A Trainable Transfer-based Machine Translation Approach for Languages with Limited Resources

Alon Lavie, Katharina Probst, Erik Peterson, Stephan Vogel,
Lori Levin, Ariadna Font-Llitjos and Jaime Carbonell

Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA, USA

Email: alavie@cs.cmu.edu

Abstract. We describe a Machine Translation (MT) approach that is specifically designed to enable rapid development of MT for languages with limited amounts of online resources. Our approach assumes the availability of a small number of bi-lingual speakers of the two languages, but these need not be linguistic experts. The bi-lingual speakers create a comparatively small corpus of word aligned phrases and sentences (on the order of magnitude of a few thousand sentence pairs) using a specially designed elicitation tool. From this data, the learning module of our system automatically infers hierarchical syntactic transfer rules, which encode how syntactic constituent structures in the source language transfer to the target language. The collection of transfer rules is then used in our run-time system to translate previously unseen source language text into the target language. We describe the general principles underlying our approach, and present results from an experiment, where we developed a basic Hindi-to-English MT system over the course of two months, using extremely limited resources.

1. Introduction

Corpus-based Machine Translation (MT) approaches such as Statistical Machine Translation (SMT) (Brown et al, 1990), (Brown et al, 1993), (Vogel and Tribble, 2002), (Yamada and Knight, 2001), (Papineni et al, 1998), (Och and Ney, 2002) and Example-based Machine Translation (EBMT) (Brown, 1997), (Sato and Nagao, 1990) have received much attention in recent years, and have significantly improved the state-of-the-art of Machine Translation for a number of different language pairs. These approaches are attractive because they are fully automated, and require orders of magnitude less human labor than traditional rule-based MT approaches. However, to achieve reasonable levels of translation performance, the corpus-based methods require very large volumes of sentence-aligned parallel text for the two languages – on the order of magnitude of a million words or more. Such resources are currently available for only a small number of language pairs. While the amount of online resources for many languages will undoubtedly grow over time, many of the languages spoken by smaller ethnic groups and populations in the world will not have such resources within the foreseeable future. Corpus-based MT approaches will therefore not be effective for such languages for some time to come.

Our MT research group at Carnegie Mellon, under DARPA and NSF funding, has been working on a new MT approach that is specifically designed to enable rapid development of MT for languages with limited amounts of online resources. Our approach assumes the availability of a small number of bi-lingual speakers of the two languages, but these need not be linguistic experts. The bi-lingual speakers create a comparatively small corpus of word aligned phrases and sentences (on the order of magnitude of a few thousand sentence pairs) using a specially designed elicitation tool. From this data, the learning module of our system automatically infers hierarchical syntactic transfer rules, which encode how constituent structures in the source language transfer to the target language. The collection of transfer rules is then used in our run-time system to translate previously unseen source language text into the target language. We refer to this system as the “Trainable Transfer-based MT System”, or in short the XFER system.

In this paper, we describe the general principles underlying our approach, and the current state of development of our research system. We then describe an extensive experiment we conducted to assess the promise of our approach for rapid ramp-up of MT for languages with limited resources: a Hindi-to-English XFER MT system was developed over the course of two months, using extremely limited resources on the Hindi side. We compared the performance of our XFER system with our in-house SMT and EBMT systems, under this limited data scenario. The results of the experiment indicate that under these extremely limited training data conditions, when tested on unseen data, the XFER system significantly outperforms both EBMT and SMT.

We are currently in the middle of yet another two-month rapid-development application of our XFER approach, where we are developing a Hebrew-to-English XFER MT system. Preliminary results from this experiment will be reported at the workshop.

2. Trainable Transfer-based MT Overview

The fundamental principles behind the design of our XFER approach for MT are: (1) that it is possible to automatically learn syntactic transfer rules from limited amounts of word-aligned data; (2) that such data can be elicited from non-expert bilingual speakers of the pair of languages; and (3) that the rules learned are useful for machine translation between the two languages. We assume that one of the two languages involved is a “major” language (such as English or Spanish) for which significant amounts of linguistic resources and knowledge are available.

The XFER system consists of four main sub-systems: elicitation of a word aligned parallel corpus; automatic learning of transfer rules; the run time transfer system; and a statistical decoder for selection of a final translation output from a large lattice of alternative translation fragments produced by the transfer system. The architectural design of the XFER system in a configuration in which translation is performed from a limited-resource language to a major language is shown in Figure 1.

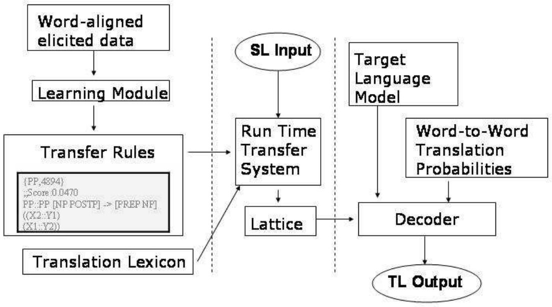
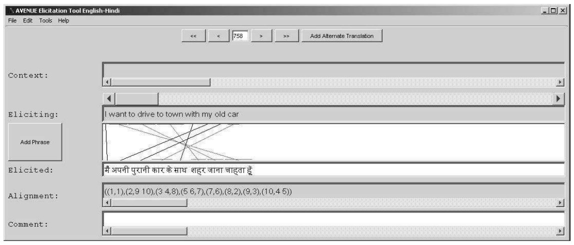


Figure 1. Architecture of the XFER MT System and its Major Components


AVENUE Elicitation Tool English-Hindi
File Edit Tools Help
<<
<
>
>>
Add Alternate Translation
Add Phrase

758

Context:

Eliciting: I want to drive to town with my old car

Elicited: मैं अपनी पुरानी कार के साथ शहर जाना चाहता हूँ

Alignment: ((1,1),(2,9 10),(3 4,8),(5 6,7),(7,6),(8,2),(9,3),(10,4 5))

Comment:

Figure 2. The Elicitation Tool as Used to Translate and Align an English Sentence to Hindi.

3. Elicitation of Word-Aligned Parallel Data

The purpose of the elicitation sub-system is to collect a high quality, word aligned parallel corpus. A specially designed user interface was developed to allow bilingual speakers to easily translate sentences from a corpus of the major language (i.e. English) into their native language (i.e. Hindi), and to graphically annotate the word alignments between the two sentences. Figure 2 contains a snapshot of the elicitation tool, as used in the translation and alignment of an English sentence to Hindi. The informant must be bilingual and literate in the language of elicitation and the language being elicited, but does not need to have knowledge of linguistics or computational linguistics.

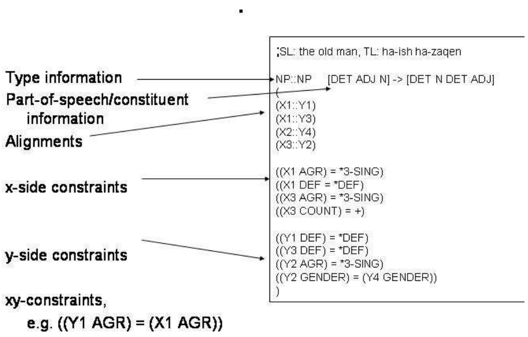
The word-aligned elicited corpus is the primary source of data from which transfer rules are inferred by our system. In order to support effective rule learning, we designed a “controlled” English elicitation corpus. The design of this corpus was based on elicitation principles from field linguistics, and the variety of phrases and sentences attempts to cover a wide variety of linguistic phenomena that the minor language may or may not possess. The elicitation process is organized along “minimal pairs”, which allows us to identify whether the minor languages possesses specific linguistic phenomena (such as gender, number, agreement, etc.). The sentences in the corpus are ordered in groups corresponding to constituent types of increasing levels of complexity. The ordering supports the goal of learning compositional syntactic transfer rules. For example, simple noun phrases are elicited before prepositional phrases and simple sentences, so that during rule learning, the system can detect cases where transfer rules for NPs can serve as components within higher-level transfer rules for PPs and sentence structures. The current controlled elicitation corpus contains about 2000 phrases and sentences. It is by design very limited in vocabulary. A more detailed description of the elicitation corpus, the elicitation process and the interface tool used for elicitation can be found in (Probst et al, 2001), (Probst and Levin, 2002).

4. Automatic Transfer Rule Learning

The rule learning system takes the elicited, word-aligned data as input. Based on this information, it then infers syntactic transfer rules. The learning system also learns the composition of transfer rules. In the compositionality learning stage, the learning system identifies cases where transfer rules for “lower-level” constituents (such as NPs) can serve as components within “higher-level” transfer rules (such as PPs and sentence structures). This process generalizes the applicability of the learned transfer rules and captures the compositional makeup of syntactic correspondences between the two languages. The output of the rule learning system is a set of transfer rules that then serve as a transfer grammar in the run-time system. The transfer rules are comprehensive in the sense that they include all information that is necessary for parsing, transfer, and generation. In this regard, they differ from “traditional” transfer rules that exclude parsing and generation information. Despite this difference, we will refer to them as transfer rules.

The design of the transfer rule formalism itself was guided by the consideration that the rules must be simple enough to be learned by an automatic process, but also powerful enough to allow manually-crafted rule additions and changes to improve the automatically learned rules.

The following list summarizes the components of a transfer rule. In general, the x-side of a transfer rules refers to the source language (SL), whereas the y-side refers to the target language (TL).



```
;SL: the old man, TL: ha-ish ha-zaqen

NP::NP    [DET ADJ N] -> [DET N DET ADJ]
(
(X1::Y1)
(X1::Y3)
(X2::Y4)
(X3::Y2)

((X1 AGR) = *3-SING)
((X1 DEF = *DEF)
((X3 AGR) = *3-SING)
((X3 COUNT) = +)

((Y1 DEF) = *DEF)
((Y3 DEF) = *DEF)
((Y2 AGR) = *3-SING)
((Y2 GENDER) = (Y4 GENDER))
)
```

Figure 3. An Example Transfer Rule along with its Components

1. **Type information:** This identifies the type of the transfer rule and in most cases corresponds to a syntactic constituent type. Sentence rules are of type “S”, noun phrase rules of type “NP”, etc. The formalism also allows for SL and TL type information to be different.
2. **Part-of speech/constituent information:** For both SL and TL, we list a linear sequence of components that constitute an instance of the rule type. These can be viewed as the “right-hand sides” of context-free grammar rules for both source and target language grammars. The elements of the list can be lexical categories, lexical items, and/or phrasal categories.
3. **Alignments:** Explicit annotations in the rule describe how the set of source language components in the rule align and transfer to the set of target language components. Zero alignments and many-to-many alignments are allowed.
4. **X-side constraints:** The x-side constraints provide information about features and their values in the source language sentence. These constraints are used at run-time to determine whether a transfer rule applies to a given input sentence.
5. **Y-side constraints:** The y-side constraints are similar in concept to the x-side constraints, but they pertain to the target language. At run-time, y-side constraints serve to guide and constrain the generation of the target language sentence.
6. **XY-constraints:** The xy-constraints provide information about which feature values transfer from the source into the target language. Specific TL words can obtain feature values from the source language sentence.

Figure 3 shows an example transfer rule along with all its components.

Learning from elicited data proceeds in three stages: the first phase, Seed Generation, produces initial "guesses" at transfer rules. The rules that result from Seed Generation are "flat" in that they specify a sequence of parts of speech, and do not contain any non-terminal or phrasal nodes. The second phase, Compositionality Learning, adds structure using previously learned rules. For instance, it learns that sequences such as "Det N PostP" and "Det Adj N PostP" can be re-written more generally as "NP PostP", as an expansion of PP in Hindi. This generalization process can be done automatically based on the flat version of the rule, and a set of previously learned transfer rules for NPs.

The first two stages of rule learning result in a collection of structural transfer rules that are context-free – they do not contain any unification constraints that limit their applicability. Each of the rules is associated with a collection of elicited examples from which the rule was created. The rules can thus be augmented with a collection of unification constraints, based on specific features that are extracted from the elicited examples. The constraints can then limit the applicability of the rules, so that a rule may succeed only for inputs that satisfy the same unification constraints as the phrases from which the rule was learned. A constraint relaxation technique known as "Seeded Version Space Learning" attempts to increase the generality of the rules by identifying unification constraints that can be relaxed without introducing translation errors. While the first two steps of rule learning are currently well developed, the learning of appropriately generalized unification constraints is still in a preliminary stage of investigation. Detailed descriptions of the rule learning process can be found in (Probst et al, 2003).

5. The Runtime Transfer System

At run time, the translation module translates a source language sentence into a target language sentence. The output of the run-time system is a lattice of translation alternatives. The alternatives arise from syntactic ambiguity, lexical ambiguity, multiple synonymous choices for lexical items in the dictionary, and multiple competing hypotheses from the rule learner.

The runtime translation system incorporates the three main processes involved in transfer-based MT: parsing of the SL input, transfer of the parsed constituents of the SL to their corresponding structured constituents on the TL side, and generation of the TL output. All three of these processes are performed based on the transfer grammar – the comprehensive set of transfer rules that are loaded into the runtime system. In the first stage, parsing is performed based solely on the "x" side of the transfer rules. The implemented parsing algorithm is for the most part a standard bottom-up Chart Parser, such as described in (Allen, 1995). A chart is populated with all constituent structures that were created in the course of parsing the SL input with the source-side portion of the transfer grammar. Transfer and generation are performed in an integrated second stage. A dual TL chart is constructed by applying transfer and generation operations on each and every constituent entry in the SL parse chart. The transfer rules associated with each entry in the SL chart are used in order to determine the corresponding constituent structure on the TL side. At the word level, lexical transfer rules are accessed in order to seed the individual lexical choices for the TL word-level entries in the TL chart. Finally, the set of generated TL output strings that corresponds to the collection of all TL chart entries is collected into a TL lattice, which is then passed on for decoding. A more detailed description of the runtime transfer-based translation sub-system can be found in (Peterson, 2002).

6. Target Language Decoding

In the final stage, a statistical decoder is used in order to select a single target language translation output from a lattice that represents the complete set of translation units that were created for all substrings of the input sentence. The translation units in the lattice are organized according the positional start and end indices of the input fragment to which they correspond. The lattice typically contains translation units of various sizes for different contiguous fragments of input. These translation units often overlap. The lattice also includes multiple word-to-word (or word-to-phrase) translations, reflecting the ambiguity in selection of individual word translations.

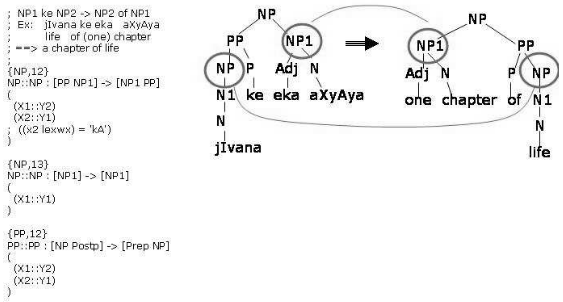
The task of the statistical decoder is to select a linear sequence of adjoining but non-overlapping translation units that maximizes the probability of the target language string given the source language string. The probability model that is used calculates this probability as a product of two factors: a translation model for the translation units and a language model for the target language. The probability assigned to translation units is based on a trained word-to-word probability model. A standard trigram model is used for the target language model.

The decoding search algorithm considers all possible sequences in the lattice and calculates the product of the language model probability and the translation model probability for the resulting sequence of target words. It then selects the sequence which has the highest overall probability. As part of the decoding search, the decoder can also perform a limited amount of re-ordering of translation units in the lattice, when such reordering results in a better fit to the target language model.

7. Construction of the Hindi-to-English System

As part of a DARPA "Surprise Language Exercise", we quickly developed a Hindi-to-English MT system based on our XFER approach over a two-month period. The training and development data for the system consisted entirely of phrases and sentences that were translated and aligned by Hindi speakers using our elicitation tool. Two very different corpora were used for elicitation: our "controlled" typological elicitation corpus and a set of NP and PP phrases that we extracted from the Brown Corpus section of the Penn Treebank. We estimated the total amount of human effort required in collecting, translating and aligning the elicited phrases based on a sample. The estimated time spent on translating and aligning a file (of 200 phrases) was about 8 hours. Translation took about 75% of the time, and alignment about 25%. We estimate the total time spent to be about 700 hours of human labor.

We acquired a transfer grammar for Hindi-to-English transfer by applying our automatic learning module to the corpus of word-aligned data. The learned grammar consists of a total of 327 rules. In a second round of experiments, we assigned probabilities to the rules based on the frequency of the rule (i.e. how many training examples produce a certain rule). We then pruned rules with low probability, resulting in a grammar of a mere 16 rules. As a point of comparison, we also developed a small manual transfer grammar. The manual grammar was developed by two non-Hindi-speaking members of our project, assisted by a Hindi language expert. Our grammar of manually written rules has 70 transfer rules. The grammar includes a rather large verb paradigm, with 58 verb sequence rules, ten recursive noun phrase rules and two prepositional phrase rules. Figure 4 shows an example of recursive NP and PP transfer rules.



```
; NP1 ke NP2 -> NP2 of NP1
; Ex:  jIvana ke eka   aXyAya
;         life   of (one) chapter
; ==> a chapter of life
;
{NP,12}
NP::NP : [PP NP1] -> [NP1 PP]
(
 (X1::Y2)
 (X2::Y1)
;  ((x2 lexwx) = 'kA')
)

{NP,13}
NP::NP : [NP1] -> [NP1]
(
 (X1::Y1)
)

{PP,12}
PP::PP : [NP Postp] -> [Prep NP]
(
 (X1::Y2)
 (X2::Y1)
)
```

Figure 4. Recursive NP and PP Transfer Rules for Hindi to English Translation

In addition to the transfer grammar, the XFER system requires a word-level translation lexicon. The Hindi-to-English lexicon we constructed contains entries from a variety of sources. One source for lexical translation pairs is the elicited corpus itself. The translations pairs can simply be read off from the alignments that were manually provided by Hindi speakers. Because the alignments did not need to be 1-to-1, the resulting lexical translation pairs can have strings of more than one word one either the Hindi or English side or both. Another source for lexical entries was an English-Hindi dictionary provided by the Linguistic Data Consortium (LDC). Two local Hindi experts "cleaned up" a portion of this lexicon, by editing the list of English translations provided for the Hindi words, and leaving only those that were "best bets" for being reliable, all-purpose translations of the Hindi word. The full LDC lexicon was first sorted by Hindi word frequency (estimated from Hindi monolingual text) and the cleanup was performed on the most frequent 12% of the Hindi words in the lexicon. The "clean" portion of the LDC lexicon was then used for the limited-data experiment. This consisted of 2725 Hindi words, which corresponded to about 10,000 translation pairs. This effort took about 3 days of manual labor. To create an additional resource for high-quality translation pairs, we used monolingual Hindi text to extract the 500 most frequent bigrams. These bigrams were then translated into English by an expert in about 2 days. Some judgment was applied in selecting bigrams that could be translated reliably out of context. Finally, our lexicon contains a number of manually written phrase-level rules.

The system we put together also included a morphological analysis module for Hindi input. The morphology module used is the IIIT Morpher (IIIT

Morphology Module). Given a fully inflected word in Hindi, Morpher outputs the root and other features such as gender, number, and tense. To integrate the IIIT Morpher with our system, we installed it as a server.

8. Hindi-to-English Translation Evaluation

The evaluation of our XFER-based Hindi-to-English MT system compares the performance of this system with an SMT system and EBMT system that were trained on the exact same training data as our XFER system. The limited training data consists of:

- 17,589 word-aligned phrases and sentences from the elicited data collection. This includes both our translated and aligned controlled elicitation corpus, and also the translated and aligned uncontrolled corpus of noun phrases and prepositional phrases extracted from the Penn Treebank.
- A Small Hindi-to-English Lexicon: 23,612 "clean" translation pairs from the LDC dictionary.
- A small amount of manually acquired lexical resources (as described above).

The limited data setup includes no additional parallel Hindi-English text. The total amount of bilingual training data was estimated to amount to about 50,000 words.

A small, previously unseen, Hindi text was selected as a test-set for this experiment. The test-set chosen was a section of the data collected at Johns Hopkins University during the later stages of the DARPA Hindi exercise, using a web-based interface. The section chosen consists of 258 sentences, for which four English reference translations are available.

The following systems were evaluated in the experiment:

1. Three versions of the Hindi-to-English XFER system:

   1a. **XFER with No Grammar:** the XFER system with no syntactic transfer rules (i.e. only lexical phrase-to-phrase matches and word-to-word lexical transfer rules, with and without morphology).

   1b. **XFER with Learned Grammar:** The XFER system with automatically learned syntactic transfer rules.

   1c. **XFER with Manual Grammar:** The XFER system with the manually developed syntactic transfer rules.
2. **SMT:** The CMU Statistical MT (SMT) system (Vogel et al, 2003), trained on the limited-data parallel text resources.
3. **EBMT:** The CMU Example-based MT (EBMT) system (Brown, 1997), trained on the limited-data parallel text resources.
4. **MEMT:** A "multi-engine" version that combines the lattices produced by the SMT system, and the XFER system with manual grammar. The decoder then selects an output from the joint lattice.

Performance of the systems was measured using the NIST scoring metric (Doddington, 2002), as well as the BLEU score (Papineni et al, 2002). In order to validate the statistical significance of the differences in NIST and BLEU scores, we applied a commonly used sampling technique over the test set: we randomly draw 258 sentences independently from the set of 258 test sentences (thus sentences can appear zero, once, or more in the newly drawn set). We then calculate scores for all systems on the randomly drawn set (rather than the original set). This process was repeated 10,000 times. Median scores and 95% confidence intervals were calculated based on the set of scores. The results for the various systems tested can be seen in Table 1 below. Figure 5 shows the NIST score results with different reordering windows within the decoder.

Table 1. System Performance Results for the Various Translation Approaches

System	BLEU	NIST
EBMT	0.058	4.22
SMT	0.102 (+/- 0.016)	4.70 (+/- 0.20)
XFER no gra	0.109 (+/- 0.015)	5.29 (+/- 0.19)
XFER learn gra	0.112 (+/- 0.016)	5.32 (+/- 0.19)
XFER man gra	0.135 (+/- 0.018)	5.59 (+/- 0.20)
MEMT	0.136 (+/- 0.018)	5.65 (+/- 0.21)

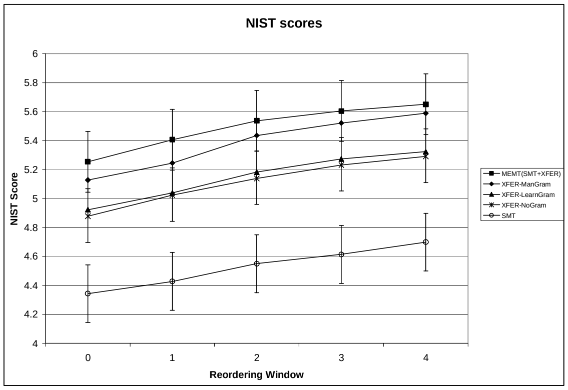


Figure 5. Results by NIST Score with Various Reordering Windows.

The results of the experiment clearly show that under the very limited data training scenario that we constructed, the XFER system, with all its variants, significantly outperformed the SMT system. While the scenario of this experiment was clearly and intentionally more favorable towards our XFER approach, we see these results as a clear validation of the utility and effectiveness of our transfer approach in other scenarios where only very limited amounts of parallel text and other online resources are available.

The results of the comparison between the various versions of the XFER system also show interesting trends, although the statistical significance of some of the differences is not very high. XFER with the manually developed transfer rule grammar clearly outperformed (with high statistical significance) XFER with no grammar and XFER with automatically learned grammar. XFER with automatically learned grammar is slightly better than XFER with no grammar, but the difference is statistically not very significant. We take these results to be highly encouraging, since both the manually written and automatically learned grammars were very limited in this experiment. The automatically learned rules only covered NPs and PPs, whereas the manually developed grammar mostly covers verb constructions. While our main objective is to infer rules that perform comparably to hand-written rules, it is encouraging that the hand-written grammar rules result in a big performance boost over the no-grammar system, indicating that there is much room for improvement. If the learning algorithms are improved, the performance of the overall system can also be improved significantly.

The significant effects of decoder reordering are also quite interesting. On one hand, we believe this indicates that various more sophisticated rules could be learned, and that such rules could better order the English output, thus reducing the need for re-ordering by the decoder. On the other hand, the results indicate that some of the "burden" of reordering can remain within the decoder, thus possibly compensating for weaknesses in rule learning.

Finally, we were pleased to see that the consistently best performing system was our multi-engine configuration, where we combined the translation hypotheses of the SMT and XFER systems together into a common lattice and applied the decoder to select a final translation. The MEMT configuration outperformed the best pure XFER system with reasonable statistical confidence. Obtaining a multi-engine combination scheme that consistently outperforms all the individual MT engines has been notoriously difficult in past research. While the results we obtained here are for a unique data scenario, we hope that the framework applied here for multi-engine integration will prove to be effective for a variety of other scenarios as well. The inherent differences between the XFER and SMT approaches should hopefully make them complementary in a broad range of data scenarios.

9. Conclusions

In summary, we feel that we have made significant steps towards the development of a statistically grounded transfer-based MT system with: (1) rules that are scored based on a well-founded probability model; and (2) strong and effective decoding that incorporates the most advanced techniques used in SMT decoding. Our work complements recent work by other groups on improving translation performance by incorporating models of syntax into traditional corpus-driven MT methods. The focus of our approach, however, is from the "opposite end of the spectrum": we enhance the performance of a syntactically motivated rule-based approach to MT, using strong statistical methods. We find our approach particularly suitable for languages with very limited data resources.

Acknowledgments

This research was funded in part by the DARPA TIDES program and by NSF grant number IIS-0121-631. We would like to thank our team of Hindi-English bilingual speakers in Pittsburgh and in India that conducted the data collection for the research work reported in this paper. Special thanks to Richard Cohen (University of Pittsburgh) for providing Hindi linguistics expertise on this project.

References

The Brown Corpus. http://www.hit.uib.no/icame/brown/bcm.html.

The Johns Hopkins University Hindi translation webpage. http://nlp.cs.jhu.edu/ hindi.

Morphology module from IIIT. http://www.iiit.net/ltrc/morph/index.htm.

The Penn Treebank. http://www.cis.upenn.edu/~treebank/home.html.

Allen, J. 1995. Natural Language Understanding, Second Edition, Benjamin Cummings.

Brown, P., Cocke, J., Della Pietra, V., Della Pietra, S., Jelinek, F., Lafferty, J., Mercer, R., and Roossin,

P. 1990. A Statistical Approach to Machine Translation. Computational Linguistics 16(2), 79-85.

Brown, P., Della Pietra, V., Della Pietra, S., and Mercer, R. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. Computational Linguistics 19(2), 263-311.

Brown, R. 1997. Automated Dictionary Extraction for Knowledge-free Example-based Translation. In Proceedings of International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-1997). 111-118.

Doddington, G. 2002. Automatic Evaluation of Machine Translation Quality using N-gram Cooccurrence Statistics. In Proceedings of Human Language Technologies Conference (HLT-2002). 128-132.

Och, F. J. and Ney, H. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In Proceedings of 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002). Philadelphia, PA.

Papineni, K., Roukos, S., Ward, T. and Zhu, W. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. . In Proceedings of 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002). Philadelphia, PA.

Papineni, K., Roukos, S., and Ward, T. 1998. Maximum Likelihood and Discriminative Training of Direct Translation Models. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP-98). 189-192.

Peterson, E. 2002. Adapting a Transfer Engine for Rapid Machine Translation Development. M.S. Thesis, Georgetown University.

Probst, K., Brown, R., Carbonell, J., Lavie, A., Levin, L., and Peterson, E. 2001. Design and Implementation of Controlled Elicitation for Machine Translation of Low-Density Languages. In Workshop MT2010 at Machine Translation Summit VIII.

Probst, K. and Levin, L. 2002. Challenges in Automated Elicitation of a Controlled Bilingual Corpus. In Theoretical and Methodological Issues in Machine Translation 2002 (TMI-02).

Probst, K., Levin, L., Peterson, E., Lavie, A., and Carbonell, J. 2003. MT for Resource-Poor Languages using Elicitation-based Learning of Syntactic Transfer Rules. Machine Translation. To appear.

Sato, S. and Nagao, M. 1990. Towards Memory-based Translation. In Proceedings of COLING-90. 247-252.

Vogel, S. and Tribble, A. 2002. Improving Statistical Machine Translation for a Speech-to-Speech Translation task. In Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP-02).

Vogel, S., Zhang, Y., Tribble, A., Huang, F., Venugopal, A., Zhao, B., and Waibel, A. 2003. The CMU Statistical Translation System. In Proceedings of MT Summit IX, New Orleans, LA..

Yamada, K. and Knight, K. 2001. A Syntax-based Statistical Translation Model. In Proceedings of the 39th Meeting of the Association for Computational Linguistics (ACL-01).