EBMT for SMT: A New EBMT-SMT Hybrid

James Smith† and Stephen Clark*
†Oxford University Computing Laboratory
*University of Cambridge Computer Laboratory

DCU, November 2009
Strengths and Weaknesses of EBMT and SMT

- **EBMT**
  - Translation is good when good example(s) exist
  - Translation is poor when no good examples

- **SMT**
  - Much better at generalising over example base
  - Not able to directly exploit a good example
A New Combination?

- Use EBMT to translate parts of sentence for which it is confident
- Pass the partial translation to Moses and let it do the rest:
  - translate the remainder
  - perform any reordering
- Moses performs the recombination/generation
A New Combination?

- Use EBMT to translate parts of sentence for which it is confident
- Pass the partial translation to Moses and let it do the rest:
  - translate the remainder
  - perform any reordering
- Moses performs the recombination/generation
- Very hard to beat the Moses baseline
This Work

- **EPSRC** Case studentship with Sharp Laboratories of Europe – Victor Poznanski, Pete Whitelock

- Focused on the matching phase for the EBMT system:
  - string-based approach
  - dependency-based approach

- Outline:
  - string-based system
  - dependency-based system
  - consequences for EBMT and SMT
General Framework

Function EBMT-SMT
Input: Word-aligned parallel corpus (example base)
  Index over source sentences in example base
  Input sentence in source language
  Moses phrased-based SMT system

Candidates = Filter(Input, Index)
Foreach Candidate in Candidates
  Score(Candidate) = Similarity(Input, Candidate)
BestMatch = argmax Score(Candidate)

Translations = Matches(Input, BestMatch, Alignment)
Output: MosesConstrained(Translations)
String-Based Matching

- Similarity measure for two sentences based on word sequences
- Levenshtein distance (edit distance)
  - counts insertions, deletions, substitutions needed to transform one sentence into another
  - standard dynamic programming algorithm
- Investigated alternatives:
  - WordNet-based substitution cost on nouns
  - Comparison of common n-gram sequences
Index-Based Filter

- Calculating edit distance for all examples is expensive
- Create an index from n-grams (length 1-5) to example sentences
- Filter score favours longer n-gram matches
- Index contains 42,071,791 n-grams for the EuroParl data set
- Saving of 98% (with an example base of 100,000 sentences and an input of 500 sentences)
Alignment-Based Translation

Input: the clever man likes reading books

- Only allow n-gram matches above a certain length
- Use translation of man likes reading

Stephen Clark
EBMT for SMT: A New EBMT-SMT Hybrid
DCU, November 2009
Moses Interface

- Moses already has an XML interface which allows part of the translation to be fixed.
- So using Moses was straightforward for the string-based system:

  Input: the clever man likes reading books
  Output: <span foreign="homme aime lire" span="2,4"/>
Results on English-French Europarl (5-15 words)

Average BLEU Score for Affected Sentences

Minimum Match Size

- Raw match
- Moses
- Our System
Summary of String-Based System

- Filter allows practical EBMT on a large parallel corpus
- String-based hybrid performs worse than Moses
- **String-based matching is similar to phrase-based SMT** (but without the advantages of generalising over many examples)
- Move away from the phrase-based SMT system by using matches based on dependency trees
• Input: On this point we are in agreement
• Can match discontinuous sequences and word order can be different
Comparison to other Work

- We use syntax on the source side only (as part of the EBMT matching phase)
- Others use syntax on both sides, eg as part of a synchronous CFG or TAG, or just use syntax on the target side
Parser

- Use the Clark and Curran CCG parser to produce dependency structures for the source side
- One advantage is that it’s fast (100s of sentences/second)
- Produces labelled dependency trees, including long-range dependencies
General Framework

Function EBMT-SMT
Input: Word-aligned parallel corpus with source sentences parsed
Dependency-based index over source sentences in example base
Parsed input sentence in source language
Moses phrased-based SMT system

Candidates = Filter(Input, Index)
Foreach Candidate in Candidates
  Score(Candidate) = Similarity(Input, Candidate)

BestMatches = Greedy(Input, Candidates)

Translations = Matches(Input, BestMatches, Alignment)
Output: MosesConstrained(Translations)
We would like to supply alternative, potentially overlapping, hypotheses to Moses and let it select the best ones.

Matches can be for discontinuous sequences on the source side.

We don't want Moses to choose one part of a discontinuous match and not the other.
The Linked Tag

```xml
<linked>
  <span1 foreign="homme intelligent" span="1,2"/>
  <span2 foreign="aime lire" span="4,5"/>
</linked>

<linked>
  <span3 foreign="aime lire" span="4,5"/>
  <span4 foreign="grands livres" span="7,8"/>
</linked>

the clever man certainly likes reading really big books
```
Results on English-French Europarl

![Graph showing BLEU score relative to pure SMT for string-based and syntax-based methods. The x-axis represents the minimum match size in words, ranging from 1 to 8. The y-axis represents the BLEU score, ranging from -9 to 0. The graph shows that the BLEU score for both methods decreases as the minimum match size increases.](image)
Effect of Dependency Threshold

![Graph showing BLEU score relative to pure SMT for different minimum match sizes (dependencies). The graph compares 1 Word, 3 Words, and 5 Words match sizes, with BLEU scores ranging from -7 to 0 and minimum match sizes from 1 to 7.](image-url)
Will there be environmental and social dumping?
Good EBMT Example
Good EBMT Example

and

conj

conj

social

environmental

... the economic , environmental and social dimensions .

... l , aspect économique , environnemental et social .

Stephen Clark

EBMT for SMT: A New EBMT-SMT Hybrid

DCU, November 2009
Really Good EBMT Example

does anyone wish to speak in favour of this request?
Conclusion

- Problems with the EBMT system:
  - noisy alignments
  - noisy parser output
  - EBMT choices need to account for the uncertainty better and be more closely integrated with Moses

- Any ideas we had to improve the EBMT system moved us closer to the SMT model!
Conclusion

- Problems with the EBMT system:
  - noisy alignments
  - noisy parser output
  - EBMT choices need to account for the uncertainty better and be more closely integrated with Moses

- Any ideas we had to improve the EBMT system moved us closer to the SMT model!

- What can EBMT offer (hierarchical) phrase-based SMT that it doesn’t already have?