Graph-based Semi-Supervised Learning of Translation Models from Monolingual Data

Avneesh Saluja[*]
Carnegie Mellon University
Pittsburgh, PA 15213, USA
avneesh@cs.cmu.edu

Hany Hassan, Kristina Toutanova, Chris Quirk
Microsoft Research
Redmond, WA 98502, USA
hanyh,kristout,chrisq@microsoft.com

Abstract

Statistical phrase-based translation learns translation rules from bilingual corpora, and has traditionally only used monolingual evidence to construct features that rescore existing translation candidates. In this work, we present a semi-supervised graph-based approach for generating new translation rules that leverages bilingual and monolingual data. The proposed technique first constructs phrase graphs using both source and target language monolingual corpora. Next, graph propagation identifies translations of phrases that were not observed in the bilingual corpus, assuming that similar phrases have similar translations. We report results on a large Arabic-English system and a medium-sized Urdu-English system. Our proposed approach significantly improves the performance of competitive phrase-based systems, leading to consistent improvements between 1 and 4 BLEU points on standard evaluation sets.

1 Introduction

Statistical approaches to machine translation (SMT) use sentence-aligned, parallel corpora to learn translation rules along with their probabilities. With large amounts of data, phrase-based translation systems (Koehn et al., 2003; Chiang, 2007) achieve state-of-the-art results in many typologically diverse language pairs (Bojar et al., 2013). However, the limiting factor in the success of these techniques is parallel data availability. Even in resource-rich languages, learning reliable translations of multiword phrases is a challenge, and an adequate phrasal inventory is crucial for effective translation. This problem is exacerbated in the many language pairs for which parallel resources are either limited or nonexistent. While parallel data is generally scarce, monolingual resources exist in abundance and are being created at accelerating rates. Can we use monolingual data to augment the phrasal translations acquired from parallel data?

The challenge of learning translations from monolingual data is of long standing interest, and has been approached in several ways (Rapp, 1995; Callison-Burch et al., 2006; Haghighi et al., 2008; Ravi and Knight, 2011). Our work introduces a new take on the problem using graph-based semi-supervised learning to acquire translation rules and probabilities by leveraging both monolingual and parallel data resources. On the source side, labeled phrases (those with known translations) are extracted from bilingual corpora, and unlabeled phrases are extracted from monolingual corpora; together they are embedded as nodes in a graph, with the monolingual data determining edge strengths between nodes (§2.2). Unlike previous work (Irvine and Callison-Burch, 2013a; Razmara et al., 2013), we use higher order n-grams instead of restricting to unigrams, since our approach goes beyond OOV mitigation and can enrich the entire translation model by using evidence from monolingual text. This enhancement alone results in an improvement of almost 1.4 BLEU points. On the target side, phrases initially consisting of translations from the parallel data are selectively expanded with generated candidates (§2.1), and are embedded in a target graph.

We then limit the set of translation options for each unlabeled source phrase (§2.3), and using a structured graph propagation algorithm, where translation information is propagated from labeled to unlabeled phrases proportional to *both* source and target phrase similarities, we estimate probability distributions over translations for

[*]This work was done while the first author was interning at Microsoft Research

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 676–686,
Baltimore, Maryland, USA, June 23-25 2014. ©2014 Association for Computational Linguistics

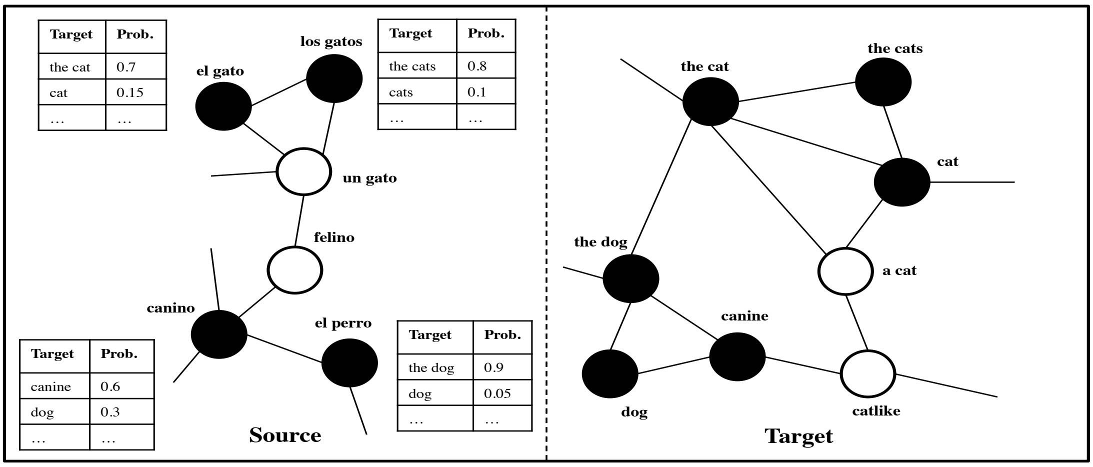

Figure 1: Example source and target graphs used in our approach. Labeled phrases on the source side are black (with their corresponding translations on the target side also black); unlabeled and generated (§2.1) phrases on the source and target sides respectively are white. Labeled phrases also have conditional probability distributions defined over target phrases, which are extracted from the parallel corpora.

the unlabeled source phrases (§2.4). The additional phrases are incorporated in the SMT system through a secondary phrase table (§2.5). We evaluated the proposed approach on both Arabic-English and Urdu-English under a range of scenarios (§3), varying the amount and type of monolingual corpora used, and obtained improvements between 1 and 4 BLEU points, even when using very large language models.

2 Generation & Propagation

Our goal is to obtain translation distributions for source phrases that are not present in the phrase table extracted from the parallel corpus. Both parallel and monolingual corpora are used to obtain these probability distributions over target phrases. We assume that sufficient parallel resources exist to learn a basic translation model using standard techniques, and also assume the availability of larger monolingual corpora in both the source and target languages. Although our technique applies to phrases of any length, in this work we concentrate on unigram and bigram phrases, which provides substantial computational cost savings.

Monolingual data is used to construct *separate* similarity graphs over phrases (word sequences), as illustrated in Fig. 1. The source similarity graph consists of phrase nodes representing sequences of words in the source language. If a source phrase is found in the baseline phrase table it is called a **labeled** phrase: its conditional empirical probability distribution over target phrases (estimated from the parallel data) is used as the label, and is subsequently never changed. Otherwise it is called an **unlabeled** phrase, and our algorithm finds labels (translations) for these unlabeled phrases, with the help of the graph-based representation. The label space is thus the phrasal translation inventory, and like the source side it can also be represented in terms of a graph, initially consisting of target phrase nodes from the parallel corpus.

For the unlabeled phrases, the set of possible target translations could be extremely large (e.g., all target language n-grams). Therefore, we first **generate** and fix a list of possible target translations for each unlabeled source phrase. We then **propagate** by deriving a probability distribution over these target phrases using graph propagation techniques. Next, we will describe the generation, graph construction and propagation steps.

2.1 Generation

The objective of the generation step is to populate the target graph with *additional* target phrases for all unlabeled source phrases, yielding the full set of possible translations for the phrase. Prior to generation, one phrase node for each target phrase occurring in the baseline phrase table is added to the target graph (black nodes in Fig. 1's target graph). We only consider target phrases whose source phrase is a bigram, but it is worth noting that the target phrases are of variable length.

The generation component is based on the observation that for structured label spaces, such as translation candidates for source phrases in SMT, even similar phrases have slightly different labels (target translations). The exponential dependence of the sizes of these spaces on the length of instances is to blame. Thus, the target phrase inventory from the parallel corpus may be inadequate for unlabeled instances. We therefore need to enrich the target or label space for unknown phrases. A naïve way to achieve this goal would be to extract all n-grams, from $n = 1$ to a maximum n-gram order, from the monolingual data, but this strategy would lead to a combinatorial explosion in the number of target phrases.

Instead, by intelligently expanding the target space using linguistic information such as morphology (Toutanova et al., 2008; Chahuneau et al., 2013), or relying on the baseline system to generate candidates similar to self-training (McClosky et al., 2006), we can tractably propose novel translation candidates (white nodes in Fig. 1's target graph) whose probabilities are then estimated during propagation. We refer to these additional candidates as "generated" candidates.

To generate new translation candidates using the baseline system, we decode each unlabeled source bigram to generate its m-best translations. This set of candidate phrases is filtered to include only n-grams occurring in the target monolingual corpus, and helps to prune passed-through OOV words and invalid translations. To generate new translation candidates using morphological information, we morphologically segment words into prefixes, stem, and suffixes using linguistic resources. We assume that a morphological analyzer which provides context-independent analysis of word types exists, and implements the functions $\text{STEM}(f)$ and $\text{STEM}(e)$ for source and target word types. Based on these functions, source and target sequences of words can be mapped to sequences of stems. The morphological generation step adds to the target graph all target word sequences from the monolingual data that map to the same stem sequence as one of the target phrases occurring in the baseline phrase table. In other words, this step adds phrases that are morphological variants of existing phrases, differing only in their affixes.

2.2 Graph Construction

At this stage, there exists a list of source bigram phrases, both labeled and unlabeled, as well as a list of target language phrases of variable length, originating from both the phrase table and the generation step. To determine pairwise phrase similarities in order to embed these nodes in their graphs, we utilize the monolingual corpora on both the source and target sides to extract distributional features based on the context surrounding each phrase. For a phrase, we look at the p words before and the p words after the phrase, explicitly distinguishing between the two sides, but not distance (i.e., bag of words on each side). Co-occurrence counts for each feature (context word) are accumulated over the monolingual corpus, and these counts are converted to pointwise mutual information (PMI) values, as is standard practice when computing distributional similarities. Cosine similarity between two phrases' PMI vectors is used for similarity, and we take only the k most similar phrases for each phrase, to create a k-nearest neighbor similarity matrix for both source and target language phrases. These graphs are distinct, in that propagation happens within the two graphs but not between them.

While accumulating co-occurrence counts for each phrase, we also maintain an inverted index data structure, which is a mapping from features (context words) to phrases that co-occur with that feature within a window of p.[1] The inverted index structure reduces the graph construction cost from $\theta(n^2)$, by only computing similarities for a subset of all possible pairs of phrases, namely other phrases that have at least one feature in common.

2.3 Candidate Translation List Construction

As mentioned previously, we construct and fix a set of translation candidates, i.e., the label set for each unlabeled source phrase. The probability distribution over these translations is estimated through graph propagation, and the probabilities of items outside the list are assumed to be zero.

We obtain these candidates from two sources:[2]

1. The union of each unlabeled phrase's labeled neighbors' labels, which represents the set of target phrases that occur as translations of source phrases that are similar to the unlabeled source phrase. For *un gato* in Fig. 1, this source would yield *the cat* and *cat*, among others, as candidates.

2. The generated candidates for the unlabeled phrase – the ones from the baseline system's

[1] The q most frequent words in the monolingual corpus were removed as keys from this mapping, as these high entropy features do not provide much information.

[2] We also obtained the k-nearest neighbors of the translation candidates generated through these methods by utilizing the target graph, but this had minimal impact.

decoder output, or from a morphological generator (e.g., *a cat* and *catlike* in Fig. 1).

The morphologically-generated candidates for a given source unlabeled phrase are initially defined as the target word sequences in the monolingual data that have the same stem sequence as one of the baseline's target translations for a source phrase which has the same stem sequence as the unlabeled source phrase. These candidates are scored using stem-level translation probabilities, morpheme-level lexical weighting probabilities, and a language model, and only the top 30 candidates are included.

After obtaining candidates from these two possible sources, the list is sorted by forward lexical score, using the lexical models of the baseline system. The top r candidates are then chosen for each phrase's translation candidate list.

In Figure 2 we provide example outputs of our system for a handful of unlabeled source phrases, and explicitly note the source of the translation candidate ('G' for generated, 'N' for labeled neighbor's label).

2.4 Graph Propagation

A graph propagation algorithm transfers label information from labeled nodes to unlabeled nodes by following the graph's structure. In some applications, a label may consist of class membership information, e.g., each node can belong to one of a certain number of classes. In our problem, the "label" for each node is actually a probability distribution over a set of translation candidates (target phrases). For a given node f, let e refer to a candidate in the label set for node f; then in graph propagation, the probability of candidate e given source phrase f in iteration $t+1$ is:

$$\mathbb{P}^{t+1}(e|f) = \sum_{j \in \mathcal{N}(f)} T_s(j|f)\mathbb{P}^t(e|j) \tag{1}$$

where the set $\mathcal{N}(f)$ contains the (labeled and unlabeled) neighbors of node f, and $T_s(j|f)$ is a term that captures how similar nodes f and j are. This quantity is also known as the propagation probability, and its exact form will depend on the type of graph propagation algorithm used. For our purposes, node f is a source phrasal node, the set $\mathcal{N}(f)$ refers to other source phrases that are neighbors of f (restricted to the k-nearest neighbors as in §2.2), and the aim is to estimate $P(e|f)$, the probability of target phrase e being a phrasal translation of source phrase f.

A classic propagation algorithm that has been suitably modified for use in bilingual lexicon induction (Tamura et al., 2012; Razmara et al., 2013) is the **label propagation** (LP) algorithm of Zhu et al. (2003). In this case, $T_s(f, j)$ is chosen to be:

$$T_s(j|f) = \frac{w^s_{f,j}}{\sum_{j' \in \mathcal{N}(f)} w^s_{f,j'}} \tag{2}$$

where $w^s_{f,j}$ is the cosine similarity (as computed in §2.2) between phrase f and phrase j on side s (the source side).

As evident in Eq. 2, LP only takes into account source language similarity of phrases. To see this observation more clearly, let us reformulate Eq. 1 more generally as:

$$\mathbb{P}^{t+1}(e|f) = \sum_{j \in \mathcal{N}(f)} T_s(j|f) \sum_{e' \in \mathcal{H}(j)} T_t(e'|e)\mathbb{P}^t(e'|j) \tag{3}$$

where $\mathcal{H}(j)$ is the translation candidate set for source phrase j, and $T_t(e'|e)$ is the propagation probability between nodes or phrases e and e' on the *target* side. We have simply replaced $\mathbb{P}^t(e|j)$ with $\sum_{e' \in \mathcal{H}(j)} T_t(e'|e)\mathbb{P}^t(e'|j)$, defining it in terms of j's translation candidate list.

Note that in the original LP formulation the target side information is disregarded, i.e., $T_t(e'|e) = 1$ if and only if $e = e'$ and 0 otherwise. As a result, LP is suboptimal for our needs, since it is unable to appropriately handle generated translation candidates for the unlabeled phrases. These translation candidates are usually not present as translations for the labeled phrases (or for the labeled phrases that neighbor the unlabeled one in question). When propagating information from the labeled phrases, such candidates will obtain no probability mass since $e \neq e'$. Thus, due to the setup of the problem, LP naturally biases away from translation candidates produced during the generation step (§2.1).

2.4.1 Structured Label Propagation

The label set we are considering has a similarity structure encoded by the target graph. How can we exploit this structure in graph propagation on the source graph? In Liu *et al.* (2012), the authors generalize label propagation to **structured label propagation** (SLP) in an effort to work more elegantly with structured labels. In particular, the definition of target similarity is similar to that of source similarity:

$$T_t(e'|e) = \frac{w^t_{e,e'}}{\sum_{e'' \in \mathcal{H}(j)} w^t_{e,e''}} \tag{4}$$

Therefore, the final update equation in SLP is:

$$\mathbb{P}^{t+1}(e|f) = \sum_{j \in \mathcal{N}(f)} T_s(j|f) \sum_{e' \in \mathcal{H}(j)} T_t(e'|e)\mathbb{P}^t(e'|j) \quad (5)$$

With this formulation, even if $e \neq e'$, the similarity $T_t(e'|e)$ as determined by the target phrase graph will dictate propagation probability. We renormalize the probability distributions after each propagation step to sum to one over the fixed list of translation candidates, and run the SLP algorithm to convergence.[3]

2.5 Phrase-based SMT Expansion

After graph propagation, each unlabeled phrase is labeled with a categorical distribution over the set of translation candidates defined in §2.3. In order to utilize these newly acquired phrase pairs, we need to compute their relevant features. The phrase pairs have four log-probability features with two likelihood features and two lexical weighting features. In addition, we use a sophisticated lexicalized hierarchical reordering model (HRM) (Galley and Manning, 2008) with five features for each phrase pair.

We utilize the graph propagation-estimated forward phrasal probabilities $\mathbb{P}(e|f)$ as the forward likelihood probabilities for the acquired phrases; to obtain the backward phrasal probability for a given phrase pair, we make use of Bayes' Theorem:

$$\mathbb{P}(f|e) = \frac{\mathbb{P}(e|f)\mathbb{P}(f)}{\mathbb{P}(e)}$$

where the marginal probabilities of source and target phrases e and f are obtained from the counts extracted from the monolingual data. The baseline system's lexical models are used for the forward and backward lexical scores. The HRM probabilities for the new phrase pairs are estimated from the baseline system by backing-off to the average values for phrases with similar length.

3 Evaluation

We performed an extensive evaluation to examine various aspects of the approach along with overall system performance. Two language pairs were used: Arabic-English and Urdu-English. The Arabic-English evaluation was used to validate the decisions made during the development of our method and also to highlight properties of the technique. With it, in §3.2 we first analyzed the impact of utilizing phrases instead of words and SLP instead of LP; the latter experiment underscores the importance of generated candidates. We also look at how adding morphological knowledge to the generation process can further enrich performance. In §3.3, we then examined the effect of using a very large 5-gram language model training on 7.5 billion English tokens to understand the nature of the improvements in §3.2. The Urdu to English evaluation in §3.4 focuses on how noisy parallel data and completely monolingual (i.e., not even comparable) text can be used for a realistic low-resource language pair, and is evaluated with the larger language model only. We also examine how our approach can learn from noisy parallel data compared to the traditional SMT system.

Baseline phrasal systems are used both for comparison and for generating translation candidates for unlabeled phrases as described in §2.1. The baseline is a state-of-the-art phrase-based system; we perform word alignment using a lexicalized hidden Markov model, and then the phrase table is extracted using the `grow-diag-final` heuristic (Koehn et al., 2003). The 13 baseline features (2 lexical, 2 phrasal, 5 HRM, and 1 language model, word penalty, phrase length feature and distortion penalty feature) were tuned using MERT (Och, 2003), which is also used to tune the 4 feature weights introduced by the secondary phrase table (2 lexical and 2 phrasal, other features being shared between the two tables). For all systems, we use a distortion limit of 4. We use case-insensitive BLEU (Papineni et al., 2002) to evaluate translation quality.

3.1 Datasets

Bilingual corpus statistics for both language pairs are presented in Table 2. For Arabic-English, our training corpus consisted of 685k sentence pairs from standard LDC corpora[4]. The NIST MT06 and MT08 Arabic-English evaluation sets (combining the newswire and weblog domains for both sets), with four references each, were used as tuning and testing sets respectively. For Urdu-English, the training corpus was provided by the LDC for the NIST Urdu-English MT evaluation, and most of the data was automatically acquired from the web, making it quite noisy. After filtering, there are approximately 65k parallel sen-

[3] Empirically within a few iterations and a wall-clock time of less than 10 minutes in total.

[4] LDC2007T08 and LDC2008T09

Parameter	Description	Value
m	m-best candidate list size when bootstrapping candidates in generation stage.	100
p	Window size on each side when extracting features for phrases.	2
q	Filter the q most frequent words when storing the inverted index data structure for graph construction. Both source and target sides share the same value.	25
k	Number of neighbors stored for each phrase for both source and target graphs. This parameter controls the sparsity of the graph.	500
r	Maximum size of translation candidate list for unlabeled phrases.	20

Table 1: Parameters, explanation of their function, and value chosen.

tences; these were supplemented by an additional 100k dictionary entries. Tuning and test data consisted of the MT08 and MT09 evaluation corpora, once again a mixture of news and web text.

Corpus	Sentences	Words (Src)
Ar-En Train	685,502	17,055,168
Ar-En Tune (MT06)	1,664	33,739
Ar-En Test (MT08)	1,360	42,472
Ur-En Train	165,159	1,169,367
Ur-En Tune (MT08)	1,864	39,925
Ur-En Test (MT09)	1,792	39,922

Table 2: Bilingual corpus statistics for the Arabic-English and Urdu-English datasets used.

Table 3 contains statistics for the monolingual corpora used in our experiments. From these corpora, we extracted all sentences that contained at least one source or target phrase match to compute features for graph construction. For the Arabic to English experiments, the monolingual corpora are taken from the AFP Arabic and English Gigaword corpora and are of a similar date range to each other (1994-2010), rendering them comparable but not sentence-aligned or parallel.

Corpus	Sentences	Words
Ar Comparable	10.2m	290m
En I Comparable	29.8m	900m
Ur Noisy Parallel	470k	5m
En II Noisy Parallel	470k	4.7m
Ur Non-Comparable	7m	119m
En II Non-Comparable	17m	510m

Table 3: Monolingual corpus statistics for the Arabic-English and Urdu-English evaluations. The monolingual corpora can be sub-divided into comparable, noisy parallel, and non-comparable components. En I refers to the English side of the Arabic-English corpora, and En II to the English side of the Urdu-English corpora.

For the Urdu-English experiments, completely non-comparable monolingual text was used for graph construction; we obtained the Urdu side through a web-crawler, and a subset of the AFP Gigaword English corpus was used for English. In addition, we obtained a corpus from the ELRA[5], which contains a mix of parallel and monolingual data; based on timestamps, we extracted a comparable English corpus for the ELRA Urdu monolingual data to form a roughly 470k-sentence "noisy parallel" set. We used this set in two ways: either to augment the parallel data presented in Table 2, or to augment the non-comparable monolingual data in Table 3 for graph construction.

For the parameters introduced throughout the text, we present in Table 1 a reminder of their interpretation as well as the values used in this work.

3.2 Experimental Variations

In our first set of experiments, we looked at the impact of choosing bigrams over unigrams as our basic unit of representation, along with performance of LP (Eq. 2) compared to SLP (Eq. 4). Recall that LP only takes into account source similarity; since the vast majority of generated candidates do not occur as labeled neighbors' labels, restricting propagation to the source graph drastically reduces the usage of generated candidates as labels, but does not completely eliminate it. In these experiments, we utilize a reasonably-sized 4-gram language model trained on 900m English tokens, i.e., the English monolingual corpus.

Table 4 presents the results of these variations; overall, by taking into account generated candidates appropriately and using bigrams ("SLP 2-gram"), we obtained a 1.13 BLEU gain on the test set. Using unigrams ("SLP 1-gram") actually does worse than the baseline, indicating the importance of focusing on translations for sparser bigrams. While LP ("LP 2-gram") does reasonably well, its underperformance compared to SLP underlines the importance of enriching the translation space with generated candidates and handling these candidates appropriately.[6] In "SLP-

[5] ELRA-W0038

[6] It is relatively straightforward to combine both unigrams and bigrams in one source graph, but for experimental clarity we did not mix these phrase lengths.

HalfMono", we use only half of the monolingual comparable corpora, and still obtain an improvement of 0.56 BLEU points, indicating that adding more monolingual data is likely to improve the system further. Interestingly, biasing away from generated candidates using all the monolingual data ("LP 2-gram") performs similarly to using half the monolingual corpora and handling generated candidates properly ("SLP-HalfMono").

	BLEU	
Setup	**Tune**	**Test**
Baseline	39.33	38.09
SLP 1-gram	39.47	37.85
LP 2-gram	40.75	38.68
SLP 2-gram	41.00	39.22
SLP-HalfMono 2-gram	40.82	38.65
SLP+Morph 2-gram	41.02	39.35

Table 4: Results for the Arabic-English evaluation. The LP vs. SLP comparison highlights the importance of target side enrichment via translation candidate generation, 1-gram vs. 2-gram comparisons highlight the importance of emphasizing phrases, utilizing half the monolingual data shows sensitivity to monolingual corpus size, and adding morphological information results in additional improvement.

Additional morphologically generated candidates were added in this experiment as detailed in §2.3. We used a simple hand-built Arabic morphological analyzer that segments word types based on regular expressions, and an English lexicon-based morphological analyzer. The morphological candidates add a small amount of improvement, primarily by targeting genuine OOVs.

3.3 Large Language Model Effect

In this set of experiments, we examined if the improvements in §3.2 can be explained primarily through the extraction of language model characteristics during the semi-supervised learning phase, or through orthogonal pieces of evidence. Would the improvement be less substantial had we used a very large language model?

To answer this question we trained a 5-gram language model on 570M sentences (7.6B tokens), with data from various sources including the Gigaword corpus[7], WMT and European Parliamentary Proceedings[8], and web-crawled data from Wikipedia and the web. Only m-best generated candidates from the baseline were considered during generation, along with labeled neighbors' labels.

	BLEU	
Setup	**Tune**	**Test**
Baseline+LargeLM	41.48	39.86
SLP+LargeLM	42.82	41.29

Table 5: Results with the large language model scenario. The gains are even better than with the smaller language model.

Table 5 presents the results of using this language model. We obtained a robust, 1.43-BLEU point gain, indicating that the addition of the newly induced phrases provided genuine translation improvements that cannot be compensated by the language model effect. Further examination of the differences between the two systems yielded that most of the improvements are due to better bigrams and trigrams, as indicated by the breakdown of the BLEU score precision per n-gram, and primarily leverages higher quality generated candidates from the baseline system. We analyze the output of these systems further in the output analysis section below (§3.5).

3.4 Urdu-English

In order to evaluate the robustness of these results beyond one language pair, we looked at Urdu-English, a low resource pair likely to benefit from this approach. In this set of experiments, we used the large language model in §3.3, and only used baseline-generated candidates. We experimented with two extreme setups that differed in the data assumed parallel, from which we built our baseline system, and the data treated as monolingual, from which we built our source and target graphs.

In the first setup, we use the noisy parallel data for graph construction and augment the non-comparable corpora with it:

- parallel: "Ur-En Train"
- Urdu monolingual: "Ur Noisy Parallel"+"Ur Non-Comparable"
- English monolingual: "En II Noisy Parallel"+"En II Non-Comparable"

The results from this setup are presented as "Baseline" and "SLP+Noisy" in Table 6. In the second setup, we train a baseline system using the data in Table 2, augmented with the noisy parallel text:

- parallel: "Ur-En Train"+"Ur Noisy Parallel"+"En II Noisy Parallel"
- Urdu monolingual: "Ur Non-Comparable"
- English monolingual: "En II Non-Comparable"

[7] LDC2011T07

[8] http://www.statmt.org/wmt13/

Ex	Source	Reference	Baseline	System
1 (Ar)	ارسال التعزيزات	sending reinforcements	strong reinforcements	sending reinforcements (N)
2 (Ar)	ل+ الاندثار	with extinction	OOV	with extinction (N)
3 (Ar)	تحبط محاولة	thwarts	address	thwarted (N)
4 (Ar)	نسبت الي	was quoted as saying	attributed to	was quoted as saying (G)
5 (Ar)	أوضح عبد المحمود	abdalmahmood said	he said abdul mahmood	mahmood said (G)
6 (Ar)	تراه منكبا	it deems	OOV	it deems (G)
7 (Ur)	پر امید	I am hopeful	this hope	I am hopeful (N)
8 (Ur)	اپنا دفاع	to defend him	to defend	to defend himself (G)
9 (Ur)	گفتگو کی۔	while speaking	In the	in conversation (N)

Figure 2: Nine example outputs of our system vs. the baseline highlighting the properties of our approach. Each example is labeled (Ar) for Arabic source or (Ur) for Urdu source, and system candidates are labeled with (N) if the candidate unlabeled phrase's labeled neighbor's label, or (G) if the candidate was generated.

The results from this setup are presented as "Baseline+Noisy" and "SLP" in Table 6. The two setups allow us to examine how effectively our method can learn from the noisy parallel data by treating it as monolingual (i.e., for graph construction), compared to treating this data as parallel, and also examines the realistic scenario of using completely non-comparable monolingual text for graph construction as in the second setup.

	BLEU	
Setup	**Tune**	**Test**
Baseline	21.87	21.17
SLP+Noisy	26.42	25.38
Baseline+Noisy	27.59	27.24
SLP	28.53	28.43

Table 6: Results for the Urdu-English evaluation evaluated with BLEU. All experiments were conducted with the larger language model, and generation only considered the m-best candidates from the baseline system.

In the first setup, we get a huge improvement of 4.2 BLEU points ("SLP+Noisy") when using the monolingual data and the noisy parallel data for graph construction. Our method obtained much of the gains achieved by the supervised baseline approach that utilizes the noisy parallel data in conjunction with the NIST-provided parallel data ("Baseline+Noisy"), but with fewer assumptions on the nature of the corpora (monolingual vs. parallel). Furthermore, despite completely unaligned, non-comparable monolingual text on the Urdu and English sides, and a very large language model, we can still achieve gains in excess of 1.2 BLEU points ("SLP") in a difficult evaluation scenario, which shows that the technique adds a genuine translation improvement over and above naïve memorization of n-gram sequences.

3.5 Analysis of Output

Figure 2 looks at some of the sample hypotheses produced by our system and the baseline, along with reference translations. The outputs produced by our system are additionally annotated with the origin of the candidate, i.e., labeled neighbor's label (N) or generated (G).

The Arabic-English examples are numbered 1 to 5. The first example shows a source bigram unknown to the baseline system, resulting in a suboptimal translation, while our system proposes the correct translation of "sending reinforcements". The second example shows a word that was an OOV for the baseline system, while our system got a perfect translation. The third and fourth examples represent bigram phrases with much better translations compared to backing off to the lexical translations as in the baseline. The fifth Arabic-English example demonstrates the pitfalls of over-reliance on the distributional hypothesis: the source bigram corresponding to the name "abd almahmood" is distributional similar to another named entity "mahmood" and the English equivalent is offered as a translation. The distributional hypothesis can sometimes be misleading. The sixth example shows how morphological information can propose novel candidates: an OOV word is broken down to its stem via the analyzer and candidates are generated based on the stem.

The Urdu-English examples are numbered 7 to 9. In example 7, the bigram "par umeed" (corresponding to "hopeful") is never seen in the baseline system, which has only seen "umeed" ("hope"). By leveraging the monolingual corpus to understand the context of this unlabeled bigram, we can utilize the graph structure to propose a syntactically correct form, also resulting in a more fluent and correct sentence as determined by the language model. Examples 8 & 9 show cases where the baseline deletes words or translates them into more common words e.g., "conversation" to "the", while our system proposes reasonable candidates.

4 Related Work

The idea presented in this paper is similar in spirit to bilingual lexicon induction (BLI), where a seed lexicon in two different languages is expanded with the help of monolingual corpora, primarily by extracting distributional similarities from the data using word context. This line of work, initiated by Rapp (1995) and continued by others (Fung and Yee, 1998; Koehn and Knight, 2002) (*inter alia*) is limited from a downstream perspective, as translations for only a small number of words are induced and oftentimes for common or frequently occurring ones only. Recent improvements to BLI (Tamura et al., 2012; Irvine and Callison-Burch, 2013b) have contained a graph-based flavor by presenting label propagation-based approaches using a seed lexicon, but evaluation is once again done on top-1 or top-3 accuracy, and the focus is on unigrams.

Razmara et al. (2013) and Irvine and Callison-Burch (2013a) conduct a more extensive evaluation of their graph-based BLI techniques, where the emphasis and end-to-end BLEU evaluations concentrated on OOVs, i.e., unigrams, and not on enriching the entire translation model. As with previous BLI work, these approaches only take into account source-side similarity of words; only moderate gains (and in the latter work, on a subset of language pairs evaluated) are obtained. Additionally, because of our structured propagation algorithm, our approach is better at handling multiple translation candidates and does not need to restrict itself to the top translation.

Klementiev et al. (2012) propose a method that utilizes a pre-existing phrase table and a small bilingual lexicon, and performs BLI using monolingual corpora. The operational scope of their approach is limited in that they assume a scenario where unknown phrase pairs are provided (thereby sidestepping the issue of translation candidate generation for completely unknown phrases), and what remains is the estimation of phrasal probabilities. In our case, we obtain the phrase pairs from the graph structure (and therefore indirectly from the monolingual data) and a separate generation step, which plays an important role in good performance of the method. Similarly, Zhang and Zong (2013) present a series of heuristics that are applicable in a fairly narrow setting.

The notion of translation consensus, wherein similar sentences on the source side are encouraged to have similar target language translations, has also been explored via a graph-based approach (Alexandrescu and Kirchhoff, 2009). Liu et al. (2012) extend this method by proposing a novel structured label propagation algorithm to deal with the generalization of propagating *sets* of labels instead of single labels, and also integrated information from the graph into the decoder. In fact, we utilize this algorithm in our propagation step (§2.4). However, the former work operates only at the level of sentences, and while the latter does extend the framework to sub-spans of sentences, they do not discover new translation pairs or phrasal probabilities for new pairs at all, but instead re-estimate phrasal probabilities using the graph structure and add this score as an additional feature during decoding.

The goal of leveraging non-parallel data in machine translation has been explored from several different angles. Paraphrases extracted by "pivoting" via a third language (Callison-Burch et al., 2006) can be derived solely from monolingual corpora using distributional similarity (Marton et al., 2009). Snover et al. (2008) use cross-lingual information retrieval techniques to find potential sentence-level translation candidates among comparable corpora. In this case, the goal is to try and construct a corpus as close to parallel as possible from comparable corpora, and is a fairly different take on the problem we are looking at. Decipherment-based approaches (Ravi and Knight, 2011; Dou and Knight, 2012) have generally taken a monolingual view to the problem and combine phrase tables through the log-linear model during feature weight training.

5 Conclusion

In this work, we presented an approach that can expand a translation model extracted from a sentence-aligned, bilingual corpus using a large amount of unstructured, monolingual data in both source and target languages, which leads to improvements of 1.4 and 1.2 BLEU points over strong baselines on evaluation sets, and in some scenarios gains in excess of 4 BLEU points. In the future, we plan to estimate the graph structure through other learned, distributed representations.

Acknowledgments

The authors would like to thank Chris Dyer, Arul Menezes, and the anonymous reviewers for their helpful comments and suggestions.

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-HLT '09, pages 119–127. Association for Computational Linguistics, June.

Ondřej Bojar, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria, August. Association for Computational Linguistics.

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 17–24, New York City, USA, June. Association for Computational Linguistics.

Victor Chahuneau, Eva Schlinger, Noah A. Smith, and Chris Dyer. 2013. Translating into morphologically rich languages with synthetic phrases. In *Proc. of EMNLP*.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228, June.

Qing Dou and Kevin Knight. 2012. Large scale decipherment for out-of-domain machine translation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 266–275. Association for Computational Linguistics, July.

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, Stroudsburg, PA, USA. Association for Computational Linguistics.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. EMNLP '08, pages 848–856, Stroudsburg, PA, USA. Association for Computational Linguistics.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL-08: HLT*, pages 771–779, Columbus, Ohio, June. Association for Computational Linguistics.

Ann Irvine and Chris Callison-Burch. 2013a. Combining bilingual and comparable corpora for low resource machine translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 262–270, Sofia, Bulgaria, August. Association for Computational Linguistics.

Ann Irvine and Chris Callison-Burch. 2013b. Supervised bilingual lexicon induction with multiple monolingual signals. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 518–523, Atlanta, Georgia, June. Association for Computational Linguistics.

Alexandre Klementiev, Ann Irvine, Chris Callison-Burch, and David 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*, pages 130–140, Avignon, France, April. Association for Computational Linguistics.

Philipp Koehn and Kevin Knight. 2002. Learning a translation lexicon from monolingual corpora. In *In Proceedings of ACL Workshop on Unsupervised Lexical Acquisition*, pages 9–16.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 48–54, 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.

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*, EMNLP '09, pages 381–390, Singapore, August. Association for Computational Linguistics.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Effective self-training for parsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 152–159, New York City, USA, June. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. pages 311–318.

Reinhard Rapp. 1995. Identifying word translations in non-parallel texts. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, ACL '95.

Sujith Ravi and Kevin Knight. 2011. Deciphering foreign language. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 12–21, Portland, Oregon, USA, June. Association for Computational Linguistics.

Majid Razmara, Maryam Siahbani, Gholamreza Haffari, and Anoop Sarkar. 2013. Graph propagation for paraphrasing out-of-vocabulary words in statistical machine translation. In *Proceedings of the 51st of the Association for Computational Linguistics*, ACL-51, Stroudsburg, PA, USA. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, and Richard Schwartz. 2008. Language and translation model adaptation using comparable corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 857–866, Stroudsburg, PA, USA. Association for Computational Linguistics.

Akihiro Tamura, Taro Watanabe, and Eiichiro Sumita. 2012. Bilingual lexicon extraction from comparable corpora using label propagation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 24–36.

Kristina Toutanova, Hisami Suzuki, and Achim Ruopp. 2008. Applying morphology generation models to machine translation. In *Proceedings of ACL-08: HLT*, pages 514–522, Columbus, Ohio, June. Association for Computational Linguistics.

Jiajun Zhang and Chengqing Zong. 2013. Learning a phrase-based translation model from monolingual data with application to domain adaptation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1425–1434, Sofia, Bulgaria, August. Association for Computational Linguistics.

Xiaojin Zhu, Zoubin Ghahramani, and John D. Lafferty. 2003. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the Twentieth International Conference on Machine Learning*, ICML '03, pages 912–919.