Graph Propagation for Paraphrasing Out-of-Vocabulary Words in Statistical Machine Translation*

Majid Razmara[1] **Maryam Siahbani**[1] **Gholamreza Haffari**[2] **Anoop Sarkar**[1]

[1] Simon Fraser University, Burnaby, BC, Canada
{razmara,msiahban,anoop}@sfu.ca
[2] Monash University, Clayton, VIC, Australia
reza@monash.edu

Abstract

Out-of-vocabulary (oov) words or phrases still remain a challenge in statistical machine translation especially when a limited amount of parallel text is available for training or when there is a domain shift from training data to test data. In this paper, we propose a novel approach to finding translations for oov words. We induce a lexicon by constructing a graph on source language monolingual text and employ a graph propagation technique in order to find translations for all the source language phrases. Our method differs from previous approaches by adopting a graph propagation approach that takes into account not only one-step (from oov directly to a source language phrase that has a translation) but multi-step paraphrases from oov source language words to other source language phrases and eventually to target language translations. Experimental results show that our graph propagation method significantly improves performance over two strong baselines under intrinsic and extrinsic evaluation metrics.

1 Introduction

Out-of-vocabulary (oov) words or phrases still remain a challenge in statistical machine translation. SMT systems usually copy unknown words verbatim to the target language output. Although this is helpful in translating a small fraction of oovs such as named entities for languages with same writing systems, it harms the translation in other types of oovs and distant language pairs. In general, copied-over oovs are a hindrance to fluent, high quality translation, and we can see evidence of this in automatic measures such as BLEU (Papineni et al., 2002) and also in human evaluation scores such as HTER. The problem becomes more severe when only a limited amount of parallel text is available for training or when the training and test data are from different domains. Even noisy translation of oovs can aid the language model to better re-order the words in the target language (Zhang et al., 2012).

Increasing the size of the parallel data can reduce the number of oovs. However, there will always be some words or phrases that are new to the system and finding ways to translate such words or phrases will be beneficial to the system. Researchers have applied a number of approaches to tackle this problem. Some approaches use pivot languages (Callison-Burch et al., 2006) while others use lexicon-induction-based approaches from source language monolingual corpora (Koehn and Knight, 2002; Garera et al., 2009; Marton et al., 2009).

Pivot language techniques tackle this problem by taking advantage of available parallel data between the source language and a third language. Using a pivot language, oovs are translated into a third language and back into the source language and thereby paraphrases to those oov words are extracted (Callison-Burch et al., 2006). For each oov, the system can be augmented by aggregating the translations of all its paraphrases and assign them to the oov. However, these methods require parallel corpora between the source language and one or multiple pivot languages.

Another line of work exploits spelling and morphological variants of oov words. Habash (2008) presents techniques for online handling of oov words for Arabic to English such as spelling expansion and morphological expansion. Huang et al. (2011) proposes a method to combine sub-lexical/constituent translations of an oov word or phrase to generate its translations.

Several researchers have applied lexicon-induction methods to create a bilingual lexicon for those oovs. Marton et al. (2009) use a monolingual text on the source side to find paraphrases to oov words for which the translations are available. The translations for these paraphrases are

*This research was partially supported by an NSERC, Canada (RGPIN: 264905) grant. The third author was supported by an early career research award from Monash University to visit Simon Fraser University.

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 1105–1115,
Sofia, Bulgaria, August 4-9 2013.

then used as the translations of the oov word. These methods are based on the *distributional hypothesis* which states that words appearing in the same contexts tend to have similar meaning (Harris, 1954). Marton et al. (2009) showed that this method improves over the baseline system where oovs are untranslated.

We propose a graph propagation-based extension to the approach of Marton et al. (2009) in which a graph is constructed from source language monolingual text[1] and the source-side of the available parallel data. Nodes that have related meanings are connected together and nodes for which we have translations in the phrase-table are annotated with target-side translations and their feature values. A graph propagation algorithm is then used to propagate translations from labeled nodes to unlabeled nodes (phrases appearing only in the monolingual text and oovs). This provides a general purpose approach to handle several types of oovs, including morphological variants, spelling variants and synonyms[2].

Constructing such a huge graph and propagating messages through it pose severe computational challenges. Throughout the paper, we will see how these challenges are dealt with using scalable algorithms.

2 Collocational Lexicon Induction

Rapp (1995) introduced the notion of a distributional profile in bilingual lexicon induction from monolingual data. A *distributional profile* (DP) of a word or phrase type is a co-occurrence vector created by combining all co-occurrence vectors of the tokens of that phrase type. Each distributional profile can be seen as a point in a $|V|$-dimensional space where V is the vocabulary where each word type represents a unique axis. Points (i.e. phrase types) that are close to one another in this high-dimensional space can represent paraphrases. This approach has also been used in machine translation to find in-vocabulary paraphrases for oov words on the source side and find a way to translate them.

2.1 Baseline System

Marton et al. (2009) was the first to successfully integrate a collocational approach to finding translations for oov words into an end-to-end SMT system. We explain their method in detail as we will compare against this approach. The method relies on monolingual distributional profiles (DPs) which are numerical vectors representing the context around each word. The goal is to find words or phrases that appear in similar contexts as the oovs. For each oov a distributional profile is created by collecting all words appearing in a fixed distance from all occurrences of the oov word in the monolingual text. These co-occurrence counts are converted to an association measure (Section 2.2) that encodes the relatedness of each pair of words or phrases.

Then, the most similar phrases to each oov are found by measuring the similarity of their DPs to that of the oov word. Marton et al. (2009) uses a heuristic to prune the search space for finding candidate paraphrases by keeping the surrounding context (e.g. $L\_\_R$) of each occurrences of the oov word. All phrases that appear in any of such contexts are collected as candidate paraphrases. For each of these paraphrases, a DP is constructed and compared to that of the oov word using a similarity measure (Section 2.2).

The top-k paraphrases that have translations in the phrase-table are used to assign translations and scores to each oov word by marginalizing translations over paraphrases:

$$p(t|o) = \sum_s p(t|s)p(s|o)$$

where t is a phrase on the target side, o is the oov word or phrase, and s is a paraphrase of o. $p(s|o)$ is estimated using a similarity measure over DPs and $p(t|s)$ is coming from the phrase-table.

We reimplemented this collocational approach for finding translations for oovs and used it as a baseline system.

Alternative ways of modeling and comparing distributional profiles have been proposed (Rapp, 1999; Fung and Yee, 1998; Terra and Clarke, 2003; Garera et al., 2009; Marton et al., 2009). We review some of them here and compare their performance in Section 4.3.

2.2 Association Measures

Given a word u, its distributional profile $DP(u)$ is constructed by counting surrounding words (in a fixed window size) in a monolingual corpus.

$$DP(u) = \{\langle A(u, w_i)\rangle \mid w_i \in V\}$$

[1] Here on by monolingual data we always mean monolingual data on the source language

[2] Named entity oovs may be handled properly by copying or transliteration.

The counts can be collected in positional[3] (Rapp, 1999) or non-positional way (count all the word occurrences within the sliding window). $A(\cdot,\cdot)$ is an association measure and can simply be defined as co-occurrence counts within sliding windows. Stronger association measures can also be used such as:

Conditional probability: the probability for the occurrence of each word in DP given the occurrence of u: $\text{CP}(u, w_i) = P(w_i|u)$ (Schütze and Pedersen, 1997)

Pointwise Mutual Information: this measure is a transformation of the independence assumption into a ratio. Positive values indicate that words co-occur more than what we expect under the independence assumption (Lin, 1998):

$$\text{PMI}(u, w_i) = log_2 \frac{P(u, w_i)}{P(u)P(w_i)}$$

Likelihood ratio: (Dunning, 1993) uses the likelihood ratio for word similarity:

$$\lambda(u, w_i) = \frac{L(P(w_i|u); p) * L(P(w_i|\neg u); p)}{L(P(w_i|u); p_1) * L(P(w_i|\neg u); p_2)}$$

where L is likelihood function under the assumption that word counts in text have binomial distributions. The numerator represents the likelihood of the hypothesis that u and w_i are independent ($P(w_i|u) = P(w_i|\neg u) = p$) and the denominator represents the likelihood of the hypothesis that u and w_i are dependent ($P(w_i|u) \neq P(w_i|\neg u)$, $P(w_i|u) = p_1$, $P(w_i|\neg u) = p_2$)[4].

Chi-square test: is a statistical hypothesis testing method to evaluate independence of two categorical random variables, e.g. whether the *occurrence* of u and w_i (denoted by x and y respectively) are independent. The test statistics $\chi^2(u, w_i)$ is the deviation of the observed counts $f_{x,y}$ from their expected values $E_{x,y}$:

$$\chi^2(u, w_i) := \sum_{x \in \{w_i, \neg w_i\}} \sum_{y \in \{u, \neg u\}} \frac{(f_{x,y} - E_{x,y})^2}{E_{x,y}}$$

2.3 Similarity Measures

Various functions have been used to estimate the similarity between distributional profiles. Given two distributional profiles $DP(u)$ and $DP(v)$, some similarity functions can be defined as follows. Note that $A(\cdot,\cdot)$ stands for the various association measures defined in Sec. 2.2.

Cosine coefficient is the cosine the angle between two vectors $DP(u)$ and $DP(v)$:

$$cos(DP(u), DP(v)) = \frac{\sum_{w_i \in V} A(u, w_i)A(v, w_i)}{\sqrt{\sum_{w_i \in V} A(u, w_i)^2}\sqrt{\sum_{w_i \in V} A(v, w_i)^2}}$$

L_1-Norm computes the accumulated distance between entries of two distributional profiles ($L_1(\cdot,\cdot)$). It has been used as word similarity measure in language modeling (Dagan et al., 1999).

$$L_1(DP(u), DP(v)) = \sum_{w_i \in V} |A(u, w_i) - A(v, w_i)|$$

Jensen-Shannon Divergence is a symmetric version of *contextual average mutual information* (KL) which is used by (Dagan et al., 1999) as word similarity measure.

$$JSD(DP(u), DP(v)) = KL(DP(u), AVG_{DP}(u, v)) + KL(DP(v), AVG_{DP}(u, v))$$

$$AVG_{DP}(u, v) = \left\{ \frac{A(u, w_i) + A(v, w_i)}{2} \mid w_i \in V \right\}$$

$$KL(DP(u), DP(v)) = \sum_{w_i \in V} A(u, w_i) log \frac{A(u, w_i)}{A(v, w_i)}$$

3 Graph-based Lexicon Induction

We propose a novel approach to alleviate the oov problem. Given a (possibly small amount of) parallel data between the source and target languages, and a large monolingual data in the source language, we construct a graph over all phrase types in the monolingual text and the source side of the parallel corpus and connect phrases that have similar meanings (i.e. appear in similar context) to one another. To do so, the distributional profiles of all source phrase types are created. Each phrase type represents a vertex in the graph and is connected to other vertices with a weight defined by a similarity measure between the two profiles (Section 2.3). There are three types of vertices in the graph: i) labeled nodes which appear in the parallel corpus and for which we have the target-side

[3] e.g., position 1 is the word immediately after, position -1 is the word immediately before etc.

[4] Binomial distribution $B(k; n, \theta)$ gives the probability of observing k heads in n tosses of a coin where the coin parameter is θ. In our context, p, p_1 and p_2 are parameters of Binomial distributions estimated using maximum likelihood.

translations[5]; ii) oov nodes from the *dev/test set* for which we seek labels (translations); and iii) unlabeled nodes (words or phrases) from the *monolingual data* which appear usually between oov nodes and labeled nodes. When a relatively small parallel data is used, unlabeled nodes outnumber labeled ones and many of them lie on the paths between an oov node to labeled ones.

Marton et al. (2009)'s approach ignores these bridging nodes and connects each oov node to the k-nearest *labeled* nodes. One may argue that these unlabeled nodes do not play a major role in the graph and the labels will eventually get to the oov nodes from the labeled nodes by directly connecting them. However based on the definition of the similarity measures using context, it is quite possible that an oov node and a labeled node which are connected to the same unlabeled node do not share any context words and hence are not directly connected. For instance, consider three nodes, u (unlabeled), o (oov) and l (labeled) where u has the same left context words with o but share the right context with l. o and l are not connected since they do not share any context word.

Once a graph is constructed based on similarities of phrases, graph propagation is used to propagate the labels from labeled nodes to unlabeled and oov nodes. The approach is based on the *smoothness assumption* (Chapelle et al., 2006) which states if two nodes are similar according to the graph, then their output labels should also be similar.

The baseline approach (Marton et al., 2009) can be formulated as a *bipartite graph* with two types of nodes: labeled nodes (L) and oov nodes (O). Each oov node is connected to a number of labeled nodes, and vice versa and there is no edge between nodes of the same type. In such a graph, the similarity of each pair of nodes is computed using one of the similarity measures discussed above. The labels are translations and their probabilities (more specifically $p(e|f)$) from the phrase-table extracted from the parallel corpus. Translations get propagated to oov nodes using a label propagation technique. However beside the difference in the oov label assignment, there is a major difference between our bipartite graph and the baseline (Marton et al., 2009): we do not use a heuristic to reduce the number of neighbor candidates and we consider all possible candidates that share at least one context word. This makes a significant difference in practice as shown in Section 4.3.1.

We also take advantage of unlabeled nodes to help connect oov nodes to labeled ones. The discussed bipartite graph can easily be expanded to a *tripartite graph* by adding unlabeled nodes. Figure 1 illustrate a tripartite graph in which unlabeled nodes are connected to both labeled and oov nodes. Again, there is no edge between nodes of the same type. We also created the *full graph* where all nodes can be freely connected to nodes of any type including the same type. However, constructing such graph and doing graph propagation on it is computationally very expensive for large n-grams.

3.1 Label Propagation

Let $G = (V, E, W)$ be a graph where V is the set of vertices, E is the set of edges, and W is the edge weight matrix. The vertex set V consists of labeled V_L and unlabeled V_U nodes, and the goal of the labeling propagation algorithm is to compute soft labels for unlabeled vertices from the labeled vertices. Intuitively, the edge weight $W(u, v)$ encodes the degree of our belief about the similarity of the soft labeling for nodes u and v. A soft label $\hat{Y}_v \in \Delta^{m+1}$ is a probability vector in $(m+1)$-dimensional simplex, where m is the number of possible labels and the additional dimension accounts for the *undefined* $\perp$ label[6].

In this paper, we make use of the *modified Adsorption* (MAD) algorithm (Talukdar and Crammer, 2009) which finds soft label vectors $\hat{Y}_v$ to solve the following unconstrained optimization problem:

$$\min_{\hat{Y}} \quad \mu_1 \sum_{v \in V_L} p_{1,v} ||Y_v - \hat{Y}_v||_2^2 + \tag{1}$$

$$\mu_2 \sum_{v,u} p_{2,v} W_{v,u} ||\hat{Y}_v - \hat{Y}_u||_2^2 + \tag{2}$$

$$\mu_3 \sum_{v} p_{3,v} ||\hat{Y}_v - R_v||_2^2 \tag{3}$$

where μ_i and $p_{i,v}$ are hyper-parameters ($\forall v$: $\sum_i p_{i,v} = 1$)[7], and $R_v \in \Delta^{m+1}$ encodes our prior belief about the labeling of a node v. The first

[5] It is possible that a phrase appears in the parallel corpus, but not in the phrase-table. This happens when the word-alignment module is not able to align the phrase to a target side word or words.

[6] Capturing those cases where the given data is not enough to reliably compute a soft labeling using the initial m *real* labels.

[7] The values of these hyper-parameters are set to their defaults in the *Junto* toolkit (Talukdar and Crammer, 2009).

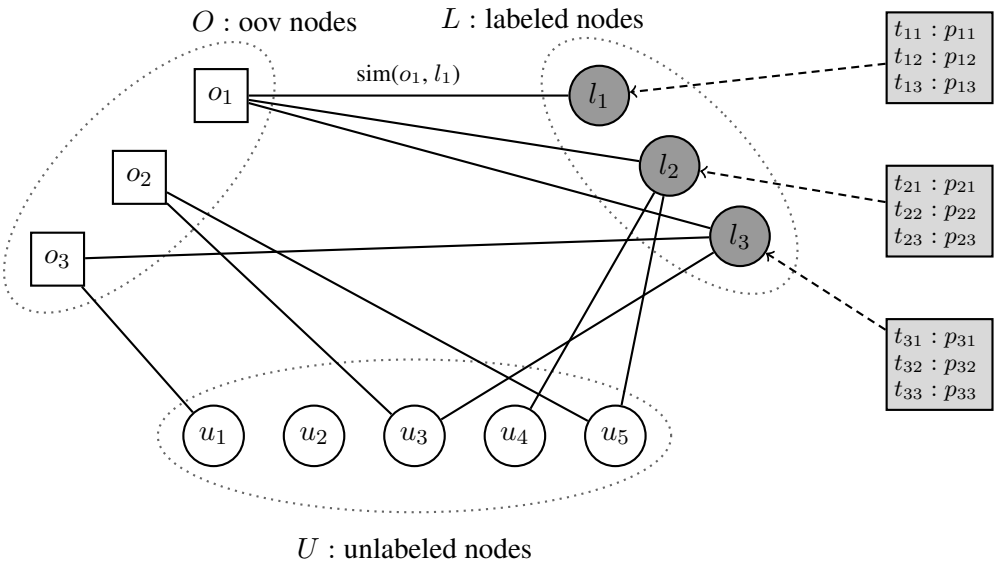

Figure 1: A tripartite graph between oov, labeled and unlabeled nodes. Translations propagate either directly from labeled nodes to oov nodes or indirectly via unlabeled nodes.

term (1) enforces the labeling of the algorithm to match the seed labeling Y_v with different extent for different labeled nodes. The second term (2) enforces the *smoothness* of the labeling according to the graph structure and edge weights. The last term (3) regularizes the soft labeling for a vertex v to match a priori label R_v, e.g. for high-degree unlabeled nodes (hubs in the graph) we may believe that the neighbors are not going to produce reliable label and hence the probability of undefined label $\perp$ should be higher. The optimization problem can be solved with an efficient iterative algorithm which is parallelized in a MapReduce framework (Talukdar et al., 2008; Rao and Yarowsky, 2009). We used the *Junto label propagation* toolkit (Talukdar and Crammer, 2009) for label propagation.

3.2 Efficient Graph Construction

Graph-based approaches can easily become computationally very expensive as the number of nodes grow. In our case, we use phrases in the monolingual text as graph vertices. These phrases are n-grams up to a certain value, which can result in millions of nodes. For each node a distributional profile (DP) needs to be created. The number of possible edges can easily explode in size as there can be as many as $O(n^2)$ edges where n is the number of nodes. A common practice to control the number of edges is to connect each node to at most k other nodes (k-nearest neighbor). However, finding the top-k nearest nodes to each node requires considering its similarity to all the other nodes which requires $O(n^2)$ computations and since n is usually very large, doing such is practically intractable. Therefore, researchers usually resort to an approximate k-NN algorithms such as *locality-sensitive hashing* (**?**; Goyal et al., 2012).

Fortunately, since we use context words as cues for relating their meaning and since the similarity measures are defined based on these cues, the number of neighbors we need to consider for each node is reduced by several orders of magnitude. We incorporate an inverted-index-style data structure which indicates what nodes are neighbors based on each context word. Therefore, the set of neighbors of a node consists of union of all the neighbors bridged by each context word in the DP of the node. However, the number of neighbors to be considered for each node even after this drastic reduction is still large (in order of a few thousands).

In order to deal with the computational challenges of such a large graph, we take advantage of the Hadoop's MapReduce functionality to do both graph construction and label propagation steps.

4 Experiments & Results

4.1 Experimental Setup

We experimented with two different domains for the bilingual data: *Europarl* corpus (v7) (Koehn,

Dataset	Domain	Sents	Tokens	
			Fr	En
Bitext	Europarl	10K	298K	268K
	EMEA	1M	16M	14M
Monotext	Europarl	2M	60M	–
Dev-set	WMT05	2K	67K	58K
Test-set	WMT05	2K	66K	58K

Table 1: Statistics of training sets in different domains.

2005), and *European Medicines Agency* documents (EMEA) (Tiedemann, 2009) from French to English. For the monolingual data, we used French side of the Europarl corpus and we used ACL/WMT 2005[8] data for dev/test sets. Table 1 summarizes statistics of the datasets used.

From the dev and test sets, we extract all source words that do not appear in the phrase-table constructed from the parallel data. From the oovs, we exclude numbers as well as named entities. We apply a simple heuristic to detect named entities: basically words that are capitalized in the original dev/test set that do not appear at the beginning of a sentence are named entities. Table 2 shows the number of oov types and tokens for Europarl and EMEA systems in both dev and test sets.

Dataset	Dev		Test	
	types	tokens	types	tokens
Europarl	1893	2229	1830	2163
EMEA	2325	4317	2294	4190

Table 2: number of oovs in dev and test sets for Europarl and EMEA systems.

For the end-to-end MT pipeline, we used Moses (Koehn et al., 2007) with these standard features: relative-frequency and lexical translation model (TM) probabilities in both directions; distortion model; language model (LM) and word count. Word alignment is done using GIZA++ (Och and Ney, 2003). We used distortion limit of 6 and max-phrase-length of 10 in all the experiments. For the language model, we used the KenLM toolkit (Heafield, 2011) to create a 5-gram language model on the target side of the Europarl corpus (v7) with approximately $54M$ tokens with Kneser-Ney smoothing.

4.1.1 Phrase-table Integration

Once the translations and their probabilities for each oov are extracted, they are added to the phrase-table that is induced from the parallel text. The probability for new entries are added as a new feature in the log-linear framework to be tuned along with other features. The value of this newly introduced feature for original entries in the phrase-table is set to 1. Similarly, the value of original four probability features in the phrase-table for the new entries are set to 1. The entire training pipeline is as follows: (i) a phrase table is constructed using parallel data as usual, (ii) oovs for dev and test sets are extracted, (iii) oovs are translated using graph propagation, (iv) oovs and translations are added to the phrase table, introducing a new feature type, (v) the new phrase table is tuned (with a LM) using MERT (Och, 2003) on the dev set.

4.2 Evaluation

If we have a list of possible translations for oovs with their probabilities, we become able to evaluate different methods we discussed. We word-aligned the dev/test sets by concatenating them to a large parallel corpus and running GIZA++ on the whole set. The resulting word alignments are used to extract the translations for each oov. The correctness of this gold standard is limited to the size of the parallel data used as well as the quality of the word alignment software toolkit, and is not 100% precise. However, it gives a good estimate of how each oov should be translated without the need for human judgments.

For evaluating our baseline as well as graph-based approaches, we use both intrinsic and extrinsic evaluations. Two intrinsic evaluation metrics that we use to evaluate the possible translations for oovs are *Mean Reciprocal Rank* (MRR) (Voorhees, 1999) and *Recall*. Intrinsic evaluation metrics are faster to apply and are used to optimize different hyper-parameters of the approach (e.g. window size, phrase length, etc.). Once we come up with the optimized values for the hyper-parameters, we extrinsically evaluate different approaches by adding the new translations to the phrase-table and run it through the MT pipeline.

4.2.1 MRR

MRR is an Information Retrieval metric used to evaluate any process that produces a ranked list of possible candidates. The reciprocal rank of a list is the inverse of the rank of the correct answer in the list. Such score is averaged over a set, oov set

[8] http://www.statmt.org/wpt05/mt-shared-task/

in our case, to get the mean-reciprocal-rank score.

$$\text{MRR} = \frac{1}{|O|}\sum_{i=1}^{|O|}\frac{1}{rank_i} \qquad O = \{oov\}$$

In a few cases, there are multiple translations for an oov word (i.e. appearing more than once in the parallel corpus and being assigned to multiple different phrases), we take the average of reciprocal ranks for each of them.

4.2.2 Recall

MRR takes the probabilities of oov translations into account in sorting the list of candidate translations. However, in an MT pipeline, the language model is supposed to rerank the hypotheses and move more appropriate translations (in terms of fluency) to the top of the list. Hence, we also evaluate our candidate translation regardless of the ranks. Since Moses uses a certain number of translations per source phrase (called the translation table limit or *ttl* which we set to 20 in our experiments) , we use the *recall* measure to evaluate the top ttl translations in the list. Recall is another Information Retrieval measure that is the fraction of correct answers that are retrieved. For example, it assigns score of 1 if the correct translation of the oov word is in the top-k list and 0 otherwise. The scores are averaged over all oovs to compute recall.

$$\text{Recall} = \frac{|\{\text{gold standard}\} \cap \{\text{candidate list}\}|}{|\{\text{gold standard}\}|}$$

4.3 Intrinsic Results

In Section 2.2 and 2.3, different types of association measures and similarity measures have been explained to build and compare distributional profiles. Table 3 shows the results on Europarl when using different similarity combinations. The measures are evaluated by fixing the window size to 4 and maximum candidate paraphrase length to 2 (e.g. bigram). First column shows the association measures used to build DPs. As the results show, the combination of PMI as association measure and cosine as DP similarity measure outperforms the other possible combinations. We use these two measures throughout the rest of the experiments.

Figure 2 illustrates the effects of different window sizes and paraphrase lengths on MRR. As the figure shows, the best MRR is reached when using window size of 4 and trigram nodes. Going from trigram to 4-gram results in a drop in MRR. One reason would be that distributional profiles for 4-grams are very sparse and that negatively affects the stability of similarity measures.

Assoc	cosine(%)		L_1norm(%)		JSD(%)	
	MRR	RCL	MRR	RCL	MRR	RCL
CP	1.66	4.16	2.18	5.55	2.33	6.32
LLR	1.79	4.26	0.13	0.37	0.5	1.00
PMI	**3.91**	**7.75**	0.50	1.17	0.59	1.21
Chi	1.66	4.16	0.26	0.55	0.03	0.05

Table 3: Results of intrinsic evaluations (MRR and Recall) on Europarl, window size 4 and paraphrase length 2

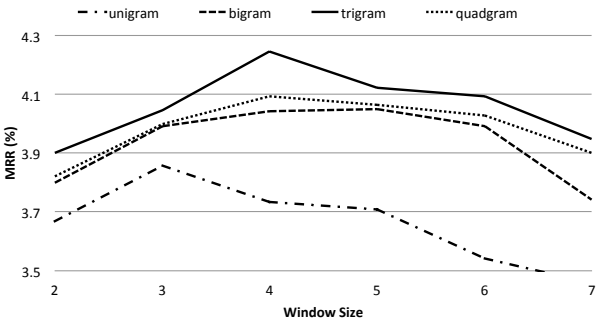

Figure 2: Effects of different window sizes and paraphrase length on the MRR of the dev set.

Figure 3 illustrates the effect of increasing the size of monolingual text on both MRR and recall. $1\times$ refers to the case of using $125k$ sentences for the monolingual text and the $16\times$ indicates using the whole Europarl text on the source side ($\approx 2M$ sentences). As shown, there is a linear correlation between the logarithm of the data size and the MRR and recall ratios. Interestingly, MRR is growing faster than recall by increasing the monolingual text size, which means that the scoring function gets better when more data is available. The figure also indicates that a much bigger monolingual text data can be used to further improve the quality of the translations, however, at the expense of more computational resources.

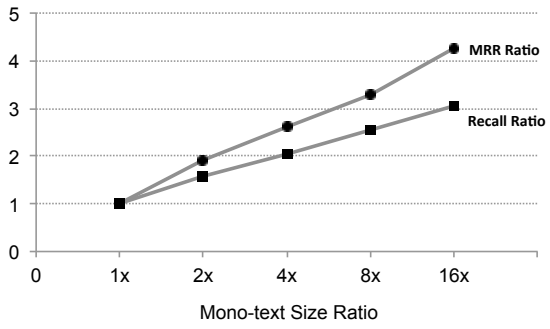

Figure 3: Effect of increasing the monolingual text size on MRR and Recall.

Graph	Neighbor	MRR %	RCL %
Bipartite	20	5.2	12.5
Tripartite	15+5	5.9	12.6
Full	20	5.1	10.9
Baseline	20	3.7	7.2

Table 4: Intrinsic results of different types of graphs when using unigram nodes on Europarl.

Type	Node	MRR %	RCL %
Bipartite	unigram	5.2	12.5
	bigram	6.8	15.7
Tripartite	unigram	5.9	12.6
	bigram	6.9	15.9
Baseline	bigram	3.9	7.7

Table 5: Results on using unigram or bigram nodes.

4.3.1 Graph-based Results

Table 4 shows the intrinsic results on the Europarl corpus when using unigram nodes in each of the graphs. The results are evaluated on the dev-set based on the gold alignment created using GIZA++. Each node is connected to at most 20 other nodes (same as the max-paraphrase-limit in the baseline). For the tripartite graph, each node is connected to 15 labeled nodes and 5 unlabeled ones. The tripartite graph gets a slight improvement over the bipartite one, however, the full graph failed to have the same increase. One reason is that allowing paths longer than 2 between oov and labeled nodes causes more noise to propagate into the graph. In other words, a paraphrase of a paraphrase of a paraphrase is not necessarily a useful paraphrase for an oov as the translation may no longer be a valid one.

Table 5 also shows the effect of using bigrams instead of unigrams as graph nodes. There is an improvement by going from unigrams to bigrams in both bipartite and tripartite graphs. We did not use trigrams or larger n-grams in our experiments.

4.4 Extrinsic Results

The generated candidate translations for the oovs can be added to the phrase-table created using the parallel corpus to increase the coverage of the phrase-table. This aggregated phrase-table is to be tuned along with the language model on the dev set, and run on the test set. BLEU (Papineni et al., 2002) is still the de facto evaluation metric for machine translation and we use that to measure the quality of our proposed approaches for MT. In these experiments, we do not use alignment information on dev or test sets unlike the previous section.

Table 6 reports the Bleu scores for different domains when the oov translations from the graph propagation is added to the phrase-table and compares them with the baseline system (i.e. Moses). Results for our approach is based on unigram tripartite graphs and show that we improve over the baseline in both the same-domain (Europarl) and domain adaptation (EMEA) settings.

Table 7 shows some translations found by our system for oov words.

oov	gold standard	candiate list
spécialement	undone particularly especially special particular	particularly specific only particular should and especially
assentiment	approval	support agreement approval accession will approve endorses

Table 7: Two examples of oov translations found by our method.

5 Related work

There has been a long line of research on learning translation pairs from non-parallel corpora (Rapp, 1995; Koehn and Knight, 2002; Haghighi et al., 2008; Garera et al., 2009; Marton et al., 2009; Laws et al., 2010). Most have focused on extracting a translation lexicon by mining monolingual resources of data to find clues, using probabilistic methods to map words, or by exploiting the cross-language evidence of closely related languages. Most of them evaluated only high-frequency words of specific types (nouns or content words) (Rapp, 1995; Koehn and Knight, 2002; Haghighi et al., 2008; Garera et al., 2009; Laws et al., 2010) In contrast, we do not consider any constraint on our test data and our data includes many low frequency words. It has been shown that translation of high-frequency words is easier than low frequency words (Tamura et al., 2012).

Some methods have used a third language(s) as pivot or bridge to find translation pairs (Mann and Yarowsky, 2001; Schafer and Yarowsky, 2002; Callison-Burch et al., 2006).

Corpus	System	MRR	Recall	Dev Bleu	Test Bleu
Europarl	Baseline	–	–	28.53	28.97
	Our approach	5.9	12.6	28.76	29.40*
EMEA	Baseline	–	–	20.05	20.34
	Our approach	3.6	7.4	20.54	20.80*

* Statistically significant with $p < 0.02$ using the bootstrap resampling significance test (in Moses).

Table 6: Bleu scores for different domains with or without using oov translations.

Context similarity has been used effectively in bilingual lexicon induction (Rapp, 1995; Koehn and Knight, 2002; Haghighi et al., 2008; Garera et al., 2009; Marton et al., 2009; Laws et al., 2010). It has been modeled in different ways: in terms of adjacent words (Rapp, 1999; Fung and Yee, 1998), or dependency relations (Garera et al., 2009). Laws et al. (2010) used linguistic analysis in the form of graph-based models instead of a vector space. But all of these researches used an available seed lexicon as the basic source of similarity between source and target languages unlike our method which just needs a monolingual corpus of source language which is freely available for many languages and a small bilingual corpora.

Some methods tried to alleviate the lack of seed lexicon by using orthographic similarity to extract a seed lexicon (Koehn and Knight, 2002; Fiser and Ljubesic, 2011). But it is not a practical solution in case of unrelated languages.

Haghighi et al. (2008) and Daumé and Jagarlamudi (2011) proposed generative models based on canonical correlation analysis to extract translation lexicons for non-parallel corpora by learning a matching between source and target lexicons. Using monolingual features to represent words, feature vectors are projected from source and target words into a canonical space to find the appropriate matching between them. Their method relies on context features which need a seed lexicon and orthographic features which only works for phylogenetically related languages.

Graph-based semi-supervised methods have been shown to be useful for domain adaptation in MT as well. Alexandrescu and Kirchhoff (2009) applied a graph-based method to determine similarities between sentences and use these similarities to promote similar translations for similar sentences. They used a graph-based semi-supervised model to re-rank the n-best translation hypothesis. Liu et al. (2012) extended Alexandrescu's model to use translation consensus among similar sentences in bilingual training data by developing a new structured label propagation method. They derived some features to use during decoding process that has been shown useful in improving translation quality. Our graph propagation method connects monolingual source phrases with oovs to obtain translation and so is a very different use of graph propagation from these previous works.

Recently label propagation has been used for lexicon induction (Tamura et al., 2012). They used a graph based on context similarity as well as co-occurrence graph in propagation process. Similar to our approach they used unlabeled nodes in label propagation process. However, they use a seed lexicon to define labels and comparable corpora to construct graphs unlike our approach.

6 Conclusion

We presented a novel approach for inducing oov translations from a monolingual corpus on the source side and a parallel data using graph propagation. Our results showed improvement over the baselines both in intrinsic evaluations and on BLEU. Future work includes studying the effect of size of parallel corpus on the induced oov translations. Increasing the size of parallel corpus on one hand reduces the number of oovs. But, on the other hand, there will be more labeled paraphrases that increases the chance of finding the correct translation for oovs in the test set.

Currently, we find paraphrases for oov words. However, oovs can be considered as n-grams (phrases) instead of unigrams. In this scenario, we also can look for paraphrases and translations for phrases containing oovs and add them to the phrase-table as new translations along with the translations for unigram oovs.

We also plan to explore different graph propagation objective functions. Regularizing these objective functions appropriately might let us scale to much larger data sets with an order of magnitude more nodes in the graph.

References

Andrei Alexandrescu and Katrin Kirchhoff. 2009. Graph-based learning for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 119–127, Stroudsburg, PA, USA. Association for Computational Linguistics.

C. Callison-Burch, P. Koehn, and M. Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 17–24. Association for Computational Linguistics.

O. Chapelle, B. Schölkopf, and A. Zien, editors. 2006. *Semi-Supervised Learning*. MIT Press, Cambridge, MA.

Ido Dagan, Lillian Lee, and Fernando C. N. Pereira. 1999. Similarity-based models of word cooccurrence probabilities. *Mach. Learn.*, 34(1-3):43–69, February.

Hal Daumé, III and Jagadeesh Jagarlamudi. 2011. Domain adaptation for machine translation by mining unseen words. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 407–412, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ted Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Comput. Linguist.*, 19(1):61–74, March.

Darja Fiser and Nikola Ljubesic. 2011. Bilingual lexicon extraction from comparable corpora for closely related languages. In *RANLP*, pages 125–131.

Pascale Fung and Lo Yuen Yee. 1998. An ir approach for translating new words from nonparallel, comparable texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98, pages 414–420. Association for Computational Linguistics.

Nikesh Garera, Chris Callison-Burch, and David Yarowsky. 2009. Improving translation lexicon induction from monolingual corpora via dependency contexts and part-of-speech equivalences. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, CoNLL '09, pages 129–137, Stroudsburg, PA, USA. Association for Computational Linguistics.

Amit Goyal, Hal Daume III, and Raul Guerra. 2012. Fast Large-Scale Approximate Graph Construction for NLP. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '12.

Nizar Habash. 2008. Four techniques for online handling of out-of-vocabulary words in arabic-english statistical machine translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 57–60. Association for Computational Linguistics.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *ACL*, pages 771–779.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197.

Chung-Chi Huang, Ho-Ching Yen, Ping-Che Yang, Shih-Ting Huang, and Jason S Chang. 2011. Using sublexical translations to handle the oov problem in machine translation. *ACM Transactions on Asian Language Information Processing (TALIP)*, 10(3):16.

Philipp Koehn and Kevin Knight. 2002. Learning a translation lexicon from monolingual corpora. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition - Volume 9*, ULA '02, pages 9–16, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

P. Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5.

Florian Laws, Lukas Michelbacher, Beate Dorow, Christian Scheible, Ulrich Heid, and Hinrich Schütze. 2010. A linguistically grounded graph model for bilingual lexicon extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 614–622, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*, ACL '98, pages 768–774, Stroudsburg, PA, USA. Association for Computational Linguistics.

Shujie Liu, Chi-Ho Li, Mu Li, and Ming Zhou. 2012. Learning translation consensus with structured label propagation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 302–310, Stroudsburg, PA, USA. Association for Computational Linguistics.

Gideon S. Mann and David Yarowsky. 2001. Multipath translation lexicon induction via bridge languages. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, NAACL '01, pages 1–8, Stroudsburg, PA, USA.

Yuval Marton, Chris Callison-Burch, and Philip Resnik. 2009. Improved statistical machine translation using monolingually-derived paraphrases. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 381–390, Stroudsburg, PA, USA. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Comput. Linguist.*, 29(1):19–51, March.

Franz Josef Och. 2003. Minimum error rate training for statistical machine translation. In *Proceedings of the 41th Annual Meeting of the ACL*, Sapporo, July. ACL.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Delip Rao and David Yarowsky. 2009. Ranking and semi-supervised classification on large scale graphs using map-reduce. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing*, TextGraphs-4. Association for Computational Linguistics.

Reinhard Rapp. 1995. Identifying word translations in non-parallel texts. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, ACL '95, pages 320–322. Association for Computational Linguistics.

Reinhard Rapp. 1999. Automatic identification of word translations from unrelated english and german corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 519–526. Association for Computational Linguistics.

Charles Schafer and David Yarowsky. 2002. Inducing translation lexicons via diverse similarity measures and bridge languages. In *proceedings of the 6th conference on Natural language learning - Volume 20*, COLING-02, pages 1–7, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hinrich Schütze and Jan O. Pedersen. 1997. A cooccurrence-based thesaurus and two applications to information retrieval. *Inf. Process. Manage.*, 33(3):307–318, May.

Partha Pratim Talukdar and Koby Crammer. 2009. New Regularized Algorithms for Transductive Learning. In *European Conference on Machine Learning (ECML-PKDD)*.

Partha Pratim Talukdar, Joseph Reisinger, Marius Paşca, Deepak Ravichandran, Rahul Bhagat, and Fernando Pereira. 2008. Weakly-supervised acquisition of labeled class instances using graph random walks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08.

Akihiro Tamura, Taro Watanabe, and Eiichiro Sumita. 2012. Bilingual lexicon extraction from comparable corpora using label propagation. In *EMNLP-CoNLL*, pages 24–36.

Egidio L. Terra and Charles L. A. Clarke. 2003. Frequency estimates for statistical word similarity measures. In *HLT-NAACL*.

Jorg Tiedemann. 2009. News from opus - a collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia.

Ellen M. Voorhees. 1999. TREC-8 Question Answering Track Report. In *Proceedings of the 8th Text Retrieval Conference*, pages 77–82.

Jiajun Zhang, Feifei Zhai, and Chengqing Zong. 2012. Handling unknown words in statistical machine translation from a new perspective. In *Natural Language Processing and Chinese Computing*, pages 176–187. Springer.