Typed Semantic and Syntactic Frames for Statistical Machine Translation

Abstract

We propose a new approach to statistical machine translation driven by typed semantic and syntactic frames. Typed translation frames build upon target and source side linguistic annotations are shown to improve the structure and readability of translations, while also producing significant improvements in BLEU score. We explore typed translation frames built over argument named entity category annotations (e.g., person, org, place), predicate-argument roles as defined by a semantically orientated dependency parse representation (e.g., nsubj, dobj) and the linkage between predicates. Our technique is shown to outperform a competitive hierarchical translation model by 1.0 to 1.5 BLEU points on Chinese to English NIST evaluation sets.

1 Introduction

• MT systems don’t do a good job of capturing predicate argument semantics

• This makes sense, given that they have many features that score how well things locally fit together but do not do a good job of scoring larger meaningful chunks
  – phrase table - local source-to-target adequacy
  – lm language model - target fluency
  – phrase based re-ordering features - local systematic reordering

• But, scoring larger meaningful chunks is hard. No longer just local target side fluency (n-gram lm) or source-to-target adequacy (phrase table), but rather structured meaning of the material.
  – structure meaning is who did what to whom: the relationship between different parts of material

• Early transfer based systems captured this, but they do not scale well to new languages and genres - depend on many hand written rules ¹
  – if possible find an old paper that specifically looks this argument structure - cite them and have one or two sentence they say what they do.

• What would it take to extract transfer rules that capture the relationship between different parts of the material?
  – rules that have argument slots - hierarchical and syntax based translation already gives us this - great!
  – but, hiero’s untyped slots reduce to just gap based rather than meaning based translation.

   * As seen in figure (1) - even if something is an agent in the source - nothing prevents it from sliding into a gap meant for a patient or other role in the sentence

   • figure (1) has a short mistranslated sentence the confuses the roles of two parts of the material. Below the sentence there are the transfer rules that were used to construct the bad translation. In the text reference the one rule where if it had typing the error go away.

   • saw stuff like this during WMT manual evaluation

¹ cite systran style hypebrid rule/stat-MT system as one way of improving this
for this work, this is motivating example, as it’s the problem that is being directly and obviously addressed by this work.

Syntactic typing, as in both phrase-structure and liben shen’s dependency MT is better. But, (basic?) syntactic argument typing such as NN still allows material to slide into inappropriate slots. For example, POS types would not fix example since the confused arguments would both have the typing NN not say agent/patient.

In this paper, we explore using typed semantic and syntactic frames for statistical machine translation

We show that this helps produce translations with better argument structure since it explicitly models the structural relationships between different parts of the material. This results in predicate argument like transfer rules, but whose extraction is completely data-driven.

The remainder of this paper is organized as follows: frame base translation model, experiments, related work, conclusion

2 Semantic Frame Based Translation

- two types of rules source drive and target driven
- both types are extracted from a type annotated bitext
- allow any source or target annotation over the bitext to inform machine translation
  - source driven - more powerful since we can also annotate the source prior to decoding.
  - target driven - depends on context free assignment to words and phrases to types.
- e.g., Syntactic Role, Semantic Role, NER categories
- example of two rules with different same set of argument types but in different order
  - as seen in figure 1 the decode can be confused / doesn’t model X
  - type annotation trivially fix this

2.1 Rule Extraction

- Like heirarchical machine translation systems first extract all phrases that are consistent with alignments and then look for instances of nesting

- as shown in figure X, transform rule into typed frame based transfer rule …
  - figure starts with two heiro style rules and then has step by step transformation to typed frame
  - one predicate like rule
  - one argument filler like rule

- Annotate gaps with types

  - Gaps in rules are checked to overlap with source or target typing information
  - if gap overlap source or target type annotation, modify the rule by changing the non-terminal from an untuped ’X’ to the typing assigned by annotation

- Assign types to rules

  - rules are check for overlap with source or target type annotations
  - if they match, change RHS of rule to type annotation

2.2 Contextualization and Generalization of Rules

- match feature fires for source side rules that fires if source side typing of rule matches typing assigned by source side analysis prior to decoding (source side analysis is a soft constraint)

- typing patch rules: if no rule matches source-side analysis typing, material can still plug into predicates that require the desired typing. Online generation of rules that convert available translation options to the desired type, flagged by generation rule penalty feature.

2.3 Decoding

- Resulting rules can be used with any algorithm SCFG based translation. In this work we use CYK with cube-pruning for LM-integration.
3 Experiments

machinery
  • decoder cdec
  • n-gram lm trained using SRILM
  • trained using MERT
  • FDA data selection
  • NER Stanford
  • dep parser Stanford

experiment
  • POS typing
  • dependency argument annotation
  • dependency linkage annotation
  • NER annotation

4 Results
  • table 1 for syntactic role and predicate linkage
  • table 2 for NER

5 Related Work
  • galley (isi) syntax based mt phrase structure non-terminal typing
  • vogal soft phrase structure non-terminal typing for heiro
  • liben shen POS typing for dependency translation
  • quirk (microsoft treelet) head based typing.

6 Conclusion
## Table 1: Syntactic Typing

<table>
<thead>
<tr>
<th>Model</th>
<th>Tune</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part-of-speech typing</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
<tr>
<td>Role typing</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
<tr>
<td>Linkage typing</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
<tr>
<td>Role + Linkage typing</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
<tr>
<td>Heirarchical baseline</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
</tbody>
</table>

## Table 2: NER Typing

<table>
<thead>
<tr>
<th>Model</th>
<th>Tune</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>NER typing</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
<tr>
<td>NER+Role+Linkage typing</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
<tr>
<td>Heirarchical baseline</td>
<td>x05</td>
<td>x06</td>
<td>x08</td>
</tr>
</tbody>
</table>