Sādhanā (2020)45:191
https://doi.org/10.1007/s12046-020-01427-w

Statistical machine translation based on weighted syntax–semantics

DEBAJYOTY BANIK[1,2,*], ASIF EKBAL[1] and PUSHPAK BHATTACHARYYA[1]

[1]Department of Computer Science and Engineering, Indian Institute of Technology Patna, Patna, India
[2]School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar, India
e-mail: debajyoty.banik@gmail.com

MS received 13 June 2018; revised 16 February 2020; accepted 2 April 2020

Abstract. In this paper, we propose an alternate technique to improve the performance of the statistical machine translation (SMT) system. Here, the phrases are re-weighted in light of linguistic knowledge as both syntactic and semantic information. Syntactic knowledge helps to increase fluency whereas semantic similarity helps to incorporate semantic meaning, which is required for adequacy of translated sentences. The scores of the phrases from the phrase-table are re-balanced by expanding and diminishing the weights of the correct phrases and the incorrect phrases, respectively. Additional knowledge in phrase-table helps in improving overall performance of translation quality. In this work, our proposed methodology achieves an impressive accuracy improvement in terms of BLEU, NIST and RIBES in different domain data. We achieve 58.54 BLEU points, 0.7759 RIBES points and 9.684 NIST points for product domain catalog.

Keywords. Natural language processing; machine translation; phrase-based; parsing.

1. Introduction

Machine translation (MT) is an interesting field of the computer science that started in the seasons of the second world war. Because of the advent of tourism, globalization, education and administration, MT gains its attention. The sheer volume of elucidation has made automation unavoidable. One explanation for the tremendous improvement of statistical machine translation (SMT) system is the disclosure of the word alignment algorithm based on expectation maximization. SMT system has an exceptionally principled path in the light of the likelihood of recognizing the probable word correspondences between parallel sentences. It is a well-established tool in natural language processing (NLP) and artificial intelligence (AI), which is getting more attention by giving better qualitative output [1, 2]. In SMT system [3], the phrase-based approach can be used to improve the quality of the translated output. Alignment template model is inherited for phrase-based MT in [4]. Hardcore statistical approach for translation without having linguistic information has a bottleneck in meeting our expectation in terms of accuracy. Thus, syntax-based MT system is discussed in [5] and phrase-based translation based on joint probability is discussed in [6]. The CMU and IBM word-based SMT systems are inflamed with phrase translation capability. After the argmax computation, preferences of a phrase are selected from the traditional phrase-table depending on its occurrences in the corpora. However, it cannot achieve better accuracy due to ignorance of linguistic properties. In this work, we try to incorporate syntactic and semantic information into the phrase-table. The fluency can be improved by incorporating syntactic information. Therefore, we incorporate both syntactic and semantic information to improve both fluency and adequacy. We incorporate dictionary and word net in SMT system to unite this point with syntactic information.

A remarkable pitfall of phrase-based statistical machine translation (PBSMT) system [3] is the equal probability distribution among wrongly aligned phrases and correctly aligned phrases. However, it is proved that pruning out linguistically incorrect phrases from the phrase-table cannot improve the system's performance [7]. It degrades the translation quality. Thus, we propose an alternative model where we need to pay more attention to linguistically correct phrases without any information loss or without pruning out linguistically incorrect phrases. According to this phenomenon , we modify various probabilistic scores (i.e., elf, fle). Our research infers that a remarkable improvement can be found with simple prioritized means. With this proposed method, the linguistically correct phrases get more priority than linguistically incorrect phrases. This priority helps to increase direct phrase translation probability and inverse phrase translation probability for linguistically correct phrases and decrease the probability scores for linguistically incorrect phrases. The target of our work is to make a better phrase-table from the existing one. A phrase-table consists of various

*For correspondence
Published online: 30 July 2020

probabilistic scores. Hence, modifying the scores by adding or subtracting with a constant is not a better option; it can violate the principle of the statistical and probabilistic approach. Thus, scores are modified by depending on the correctness of the phrases (as per linguistic knowledge) in the phrase-table. Joint probability helps in calculating the scores that need to be added or subtracted from original phrase-table. The proposed architecture is a generic model for phrase-table modification to have better translation quality. There are some existing approaches for incorporating syntactic information into SMT system. For instance [8, 9] focused on incorporating syntactic information into the hierarchical phrase-based translation.

This paper is organized as follows. Related works are described in Section 3. Correct/incorrect phrase detection is described in Section 4. Architecture for phrase-table readjustment with syntactic and semantic knowledge is discussed in Section 5. Section 6 presents a discussion on datasets, experimental set-up, results and analysis. Finally, we conclude our paper in Section 7.

2. Motivation

The phrase-table is a compelling sub-module in SMT system. The accuracy of the translated output depends mainly on the quality of phrase-table. Hence, qualitative phrase-table preparation is the the most important task in SMT. Here, we propose an interesting approach to improve quality of the phrase-table to achieve better accuracy in translated sentences. Traditional PBSMT does not have any linguistic knowledge. Therefore, it is tough to improve the accuracy of the translated output after a certain limit.

The statistical approach assumes the same priority for all phrases (irrespective of the correctness of the phrases) in phrase-table for translation, which is the primary problem for accuracy improvement. Though the hierarchical MT system considers sentence structures, they are not linguistic structure [10]. Even syntax-based MT uses partially linguistic phrases information, but neither considers semantic phenomenon nor provides better weight for the linguistically (syntactically) correct phrases [5]. Considering the following examples of syntax-based MT, both expression (a) and expression (b) tuples should not have equal priority because expression (b) is linguistically correct whereas expression (a) is linguistically incorrect. Hence, expression [b] should have better priority. Similarly, semantically correct phrase pairs should have better priority/probability than the semantically incorrect phrase pairs.

Doing work [.]|||काम हूँ[1] (a)
Doing work [VP]|||काम कर रहा हूँ[2] (b)

[1]HT: kaam hun.

[2]HT: kaam kar raha hun.

Traditional phrase-based SMT system does not have any linguistic knowledge. Thus, various probability scores are uniformly distributed over linguistically correct phrases and linguistically incorrect phrases. It is also assumed that all phrases have the same probability distribution depending upon appearing data in the parallel corpus. Phrase-table structure of phrase-based machine translation (PBSMT) system is as follows:

sun ||| सूरज[3]||| 0.0819672 0.0230415 0.00086103 0.0010026 ||| 0-0 ||| 61 5807 5 ||| |||.

The phrase-table consists of a maximum of seven columns: Source phrase, Target phrase, Scores, Alignment, Counts, Sparse feature scores and Key-value properties. There are four different phrase translation scores inside this: inverse phrase translation probability $\phi(f|e)$, inverse lexical weighting $lex(f|e)$, direct phrase translation probability $\phi(e|f)$ and direct lexical weighting $lex(e|f)$.

Due to information loss, accuracy of translated output could not be improved after pruning out linguistically incorrect phrases from phrase-table. Thus, we proposed a novel model where we used all of the information of linguistically correct phrases and linguistically incorrect phrases with different priorities. Depending upon the priorities, probabilistic values have been updated. We assume that the linguistically correct phrases have better priorities than the linguistically incorrect phrases. Based on the priority, inverse phrase translation probabilities and direct phrase translation probabilities are improved with some weights. Finally, obtained weighted phrase-table is used to decode the source sentences into translated sentences.

Here, parser takes an important role in our work. Hence, its performance is crucial. Due to parser inefficiency we may not detect correct/incorrect phrase for some cases. The examples given here are c and d. This is the limitation of the proposed work. In example d, the parser detects the phrase as NNP but it should be VP:

! [.] ||| अच्छा है[4] [X] (c)
Good [NNP] ||| अच्छा[5] [X] (d)

3. Related works

In [11], researchers addressed the problem of syntax-based MT-like mistranslated predicate–argument structures. For translating ambiguous predicates and arguments, knowledge about semantic relation among target predicates and their argument fillers is useful. Authors in [12] proposed string-to-tree translation system using noun class information to model selection preferences of prepositions. To

[3]HT: suraj.

[4]HT: achhya hun.

[5]HT: achhya.

restrict their applicability to specific semantic classes, they used the noun class information to annotate PP translation rules.

In [13], parse-tree-based alignment was incorporated to permit alignment of non-constituent sub-phrases on the source side; later a separate phrase-based model translate modifier was used. To improve phrasal coverage, syntax-based extraction and binarized syntax trees [14] are modified in [15]. To avoid loss of non-syntactic phrase pair, syntax-based system has been demonstrated after re-writing target side parse trees in [16]. Instead of the system based on linguistically motivated parse trees, authors introduced a model in [9] that was based on syntactic nature in formal sense. The good quality language model can increase fluency of translated output. Language model based on syntactic information is discussed in [17]. In [18], shallow phrases (chunks) are added into the translation model of PBSMT system for English–Bangla MT. Recently, researchers focused to improve PBSMT using decoder's parameters tuning [19], data augmentation into PBSMT system [20], hybridization [21] and system combination [22].

In [23] researchers applied coarse-grained features and fine-grained features to tune the development model based on syntactic information, which requires lots of iteration to converge. Moreover, nonterminal labels have been selected that appear more than 100 times. Hence some information is lost, whereas our approach is quite simple and straight forward. Instead of pruning out any information, it updates the weight for all phrases dynamically and is not an iterative process. Thus, the proposed approach is more efficient. Our proposed method uses traditional SMT system. The obtained phrase-table is the main ingredient of our approach. Based on linguistics information of every phrase, probabilistic scores are modified. It does not need to train the model again. Thus, it is easy to incorporate our algorithm to any existing SMT system like PBSMT system, hierarchical SMT system, etc. We are trying to cover maximum linguistic phrases present in phrase-table to update their weight.

Very few works have been done in SMT model based on semantic information in the literature. Some research is done in [24, 25]. Using our proposed algorithm, we have incorporated syntactic as well as semantic knowledge from synsets in SMT system. The syntactic and semantic information is responsible for adequacy and fluency.

4. Correct phrase detection

Correctness of phrases could be defined in two ways: syntactic correctness and semantic correctness. Any phrase that is correct in linguistic structure is known as syntactically correct. Any aligned phrases pair that preserves similar meaning is known as semantically correct aligned phrases. We have considered both of the types one after another to incorporate maximum linguistic knowledge. Syntactically incorrect phrases cannot follow proper syntax according to the linguistic phenomenon of that language. Thus it may not have proper parts of speech (PoS). For example, in English, "is reading" is a syntactically correct phrase whereas "read am" is a syntactically incorrect phrase. In Hindi, "पढ़ रहा हूँ" (HT: pad raha hun) is a syntactically correct phrase whereas "पढ़ हूँ" (HT: pad hun) is syntactically incorrect.

To detect syntactically correct and incorrect phrases, we took help from PoS information. We use the shallow parser[6] available for the Indian languages and the stanford parser[7] available for English to extract PoS.

Sentence: "flat sandals with buckle strap and toe separator"

Parsed tree:

```
(ROOT
    (NP
      (NP (JJ flat) (NNS sandals))
      (PP (IN with)
        (NP (JJ buckle) (NN strap)
          (CC and)
              (NN toe) (NN separator))))).
```

From this information, we tried to retrieve all possible phrases with proper tagging following a bottom up approach. If a valid sub-tree is found at the tree for that particular string, then the phrase will be considered as a correct syntactic phrase. Hence, syntactically correct phrases for this sentence are the following: separator, toe, flat, sandals, buckle, and, with, flat, strap, sandals, buckle strap, toe separator, with buckle strap, with toe separator, flat sandals with buckle strap and toe separator. To achieve better accuracy, the parser could be trained with domain-specific data. To identify all correct/incorrect phrases, we used the same procedure for both languages. This helped to detect syntactic correctness of phrases. Error of the parse tree has propagated to our system, which affects the translation quality. Hence, the proposed system can be improved by updating the parse tree. There are also cases where translation quality suffers as the source and target languages differ in their syntactic structures. English follows subject–verb–object (SVO), whereas Hindi follows subject–object–verb (SOV) notation. Initially we reordered the source side at sentence level with linguistic rules, according to target order. Then we use weight re-adjustment process to update the weights of correct and incorrect phrases. Researchers [26, 27] have shown that the source-

[6] http://ltrc.iiit.ac.in/showfile.php?filename=downloads/shallow_parser.php.

[7] http://nlp.stanford.edu:8080/parser/.

side syntactic reordering to conform to the syntax of the target-side mitigates structural divergence and improves quality of the translation significantly.

On the other hand, lexicon for English–Hindi bilingual mapping[8] and IndoWordnet [28] are used to gather semantic similarity between phrase pairs.

In our algorithm, we assume that if similarity is found between any word pair inside a phrase pair, then that phrase pair will be considered to be semantically correct phrase. For example, there are two phrase pairs, first "is playing" and "खेल रहे" (HT: khel rahe); second "is playing", "जा रहे हैं" (HT: ja rahe hun). Here, all phrases are syntactically correct but the second pair is not a semantically correct pair because it does not preserve the meaning for any of the word pairs. Along with the first phrase pair ("is playing" and "खेल रहे" (HT: khel rahe)), every synonym of any phase of this pair will be considered as semantically correct.

Using the proposed methodology, we have measured semantic similarity for phrase pair. In phrase pairs, if there is any closeness in semantic similarity in any word pair then it will be considered as a semantically correct phrase. Hence, in the first example, "playing" in source phrase is similar to "खेल" (HT: khel) in target phrase, so this phrase pair ("is playing" and "खेल रहे" (HT: khel rahe);) is semantically correct phrase. In the second phrase pair, if we traverse all words in the target phrase for every word in source phrase then we cannot find any similarity. Thus, the second phrase pair ("is playing" and "जा रहे हैं" (HT: ja rahe hun)) is semantically incorrect.

For a phrase in a phrase-table, even if it does not appear as a constituent in a parse tree, it can still have a good translation counterpart, which might be useful for the SMT system. Hence, instead of pruning out the incorrect phrases, we update the weight as per nature of the phrases.

5. Weight re-adjustment of phrase-table using proposed approach based on weighted syntax–semantics

The basic phrase-table is found using standard PBSMT system. Based on linguistically (syntactically) correct/incorrect phrases, the basic phrase-table is divided into four sets (A, B, C, D). Set A consists of both linguistically correct source phrases ($\bar{f}$) and target phrases ($\bar{e}$). Set B consists of linguistically correct source phrases and linguistically incorrect target phrases. Set C consists of linguistically incorrect source phrases and linguistically correct target phrases. Set D consists of linguistically incorrect source phrases and linguistically incorrect target phrases.

[8] http://www.cfilt.iitb.ac.in/~sudha/bilingual_mapping.tar.gz.

5.1 *Frequently used mathematical notations*

The basic mathematical notation for proposed algorithms is described and shown here:

$\bar{e_A}$ = Target phrase in set A
$\bar{e_B}$ = Target phrase in set B
$\bar{e_C}$ = Target phrase in set C
$\bar{e_D}$ = Target phrase in set D
$\bar{f_A}$ = Source phrase in set A
$\bar{f_B}$ = Source phrase in set B
$\bar{f_C}$ = Source phrase in set C
$\bar{f_D}$ = Source phrase in set D
$|\bar{f_A}|$ = Number of source phrases (f) in set A
$|\bar{e_A}|$ = Number of target phrases (e) in set A
$|\bar{f_B}|$ = Number of source phrases (f) in set B
$|\bar{e_B}|$ = Number of target phrases (e) in set B
$|\bar{f_C}|$ = Number of source phrases (f) in set C
$|\bar{e_C}|$ = Number of target phrases (e) in set C
$|\bar{f_D}|$ = Number of source phrases (f) in set D
$|\bar{e_D}|$ = Number of target phrases (e) in set D
$\varphi(e|f)_A$ = Direct phrase translation probability $\varphi(e|f)$ in A
$\varphi(f|e)_A$ = Inverse phrase translation probability $\varphi(f|e)$ in A
$\varphi(e|f)_B$ = Direct phrase translation probability $\varphi(e|f)$ in B
$\varphi(f|e)_B$ = Inverse phrase translation probability $\varphi(f|e)$ in B
$\varphi(e|f)_C$ = Direct phrase translation probability $\varphi(e|f)$ in C
$\varphi(f|e)_C$ = Inverse phrase translation probability $\varphi(f|e)$ in C
$\varphi(e|f)_D$ = Direct phrase translation probability $\varphi(e|f)$ in D
$\varphi(f|e)_D$ = Inverse phrase translation probability $\varphi(f|e)$ in D
Δe_f = Improvement in score of direct phrase translation probability in set A
$\Delta \bar{e}_f$ = Decrease in score for direct phrase translation probability in B
Δf_e = Improvement in score of inverse phrase translation probability in set A
$\Delta \bar{f}_e$ = Decrease in score for inverse phrase translation probability in C
$\Delta f_{\bar{e}}$ = Improvement in score of inverse phrase translation probability in set B
$\Delta \bar{f}_{\bar{e}}$ = Decrease in score of inverse phrase translation probability in set D
$\Delta e_{\bar{f}}$ = Improvement in score of direct phrase translation probability in set C
$\Delta \bar{e}_{\bar{f}}$ = Decrease in score of direct phrase translation probability in set D

5.2 *Proposed approach for phrase-table update*

To improve the phrase-table quality, we distributed joint probability scores among different phrase translation

probabilities in an interesting fashion. This joint probability is added to phrase translation probabilities for correct linguistic phrase. On the other hand, joint probability is subtracted from phrase translation probabilities for incorrect linguistic phrase. The basic intuition behind weight re-adjustment is sentence formation with linguistically correct phrases, which can be better than linguistically incorrect phrases during translation. Hence, by following some constraint, we have increased the direct phrase translation probability $\varphi(e|f)$ and/or inverse phrase translation probability $\varphi(f|e)$ if it is linguistically correct phrase and decreased the scores for linguistically incorrect phrases. To accomplish this task, the statistical concept of joint probability is used. The joint probability model can be easily marginalized in order to yield conditional probability models for both target-from-source and source-from-target alignments. If there is any syntactically correct source phrase that is aligned with syntactically correct target phrases (in set A) and when that very source phrase is aligned with syntactically incorrect target phrases (in set B), then $\varphi(e|f)_A$ (i.e., (elf) of aligned syntactically correct target phrases) will be added to Δe_f and $\Delta\bar{e}_f$ will be subtracted from $\varphi(e|f)_B$ (i.e., (elf) of aligned syntactically incorrect target phrases), respectively.

$$\Delta e_f = \frac{\prod^{f_A}\prod^{f_B}\varphi(e|f)}{|f_A|}, \tag{1}$$

$$\Delta\bar{e}_f = \frac{\prod^{f_A}\prod^{f_B}\varphi(e|f)}{|f_B|}. \tag{2}$$

When there is any syntactically correct target phrase that is aligned with syntactically correct source phrases (in set A) and then that very target phrase is aligned with syntactically incorrect source phrases (in set C), $\varphi(f|e)_A$ (i.e., (fle) of aligned syntactically correct source phrases) will be added to Δf_e and $\Delta\bar{f}_e$ will be subtracted from $\varphi(f|e)_C$ (i.e., (flc) of aligned syntactically incorrect source phrases), respectively.

$$\Delta f_e = \frac{\prod^{e_A}\prod^{e_C}\varphi(f|e)}{|e_A|}, \tag{3}$$

$$\Delta\bar{f}_e = \frac{\prod^{e_A}\prod^{e_C}\varphi(f|e)}{|e_C|}. \tag{4}$$

Whenever there is any syntactically incorrect target phrase that is aligned with syntactically correct source phrases (in set B) and when that very target phrase is aligned with syntactically incorrect source phrases (in set D), $\varphi(f|e)_B$ (i.e., (fle) of aligned syntactically correct source phrases) will be added to $\Delta f_{\bar{e}}$ and $\Delta\bar{f}_{\bar{e}}$ will be subtracted from $\varphi(f|e)_B$ (i.e., (fle) of aligned syntactically correct source phrases), respectively.

$$\Delta f_{\bar{e}} = \frac{\prod^{e_B}\prod^{e_D}\varphi(f|e)}{|e_B|}, \tag{5}$$

$$\Delta\bar{f}_{\bar{e}} = \frac{\prod^{e_B}\prod^{e_D}\varphi(f|e)}{|e_D|}. \tag{6}$$

If there is any syntactically incorrect source phrase that is aligned with syntactically correct target phrases (in set C) and when that very source phrase is aligned with incorrect target phrases (in set D), then $\varphi(e|f)_C$ (i.e., (elf) of aligned syntactically correct target phrases) will be added to $\Delta e_{\bar{f}}$ and $\Delta\bar{e}_{\bar{f}}$ will be subtracted from $\varphi(e|f)_D$ (i.e., (elf) of aligned syntactically incorrect target phrases), respectively.

$$\Delta e_{\bar{f}} = \frac{\prod^{f_C}\prod^{f_D}\varphi(e|f)}{|f_C|}, \tag{7}$$

$$\Delta\bar{e}_{\bar{f}} = \frac{\prod^{f_C}\prod^{f_D}\varphi(e|f)}{|f_D|}. \tag{8}$$

The changing of probabilities depends on the joint probability model. The total amounts of increased probability score and decreased probability score for a specific source phrase are equal; thus total probability score for every phrase remains one. Hence, $\sum\varphi(e|f)_{\bar{f}_i} = 1$ where $\varphi(e|f)_{\bar{f}_i}$ denotes direct phrase translation probability for i^{th} source phrase and $\sum\varphi(f|e)_{\bar{f}_i} = 1$ where $\varphi(f|e)_{\bar{f}_i}$ denotes inverse phrase translation probability for i^{th} source phrase. The detailed algorithm for phrase-table update is shown in Algorithm 1.

ALGORITHM 1: phrase-table update based on syntactic information (Phase 1)

```
Input: phrase-table;
Output: modified phrase-table;
Supporting resources: standford parser and
shallow parser;
  for each f in A
    if f_A ∈ f_B
      Update φ(e|f)_A by +Δe_f
      Update φ(e|f)_B by -Δē_f
    else if e_A ∈ e_C
      Update φ(f|e)_A by +Δf_e
      Update φ(f|e)_C by -Δf̄_e
  for each e in B
    if e_B ∈ e_D
      Update φ(f|e)_B by +Δf_ē
      Update φ(f|e)_D by -Δf̄_ē
  for each f in C
    if f_C ∈ f_D
      Update φ(e|f)_C by +Δe_f̄
      Update φ(e|f)_D by -Δē_f̄
```

After this procedure, we will get modified phrase-table with syntactic knowledge. To achieve complete

linguistic knowledge, we have to consider semantic similarity also. Algorithm 2 fulfills this crisis by introducing semantic similarity at phrase pair of phrase-table. Generated phrase-table of Algorithm 1 is considered as an ingredient of Algorithm 2. Prepared compact phrase-table after this model is used for decoding to have better quality translated output. Using Algorithm 3 we can gather semantical similarity between phrase pairs, which is described in Section 4. Semantic similarity in phrase level will help Algorithm 2 to proceed towards its goal. According to this model, for any target phrase, if phrase pair is semantically similar, then (fle) will be added to $\Delta Semantic_{(f|e)}$. $\Delta Semantic_{(f|e)}$ is the normalized product of (fle) for that very target phrase.

$$\Delta Semantic_{(f|e)} = \frac{\prod_{j=1}^{n} \varphi((f_i|e_j))}{|Sem|}. \tag{9}$$

For any source phrase, if phrase pair is semantically similar then (elf) will be added to $\Delta Semantic_{(e|f)}$. $\Delta Semantic_{(e|f)}$ is the normalized product of all (elf) for that very source phrase:

$$\Delta Semantic_{(e|f)} = \prod_{j=1}^{n} \frac{\varphi((e_j|f_i))}{|Sem|}. \tag{10}$$

On the other hand, for any target phrase, if phrase pair is not semantically similar, then $\Delta nonSemantic_{(f|e)}$ will be decreased from (fle). $\Delta nonSemantic_{(f|e)}$ is the normalized product of all (fle) for that very target phrase:

$$\Delta nonSemantic_{(f|e)} = \frac{\prod_{j=1}^{n} \varphi((f_i|e_j))}{|nonSem|}. \tag{11}$$

For a certain source phrase if phrase pair is not semantically similar, then $\Delta nonSemantic_{(e|f)}$ will be decreased from (elf). $\Delta nonSemantic_{(e|f)}$ is the normalized product of all (elf) for that very source phrase:

$$\Delta nonSemantic_{(e|f)} = \frac{\prod_{j=1}^{n} \varphi((e_j|f_i))}{|nonSem|} \tag{12}$$

where $|nonSem|$ and $|Sem|$ denote number of semantically incorrect phrases and number of semantically correct phrases, respectively, for specific source or target phrase for which phrase semantic similarity will be measured.

ALGORITHM 2: phrase-table update based on lexicon knowledge (Phase 2).

Input: Modified phrase-table;
Output: Updated phrase-table;
Supporting resource: Lexicon;
$\bar{f}_i \rightarrow \bar{t_1} \,\|\, \bar{t_2} \,\|\, ... \,\|\, \bar{t_n}$
$T := \{T_i | i = 1\,(1)\,n\}$
$|Sem| \leftarrow$ Number of linguistic phrases;
$|nonSem| \leftarrow$ Number of non-linguistic phrases;
for each $\bar{f}_i$ in modified phrase-table
  $product_{(e|f)} \leftarrow \prod_{j=1}^{n} \varphi((e_j|f_i))$
  $product_{(f|e)} \leftarrow \prod_{j=1}^{n} \varphi((f_i|e_j))$
for each $\bar{e}_j$ in T
  if (isSemantic ($\bar{e}_j$ *and* $\bar{f}_i$))
    Sem ++;
  else
    nonSem ++;
for each $\bar{e}_j$ in T
  if isSemantic ($\bar{e}_j$ *and* $\bar{f}_i$)
    Update $\varphi(e_j|f_i)$ by + $\frac{product_{(e|f)}}{|Sem|}$
    $Update \varphi(f_i|e_j)$ by + $\frac{product_{(f|e)}}{|Sem|}$
  else
    Update $\varphi\,(e_j|f_i)$ by - $\frac{product_{(e|f)}}{|nonSem|}$
    Update $\varphi\,(f_i|e_j)$ by - $\frac{product_{(f|e)}}{|nonSem|}$

ALGORITHM 3: isSemantic $(\bar{e}, \bar{f})$: to cheek at least one word from phrase pair is linguistically correct or not

Input: Phrase pair;
Output: linguistically correct or not;
Supporting resource: Lexicon;
for each $\bar{f}_i$ *in* $\bar{f}$
  for each $\bar{e}_j$ in $\bar{e}$
    if $\bar{f}_i$ *and* $\bar{e}_j$ are present in lexicon as semantic pair
      return true and break;
    else
      continue;
return false if not match any sub pair;

Lemma 5.1 *The summation of total probability for direct phrase translation ($\varphi(e|f)$) is equal to total probability for inverse phrase translation scores $\varphi(f|e)$ after processing of the phrase-table.*

It is shown that the total probability scores for any phrase remains one after the first phase of operation. The preliminary phrase-table is formed using statistical translation model that follows total probability theorem, i.e. $\sum_{i=1}^{n} \varphi(e|f) = 1$ and $\sum_{i=1}^{n} \varphi(f|e) = 1$ where n is total number of translated phrases $(\bar{e_1}, \bar{e_2}, \ldots, \bar{e_n})$ for a specific source phrase $(\bar{f})$.

After processing of the phrase-table with the first phase of operation, the total probability scores will remain the same because total increased probability score $(\sum \Delta(e|f)_{increase})$ is equal to total decreased direct phrase translation probability $(\sum \Delta(e|f)_{decrease})$ score:

$$\sum \Delta(e|f)_{increase} = \sum \Delta(e|f)_{decrease}$$

where $\Delta(e|f)_{increase}$ and $\Delta(e|f)_{decrease}$ are increased direct phrase translation probability and decreased direct phrase translation probability, respectively, for any source phrase $\bar{f_i}$.

$$\Delta(e|f)_{increase} = \frac{\prod^{f_A} \prod^{f_B} \varphi(e|f)}{|f_A|},$$
$$\Delta(e|f)_{decrease} = \frac{\prod^{f_A} \prod^{f_B} \varphi(e|f)}{|f_B|}.$$

$|f_A|$ is number of a specific linguistically correct source phrase (f_A) with respect to source and target phrase, and $|f_B|$ is number of a specific linguistically correct source phrase (f_B) with respect to source phrase only. Hence, denominators of these equations are used to distribute the score uniformly so that total incremental score and total decremental score will be equal. Thus,

$$\varphi(e|f)_A + \varphi(e|f)_B = 1$$

where $\varphi(e|f)_A = \varphi(e|f)_A + \frac{\prod^{f_A} \prod^{f_B} \varphi(e|f)}{|f_A|}$ and

$$\varphi(e|f)_B = \varphi(e|f)_B - \frac{\prod^{f_A} \prod^{f_B} \varphi(e|f)}{|f_B|}.$$

Inverse phrase translation probability is updated in the same manner as follows:

$$\varphi(f|e)_A = \varphi(f|e)_A + \frac{\prod^{e_A} \prod^{e_C} \varphi(f|e)}{|e_A|}$$
$$\varphi(f|e)_C = \varphi(f|e)_C - \frac{\prod^{e_A} \prod^{e_C} \varphi(f|e)}{|e_C|}$$

where $\varphi(f|e)_A$ and $\varphi(f|e)_C$ are, respectively, direct phrase translation probability with linguistically correct phrase in both languages and linguistically correct phrase in target language only. From this discussion it is easily understood that after first phase of operation the total probability score is preserved.

For all other various types of phrases from different sets this can be easily proved using a similar strategy.

Lemma 5.2 *After updating the phrase-table with semantic bilingual knowledge, it satisfies the total probability theorem.*

The basic logic behind the second phase of the proposed approach is increment of the direct phrase translation probability scores $(\varphi(f|e))$ and inverse phrase translation probability scores $(\varphi(e|f))$ for aligned semantically correct phrases. It also decreases the direct phrase translation probability scores $(\varphi(f|e))$ and inverse phrase translation probability scores $(\varphi(e|f))$ for aligned semantically incorrect phrases. Total number of semantically correct aligned phrases and incorrect aligned phrases is $|Sem|$ and $|nonSem|$, respectively. Joint probability is denoted by product. Thus individual increment for semantically correct phrases will be $\frac{product}{Sem}$, and decrease for semantically correct phrases will be $\frac{product}{nonSem}$ for both $(\varphi(f|e))$ and $(\varphi(e|f))$. Finally both total increased and decreased probability scores will be the same, which can be proved using the same logic as in Lemma 5.1. As the increased and the decreased probabilities are the same the total probability will remain the same, which is equal to 1.

6. Datasets, experimental set-up, results and analysis

This section deals with various implementation requirements such as experimental set-up and used datasets. Results and analysis are reported here.

6.1 *Datasets*

We have evaluated proposed approach on the in-house product catalogs (Dataset 1) and Launchpad data (Dataset 2) (which is mined from the HindEnCorp corpora [29]) for English–Hindi pair. The data distribution is described in Tables 1 and 2 for Dataset 1 and Dataset 2, respectively. In Dataset 1, the training set and development set comprise

Table 1. Statistics of Dataset 1; En and Hi represent English and Hindi language, respectively; #Sentences and #Tokens denote number of sentences and number of tokens, respectively.

		#Tokens	
Set	#Sentences	En	Hi
Train	111,580 × 2	637,966	704,185
Test	5640	49,394	57,037
Development	599 × 2	7,041	8,604
Monolingual LM corpus	111,580	708,275	

Table 2. Statistics of Dataset 2. It is Launchpad technical domain Dataset collected from HindEnCorp corpora. LM stands for language model.

Set	#Sentences	#Tokens	
		En	Hi
Train	64,724 × 2	458,831	532,985
Test	10, 02	7,229	8,519
Development	1,001 × 2	7,241	8,271
Monolingual Hindi corpus for LM	64,724	532,985	
Used model	3-gram language model		

111,580 and 599 English–Hindi parallel sentence pairs, respectively. To test the model, 5,640 sentences are used from the product domain English–Hindi language pair. The chosen corpora consists of sentences with varying lengths. Minimum and maximum lengths of the sentences vary from 3 to 80 tokens. Approximately 10 is the average length of the sentences.

In Dataset 2, the training set and development set comprise 64724 and 1001 English–Hindi parallel sentence pairs, respectively. We used 1002 sentence pairs from English–Hindi technical domain set to test the model.

Along with English to Hindi translation, ILCI corpus [30] (Dataset 3) has been used for Hindi to Konkoni and Bengali to Hindi translations. Detailed distribution for various datasets is presented in Table 3. Finally it shows significant improvement in quality, which proves that the proposed methodology works also for other language pairs.

Finally, we did the same experiments using popular benchmark Dataset WMT 14 [31]. The dataset distribution for our experiments is shown in Table 6.

6.2 *Experimental set-up*

Moses[9] set-up is used as the preliminary system for basic phrase-table creation of SMT [32]. With the help of English–Hindi parallel corpora, SMT model is trained to get the basic phrase-table. To get the updated phrase-table for better performance our proposed model is applied on this phrase-table, which is generated by typical PBSMT system. The proposed model is divided into two phases. The first phase of the model focuses on syntactic knowledge, based on PoS information. The second phase is concerned with the semantic knowledge by the lexicon. Finally, the modified phrase-table is used for decoding. Development set is used to tune the model minutely. Standford parser[10] is used to acquire PoS information of English sentences. Shallow parser is used to acquire PoS information of Hindi sentences[11], Bengali sentences[12], etc. For preparing lexicon we use bilingual mapping (such as English–Hindi), which is a gathered information from IndoWordnet [28]. Thus, there is no problem in handling synonyms or similar phrases.

6.3 *Results and analysis*

Total work is concentrated to build better qualitative phrase-table, for a better quality of translated output. The phrase-table (which is generated after training the statistical model) is used to update the phrase-table itself. Based on various linguistic knowledge, phrase-table is divided into four sets as discussed earlier. After the update, all the phrase-tables are combined and sorted accordingly. The final modified phrase-table updated after the second phase is used for decoding to have better translated output. Statistical analysis for accuracy with a proper comparison is shown in Table 4 (for English–Hindi translation) and Table 5 (for Hindi–Konkoni and Bengali–Hindi translation). Our proposed methodology (Table 6) outperforms over the base-line method [3]. We observe continuous improvements in the accuracy with syntactic information. We can achieve further improvements in translation accuracy with joint incorporation of syntactic and semantic information. The improvement score is based on the joint probability model, which is small compared with total probabilistic score. The heuristic model is capable of increasing this score intuitively, which makes the translation better. Since it assigns more priority for linguistically correct phrases over linguistically incorrect phrases, its accuracy is improved. As a result, adequacy and fluency of the translated sentences are increased.

Finally, surprisingly, we find more improvement if we add heuristic knowledge into it. The heuristic knowledge adds/reduces some (ξ) percentages of calculated improved score. This calculated improved score (Δ) is the total score difference between after and before syntactic and semantic operation. We randomly assume ξ as 75 for our experiments. We think that changing of scores is very small, so this heuristics values help to improve the scores based on their correctness. Finally, better phrases gain more priority and chance of selection of incorrect phrases again is reduced.

The accuracy of the translated output is evaluated using various evaluation metrics, such as BLEU [33], NIST [34] and RIBES [35] MT evaluation metrics. The improved scores are 11.88 BLEU points, 0.1213 RIBES points and 0.568 NIST points with respect to the base-line (PBSMT) system for Dataset 1. Finally, we found 58.54 BLEU points, 0.7759 RIBES points and 9.684 NIST points by syntactic+semantic+heuristic approach.

The improved scores are 3.33 BLEU points, 0.0275 RIBES points and 0.221 NIST points with respect to the base-line (PBSMT) system for Dataset 2.

[9] http://www.statmt.org/moses/.

[10] http://nlp.stanford.edu/software/lex-parser.html.

[11] http://ltrc.iiit.ac.in/analyzer/hindi/.

[12] http://ltrc.iiit.ac.in/analyzer/bengali/.

Table 3. Statistics of Dataset 3, which is Indian–Indian language pairs (ILCI corpus).

	Train			Development			Test	
		#Token			#Token			
Language pairs	#Sentence	Source	Target	#Sentence	Source	Target	#Sentence	#Token
Hindi-Konkoni	38998	696786	541158	1000	18365	14355	400	8047
Bengali-Hindi	38998	571653	696786	1000	14788	18365	400	6554

Table 4. Evaluation results for English–Hindi translation, incorporated with syntactic and semantic information.

	Dataset 1					Dataset 2				
Evaluation metrics	PBSMT system [3]	Pre-order SMT [27]	Syntactic knowledge	Syntactic + semantic knowledge	Syntactic + semantic with heuristic	PBSMT system [3]	Pre-order SMT [27]	Syntactic knowledge	Syntactic + semantic knowledge	Syntactic + Semantic with heuristic
BLEU	46.66	55.14	57.43	58.01	58.54	36.15	37.21	38.12	38.87	39.48
RIBES	0.6546	0.7423	0.7659	0.7211	0.7759	0.6048	0.6121	0.6200	0.6235	0.6323
NIST	9.116	9.412	9.504	9.164	9.684	7.4511	7.5211	7.5921	7.6113	7.6721

Table 5. Evaluation results for Indian language pairs with Dataset 3.

Language pairs	PBSMT system [3]	Syntactic knowledge	Syntactic + symantic knowledge	Syntactic + semantic with heuristic	Evaluation metrics
Hindi–Konkani	5.2112	5.3214	5.3944	5.4121	NIST
	20.12	21.09	21.95	22.12	BLEU
	0.4128	0.4313	0.4867	0.5123	RIBES
Bengali–Hindi	4.4213	4.4501	4.5011	4.5123	NIST
	16.02	17.19	17.8949	18.2174	BLEU
	0.3251	0.3343	0.3401	0.3486	RIBES

Table 6. Statistics of prepossessed WMT 14 Dataset.

		#Tokens	
Set	#Sentences	En	Hi
Train	1458686	19175506	20841335
Test	1001	21563	22383
Development	520	10656	10524
Monolingual Hindi corpus	1458686	20841345	

The improved scores are 2 BLEU points, 0.0995 RIBES points and 0.2009 NIST points with respect to the base-line (PBSMT) system for Hindi–Konkoni translation using Dataset 3.

The improved scores are 2.1974 BLEU points, 0.0235 RIBES points and 0.091 NIST points with respect to the base-line (PBSMT) system for Bengali–Hindi translation using Dataset 3.

Moreover, we did the same experiments using the WMT 14 Dataset [31]. We found an improvement over the base-line phrase-based MT system. The performance analysis for this dataset is shown in Table 7. We find similar behaviour here. It reflects continuous improvements of the translation accuracy with incorporation of linguistic information into SMT system.

We picked some random source sentences (SS) and their translated output from our proposed system (PS) and also from the PBSMT system.

SS: A complete bat ball set for your kid.
PBSMT:
एक पूर्ण चमगादड़ का बॉल आपके बच्चे के लिए सेट।
HT: Ek purna chamgaadar ka ball aapke bachche ke liye.

Table 7. Performance analysis with WMT 14 Dataset.

	PBSMT System [3]	Pre-order SMT [27]	Syntactic knowledge	Syntactic + Semantic knowledge	Syntactic+Semantic with heuristic
BLEU	16.57	17.12	17.87	18.11	19.14
RIBES	24.89	25.52	25.92	26.32	27.23
NIST	18.02	18.65	19.01	19.26	20.13

Table 8. Human evaluation for fluency and adequacy checking.

	PBSMT system	Pre-order SMT	Syntactic knowledge	Syntactic + semantic knowledge	Syntactic + semantic with heuristic
Fluency	2.5	3	3.5	3.9	4.1
Adequacy	2.7	3	3.1	3.3	3.7
Inter-annotator agreement (fluency)	45.45%	54.54%	54.54%	63.63%	45.45%
Inter-annotator agreement (fluency)	63.63%	54.54%	45.45%	45.45%	63.63%

PS: पूरा बैट बॉल आपके बच्चे के लिए।
HT: Puraa bat ball aapke bachche ke liye.

Output of PS is better than PBSMT system. The PBSMT system translates the word "bat" as "चमगादड़" (HT: chamgaadar), which is not the meaning of the bat in the sentence (bat is an ambiguous word here, which has two meanings) and PS just transliterates the word as बैट (HT: bat). The PBSMT system does not give the sense of source sentence while PS conveys the meaning of source text.

SS: This nail colour is known for its long lasting effect.
PBSMT:

के लिए जाना जाता है यह नेल कलर अपने लंबे समय तक चलने वाले प्रभाव ।

HT: Ke liye jaanaa jaataa hai yeh nel kalar apne lambe samay tak chalne vaale prabhav.
PS:

यह नेल कलर अपने लंबे समय तक चलने वाले प्रभाव के लिए जाना जाता है ।

HT: Yeh nel kalar apne lambe samay tak chalne vaale prabhav ke liye jaanaa jaataa hai.

The syntactic structure of PBSMT system is incorrect. In the PBSMT system the sentence starts with "के लिए जाना जाता है" (HT: ke liye jaanaa jaataa hai), which is not welcome in the Hindi structure. Moreover, the word "के लिए" (HT: ke liye) in PBSMT system is for the word in source sentence for its long lasting effect. The translation should be "अपने लंबे समय तक चलने वाले प्रभाव के लिए" (HT: apne lambe samay tak chalne vaale prabhav ke liye), generated by the proposed system. Hence, the proposed MT system is better than existing systems (i.e., PBSMT system, PBSMT+syntactic).

Finally, 11 linguists (translation experts) evaluated (manually) the performance analysis considering adequacy and fluency for WMT 14 Dataset. They scored in a scale of 1 to 5 for randomly chosen specific 150 sentence pairs. Hence, every sentence was manually evaluated (scored) 11 times. One is assigned for the worst and 5 is assigned for the best output. Averages of their scores are shown in Table 8. In this manual evaluation we found performance similar to that of automated evaluation for injecting syntactic and semantic information, which was discussed earlier. Percentage of the inter-annotator agreement is also mentioned here, which is really impressive.

7. Conclusion

We have proposed to improve the phrase-table of the SMT system. The phrase translation probability scores have been modified depending on the correctness of the phrases. Instead of pruning out any linguistically incorrect phrase, we modify the scores to have qualitative phrase-table for better translated accuracy. Thus, instead of information loss, additional information is incorporated into the phrase-table. Syntactic information improves the fluency whereas semantic information improves the adequacy in our proposed approach. For this reason, adequacy and fluency of the translated output have been increased. This approach can be applied to several translation systems for better

performance. Thus, we can conclude that accuracy of translated output can be improved using collaboration of the linguistic knowledge with statistical model. We are planning to update other probabilistic scores, along with the direct and inverse phrase translation probability, which are present in the phrase-table, for further improvement.

References

[1] Callison-Burch C and Koehn P 2005 Introduction to statistical machine translation. *Language 1*: 1

[2] Koehn P and Monz C 2006 Shared task: Exploiting parallel texts for statistical machine translation. In: *Proceedings of the NAACL 2006 workshop on statistical machine translation, New York City (June 2006)*

[3] Koehn P, Och F J and Marcu D 2003 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-Volume 1*, pp. 48–54. Association for Computational Linguistics

[4] Och F J, Tillmann C, Ney H, et al 1999 Improved alignment models for statistical machine translation. In: *Proc. of the Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 20–28

[5] Yamada K and Knight K 2001 A syntax-based statistical translation model. In: *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pp. 523–530. Association for Computational Linguistics

[6] Marcu D and Wong W 2002 A phrase-based, joint probability model for statistical machine translation. In: *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pp. 133–139. Association for Computational Linguistics

[7] Hanneman G and Lavie A 2009 Decoding with syntactic and non-syntactic phrases in a syntax-based machine translation system. In: *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation*, pp. 1–9. Association for Computational Linguistics

[8] Vilar D, Stein D and Ney H 2008 Analysing soft syntax features and heuristics for hierarchical phrase based machine translation. In: *IWSLT*, pp. 190–197

[9] Marton Y and Resnik P 2008 Soft syntactic constraints for hierarchical phrased-based translation. In: *ACL*, pp. 1003–1011

[10] Chiang D 2005 A hierarchical phrase-based model for statistical machine translation. In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 263–270. Association for Computational Linguistics

[11] Nadejde M, Birch A and Koehn P 2016 Modeling selectional preferences of verbs and nouns in string-to-tree machine translation. In: *WMT*, pp. 32–42

[12] Weller M, Walde S S I and Fraser A 2014 Using noun class information to model selectional preferences for translating prepositions in smt. In: *Proceedings of AMTA*

[13] Wang C, Collins M and Koehn P 2007 Chinese syntactic reordering for statistical machine translation. In: *EMNLP-CoNLL*, pp. 737–745

[14] Wang W, Knight K and Marcu D 2007 Binarizing syntax trees to improve syntax-based machine translation accuracy. In: *EMNLP-CoNLL*, pp. 746–754

[15] DeNeefe S, Knight K, Wang W and Marcu D 2007 What can syntax-based mt learn from phrase-based mt? In: *EMNLP-CoNLL*, pp. 755–763

[16] Marcu D, Wang W, Echihabi A and Knight K 2006 Spmt: Statistical machine translation with syntactified target language phrases. In: *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pp. 44–52. Association for Computational Linguistics

[17] Charniak E, Knight K and Yamada K 2003 Syntax-based language models for statistical machine translation. In: *Proceedings of MT Summit IX*, pp. 40–46

[18] Pal S, Hasanuzzaman M, Naskar S K and Bandyopadhyay S 2013 Impact of linguistically motivated shallow phrases in pb-smt. *ICON*

[19] Banik D, Ekbal A and Bhattacharyya P 2018 Machine learning based optimized pruning approach for decoding in statistical machine translation. *IEEE Access 7*: 1736–1751

[20] Sen S, Banik D, Ekbal A and Bhattacharyya P 2016 Iitp English-Hindi machine translation system at wat 2016. In: *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pp. 216–222

[21] Banik D, Sen S, Ekbal A and Bhattacharyya P 2016 Can smt and rbmt improve each other's performance?-an experiment with english-hindi translation. In: *13th International Conference on Natural Language Processing*, p. 10

[22] Banik D, Ekbal A, Bhattacharyya P and Bhattacharyya S 2019 Assembling translations from multi-engine machine translation outputs. *Applied Soft Computing 78*: 230–239

[23] Chiang D, Marton Y and Resnik P 2008 Online large-margin training of syntactic and structural translation features. In: *Proceedings of the conference on empirical methods in natural language processing*, pp. 224–233. Association for Computational Linguistics

[24] Jones B, Andreas J, Bauer D, Hermann Karl Moritz and Knight Kevin 2012 Semantics-based machine translation with hyperedge replacement grammars. In: *COLING*, pp. 1359–1376

[25] Hermann K M 2012 Semantics-based machine translation with hyperedge replacement grammars

[26] Collins M, Koehn P and Kučerová I 2005 Clause restructuring for statistical machine translation. In: *Proceedings of the 43rd annual meeting on association for computational linguistics*, pp. 531–540. Association for Computational Linguistics

[27] Ramanathan A, Hegde J, Shah R M, Bhattacharyya P and Sasikumar M 2008 Simple syntactic and morphological processing can help English-Hindi statistical machine translation. In: *IJCNLP*, pp. 513–520

[28] Bhattacharyya P 2017 Indowordnet. In: *The WordNet in Indian Languages*, pp. 1–18. Springer

[29] Bojar O, Diatka V, Rychlỳ P, Stranák P, Suchomel V, Tamchyna A and Zeman D 2014 Hindencorp-Hindi-English and Hindi-only corpus for machine translation. In: *LREC*, pp. 3550–3555

[30] Jha G N 2010 The tdil program and the Indian language corpora intiative (ilci). In: *LREC*

[31] Kunchukuttan A, Mehta P and Bhattacharyya P 2017 The IIT Bombay English-Hindi parallel corpus. *arXiv preprint* arXiv:1710.02855

[32] Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R, et al 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*, pp. 177–180. Association for Computational Linguistics

[33] Papineni K, Roukos S, Ward T and Zhu W-J 2002 Bleu: a method for automatic evaluation of machine translation. In: *Proceedings of the 40th annual meeting on association for computational linguistics*, pp. 311–318. Association for Computational Linguistics

[34] Callison-Burch C 2005 Linear b system description for the 2005 nist mt evaluation exercise. In: *Proceedings of the NIST 2005 Machine Translation Evaluation Workshop*. Citeseer

[35] Isozaki H, Hirao T, Duh K, Sudoh K and Tsukada H 2010 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. Association for Computational Linguistics