Models and Inference for Prefix-Constrained Machine Translation

Joern Wuebker, Spence Green, John DeNero, Saša Hasan
Lilt, Inc.
first_name@lilt.com

Minh-Thang Luong
Stanford University
lmthang@stanford.edu

Abstract

We apply phrase-based and neural models to a core task in interactive machine translation: suggesting how to complete a partial translation. For the phrase-based system, we demonstrate improvements in suggestion quality using novel objective functions, learning techniques, and inference algorithms tailored to this task. Our contributions include new tunable metrics, an improved beam search strategy, an n-best extraction method that increases suggestion diversity, and a tuning procedure for a hierarchical joint model of alignment and translation. The combination of these techniques improves next-word suggestion accuracy dramatically from 28.5% to 41.2% in a large-scale English-German experiment. Our recurrent neural translation system increases accuracy yet further to 53.0%, but inference is two orders of magnitude slower. Manual error analysis shows the strengths and weaknesses of both approaches.

1 Introduction

A core prediction task in interactive machine translation (MT) is to complete a partial translation (Ortiz-Martínez et al., 2009; Koehn et al., 2014). Sentence completion enables interfaces that are richer than basic post-editing of MT output. For example, the translator can receive updated suggestions after each word typed (Langlais et al., 2000). However, we show that completing partial translations by naïve constrained decoding—the standard in prior work—yields poor suggestion quality. We describe new phrase-based objective functions, learning techniques, and inference algorithms for the sentence completion task.[1] We then compare this improved phrase-based system to a state-of-the-art recurrent neural translation system in large-scale English-German experiments.

A system for completing partial translations takes as input a source sentence and a prefix of the target sentence. It predicts a suffix: a sequence of tokens that extends the prefix to form a full sentence. In an interactive setting, the first words of the suffix are critical; these words are the focus of the user's attention and can typically be appended to the translation with a single keystroke. We introduce a tuning metric that scores correctness of the whole suffix, but is particularly sensitive to these first words.

Phrase-based inference for this task involves aligning the prefix to the source, then generating the suffix by translating the unaligned words. We describe a beam search strategy and a hierarchical joint model of alignment and translation that together improve suggestions dramatically. For English-German news, next-word accuracy increases from 28.5% to 41.2%.

An interactive MT system could also display multiple suggestions to the user. We describe an algorithm for efficiently finding the n-best next words directly following a prefix and their corresponding best suffixes. Our experiments show that this approach to n-best list extraction, combined with our other improvements, increased next-word suggestion accuracy of 10-best lists from 33.4% to 55.5%.

We also train a recurrent neural translation system to maximize the conditional likelihood of the next word following a translation prefix, which is both a standard training objective in neural translation and an ideal fit for our task. This neural system provides even more accurate predictions than our improved phrase-based system. However, inference is two orders of magnitude slower, which is prob-

[1] Code available at:
https://github.com/stanfordnlp/phrasal

lematic for an interactive setting. We conclude with a manual error analysis that reveals the strengths and weaknesses of both the phrase-based and neural approaches to suffix prediction.

2 Evaluating Suffix Prediction

Let $\mathcal{F}$ and $\mathcal{E}$ denote the set of all source and target language strings, respectively. Given a source sentence $f \in \mathcal{F}$ and target prefix $e_p \in \mathcal{E}$, a predicted suffix $e_s \in \mathcal{E}$ can be evaluated by comparing the full sentence $e = e_p e_s$ to a reference e^*. Let e^*_s denote the suffix of the reference that follows e_p.

We define three metrics below that score translations by the characteristics that are most relevant in an interactive setting: the accuracy of the first words of the suffix and the overall quality of the suffix. Each metric takes example triples (f, e_p, e^*) produced during an interactive MT session in which e_p was generated in the process of constructing e^*.

A *simulated* corpus of examples can be produced from a parallel corpus of (f, e^*) pairs by selecting prefixes of each e^*. An *exhaustive* simulation selects all possible prefixes, while a *sampled* simulation selects only k prefixes uniformly at random for each e^*. Computing metrics for exhaustive simulations is expensive because it requires performing suffix prediction inference for every prefix: $|e^*|$ times for each reference.

Word Prediction Accuracy (WPA) or *next-word accuracy* (Koehn et al., 2014) is 1 if the first word of the predicted suffix e_s is also the first word of reference suffix e^*_s, and 0 otherwise. Averaging over examples gives the frequency that the word following the prefix was predicted correctly. In a sampled simulation, all reference words that follow the first word of a sampled suffix are ignored by the metric, so most reference information is unused.

Number of Predicted Words (#prd) is the maximum number of contiguous words at the start of the predicted suffix that match the reference. Like WPA, this metric is 0 if the first word of e_s is not also the first word of e^*_s. In a sampled simulation, all reference words that follow the first mis-predicted word in the sampled suffix are ignored. While it is possible that the metric will require the full reference suffix, most reference information is unused in practice.

Prefix-BLEU (pxBLEU): BLEU (Papineni et al., 2002) is computed from the geometric mean of clipped n-gram precisions $prec_n(\cdot, \cdot)$ and a brevity penalty $BP(\cdot, \cdot)$. Given a sequence of references $E^* = e^*_1, \ldots, e^*_t$ and corresponding predictions $E = e_1, \ldots, e_t$,

$$\textsc{Bleu}(E, E^*) = BP(E, E^*) \cdot \prod_{n=1}^{4} prec_n(E, E^*)^{\frac{1}{4}}$$

Ortiz-Martínez et al. (2010) use BLEU directly for training an interactive system, but we propose a variant that only scores the predicted suffix and not the input prefix. The pxBLEU metric computes $\textsc{Bleu}(\hat{E}, \hat{E}^*)$ for the following constructed sequences $\hat{E}$ and $\hat{E}^*$:

- For each (f, e_p, e^*) and suffix prediction e_s, $\hat{E}$ includes the full sentence $e = e_p e_s$.
- For each (f, e_p, e^*), $\hat{E}^*$ is a masked copy of e^* in which all prefix words that do not match any word in e are replaced by null tokens.

This construction maintains the original computation of the brevity penalty, but does not include the prefix in the precision calculations. Unlike the two previous metrics, the pxBLEU metric uses all available reference information.

In order to account for boundary conditions, the reference e^* is masked by the prefix e_p as follows: we replace each of the first $|e_p - 3|$ words with a null token e_{null}, unless the word also appears in the suffix e^*_s. Masking retains the last three words of the prefix so that the first words after the prefix can contribute to the precision of all n-grams that overlap with the prefix, up to $n = 4$. Words that also appear in the suffix are retained so that their correct prediction in the suffix can contribute to those precisions, which would otherwise be clipped.

2.1 Loss Functions for Learning

All of these metrics can be used as the tuning objective of a phrase-based machine translation system. Tuning toward a sampled simulation that includes one or two prefixes per reference is much faster than using an exhaustive set of prefixes. A linear combination of these metrics can be used to trade off the relative importance of the full suffix and the words immediately following the prefix. With a combined metric, learning can focus on these words while using all available information in the references.

2.2 Keystroke Ratio (KSR)

In addition to these metrics, suffix prediction can be evaluated by the widely used keystroke ratio (KSR) metric (Och et al., 2003). This ratio assumes that

any number of characters from the beginning of the suggested suffix can be appended to the user prefix using a single keystroke. It computes the ratio of key strokes required to enter the reference interactively to the character count of the reference. Our MT architecture does not permit tuning to KSR.

Other methods of quantifying effort in an interactive MT system are more appropriate for user studies than for direct evaluation of MT predictions. For example, measuring pupil dilation, pause duration and frequency (Schilperoord, 1996), mouse-action ratio (Sanchis-Trilles et al., 2008), or source difficulty (Bernth and McCord, 2000) would certainly be relevant for evaluating a full interactive system, but are beyond the scope of this work.

3 Phrase-Based Inference

In the log-linear approach to phrase-based translation (Och and Ney, 2004), the distribution of translations $e \in \mathcal{E}$ given a source sentence $f \in \mathcal{F}$ is:

$$p(e|f;w) = \sum_{\substack{r: \\ src(r)=f \\ tgt(r)=e}} \frac{1}{Z(f)} \exp\left[w^\top \phi(r)\right] \quad (1)$$

Here, r is a phrasal derivation with source and target projections $src(r)$ and $tgt(r)$, $w \in \mathbb{R}^d$ is the vector of model parameters, $\phi(\cdot) \in \mathbb{R}^d$ is a feature map, and $Z(f)$ is an appropriate normalizing constant.

For the same model, the distribution over suffixes $e_s \in \mathcal{E}$ must also condition on a prefix $e_p \in \mathcal{E}$:

$$p(e_s|e_p, f;w) = \sum_{\substack{r: \\ src(r)=f \\ tgt(r)=e_p e_s}} \frac{1}{Z(f)} \exp\left[w^\top \phi(r)\right] \quad (2)$$

In phrase-based decoding, the best scoring derivation r given a source sentence f and weights w is found efficiently by beam search, with one beam for every count of source words covered by a partial derivation (known as the *source coverage cardinality*). To predict a suffix conditioned on a prefix by constrained decoding, Barrachina et al. (2008) and Ortiz-Martínez et al. (2009) modify the beam search by discarding hypotheses (partial derivations) that do not match the prefix e_p.

We propose *target beam search*, a two-step inference procedure. The first step is to produce a phrase-based alignment between the target prefix and a subset of the source words. The target is aligned left-to-right by appending aligned phrase pairs. However, each beam is associated with a target word count, rather than a source word count. Therefore, each beam contains hypotheses for a fixed prefix of target words. Phrasal translation candidates are bundled and sorted with respect to each target phrase rather than each source phrase. Crucially, the source distortion limit is not enforced during alignment, so that long-range reorderings can be analyzed correctly.

The second step generates the suffix using standard beam search.[2] Once the target prefix is completely aligned, each hypothesis from the final target beam is copied to an appropriate source beam. Search starts with the lowest-count source beam that contains at least one hypothesis. Here, we re-instate the distortion limit with the following modification to avoid search failures: The decoder can always translate any source position before the last source position that was covered in the alignment phase.

3.1 Synthetic Phrase Pairs

The phrase pairs available during decoding may not be sufficient to align the target prefix to the source. Pre-compiled phrase tables (Koehn et al., 2003) are typically pruned, and dynamic phrase tables (Levenberg et al., 2010) require sampling for efficient lookup.

To improve alignment coverage, we include additional *synthetic phrases* extracted from word-level alignments between the source sentence and target prefix inferred using unpruned lexical statistics.

We first find the intersection of two directional word alignments. The directional alignments are obtained similar to IBM Model 2 (Brown et al., 1993) by aligning the most likely source word to each target word. Given a source sequence $f = f_1 \dots f_{|f|}$ and a target sequence $e = e_1 \dots e_{|e|}$, we define the alignment $a = a_1 \dots a_{|e|}$, where $a_i = j$ means that e_i is aligned to f_j. The likelihood is modeled by a single-word lexicon probability that is provided by our translation model and an alignment probability modeled as a Poisson distribution $Poisson(k, \lambda)$ in the distance to the diagonal.

$$a_i = \operatorname*{arg\,max}_{j \in \{1,\dots,|f|\}} p(a_i = j|f, e) \quad (3)$$

$$p(a_i = j|f, e) = p(e_i|f_j) \cdot p(a_i|j) \quad (4)$$

$$p(e_i|f_j) = \frac{cnt(e_i, f_j)}{cnt(f_j)} \quad (5)$$

$$p(a_i|j) = Poisson(|a_i - j|, 1.0) \quad (6)$$

[2] We choose cube pruning (Huang and Chiang, 2007) as the beam-filling strategy.

Here, $cnt(e_i, f_j)$ is the count of all word alignments between e_i and f_j in the training bitext, and $cnt(f_j)$ the monolingual occurrence count of f_j.

We perform standard phrase extraction (Och et al., 1999; Koehn et al., 2003) to obtain our synthetic phrases, whose translation probabilities are again estimated based on the single-word probabilities $p(e_i|f_j)$ from our translation model. Given a synthetic phrase pair (e, f), the phrase translation probability is computed as

$$p(e|f) = \prod_{1 \leq i \leq |e|} \max_{1 \leq j \leq |f|} p(e_i|f_j) \quad (7)$$

Additionally, we introduce three indicator features that count the number of synthetic phrase pairs, source words and target words, respectively.

4 Tuning

In order to tune the model for suffix prediction, we optimize the weights w in Equation 2 to maximize the metrics introduced in Section 2. Model tuning is performed with AdaGrad (Duchi et al., 2011), an online subgradient method. It features an adaptive learning rate and comes with good theoretical guarantees. See Green et al. (2013) for the details of applying AdaGrad to phrase-based translation.

The same model scores both alignment of the prefix and translation of the suffix. However, different feature weights may be appropriate for scoring each step of the inference process. In order to learn different weights for alignment and translation within a unified joint model, we apply the hierarchical adaptation method of Wuebker et al. (2015), which is based on *frustratingly easy domain adaptation* (FEDA) (Daumé III, 2007). We define three sub-segment domains: PREFIX, OVERLAP and SUFFIX. The PREFIX domain contains all phrases that are used for aligning the prefix with the source sentence. Phrases that span both prefix and suffix additionally belong to the OVERLAP domain. Finally, once the prefix has been completely covered, the SUFFIX domain applies to all phrases that are used to translate the remainder of the sentence. The ROOT domain spans the entire phrasal derivation.

Formally, given a set of domains $\mathcal{D} = \{\text{ROOT}, \text{PREFIX}, \text{OVERLAP}, \text{SUFFIX}\}$, each feature is replicated for each domain $d \in \mathcal{D}$. These replicas can be interpreted as domain-specific "offsets" to the baseline weights. For an original feature vector ϕ with a set of domains $D \subseteq \mathcal{D}$, the replicated feature vector contains $|\mathcal{D}|$ copies f_d of each feature $f \in \phi$, one for each $d \in \mathcal{D}$.

$$f_d = \begin{cases} f, & d \in D \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

The weights of the replicated feature space are initialized with 0 except for the ROOT domain, where we copy the baseline weights w.

$$w_d = \begin{cases} w, & d \text{ is ROOT} \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

All our phrase-based systems are first tuned without prefixes or domains to maximize BLEU. When tuning for suffix prediction, we keep these baseline weights w_{ROOT} fixed to maintain baseline translation quality and only update the weights corresponding to the PREFIX, OVERLAP and SUFFIX domains.

5 Diverse n-best Extraction

Consider the interactive MT application setting in which the user is presented with an autocomplete list of alternative translations (Langlais et al., 2000). The user query may be satisfied if the machine predicts the correct completion in its top-n output. However, it is well-known that n-best lists are poor approximations of MT structured output spaces (Macherey et al., 2008; Gimpel et al., 2013). Even very large values of n can fail to produce alternatives that differ in the first words of the suffix, which limits n-best KSR and WPA improvements at test time. For tuning, WPA is often zero for every item on the n-best list, which prevents learning.

Fortunately, the prefix can help efficiently enumerate diverse next-word alternatives. If we can find all edges in the decoding lattice that span the prefix e_p and suffix e_s, then we can generate diverse alternatives in precisely the right location in the target. Let $G = (V, E)$ be the search lattice created by decoding, where V are nodes and E are the edges produced by rule applications. For any $w \in V$, let $\text{parent}(w)$ return v s.t. $v, w \in E$, $\text{target}(w)$ return the target sequence e defined by following the next pointers from w, and $\text{length}(w)$ be the length of the target sequence up to w. During decoding, we set parent pointers and also assign monotonically increasing integer ids to each w.

To extract a full sentence completion given an edge $v, w \in E$ that spans the prefix/suffix boundary, we must find the best path to a goal node efficiently. To do this, we sort V in reverse topological order and set forward pointers from each node v to the

Algorithm 1 Diverse n-best list extraction

Require: Lattice $G = (V, E)$, prefix length P
$M = []$ ▷ Marked nodes
for $w \in V$ in reverse topological order **do**
  $v = \text{parent}(w)$ ▷ $v, w \in E$
  if $\text{length}(v) \leq P$ and $\text{length}(w) > P$ **then**
    Add w to M ▷ Mark node
  end if
  $v.\text{child} = v.\text{child} \oplus w$ ▷ Child pointer update
end for
$N = []$ ▷ n-best target strings
for $m \in M$ **do**
  Add $\text{target}(m)$ to N
end for
return N

child node on the best goal path. During this traversal, we also mark all child nodes of edges that span the prefix/suffix boundary. Finally, we use the parent and child pointers to extract an n-best list of translations. Algorithm 1 shows the full procedure.

6 Neural machine translation

Neural machine translation (NMT) models the conditional probability $p(e|f)$ of translating a source sentence f to a target sentence e. In the *encoder-decoder* NMT framework (Sutskever et al., 2014; Cho et al., 2014), an *encoder* computes a representation $\boldsymbol{s}$ for each source sentence. From that source representation, the *decoder* generates a translation one word at a time by maximizing:

$$\log p(e|f) = \sum_{i=1}^{|e|} \log p\left(e_i|e_{<i}, f, \boldsymbol{s}\right) \qquad (10)$$

The individual probabilities in Equation 10 are often parameterized by a recurrent neural network which repeatedly predicts the next word e_i given all previous target words $e_{<i}$. Since this model generates translations by repeatedly predicting next words, it is a natural choice for the sentence completion task. Even in unconstrained decoding, it predicts one word at a time conditioned on the most likely prefix.

We modified the state-of-the-art English-German NMT system described in (Luong et al., 2015) to conduct a beam search that constrains the translation to match a fixed prefix.[3] As we decode from left to right, the decoder transitions from a constrained prefix decoding mode to unconstrained beam search. In the constrained mode—the next word to predict e_i is known—we set the beam size to 1, aggregate the score of predicting e_i immediately without having to sort the softmax distribution over all words, and feed e_i directly to the next time step. Once the prefix has been consumed, the decoder switches to standard beam search with a larger beam size (12 in our experiments). In this mode, the most probable word e_i is passed to the next time step.

7 Experimental Results

We evaluate our models and methods for English-French and English-German on two domains: *software* and *news*.

The phrase-based systems are built with Phrasal (Green et al., 2014), an open source toolkit. We use a dynamic phrase table (Levenberg et al., 2010) and tune parameters with AdaGrad. All systems have 42 dense baseline features. We align the bitexts with *mgiza* (Gao and Vogel, 2008) and estimate 5-gram language models (LMs) with *KenLM* (Heafield et al., 2013).

The English-French bilingual training data consists of 4.9M sentence pairs from the Common Crawl and Europarl corpora from WMT 2015 (Bojar et al., 2015). The LM was estimated from the target side of the bitext.

For English-German we run large-scale experiments. The bitext contains 19.9M parallel segments collected from WMT 2015 and the OPUS collection (Skadiņš et al., 2014). The LM was estimated from the target side of the bitext and the monolingual Common Crawl corpus (Buck et al., 2014), altogether 37.2B running words.

The software test set includes 10k sentence pairs from the Autodesk post editing corpus[4]. For the news domain we chose the English-French *newstest2014* and English-German *newstest2015* sets provided for the WMT 2016[5] shared task. The translation systems were tuned towards the specific domain, using another 10k segments from the Autodesk data or the *newstest2013* data set, respectively. On the English-French tune set we randomly select one target prefix from each sentence pair for rapid experimentation. On all other test and tune sets we select two target prefixes at random.[6] The

[3] We used the trained models provided by the authors of (Luong et al., 2015) using the codebase at `https://github.com/lmthang/nmt.matlab`.

[4] `https://autodesk.app.box.com/Autodesk-PostEditing`

[5] `http://www.statmt.org/wmt16`

[6] We briefly experimented with larger sets of prefixes and also exhaustive simulation in tuning, but did not observe significant improvements.

selected prefixes remain fixed throughout all experiments.

For NMT, we report results both using a single network and an ensemble of eight models using various attention mechanisms (Luong et al., 2015).

7.1 Phrase-based Results

Tables 1 and 2 show the main phrase-based results. The baseline system corresponds to constrained beam search, which performed best in (Ortiz-Martínez et al., 2009) and (Barrachina et al., 2008), where it was referred to as *phrase-based (PB)* and *phrase-based model (PBM)*, respectively. Our *target beam search* strategy improves all metrics on both test sets.

For English-French, we observe absolute improvements of up to 3.2% pxBLEU, 11.4% WPA and 10.6% KSR. We experimented with four different prefix-constrained tuning criteria: pxBLEU, WPA, #prd, and the linear combination $\frac{(\text{pxBLEU}+\text{WPA})}{2}$. We see that tuning towards prefix decoding increases all metrics. Across our two test sets, the combined metric yielded the most stable results. Here, we obtain gains of up to 3.0% pxBLEU, 3.1% WPA and 2.1% KSR. We continue using the linear combination criterion for all subsequent experiments.

For English-German—the large-scale setting—we observe similar total gains of up to 3.9% pxBLEU, 11.2% WPA and 8.2% KSR. The target beam search procedure contributes the most gain among our various improvements. Table 3 illustrates the differences in the translation output on three example sentences taken from the *newstest2015* test set. It is clearly visible that both target beam search and prefix tuning improve the prefix alignment, which results in better translation suffixes.

7.2 Diverse n-best Results

To improve recall in interactive MT, the user can be presented with multiple alternative sentence completions (Langlais et al., 2000), which correspond to an n-best list of translation hypotheses generated by the prefix-constrained inference procedure. The diverse extraction scheme introduced in section 5 is particularly designed for next-word prediction recall. Table 4 shows results for 10-best lists.

We see that WPA is increased by up to 15.3% by including the 10-best candidates, 11.3% being contributed by our novel diverse n-best extraction. Jointly, target beam search, prefix tuning and diverse n-best extraction lead to an absolute improvement of up to 23.5% over the baseline 10-best oracle. We believe that $n = 10$ suggestions are the maximum number of candidates that should be presented to a user, but we also ran experiments with $n = 3$ and $n = 5$, which would result in an interface with reduced cognitive load. These settings yield 5.5% and 10.0% WPA gains respectively on English-German news.

7.3 Comparison with NMT

We compare this phrase-based system to the NMT system described in Section 6 for English-German. Table 5 shows the results. We observe a clear advantage of NMT over our best phrase-based system when comparing WPA. For pxBLEU, the phrase-based model outperforms the single neural network system on the Autodesk set, but underperforms the ensemble. The neural system substantially outperforms the phrase-based system for both metrics in the *news* domain. This advantage is likely related to baseline full-sentence translation quality. Unconstrained translation quality is 22.4% BLEU for the phrase-based system, 23.2% BLEU for the NMT single network and 26.3% BLEU for the NMT ensemble on English-German news.

In an interactive setting, the system must make predictions in near real-time, so we report average decoding times. We observe a clear time vs. accuracy trade-off; the phrase-based is 10.6 to 31.3 times faster than the single network NMT system and more than 100 times faster than the ensemble. Crucially, the phrase-based system runs on a CPU, while NMT requires a GPU for these speeds. Further, the 10-best oracle WPA of the phrase-based system is higher than the NMT ensemble in both genres.

Following the example of Neubig et al. (2015), we performed a manual analysis of the first 100 segments on the newstest2015 data set in order to qualitatively compare the constrained translations produced by the phrase-based and single network NMT systems. We observe four main error categories in which the translations differ, for which we have given examples in Table 6. NMT is generally better with long-range verb reorderings, which often lead to the verb being dropped by the phrase-based system. E.g. the word *erscheinen* in Ex. 1 and *veröffentlicht* in Ex. 2 are missing in the phrase-based translation. Also, the NMT engine often produces better German grammar and morphological agreement, e.g. *kein* vs. *keine* in Ex. 3 or the verb conjugations in Ex. 4. Especially interesting is that

	tuning criterion	autodesk pxBLEU	WPA	#prd	KSR	newstest2014 pxBLEU	WPA	#prd	KSR
baseline	BLEU	57.9	41.1	1.49	57.8	40.9	38.0	0.96	61.7
target beam search	BLEU	61.0	47.2	1.74	50.3	44.1	49.4	1.35	51.1
+ prefix tuning	$\frac{(\text{pxBLEU}+\text{WPA})}{2}$	64.0	50.3	1.95	48.2	44.7	50.9	1.40	50.5
	pxBLEU	64.0	50.1	1.95	48.2	44.9	50.3	1.38	50.8
	WPA	62.4	50.2	1.88	48.1	43.3	50.5	1.34	51.7
	#prd	63.8	49.7	1.95	48.4	44.1	50.3	1.37	50.7

Table 1: Phrase-based results on the English-French task. We compare the baseline with the target beam search proposed in this work. Prefix tuning is evaluated with four different tuning criteria.

	autodesk pxBLEU	WPA	#prd	KSR	newstest2015 pxBLEU	WPA	#prd	KSR
baseline	58.5	37.8	1.54	64.7	32.1	28.5	0.61	72.7
target beam search	61.2	44.6	1.78	58.0	36.0	39.7	0.84	64.5
+ prefix tuning	62.2	46.0	1.85	57.2	36.0	41.2	0.88	63.7

Table 2: Phrase-based results on English-German, tuned to the linear combination of pxBLEU and WPA.

the NMT system generated the negation *nicht* in the second half of Ex. 3. This word does not have a direct correspondence in the English source, but makes the sentence feel more natural in German. On the other hand, NMT sometimes drops content words, as in Ex. 5, where *middle-class jobs*, *Minnesota* and *Progressive Caucus co-chair* remain entirely untranslated by NMT. Finally, incorrect prefix alignment sometimes leads to incorrect portions of the source sentence being translated after the prefix or even superfluous output by the phrase-based engine, like *, die* in Ex. 6. Table 7 summarizes how many times each of the systems produced a better output than the other, broken down by category.

8 Related Work

Target-mediated interactive MT was first proposed by Foster et al. (1997) and then further developed within the TransType (Langlais et al., 2000) and TransType2 (Esteban et al., 2004; Barrachina et al., 2008) projects. In TransType2, several different approaches were evaluated. Barrachina et al. (2008) reports experimental results that show the superiority of phrase-based models over stochastic finite state transducers and alignment templates, which were extended for the interactive translation paradigm by Och et al. (2003). Ortiz-Martínez et al. (2009) confirm this observation, and find that their own suggested method using partial statistical phrase-based alignments performs on a similar level on most tasks. The approach using phrase-based models is used as the baseline in this paper.

In order to make the interaction sufficiently responsive, Barrachina et al. (2008) resort to search within a word graph, which is generated by the translation decoder without constraints at the beginning of the workflow. A given prefix is then matched to the paths within the word graph. This approach was recently refined with more permissive matching criteria by Koehn et al. (2014), who report strong improvements in prediction accuracy.

Instead of using a word graph, it is also possible to perform a new search for every interaction (Bender et al., 2005; Ortiz-Martínez et al., 2009), which is the approach we have adopted. Ortiz-Martínez et al. (2009) perform the most similar study to our work in the literature. The authors also define prefix decoding as a two-stage process, but focus on investigating different smoothing techniques, while our work includes new metrics, models, and inference.

9 Conclusion

We have shown that both phrase-based and neural translation approaches can be used to complete partial translations. The recurrent neural system provides higher word prediction accuracy, but requires lengthy inference on a GPU. The phrase-based system is fast, produces diverse n-best lists, and provides reasonable prefix-BLEU performance. The complementary strengths of both systems suggest future work in combining these techniques.

We have also shown decisively that simply performing constrained decoding for a phrase-based model is not an effective approach to the task of

1.	**source**	Suddenly I'm at the National Theatre and I just couldn't quite believe it.
	reference	*"Plötzlich war ich im* Nationaltheater und ich konnte es kaum glauben.
	baseline	*"Plötzlich war ich im* Nationaltheater bin und ich konnte es einfach nicht glauben.
	target beam search	*"Plötzlich war ich im* National Theatre und das konnte ich nicht ganz glauben.
	+ prefix tuning	*"Plötzlich war ich im* National Theatre, und ich konnte es einfach nicht glauben.
2.	**source**	"A little voice inside me said, 'You're going to have to do 10 minutes while they fix the computer." "
	reference	*"Eine kleine Stimme sagte mir "Du musst jetzt 10 Minuten überbrücken*, während sie den Computer reparieren." "
	baseline	*"Eine kleine Stimme sagte mir "Du musst jetzt 10 Minuten überbrücken*, sie legen die müssen, während der Computer."
	target beam search	*"Eine kleine Stimme sagte mir "Du musst jetzt 10 Minuten überbrücken* zu tun, während sie den Computer reparieren".
	+ prefix tuning	*"Eine kleine Stimme sagte mir "Du musst jetzt 10 Minuten überbrücken*, während sie den Computer reparieren." "
3.	**source**	Yemeni media report that there is traffic chaos in the capital.
	reference	*Jemenitische Medien berichten von einem Verkehrschaos* in der Hauptstadt.
	baseline	*Jemenitische Medien berichten von einem Verkehrschaos* ist der Verkehr in der Hauptstadt.
	target beam search	*Jemenitische Medien berichten von einem Verkehrschaos* gibt es in der Hauptstadt.
	+ prefix tuning	*Jemenitische Medien berichten von einem Verkehrschaos* in der Hauptstadt.

Table 3: Translation examples from the English-German *newstest2015* test set. We compare the prefix decoding output of the baseline against target beam search both with and without prefix tuning. The prefix is printed in *italics*.

		English-French				**English-German**			
		autodesk		**newstest2014**		**autodesk**		**newstest2015**	
		WPA	KSR	WPA	KSR	WPA	KSR	WPA	KSR
baseline	1-best	41.1	57.8	38.0	61.7	37.8	64.7	28.5	72.7
	10-best	48.6	53.3	42.7	58.5	43.9	60.2	33.4	69.5
target beam search	1-best	50.3	48.2	50.9	50.5	46.0	57.2	41.2	63.7
	10-best	56.8	43.7	54.9	47.3	51.1	53.2	46.6	60.3
	10-best diverse	64.5	39.1	66.2	41.4	57.3	48.4	55.5	54.5

Table 4: Oracle results on the English-French and English-German tasks. We compare the single best result with oracle scores on 10-best lists with standard and diverse n-best extraction on both target beam search with prefix tuning and the phrase-based baseline system.

	autodesk			**newstest2015**		
English-German	pxBleu	WPA	secs / segment	pxBleu	WPA	secs / segment
target beam search	62.2	46.0	0.051	36.0	41.2	0.089
10-best diverse	65.1	57.3		39.5	55.5	
NMT single	61.2	52.3	1.6	39.2	50.4	1.3
NMT ensemble	64.7	54.9	7.7	42.1	53.0	10.0

Table 5: English-German results for the phrase-based system with target beam search and tuned to a combined metric, compared with the recurrent neural translation system. The *10-best diverse* line contains oracle scores from a 10-best list; all other scores are computed for a single suffix prediction per example. The phrase-based timing results include prefix alignment and synthetic phrase extraction.

completing translations. Instead, the learning objective, model, and inference procedure should all be tailored to the task. The combination of these changes can adapt a phrase-based translation system to perform prefix alignment and suffix prediction jointly with fewer search errors and greater accuracy for the critical first words of the suffix. In light of the dramatic improvements in prediction quality that result from the techniques we have described, we look forward to investigating the effect on user experience for interactive translation systems that employ these methods.

1. **source**	He is due to appear in Karratha Magistrates Court on September 23.
reference	*Er* soll am 23. September vor dem Amtsgericht in Karratha erscheinen.
phrase-based	*Er* ist aufgrund der in Karratha Magistrates Court am 23. September.
NMT	*Er* wird am 23. September in Karratah Magistrates Court erscheinen.
2. **source**	The research, funded by the [...], will be published today in the Medical Journal of Australia.
reference	*Die von [...] finanzierte Studie wird heute im Medical Journal of Australia* veröffentlicht.
phrase-based	*Die von [...] finanzierte Studie wird heute im Medical Journal of Australia*.
NMT	*Die von [...] finanzierte Studie wird heute im Medical Journal of Australia* veröffentlicht.
3. **source**	But it is certainly not a radical initiative - at least by American standards.
reference	*Aber es ist* mit Sicherheit keine radikale Initiative - jedenfalls nicht nach amerikanischen Standards.
phrase-based	*Aber es ist* sicherlich kein radikale Initiative - zumindest von den amerikanischen Standards.
NMT	*Aber es ist* gewiss keine radikale Initiative - zumindest nicht nach amerikanischem Maßstab.
4. **source**	Now everyone knows that the labor movement did not diminish the strength of the nation but enlarged it.
reference	*Jetzt wissen alle, dass die Arbeiterbewegung die Stärke der Nation nicht* einschränkte, sondern sie vergrößerte.
phrase-based	*Jetzt wissen alle, dass die Arbeiterbewegung die Stärke der Nation nicht* schmälern, aber vergrößert .
NMT	*Jetzt wissen alle, dass die Arbeiterbewegung die Stärke der Nation nicht* verringert, sondern erweitert hat.
5. **source**	"As go unions, so go middle-class jobs," says Ellison, the Minnesota Democrat who serves as a Congressional Progressive Caucus co-chair.
reference	*"So wie Gewerkschaften sterben*, sterben auch die Mittelklassejobs," sagte Ellison, ein Demokrat aus Minnesota und stellvertretender Vorsitzender des Progressive Caucus im Kongress.
phrase-based	*"So wie Gewerkschaften sterben*, so Mittelklasse-Jobs", sagt Ellison, der Minnesota Demokrat, dient als Congressional Progressive Caucus Mitveranstalter.
NMT	*"So wie Gewerkschaften sterben*, so gehen die gehen," sagt Ellison, der Liberalen, der als Kongresses des eine dient.
6. **source**	The opposition politician, Imran Khan, accuses Prime Minister Sharif of rigging the parliamentary elections, which took place in May last year.
reference	*Der Oppositionspolitiker Imran Khan wirft Premier Sharif vor, bei der Parlamentswahl im Mai vergangenen Jahres betrogen zu haben.*
phrase-based	*Der Oppositionspolitiker Imran Khan wirft Premier Sharif vor, bei der Parlamentswahl im Mai vergangenen Jahres betrogen zu haben.* , die
NMT	*Der Oppositionspolitiker Imran Khan wirft Premier Sharif vor, bei der Parlamentswahl im Mai vergangenen Jahres betrogen zu haben.*

Table 6: Example sentences from the English-German *newstest2015* test set. We compare the prefix decoding output of phrase-based target beam search against the single network neural machine translation (NMT) engine, printing the prefix in *italics*. The examples illustrate the four error categories *missing verb* (Ex. 1 and 2), *grammar / morphology* (Ex. 3 and 4), *missing content words* (Ex. 5) and *alignment* (Ex. 6).

#better	**phrase-based**	**NMT**
missing verb	1	19
grammar / morphology	0	15
missing content words	17	3
alignment	0	6

Table 7: Result of the manual analysis on the first 100 segments of the English-German *newstest2015* test set. For each of the four error categories we count how many times one of the systems produced a better output.

Acknowledgments

Minh-Thang Luong was partially supported by NSF Award IIS-1514268 and partially supported by a gift from Bloomberg L.P.

References

Sergio Barrachina, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, et al. 2008. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28.

Oliver Bender, Saša Hasan, David Vilar, Richard Zens, and Hermann Ney. 2005. Comparison of generation strategies for interactive machine translation. In *EAMT*.

Arendse Bernth and Michael C. McCord. 2000. The effect of source analysis on translation confidence. In *AMTA*.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, et al. 2015. Findings of the 2015 Workshop on Statistical Machine Translation. In *WMT*.

Peter F. Brown, Stephan A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The

Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

Christian Buck, Kenneth Heafield, and Bas van Ooyen. 2014. N-gram counts and language models from the common crawl. In *LREC*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *EMNLP*.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *ACL*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, July.

José Esteban, José Lorenzo, Antonio S. Valderrábanos, and Guy Lapalme. 2004. TransType2 - an innovative computer-assisted translation system. In *ACL*.

George Foster, Pierre Isabelle, and Pierre Plamondon. 1997. Target-Text Mediated Interactive Machine Translation. *Machine Translation*, 12(1–2):175–194.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*.

Kevin Gimpel, Dhruv Batra, Chris Dyer, and Gregory Shakhnarovich. 2013. A systematic exploration of diversity in machine translation. In *EMNLP*.

Spence Green, Sida Wang, Daniel Cer, and Christopher D. Manning. 2013. Fast and adaptive online training of feature-rich translation models. In *ACL*.

Spence Green, Daniel Cer, and Christopher D. Manning. 2014. Phrasal: A toolkit for new directions in statistical machine translation. In *WMT*.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *ACL*.

Liang Huang and David Chiang. 2007. Forest rescoring: Faster decoding with integrated language models. In *ACL*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *NAACL*.

Philipp Koehn, Chara Tsoukala, and Herve Saint-Amand. 2014. Refinements to interactive translation prediction based on search graphs. In *ACL*.

Philippe Langlais, George Foster, and Guy Lapalme. 2000. TransType: a Computer-Aided Translation Typing System. In *NAACL Workshop on Embedded Machine Translation Systems*.

Abby Levenberg, Chris Callison-Burch, and Miles Osborne. 2010. Stream-based translation models for statistical machine translation. In *NAACL*.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*.

Wolfgang Macherey, Franz Josef Och, Ignacio Thayer, and Jakop Uszkoreit. 2008. Lattice-based minimum error rate training for statistical machine translation. In *EMNLP*.

Graham Neubig, Makoto Morishita, and Satoshi Nakamura. 2015. Neural reranking improves subjective quality of machine translation: NAIST at WAT2015. In *2nd Workshop on Asian Translation (WAT2015)*.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–450.

Franz Josef Och, Christoph Tillmann, and Hermann Ney. 1999. Improved alignment models for statistical machine translation. In *EMNLP*.

Franz Josef Och, Richard Zens, and Hermann Ney. 2003. Efficient search for interactive statistical machine translation. In *EACL*.

Daniel Ortiz-Martínez, Ismael García-Varea, and Francisco Casacuberta. 2009. Interactive machine translation based on partial statistical phrase-based alignments. In *RANLP*.

Daniel Ortiz-Martínez, Ismael García-Varea, and Francisco Casacuberta. 2010. Online learning for interactive statistical machine translation. In *NAACL*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*.

Germán Sanchis-Trilles, Daniel Ortiz-Martínez, Jorge Civera, Francisco Casacuberta, Enrique Vidal, and Hieu Hoang. 2008. Improving interactive machine translation via mouse actions. In *EMNLP*.

Joost Schilperoord. 1996. *It's about Time: Temporal Aspects of Cognitive Processes in Text Production*. Rodopi.

Raivis Skadiņš, Jörg Tiedemann, Roberts Rozis, and Daiga Deksne. 2014. Billions of parallel words for free: Building and using the EU bookshop corpus. In *LREC*.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

Joern Wuebker, Spence Green, and John DeNero. 2015. Hierarchical incremental adaptation for statistical machine translation. In *EMNLP*.