Applied Artificial Intelligence
An International Journal

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/uaai20

Fully Unsupervised Machine Translation Using Context-Aware Word Translation and Denoising Autoencoder

Shweta Chauhan, Philemon Daniel, Shefali Saxena & Ayush Sharma

To cite this article: Shweta Chauhan, Philemon Daniel, Shefali Saxena & Ayush Sharma (2022) Fully Unsupervised Machine Translation Using Context-Aware Word Translation and Denoising Autoencoder, Applied Artificial Intelligence, 36:1, 2031817, DOI: 10.1080/08839514.2022.2031817

To link to this article: https://doi.org/10.1080/08839514.2022.2031817

Published online: 04 Feb 2022.

APPLIED ARTIFICIAL INTELLIGENCE
2022, VOL. 36, NO. 1, e2031817 (1795 pages)
https://doi.org/10.1080/08839514.2022.2031817

Taylor & Francis
Taylor & Francis Group

Fully Unsupervised Machine Translation Using Context-Aware Word Translation and Denoising Autoencoder

Shweta Chauhan, Philemon Daniel, Shefali Saxena, and Ayush Sharma

Department of Electronics and Communication Engineering, National Institute of Technology, Hamirpur, INDIA

ABSTRACT

Learning machine translation by using only monolingual data sets is a complex task as there are many possible ways to connect or associate target sentences with source sentences. The monolingual word embeddings are linearly mapped on a common shared space through robust learning or adversarial training in an unsupervised way, but these learning techniques have fundamental limitations in translating sentences. In this paper, a simple yet effective method has been proposed for fully unsupervised machine translation that is based on cross-lingual sense to word embedding instead of cross-lingual word embedding and language model. We have utilized word sense disambiguation to incorporate the source language context in order to select the sense of a word more appropriately. A language model for considering target language context in lexical choices and denoising autoencoder for language insertion, deletion, and reordering are integrated. The proposed approach eliminates the problem of noisy target language context due to erroneous word translations. This work takes into account the challenge of homonyms and polysemous words in the case of morphologically rich languages. The experiments performed on English-Hindi and Hindi-English using different evaluation metrics show an improvement of +3 points in BLEU and METEOR-Hindi over the baseline system.

ARTICLE HISTORY
Received 22 February 2021
Revised 25 December 2021
Accepted 18 January 2022

Introduction

Machine Translation (MT) helps in breaking the language barrier but requires parallel data sets. However, bilingual corpora are restricted to high-resource languages like English or Chinese as compared to low-resource language. Unsupervised machine learning is an alternative to this approach, where the machine can be trained using monolingual corpora. Neural Machine Translation (NMT) models (Bahdanau et al., 2015) are the current standard, and the most challenging part is to train the system without substantial parallel corpora (Koehn and Knowles 2017).

CONTACT Shweta Chauhan shweta@nith.ac.in Department of Electronics and Communication Engineering, National Institute of Technology Hamirpur 177005, India

{shweta, phildani7, Shefali, ec16mi439}@nith.ac.in

The recent approach for Unsupervised NMT (Artetxe, Labaka, and Agirre 2018; Lample et al., 2018) trains sequence-to-sequence MT models for both source-to-target and target-to-source translation using only monolingual corpora of each language. But the problem with these models is that they use back translation (Sennrich, Haddow, and Birch 2015) back and forth for each iteration or batch and that they require a longer training period and require more precise tuning of hyperparameters for huge monolingual data. This problem is overcome by another approach that is based on word-by-word translation using cross-lingual word embeddings (CLWEs) (Kim, Geng, and Ney 2019) without back translation for an unsupervised MT system. But, in this approach, the target side context depends on the quality of translation of the previous n words, which may be corrupted due to selection of wrong words. Moreover, the first word has no context information available and if it is not properly translated, it may lead to inaccurate translation for further words. Also, this approach does not take into account the challenge of homonyms and polysemous words in the case of morphologically rich languages.

Morphologically, languages are problematic in MT especially if the translation is from a morphologically less to a richer language. Furthermore, to improve the output quality of MT especially for morphologically richer languages is another tedious and complex task. Morphological distinctions, which are not present in the source language, need to be generated in the target language. Mostly, Indian languages fall under the category of morphologically richer languages, and we have considered Hindi because it is the fourth most-spoken language in the world and is morphologically richer. When a source sentence is translated into target, it is structurally different. Ananthakrishnan et al.,2007 analyzes the English-Hindi language pair in terms of divergence problems, which are faced in their MT task. The two main tasks of a MT system are lexical and structural transfers. Lexical transfer in the case of English-Hindi includes the challenges of identifying the sense of a word and dealing with the case of two or more than two words in one language, which are realized by one word in the other language. In sentence structuring, English uses subject-verb-object word order, whereas Hindi uses subject-object-verb word order.

In this work, we propose an improved unsupervised MT system based on word-to-word translation using cross-lingual sense to word embeddings (CLSWEs) and language model (LM). CLSWEs are generated from source language corpus to learn multiple possible senses or meanings for each word of source language, and each sense is mapped to its translation in target language. CLSWE is used to find the correct sense according to source sentence context and thus to find source-to-target word translation for that sense of the word. For utilizing target language context, we integrate LM, which is learnt on target language corpus and is used to score the possible target translations.

The proposed work combined CLSWE and LM to make use of both source and target language context and thus gives an improved context-aware word translation. Each word in the source sentence is first disambiguated using the context, and hence, sense probabilities are calculated for every sense of the word. Nearest target words are retrieved from CLSWE for each word in the source sentence. Then, a pretrained LM is used to score the possible translations, which are generated by multiple combinations of retrieved words using beam search. Sense probabilities and LM scores are scaled and added to the word similarity score. Finally, the translated word in the target language is the word with the highest score. Even if the words are translated correctly for each position, the output is still not a satisfactory translation. Thus, to enhance or improve the output, we use a denoising autoencoder, which is a transformer (Vaswani et al. 2017) encoder-decoder, that further improves the translation as per target language rules. Sockeye toolkit (Hieber et al. 2017) is used to implement the encoder-decoder architecture for denoising. Different evaluation metrics BLEU, METEOR, and METEOR-HINDI depict an improved performance for English and Hindi languages.

The major contributions of this paper are as follows:

- To overcome the limitation posed by noisy target language context due to inaccurate source-to-target word translations, an improved unsupervised MT system is proposed that combines CLSWE and LM.
- To determine the appropriate or correct sense of polysemous words according to source language context, word sense disambiguation (WSD) has been incorporated in this approach.
- We propose a LM for considering target language context in lexical choices. The combination of CLSWE and LM utilizes source and target language context optimally in order to produce an improved context-aware word translation.
- A postprocessing method is used for handling insertion, deletion, and reordering of contextualized word translation outputs.
- Mostly, Indian languages fall under the category of morphologically richer languages, and they are always problematic in MT especially if the translation is from a morphologically less to a morphologically more complex language. We have considered Hindi because it is the fourth most-spoken language in the world and is morphological richer and it is our mother tongue.

The rest of this paper is organized as follows. We analyze previous work in Section 2. Section 3 describes the unsupervised word-to-word translation methods. Section 4 discusses our proposed word in unsupervised sentence translation. We have our experimental setup in section 5 and the results and discussion in Section 6. Section 7 concludes this paper and lists future work.

Related Work

Machine translation systems depend upon the availability of parallel sentences, which is time-consuming for most language pairs. Unsupervised learning is an alternative, where we can train an MT system with only monolingual corpora. Ravi and Knight 2011 used prior linguistic information in their seminal work to re-evaluate the unsupervised MT challenge as breaking down and demonstrate the feasibility of short sentences with a small vocabulary. While earlier work (Carbonell et al. 2006) also aimed at unsupervised machine translation, a bilingual dictionary has been used to seed the translation. Both works depend on a target side language model to correct the fluency of the translation. Subsequent research (Irvine and Callison-Burch 2017; Klementiev et al. 2012) depends upon bilingual dictionaries, less parallel corpora, and linguistically motivated features to prune the search space. Back translation has gained popularity in recent years for augmenting training sets on the target side with the monolingual data set (Sennrich, Haddow, and Birch 2015).

For fully unsupervised MT (Artetxe et al. 2017; Lample and Denoyer 2017), promising results have been obtained in standard machine translation tests using only monolingual corpora for the first time, but they have used back translation that has a longer training period. The working principle of both the methods is based upon the recent work on the unsupervised CLWE mappings. It is capable of training embeddings in two languages independently. It develops a linear transformation to be mapped on a shared space by self-learning (Artetxe, Labaka, and Agirre 2018) or adversarial training (Conneau et al. 2017). Thus, by using the only monolingual data set, we have word-by-word translation of cross-lingual embeddings. Here (Kim, Geng, and Ney 2019), an unsupervised MT system based on word translation using CLWE but without any back translation is proposed, but the problem is that it does not work well for morphologically rich languages. CLWE cannot capture the complexity of words with multiple meanings, such as homonyms or polysemous words. The solution of this limitation is Word Sense Disambiguation (WSD) (Pelevina et al. 2017), which identifies the correct sense of a given word in a particular sentence. Many techniques (Ahmed and Hawraa,2019) were used in WSD on different corpora for all languages. The BERT (Devlin et al. 2019), ELMo (Ilić et al. 2018), Bart (Lewis et al. 2019), and GPT (Brown et al. 2020) models are widely used in MT. Many models on various leader boards use models from the BERT family, but BERT fails (Ettinger 2020; Kodge and Roy 2021). GLUE (Wang et al. 2018) benchmarks are bart variant except for T5 (Raffel et al. 2019), which uses the transformer architecture.

BLEU (Papineni et al. 2002) has been mostly used in MT evaluation due to its easy implementation, competitive performance to capture the fluency of translation, and language independence. It depends upon the n-gram

matching of the hypothesis and reference translation. Other metrics have also been used for evaluation like WER (Su, Wu, and Chang 1992), PER (Tillmann et al., 1997), NIST (George Doddington 2002), TER (Snover et al. 2009), and ROUGE (Lin 2004). They mainly depend on the exact matches of the surface words in the output machine translation. WER, PER, and TER measure the edit distance between the reference and hypothesis by estimating the minimum total number of editing steps to transform the hypothesis to reference translation. Like BLEU, NIST calculates the degree of the n-gram overlapping between the hypothesis and reference translation. METEOR-Hindi (Gupta, Venkatapathy, and Sangal 2010) has extended the implementation of METEOR (Lavie and Agarwal, 2005) to support the evaluation of translations into Hindi. As the properties of other Indian languages are very similar to those of Hindi, METEOR-Hindi can be easily extended to different Indian languages.

Word-to-Word Translation

The first step of the unsupervised MT system is to learn a word translation model from the monolingual corpora of each language. Current state-of-the-art word translation models in unsupervised neural MT exploit CLWE (Artetxe, Labaka, and Agirre 2018). CLWEs are the same as common word embeddings but try to capture words from multiple languages into one embedding space. They use the additional insight that many, perhaps most, words in all languages refer to common concepts. So, for example, king (English) and राजा (Hindi) have the same meaning (Chauhan, Saxena, and Daniel 2021a Chauhan, Saxena, and Daniel 2021b).

In other words, CLWE represents continuous words in vector or real numbers in a vector space that is shared across multiple languages. This helps in measuring the distance between word embeddings across multiple languages, for finding possible word translation. First, learning both source and target embeddings from their monolingual corpora is done independently. Second, source embedding space is linearly mapped with the target embedding space by the fully unsupervised robust self-learning method (Artetxe, Labaka, and Agirre 2018).

Word Retrieval Techniques with CLWE

Retrieval techniques are methods for identifying word translation pairs across the two languages. They use different kinds of similarity measures to produce words in the target language, which are most similar to a given word in source language. Four different types of retrieval techniques are used to retrieve words from the CLWE. The four methods are nearest neighbor retrieval, inverted nearest neighbor retrieval (Dinu, Lazaridou, and Baroni 2014),

inverted softmax (Smith et al. 2017), and cross-domain similarity local scaling (CSLS) retrieval. CSLS improves the accuracy of retrieving word translation significantly while not requiring any parameter tuning across all retrieval methods (Azarbonyad, Shakery, and Faili 2019).

Limitation of Word Retrieval from CLWE

In most cases, the correct translation is not the closest target word but some other adjacent word, which is a synonym or morphological variation of the closest word. The reason is that training of word embedding is such that it places nearby semantically related words, even if they have opposite meanings. Moreover, the context around the current word is not included in word retrieval technique-based translations.

Contextual Word Translation Using LM

As word retrieval methods are unaware of context information, word-to-word translation using CLWE may not provide accurate results in many cases. To overcome this drawback, context information is integrated with the word-to-word translation by combining a LM with the CLWE.

LM can be used to choose the best target word depending on the context, which is a sequence of previous target words. All combinations of this context and nearest neighbors of the source word on the target side are formed. These candidate translation sequences are scored, and their probabilities are calculated based on counts of various n grams in the training corpus.

Limitation of LM with CLWE

LM helps in choosing the best target word using the target side context. But target side context depends on the accuracy of translation of words preceding the current word, which may get corrupted due to erroneous word translations. Moreover, as the first word has no context information available and if it is not properly translated, it may lead to inaccurate translation of further words.

Fully Unsupervised Sentence Translation

Figure 1 shows the toy illustration of the fully unsupervised MT method. Initially, we take two monolingual data sets (English and Hindi). The sense embeddings are generated from the word embedding of source language. The target language has word embedding indicated in red dots. Each dot will represent a word in n-dimensional space. The next step is to linearly map source sense embedding space to target embedding space (Artetxe, Labaka, and Agirre 2018), which roughly aligns the two distributions, and mapped source and target word embeddings are produced. For a clear understanding,

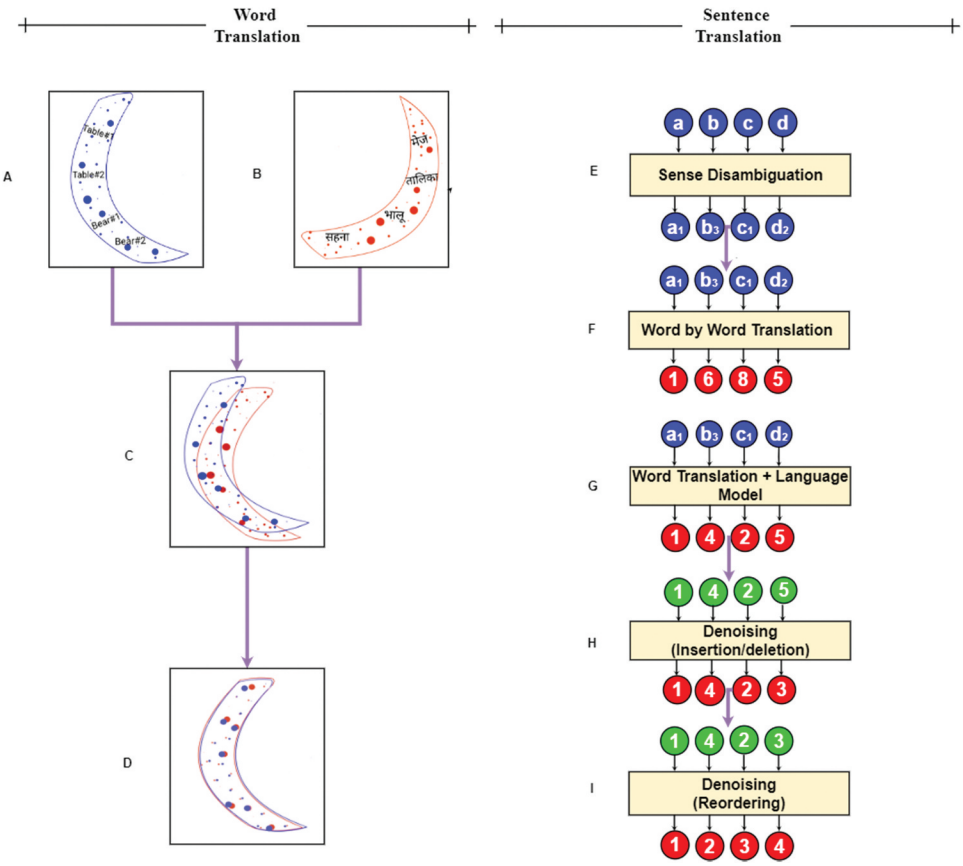

Figure 1. Toy illustration of the method. (A) The sense embeddings created from the word embeddings of the source language are denoted in blue. Each dot represents a sense vector for a word. (B) The word embeddings of the target language are denoted in red. (C) A linearly mapped source embedding space to target embedding space, which aligns two distributions. (D) The aligned or mapped source and target word embedding (E). For a clear understanding, assume a, b, c, d as words in the source language sentence. For each word in a sentence, a sense is selected, which best fits the source side context, which is assumed as a_1, b_3, c_1, d_2. (F) Different word retrieval techniques are used with CLSWE to find the nearest target words for each source sense word, which are assumed as 1, 6, 8, 5. (G) A LM along with CLSWE and different retrieval techniques is used to incorporate target context information. (H) To reconstruct a sentence, the encoder-decoder model is trained with insertion and deletion noise. It takes a noisy input sequence and outputs a clean, denoised sequence. (I) To form a proper order of words, an encoder-decode model is used, which is trained using random reordering noise.

we are taking a, b, c, and d as the source words. For each word in a sentence, a sense is selected, which best fits the source language context assumed as a_1, b_3, c_1, and d_2.

After that, different word retrieval techniques like nearest neighbor retrieval, inverted nearest neighbor retrieval, inverted softmax, and cross-lingual word scaling with CLSWE can be used to find the nearest target word for each source sense word. A KenLM and CLSWE are used to incorporate context

information in the word-to-word lexical translation. After applying this, we get 1, 4, 2 , 5 (assumed output). 1, 4, 2, 5 are more suitable according to the context as compared to the previous output. This output (1, 4, 2, 5) is far from an acceptable translation. To reconstruct the correct sentence, the encoder-decoder model is trained. The motive is to develop a sequence-to-sequence neural network that takes a noisy input and outputs a clean, denoised sequence. The training label sequences are considered as the clean target monolingual sentences, and the training input is the noisy version of it. We added insertion and deletion noise in input sentences. After passing 1, 4, 2, 5 in the system, 3 is added, and 5 is deleted; we get a more suitable output sequence as 1, 4, 2, 3.

In the final step, encoder-decoder is trained in such a way to get a proper word ordering. The training label sequences are taken as the clean target monolingual sentences, and the training input sentences are taken with randomly reordered words. At the output of the decoder, after word reordering, we get the translated output 1, 2, 3, 4.

Unsupervised Sense Embeddings

CLWE cannot capture the complexity of words with multiple meanings, such as homonyms or polysemous words. A solution to this limitation is learning separate representations for each meaning of the word that is word senses. Traditional techniques for this task rely on lexical resources built by humans, such as WordNet. These resources are like a dictionary or thesaurus and include a list of all the possible meanings for each word. These knowledge-based techniques give rise to an additional challenge of creation of such resources and sense-annotated corpora. The time-consuming and expensive nature of the task limits these approaches to a very few well-studied languages; thus, it is not scalable to other languages. Identification of word senses and learning their sense representation can also be automated by analyzing the contexts in which it appears. An unsupervised method (Pelevina et al. 2017) to learn the sense vector space uses a semantic graph, which is constructed by connecting each word to the set of its semantically similar words. In our approach, we use the following steps to learn the sense vector representations.

- First, a semantic graph of word similarities is built. Each word is connected to its nearest neighbors, and the weights of branches are set as the similarity score of the retrieved nearest neighbor with the word under consideration. In this case, nearest neighbors are words with the highest cosine similarity of their respective word vectors.
- Second, the sense induction step is a step in which an ego network is constructed for every word in the vocabulary. In this ego network, words (nodes) referring to the same sense tend to be tightly connected, while

having fewer connections to words refers to a different sense. A word sense can be represented by a group of these tightly connected words or word clusters. For instance, the cluster "chair, bed, bench, stool, sofa, desk, cabinet" can represent the sense "table (furniture)." This ego network is then clustered with the Chinese Whispers algorithm. The number of senses induced may vary for each word as the clustering algorithm used is parameter free.

- Finally, the sense vectors are calculated for each induced sense of the words present in the vocabulary. It is assumed that word sense should be represented by a combination of words in the cluster corresponding to that sense. Thus, sense vectors are calculated as the weighted average of the word vectors present in the cluster of the corresponding sense.

Context-Based Sense Identification

This section describes the task of identifying and assigning the correct sense to a polysemous word according to the context in which it appears. The disambiguation strategy is based on similarity between sense vectors of a word and the context. The context is represented by the average of word vectors corresponding to the context words present in the source language sentence. We also apply context filtering to improve the disambiguation performance. Typically, only several words in context are relevant for sense disambiguation, like "chairs" and "kitchen" are for "table" in "They bought a table and chairs for kitchen." A score is calculated for each word in the context, which quantifies its relevance in distinguishing between the senses. Only those context words are used for sense identification, which are most useful in distinguishing between the senses.

Cross-Lingual Sense to Word Embeddings

The words of the source sentence have been disambiguated into appropriate word sense representation. In order to get word translation for these word senses, a mapping is required from the source sense vectors to the word vectors of their corresponding translations. Thus, it requires to create a cross-lingual embedding space. The unsupervised technique (Artetxe, Labaka, and Agirre 2018) of mapping two vector spaces maps words based on their similarity distributions. It builds an initial translation based on the assumption that the words, which are translations of each other, should have almost identical similarity distribution with other words of their own language. This initial translation is further improved using iterative self-learning. The iterative self-learning technique does not work when starting from a completely random solution and gets stuck in a local optimal. For this reason, even though this initial translation is not accurate on its own, it is required to capture some

cross-lingual signal, which can be used to initialize the self-learning process. As the sense vector of a word sense is a function of words vectors, which identify that word sense, their similarity distributions are restructured to a form similar to their translation in target language.

Thus, we construct a cross-lingual sense vector space by learning the mapping between sense embeddings of the source language and the word embeddings of the target language. This CLSWE makes it possible to utilize the source side context for word-to-word translation. Appropriate word sense is identified using the source sentence context words. Following which, CLSWE is used to find the translation of the word senses in the target language. We experimented with four different word retrieval techniques to get the best word translations from CLSWE.

Figure 2 shows an example of similarity distribution of words in source and target domains. Here, we have the word foot in English for which two senses are learnt, i.e., foot#1 (unit of measurement, scale) is the first sense, and second sense is foot#2 (body part, the lower extremity of the leg below the ankle, on which a person stands or walks). The similarity distribution of the two-word senses is different from each other as can be seen in (B) of Figure 2. The similarity distribution of foot #1 and their target word translation फुट have very similar plots, whereas the word मेज in (C) is unrelated to them and have a different plot.

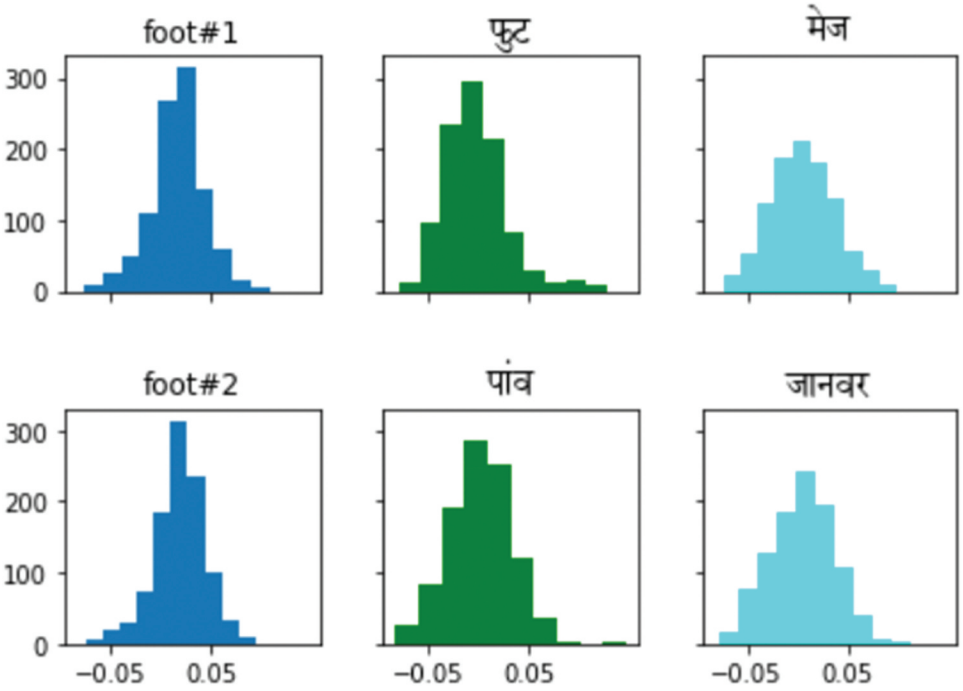

Figure 2. Example of similarity distribution of two senses of word "foot."

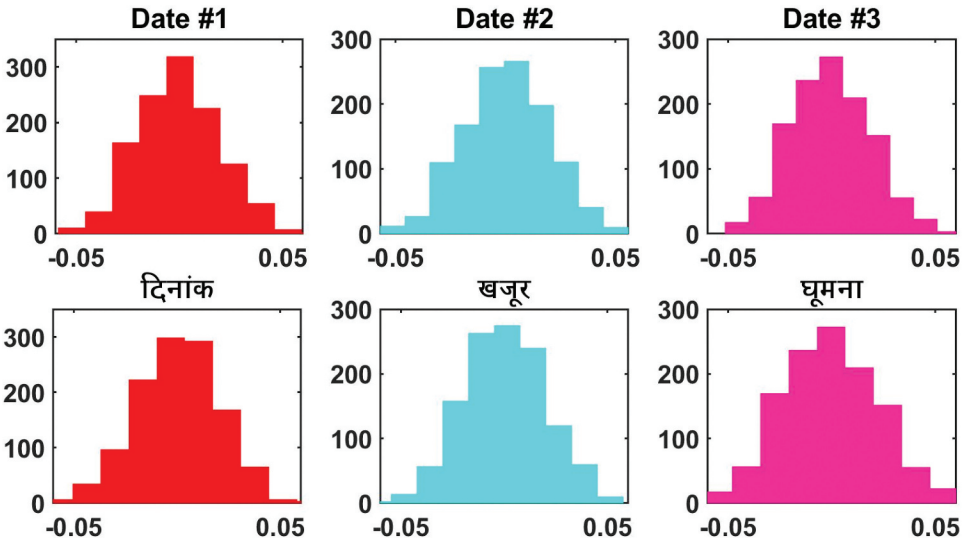

Figure 3. Similarity distribution of three senses of the word "date" for English-Hindi.

Similarly, another word foot #2 has sense word पांव, which has a very similar plot as compared to जानवर, which has totally different distribution. This will be used to build an initial dictionary and to capture some cross-lingual signal (Artetxe, Labaka, and Agirre 2018). We analyzed that the word, which has similar sense, has a similar plot or distribution as in foot#1 and फुट, and foot# 2 and पांव have similar distribution.

Figure 3 shows of similarity distribution of three words in source to target domains. Here, the word date has three senses date#1 (a particular day of a particular month and year), date#2 (fruit), and date#3 (to meet someone socially or romantically). It is observed that a date#1 and दनिांक in (A), whereas the date#2 and खजूर in (B) and date#3 have the घूमना in (C), has similar distribution as compared to each other. This will be used to build an initial dictionary and to capture some cross-lingual signal, which is then utilized in the robust self-learning process to build cross-lingual sense to word embeddings. For example, in the source sentence "Her favorite fruit to eat is a date," date#2 sense is more suitable according to the context of the sentence. Now, CLSWE can be used to find the translation for Date#2, which in this case is खजूर.

Cross-Lingual Sense to Word Embedding with the Language Model

The proposed system combined CLSWE with LM and makes use of both source and target side context in order to give a truly context-aware word translation. Each word in the source sentence is first disambiguated using the context, and sense probabilities are calculated. Top K target words are

retrieved from CLSWE for each word in the source sentence. Then, a pretrained LM is used to score various possible target translations, which are constructed from all the retrieved words using the beam search algorithm.

Figure 4 shows word-by-word translation using CLSWE with the language model for source sentence "He is going to school with bat." Firstly, the sense embedding of the source sentence is calculated and only bat has two senses, first sense the bat#1 (an implement with a handle and a solid surface, usually of wood, used for hitting the ball in games) is used and in other sense bat #2 (rearmouse, a mainly nocturnal mammal capable of sustained flight) is used. Although the above examples demonstrate words with only two senses, there is no restriction on the number of senses a word may have, and some words can have 3 or 4 senses. It will depend upon our vocabulary and word embedding of corpus. We are predicting the nearest top thirty words with their similarity score for target language, but only top three scores are shown in Figure 4. We are considering six-gram LM as it is used to capture the diversity of context for the word. The language model will give the probability score for a word depending upon the immediately preceding words of the source sentence and combining it with the CLSWE score and the target word with the best score will be selected.

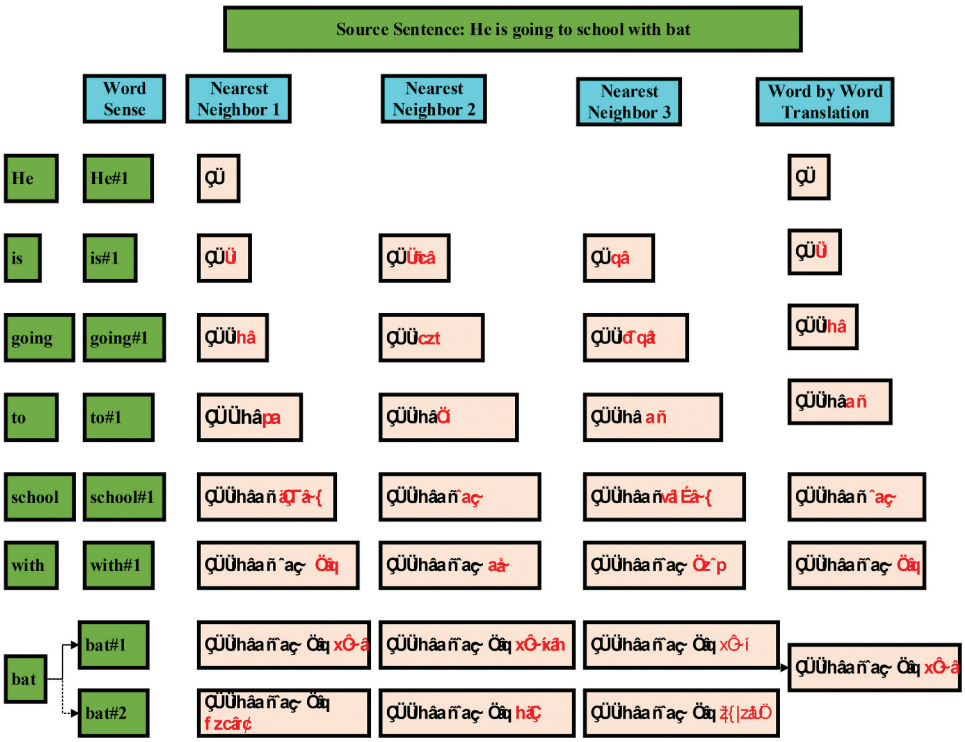

Figure 4. Example of word-by-word translation using CLSWE with the language model.

We have source sentence "he is going to school with bat". For the first word the translation is "वह," for the second word is "है". There are top 3 possibility that वह है, वह होगा, वह था, will come. वह है is more appropriate in the context and selected by LM at final word-to-word translation and so on. The last word bat has two senses; bat #1 sense is more suitable according to the context of the sentence. Now, CLSWE can be used to find the translation for bat #1, which in this case, is बल्ला. Finally, word-to-word translation score is "वह है जा को स्कूल साथ बल्ला."

In this paper, we integrate the source and target language context by scaling and adding target LM score and source word sense probability score to the word similarity score. Hence, the target words with best scores are selected as word translations.

We have taken the following assumptions:

S: source word.

$\{S_1, S_2 \ldots S_i \ldots S_n\}$: possible senses of the source word (S) learnt through sense embedding training.

z: possible target word.

$C = \{C_1, C_2 \ldots C_m\}$: source sentence context words.

h: History of the target word.

S_{sense}: Sense likelihood score.

Y: Word score

$\overline{C_w}$: Mean context vector

Let S be the set of possible senses of a word, and $S = \{S_1, S_2 \ldots S_i \ldots S_n\}$ are the possible senses of the source word. Given the source sentence context words $C = \{C_1, C_2 \ldots C_m\}$ and a history (h) of target words before z, the sense likelihood score helps in identifying the correct sense according to the context and is calculated as in equation 1. h is the history of words, which occur before z,

$$S_{sense}(i) = \text{cosim}(S_i, \overline{C_w}), \tag{1}$$

where $\overline{C_w} = \frac{\sum C_{wi}}{N}$.

$\overline{C_w}$ is the mean context vector, which is calculated as the average of word vectors of the words, which are present in the context, i.e., the current sentence. Cosim is cosine similarity, with the measured cosine similarity score of 1, which means that two vectors have the same orientation. The value closer to 0 indicates that the two documents have less similarity.

Final sense to word translation score is defined as

$$Y(i, z|\overline{C_w}, h) = \lambda_{senseemb}.\eta(S_i, \overline{C_w}).logr(S_i, z) + \lambda_{LM}\log(z|h), \tag{2}$$

where

$$\eta(S_i, \overline{C_w}) = \frac{S_{sense}(i) + 1}{2}$$

$$r(S_i, z) = \frac{\text{cosim}(S_i, z) + 1}{2}.$$

S_{sense} and cosim are in the range [−1,1] so they are transformed to range [0,1] by using linear scaling. This is because these scores are on the same scale as the language model score.

$\eta(S_i, \overline{C_w})$ = linearly scaled sense likelihood score.

$r(S_i, z)$ = linearly scaled cosine similarity score between source language sense and target language word.

where $\eta(S_i, \overline{C_w}) \in [0, 1]$ and $r(S_i, z) \in [0, 1]$, 1 < i < n such that $S_i \in S$.

Denoising Autoencoder

The above steps produce the word-to-word translation with relevance to the context. Even if the words are translated correctly for each position, the sentence is still not properly structured according to the target language. We use a denoising autoencoder, which is a transformer (Vaswani et al. 2017) encoder-decoder, to further improve the translation as per target language rules. The motive behind this is to develop a sequence-to-sequence neural network, which takes a noisy input and outputs a clean, denoised sequence. The model has two submodels that are encoder and decoder. The encoder is responsible for encoding the entire sequence into a vector of fixed length as a context vector. By keeping the context vector into account, the decoder is responsible for making the output steps.

The output label sequence for the denoising network would be the monolingual sentences in the target language. Ideally, the input is a word-to-word translation of the previous step, but such parallel data sets are not available in our case. The input sequences are thus generated by adding noise to the available target monolingual data set to stimulate a cleanup network.

Noise Types

Mainly three types of noises, that is insertion, deletion, and reordering noises, are used as explained below:

- ***Insertion Noise***. In some cases, multiple words of the source language sentence must be translated into a single word in the target language. In other cases, some words are omitted to make the translation fluent. To train the denoising network (Kim, Geng, and Ney 2019), some common words are inserted in the target sentence to form noisy sentences. These new noisy sentences are given as input to the network.
- ***Deletion Noise***. In some cases, the source word must be translated into more than one target word. These nmihgt have to be produced even when there is no corresponding target word. Thus, noisy sentences are

generated with some words deleted or dropped. We remove several terms randomly from a clean target sentence to simulate these situations (Hill, Cho, and Korhonen 2016).

- ***Reordering Noise***. In some cases, the order generated by the word-to-word translation is not in accordance with the grammar of the target language. We add reordering noise (Lample and Denoyer 2017) into a clean sentence by random permutations of the word order. A threshold limits the maximum distance between the two positions before and after the noise.

The word bank is a homonym, as we can observe from the two example sentences given in Figure 5. The word bank has two meanings; the two senses of bank are bank#1 (land alongside a river) and bank#2 (financial establishment called bank). Similarly, we take another example of "book" here: in first sense, book#1 (a written or printed work consisting of pages) is used and in other sense book#2 (reserve a place) is used. Although the above examples demonstrate words with only two senses, there is no restriction on the number of senses a word may have, and some words can have 3 or 4 senses. It will depend upon our vocabulary and word embedding of corpus.

In Figure 5, the first sentence it refers to the land alongside a river and in the second sentence, it refers to a financial establishment called bank. Thus, the two senses of bank are bank#1 (river)and bank#2 (money). For both the sentences, appropriate word sense for each word is identified using its context. After identifying the correct sense, cross-lingual sense embeddings have been used for source-to-target word translations.

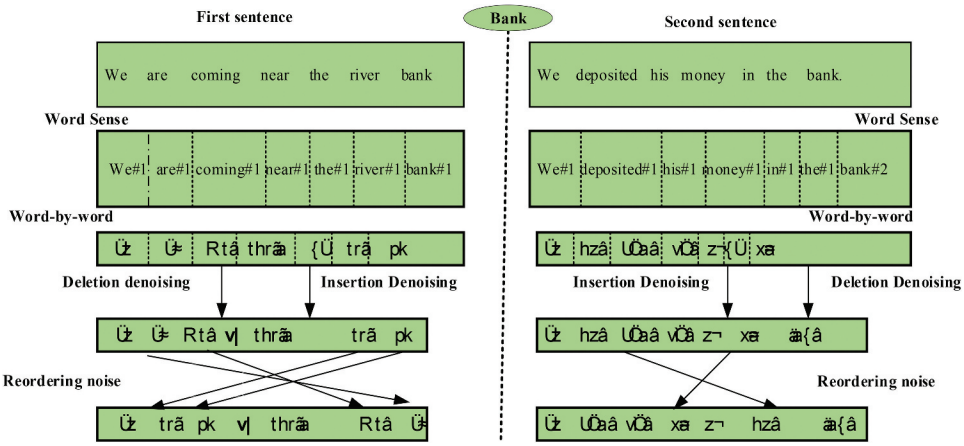

Figure 5. Example of word-by-word translation with denoising an insertion noise, deletion noise, and reordering noise.

After identifying correct sense of a word, the sense word to word translation of English sentence "We are coming near the river bank" to Hindi is "हम हैं आना जदीक यह नदी तट," but the denoised translation " हम नदी तट पर नजदीक आना हैं" is more natural in Hindi. Here, by using insertion denoising "यह," which is an extra target word that might be a translation of some redundant translation, the word is dropped. Similarly, on using deletion denoising, the word "पर " is inserted, which forms a connection of sentence and finally forms a clean target sentence by using reordering denoising word order that is manipulated as "आना हैं" is moved to the end of the sentence and "नदी तट" are shifted to an appropriate position to form a more meaningful translation. The noise is added to the available target monolingual data set to stimulate a cleanup network.

The second sentence has word to word translation of English source sentence " We deposited his money in the bank" in Hindi "हम जमा उसका पैसा में यह बैंक." So, by using insertion denoising, यह, which is an extra target word, is removed and कयिा is inserted using deletion denoising, and by using reordering denoising, बैंक and जमा are rearranged. The final cleaned sentence is "हम उसका पैसा बैंक में जमा कयिा."

Experimental Setup

In the following subsections, we describe the Data Set and Human evaluation, Preprocessing, Training, and Architectural Choice.

Data Set

We have taken a monolingual corpus of AI4Bharat (Kunchukuttan et al. 2020) and IIT Bombay (Kunchukuttan, Mehta, and Bhattacharyya 2017), IMDB (Maas et al. 2011), and WMT (Post, Callison-Burch, and Osborne 2012) to generate CLWE and CLSWE for Hindi and English. For Hindi, there are 62,961,411 sentences and 1,01,882,012 tokens, and for English, we have 50,22,111 sentences and 28,122,199 tokens are used for training. The test data set with 1000 sentences have been taken, out of which 500 sentences were from the WMT test data set and another 500 sentences from IIT Bombay corpus. For the human evaluation score, 50 native speakers manually performed the evaluation. We have a team that daily took the evaluation of 50 human speakers who evaluated the sentences in two months. Each evaluator has given rating from zero to five of translated sentences, with 0 being worse, 1 poor, 2 a medium, 3 average, 4 good, and 5 excellent. Each translator had a different perspective regarding the quality of translation. The reference translations used for automatic evaluation were kept hidden from the human translator. This ensures that the human judgment is not biased toward a single reference sentence.

Preprocessing

Preprocessing is a significant step for any machine translation task. The quality of word embedding directly depends upon the corpus provided. Sentences are preprocessed using tokenization, and any invalid tokens or tokens of other languages are removed. Texts written in Hindi have other challenges due to varying input methods, multiple representations for the same characters, etc. Thus, the Hindi data set is normalized so that text can be handled in a consistent manner.

Training and Architectural Choice

Word embedding can be prepared using simple skip-gram, CBOW, and Fast text methods. But to handle the vocabulary problem, we used FastText with a skip-gram model (Joulin et al. 2016) for the generation of word vectors. The training parameters for word embedding are as follows:

- Number of epochs is 10.
- Learning rate is 0.05.
- Number of active threads is 12.
- Word Embedding dimension is 300.

For word sense induction, we retrieve 200 nearest neighbors for each word to form its ego network. The maximum number of connections that a node is allowed to have within the network is set as 100. The minimum cluster size is kept as 5.

For learning a linear transformation, which can convert source sense embedding space into target word embedding, we are applying the robust self-learning method (Artetxe, Labaka, and Agirre 2018). We have generated mapped source and target word embeddings, which has the related words as the nearest neighbor. The source word embedding, which has maximum similarity with the target word embedding, will be taken by using four different retrieval techniques.

We have taken KenLM (Heafield 2011) LM implementation with its default settings. For denoising purposes, an autoencoder was trained with a sockeye toolkit (Hieber et al. 2017) on monolingual data sets. The encoder and decoder are composed of a stack of six layers with encoder layers. The encoder layers consist of two other sublayers called a multihead self-attention mechanism and a position-wise feed-forward fully connected network. Each decoder layer composed of three sublayers, two of which are the same as of the encoder, and the third layer performs multihead attention to the output of each encoder stack. The illustration of the denoising autoencoder network is shown in Figure 6.

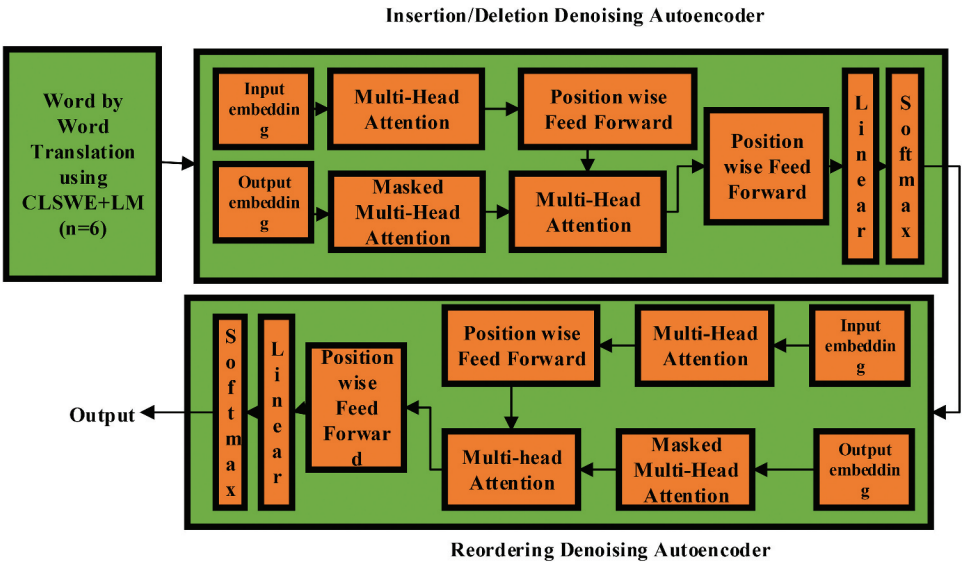

Figure 6. Illustration of the denoising autoencoder networks.

Table 1. BLEU Score for different retrieval methods and CLSWE.

Retrieval Method	English-Hindi BLEU	Hindi-English BLEU
Nearest Neighbor	14.26	14.11
Inverted Nearest Neighbor	15.39	16.26
Inverted SoftMax	19.85	20.85
Cross-Lingual Word Scaling	19.01	18.01

Table 1 analyzes the impact of using different retrieval techniques for obtaining the word translation from CLSWE. We observe that inverted softmax outperforms the three techniques with an improvement of 5.59 over nearest neighbor retrieval, followed by cross-lingual word scaling (+4.75) and inverted nearest neighbor (+1.13) in the case of English to Hindi translation. Similarly, for Hindi to English translation, inverted softmax has the highest score of 20.85.

Table 2. Evaluation scores for English-Hindi.

	English-Hindi					
System	BLEU	METEOR	METEOR-HINDI	TER	NIST	ROUGE
Word-by-word (W)	10.98	5.88	13.58	2.54	3.21	3.01
Word- by- word+ Language Model Sense (WLS)	19.85	9.70	21.41	5.65	8.76	5.01
Word- by- word+ Language Model Sense +Insertion +Deletion (WLSID)	21.99	10.23	22.90	6.82	9.01	6.78
Word- by- word+ Language Model Sense +Insertion +Deletion + Reordering (WLSIDR)	22.01	10.41	23.90	6.89	9.43	7.29
(Baseline System)	7.01	2.10	10.28	0.24	1.01	0.29
Word translation	17.20	7.30	18.01	4.87	5.01	3.43
+LM	19.26	8.01	20.91	5.40	7.98	5.99
+Denoising						

- n =6, where n is the n-gram count base language model for the target language, and it is found to be the most optimal value.
- We used a transformer encoder/decoder with six layers (Vaswani et al. 2017).
- Adam m (Kingma and Ba 2014) was used for the optimization of the denoising autoencoder model with an initial learning rate set as 0.0001.
- The Gated Linear Unit (Glu) was used as the activation function for every convolution layer.
- The number of hidden units in the transformer model is set as 256.
- The batch size is set as 3000 samples for the training process.
- For decoding, we used $\lambda_{sense\_embedding} = 1$ and $\lambda_{language\ model} = 0.1$ with the beam size set as 10. For each position, 30 nearest target words were considered.
- Total training time for denoising on the Titan XP GPU for 14 hours.

Results and Discussion

Section 6.1 presents the result of word translation by using different retrieval techniques. Section 6.2 shows the evaluation score for English and Hindi. Sample sentences are discussed in section 6.3.

Word Translation Using Different Retrieval Techniques

Machine Translation Evaluation Score

In this section, BLEU, METEOR, METEOR-HINDI, TER, NIST, and ROUGE metrics have been used for MT evaluation for English and Hindi.

English-Hindi

Table 2 shows the sentence translation results using all evaluation metrics for the English-Hindi test data sets. As we can see from the results, it increases consistently in all four steps from word-by-word (W) baselines, giving + 12% BLEU for English-Hindi. When the language model and word sense identification (WLS) are applied, then there is an increase in + 9% BLEU of translation for English-Hindi. If the denoising model (insertion and deletion) (WLSID) is implemented additionally, we have an additional benefit of around + 2% BLEU from the previous state. Again, when we applied reordering noise (WLSIDR), there was a + 1% improvement in the BLEU score. The step-to-step comparison is provided for both proposed and baseline systems.

Our methods utilize noise-free source side context for identifying the correct word sense and outperform (Kim, Geng, and Ney 2019) that is dependent only on the target side context by up to +3% BLEU.

METEOR is dependent on the unigram matching of the machine and human-produced reference translation. As we can see in Table 2, the METEOR score is 6.41% for English-Hindi and Meteor scores show an improvement of +2% over the baseline paper. Here is to be mentioned that BLEU is not suitable for morphologically rich language like Hindi (Ananthakrishnan et al.,2007; Sulem, Abend, and Rappoport 2018). METEOR-Hindi is a modified version of METEOR, containing features specific to Hindi, and as we can see from Table 2, evaluation scores for English-Hindi show an improvement of +3% over the baseline approach. As we can observe, METEOR-HINDI gives better evaluation results as compared to BLEU when the target language is Hindi. TER, NIST, and ROUGE also show around 1% improvement over the baseline approach.

Figure 6 shows the comparison of the proposed work and baseline stem using different evaluation metrics. Our proposed work is + 3% for BLEU, $_+$ 2% for METEOR, +3% for METEOR-Hindi, + 1% for TER, + 2% for NIST, and + 2% for ROUGE, which are higher for the baseline system.

Hindi-English

Table 3 shows the sentence translation results for English to Hindi using different evaluation metrics. The results show that our proposed work increases consistently in all four steps from word-by-word baselines, giving + 10% BLEU for Hindi-English. There is + 7 point BLEU score when word-by-word translation and language model sense are applied. After applying the denoising model, there is again + 2 point improvement and again + 1 more increment when reordering noise is applied. It outperforms from the baseline system in each step of around + 3 point BLEU and + 2 point for METEOR, NIST, TER, and ROUGE.

Figure 7 and Figure 8 shows the comparison of the proposed approach with the baseline paper. There is almost +3 point BLEU improvement in the proposed approach and +2 point for METEOR, NIST, and ROUGE. The results for different evaluation metrics show that by incorporating the word

Table 3. Evaluation scores for English-Hindi.

System	Hindi-English				
	BLEU	METEOR	TER	NIST	ROUGE
Word-by-word (W)	13.61	8.76	1.21	4.72	2.20
Word-by- word+ Language Model Sense (WLS)	20.84	14.05	4.15	6.56	4.90
Word-by- word+ Language Model Sense +Insertion +Deletion (WLSID)	23.03	15.01	5.21	7.46	6.00
Word-by- word+ Language Model Sense +Insertion +Deletion + Reordering (WLSIDR)	23.95	15.55	5.80	7.95	6.19
(Baseline System)	10.10	4.10	0.12	2.12	0.45
Word translation	17.98	10.21	3.12	4.98	3.13
+LM	20.43	13.67	4.43	5.98	4.90
+Denoising					

Table 4. Examples of unsupervised sentence translations (English-Hindi).

Source sentence 1		The nurse is very kind and polite.
Word sense		The #1 nurse #3 is #1 very #1 kind #3 and #1 polite #2
Translated sentence	W	यह नर स है बह त या र निम र
	WLS	नर स है बह त या र निम र यह
	WLSID	नर स है बह त या र निम र
	WLSIDR	नर स बह त या र निम र है
Reference sentence	नर स बह त या र निम र है	
Source sentence 2		The dog is trained to bark at strangers
Word sense		The#1 dog#1 is#1 trained#2 to#1 bark#2 at#1 strangers#1
Translated sentence	W	यह क त ता है प र कि ति के कना पर जनवयि ं
	WLS	यह क त ता है कना पर जनवयि ं प र कि ति
	WLSID	क त ता है कना पर जनवयि ं प र कि ति कयिा
	WLSIDR	क त ते जनवयि ं पर ं कने के ि प र कि ति कयिा है
Reference sentence	क त ते क जनवयि ं क ं कने के ि प र कि ति कयिा जाता है	
Source sentence 3		I want to quit my current job.
Word sense		I#1 want#1 to#1 quit#1 my#1 current#1 job#2.
Translated sentence	W	मैं ाहता ह ं ना मेरी र तमान न करी
	WLS	मैं ना ाहता ह ं मेरी र तमान न करी
	WLSID	मैं ना ाहता ह ं मेरी र तमान न करी
	WLSIDR	मैं मेरी र तमान न करी ना ाहता ह ं
Reference sentence	मैं पनी र तमान न करी ोड ना ाहता हूं	
Source sentence 4		I have a bat for playing.
Word sense		I#1 have#1 a#1 bat#2 for#1 playing#2
Translated sentence	W	मैं पास क ब ा के ं ने
	WLS	मैं पास के ं ने क ब ा
	WLSID	मैं पास के ं ने के ि क ब ा
	WLSIDR	मैं पास ं ने के ि क ब ा
Reference sentence	मेरे पास ं ने के ि क ब ा है	
Source sentence 5		The internet is flooded with news stories and gossips.
Word sense		The#1 internet#1 is#1 flooded#2 with#1 news#1 stories#1 and#1 gossips#1.
Translated sentence	W	यह ं टरनेट है रा सा समा ार ं कहानयि ं र सपि
	WLS	यह ं टरनेट है रा सा समा ार ं कहानयि ं र सपि
	WLSID	ं टरनेट है से रा सा समा ार ं कहानयि ं र सपि
	WLSIDR	ं टरनेट समा ार ं कहानयि ं र सपि सा से रा है
Reference sentence	ं टरनेट समा ार ं र सपि से र या है	

Table 4 shows English-Hindi examples of fully unsupervised sentence translations. For each word in the source sentence, its appropriate sense is identified considering the context of that word. The word-to-word translated sentences of the all source sentences are produced using cross-lingual sense embeddings, which are further improved with the application of the language model. The sentence is further improved by applying insertion and deletion denoising autoencoder. Finally, on implementing the reordering, we get a more meaningful word order in the translated sentence.

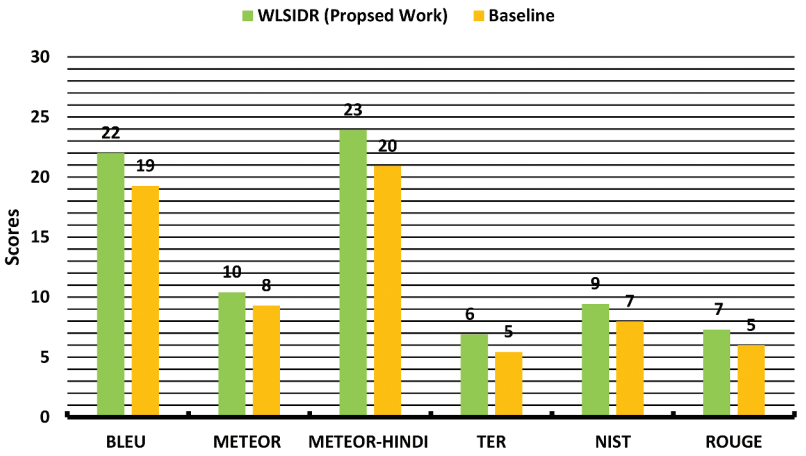

Figure 7. Evaluation Score for English-Hindi for different evaluation metrics.

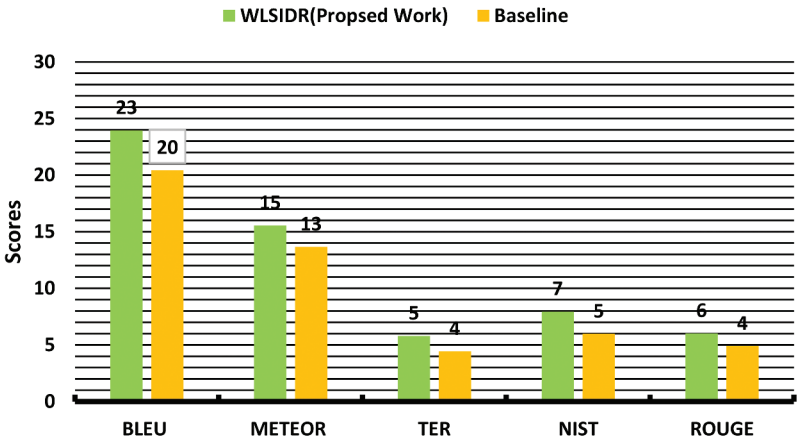

Figure 8. The comparison of the proposed approach with the baseline paper. There is almost +3 point BLEU improvement in the proposed approach and +2 point for METEOR, NIST, and ROUGE. The results for different evaluation metrics show that by incorporating the word sense disambiguation to the source language context, the sense of a word is selected more appropriately and provides improved results from the baseline approach.

sense disambiguation to the source language context, the sense of a word is selected more appropriately and provides improved results from the baseline approach.

Conclusion

In this paper, we have presented a simple pipeline to greatly improve sentence translation especially for morphologically rich ,languages which is based on CLSWE and LM. The integration of CLSWE and LM utilizes source and target

language context optimally and shows an improvement in context-aware word translation. WSD has been incorporated successfully in this approach to resolve the problem of polysemous words in morphologically rich languages. Furthermore, insertion, deletion, and reordering problems are tackled using denoising autoencoder. The experiments show the effectiveness of our proposal, obtaining significant improvements in the BLEU and METEOR- Hindi score up to + 3 points over a baseline system.

In the future, we would like to explore other neighborhood functions for denoising and analyze their effect in relation to the typological divergences of different language pairs. Second, the translation quality can further be improved by utilizing the higher quality word embedding, such as the recently proposed BERT, which are proved to be powerful and promising. Finally, we would further like to extend this proposed approach to other morphologically rich and low-resource languages.

Disclosure Statement

The author does not have any conflict of interest.

ORCID

Shweta Chauhan http://orcid.org/0000-0002-6598-1992
Philemon Daniel http://orcid.org/0000-0002-7133-9488
Shefali Saxena http://orcid.org/0000-0001-7590-7940

References

Aliwy, A. H., and H. A. Taher. 2019. Word sense disambiguation: Survey study. *Journal of Computer Science* 15 (7):1004–11. doi:10.3844/jcssp.2019.1004.1011.

Ananthakrishnan, R., P. Bhattacharyya, M. Sasikumar, and R. M. Shah,2007. Some issues in automatic evaluation of English-Hindi MT: More blues for bleu. In Proceedings of the ICON, IIT Bombay, India, 1–8.

Artetxe, M., G. Labaka, and E. Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia.

Artetxe, M., G. Labaka, E. Agirre, and K. Cho. 2017. Unsupervised neural machine translation. Paper preseneted at the 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada. April 30 - May 3.

Azarbonyad, H., A. Shakery, and H. Faili. 2019. A learning to rank approach for cross-language information retrieval exploiting multiple translation resources. *Natural Language Engineering* 25 (3):363–84. doi:10.1017/S1351324919000032.

Brown, T. B., B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, . . . D. Amodei. 2020. Language models are few-shot learners. *arXiv Preprint arXiv* 2005.14165.

Carbonell, J. G., S. Klein, D. Miller, M. Steinbaum, T. Grassiany, and J. Frey. 2006. Context-based machine translation. In Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Cambridge, Massachusetts, USA.

Chauhan, S., S. Saxena, and P. Daniel. 2021a. Fully unsupervised word translation from cross-lingual word embeddings especially for healthcare professionals. *International Journal of System Assurance Engineering and Management* 12:1–10.

Conneau, A., G. Lample, M. A. Ranzato, L. Denoyer, and H. Jégou. 2017. Word translation without parallel data. Paper presented at the 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada. April 30 - May 3.

Devlin, J., M. W. Chang, K. Lee, and K. Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7.

Dinu, G., A. Lazaridou, and M. Baroni. 2014. Improving zero-shot learning by mitigating the hubness problem. International Conference on Learning Representations, Workshop Track, The Hilton San Diego.

Doddington, G., 2002, March. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In Proceedings of the second international conference on Human Language Technology Research, San Diego California. (pp. 138–45).

Ettinger, A. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics* 8:34–48. doi:10.1162/tacl_a_00298.

Gupta, A., S. Venkatapathy, and R. Sangal, 2010. METEOR-Hindi: Automatic MT evaluation metric for Hindi as a target language. In Proceeding of the International Conference on Natural Language Processing Language. IIT Kharagpur, India, 1–10.

Heafield, K. 2011. KenLM: Faster and smaller language model queries. In Proceedings of the sixth workshop on statistical machine translation, Edinburgh, Scotland. (pp. 187–97).

Hieber, F., T. Domhan, M. Denkowski, D. Vilar, A. Sokolov, A. Clifton, and M. Post. 2017. Sockeye: A toolkit for neural machine translation. *arXiv Preprint arXiv* 1712.05690.

Hill, F., K. Cho, and A. Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016), 1367–77, San Diego, CA, USA.

Ilić, S., E. Marrese-Taylor, J. A. Balazs, and Y. Matsuo. 2018. Deep contextualized word representations for detecting sarcasm and irony. In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Brussels, Belgium, October 31. (pp. 2–7).

Irvine, A., and C. Callison-Burch. 2017. A comprehensive analysis of bilingual lexicon induction. *Computational Linguistics* 43 (2):273–310. doi:10.1162/COLI_a_00284.

Joulin, A., E. Grave, P. Bojanowski, and T. Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv Preprint arXiv* 1607.01759.

Kim, Y., J. Geng, and H. Ney. 2019. Improving unsupervised word-by-word translation with language model and denoising autoencoder. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium. October 31 - November 4. (pp. 862–868).

Kingma, D. P., and J. Ba. 2014. Adam: A method for stochastic optimization. Paper presented at the 3rd International Conference on Learning Representations, {ICLR} 2015, San Diego, CA, USA, May 7-9.

Klementiev, A., A. Irvine, C. Callison-Burch, and D. Yarowsky. 2012. Toward statistical machine translation without parallel corpora. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France. (pp. 130–40).

Kodge, S., and K. Roy. 2021. BERMo: What can BERT learn from ELMo? *arXiv Preprint arXiv* 2110.15802.

Koehn, P., and R. Knowles. 2017. Six challenges for neural machine translation. In Proceedings of the First Workshop on Neural Machine Translation, Vancouver. (pp. 28–39).

Kunchukuttan, A., D. Kakwani, S. Golla, A. Bhattacharyya, M. M. Khapra, and P. Kumar. 2020. AI4Bharat-IndicNLP Corpus: Monolingual corpora and word embeddings for indic languages. *arXiv Preprint arXiv* 2005.00085.

Kunchukuttan, A., P. Mehta, and P. Bhattacharyya. 2017. The iit bombay English-hindi parallel corpus. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan.

Lample, G., and L. M. R. Denoyer. 2017. Unsupervised machine translation using monolingual corpora only. In 6th International Conference on Learning Representations, {ICLR} 2018, Vancouver, BC, Canada, April 30 - May 3.

Lewis, M., Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, and L. Zettlemoyer. 2019. Bart: Denoising sequence-to-
sequence pre-training for natural language generation, translation, and comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL. (pp. 7871–7880).

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. Text Summarization Branches Out. Association for Computational Linguistics, Barcelona, Spain. (pp. 74–81).

Maas, A. L., R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts 2011, June. Learning word vectors for sentiment analysis. In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1, Portland, Oregon, USA. (pp. 142–50). Association for Computational Linguistics.

Papineni, K., S. Roukos, T. Ward, and W. J. Zhu, 2002, July. BLEU: A method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics, Philadelphia, Pennsylvania, USA. (pp. 311–18). Association for Computational Linguistics.

Pelevina, M., N. Arefyev, C. Biemann, and A. Panchenko. 2017. Making sense of word embeddings. In Proceedings of the 1st Workshop on Representation Learning for NLP, Berlin, Germany.

Post, M., C. Callison-
Burch, and M. Osborne, 2012, June. Constructing parallel corpora for six indian languages via crowdsourcing. In Proceedings of the Seventh Workshop on Statistical Machine Translation, Montréal, Canada. (pp. 401–09). Association for Computational Linguistics.

Raffel, C., N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, , and P. J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21 (140):1–67. .

Ravi, S., and K. Knight, 2011, June. Deciphering foreign language. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, Oregon, USA. (pp. 12–21).

Sennrich, R., B. Haddow, and A. Birch. 2015. Improving neural machine translation models with monolingual data. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics Volume 1, Berlin, Germany. (pp. 86–96).

Smith, S. L., D. H. Turban, S. Hamblin, and N. Y. Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26.

Snover, M., N. Madnani, B. J. Dorr, and R. Schwartz, 2009, March. Fluency, adequacy, or HTER?: Exploring different human judgments with a tunable MT metric. In Proceedings of the Fourth Workshop on Statistical Machine Translation, Athens, Greece. (pp. 259–68). Association for Computational Linguistics.

Su, K. Y., M. W. Wu, and J. S. Chang, 1992. A new quantitative quality measure for machine translation systems. In Proceedings of the 14th conference on Computational linguistics-Volume 2, Nantes, France. (pp. 433–39). Association for Computational Linguistics.

Sulem, E., O. Abend, and A. Rappoport. 2018. Bleu is not suitable for the evaluation of text simplification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium. (pp. 738–744).

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, and I. Polosukhin. 2017. Attention is all you need. Advances in Neural Information Processing Systems. In 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA. (pp. 5998–6008).

Wang, A., A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium. (pp. 353–355).