp. 6000–6010 (2017)