Lexical Syntax for Statistical Machine Translation

Hany Hassan

DCU & IBM
In collaboration with: Andy Way and Khalil Sima’an
Outline

Introduction

Syntax for Phrase-based SMT

Supertagged Phrase-based SMT

From Supertagged to Dependency-based Language Models

Incremental Dependency-based Language Model (IDLM)

DTM2

Dependency-based SMT

Future Work

Conclusion and Discussion
Outline

Introduction

Syntax for Phrase-based SMT

Supertagged Phrase-based SMT

From Supertagged to Dependency-based Language Models

Incremental Dependency-based Language Model (IDLM)

DTM2

Dependency-based SMT

Future Work

Conclusion and Discussion
Can linguistic syntax improve PBSMT?

(Koehn et al 2003) tried to impose syntactic constituents on phrase extraction

Hierarchical Phrase structure (Chiang 2005)

- Allows for hierarchical phrases
- Handles a range of reordering problems
- The syntax induced is not linguistically motivated.

Syntactified target phrases (Marcu et. al. 2006)

- Induces millions of xRs rules from parallel corpus
- Mismatch between constituent (xRs) and phrase
- Subtrees for phrases: leads to spurious ambiguity in phrase table

Do subtrees/constituents fit well with phrases?
Can linguistic syntax improve PBSMT?

(Koehn et al 2003) tried to impose syntactic constituents on phrase extraction

Hierarchical Phrase structure (Chiang 2005)
- Allows for hierarchical phrases
- Handles a range of reordering problems
- The syntax induced is not linguistically motivated.

Syntactified target phrases (Marcu et. al. 2006)
- Induces millions of xRs rules from parallel corpus
- Mismatch between constituent (xRs) and phrase
- Subtrees for phrases: leads to spurious ambiguity in phrase table

Do subtrees/constituents fit well with phrases?
Do subtrees/constituents fit well with phrases?

The president meets Saudi economic officials

- S
  - VP
    - V
    - NP
    - NP
  - NP
- تَستقبل الرئيس
- اقتصاديين مسؤولين
- سعوديين
Spurious Ambiguity:
Do subtrees/constituents fit well with phrases?

Why subtress do not match SMT phrases?

- Syntactic constituents mismatch phrase concept
- Which level of tree structure should be incorporated?
- This leads to spurious ambiguity

Can linguistic syntax improve PBSMT?

Trees/constituents do NOT fit well with phrases

What syntax does fit then?
Lexical Syntax (Supertags) Matches Phrases

- **NP**
  - **D**
  - **NP***
  - **the**
  - **β1**

- **N**
  - **N**
  - **purchase**
  - **β2**

- **NP**
  - **S**
  - **VP**
  - **includes**
  - **α2**

- **NP**
  - **S**
  - **V**
  - **price**
  - **α1**

- **NP**
  - **N**
  - **taxes**
  - **α1**
Lexical Syntax (Supertags)

Linguistics offers lexical-syntax (Supertags):

- Lexicalized Tree Adjoining Grammar (LTAG) : (Joshi & Schabes, 1992) & (Srinivas & Joshi, 1999)
- Combinatory Categorical Grammar (CCG) (Steedman, 2000)

Rich lexical categories

- Localizing syntactic dependencies
- Representing predicate argument constraints on the word level
- Markovian language model on the sequence produce almost parsing
- Handful of Combination Operators are used to construct dependency tree
Lexical Syntax (Supertags)

Linguistics offers lexical-syntax (Supertags):

- Lexicalized Tree Adjoining Grammar (LTAG) : (Joshi & Schabes, 1992) & (Srinivas & Joshi, 1999)
- Combinatory Categorical Grammar (CCG) (Steedman, 2000)

Rich lexical categories

- Localizing syntactic dependencies
- Representing predicate argument constraints on the word level
- Markovian language model on the sequence produce almost parsing
- Handful of Combination Operators are used to construct dependency tree
The purchase price includes taxes
The purchase price includes taxes

NP/NP (NP) NP (S\NP)/NP NP
The purchase price includes taxes.
The purchase price includes taxes

\[
\begin{align*}
\text{NP/NP} & \quad (\text{NP}) & \quad \text{NP} & \quad (S\backslash\text{NP})/\text{NP} & \quad \text{NP} \\
\text{NP} & \quad \rightarrow_{\text{FA}} & \quad \text{NP} & \quad \rightarrow_{\text{FA}} & \quad S\backslash\text{NP} \\
\text{NP} & \quad \rightarrow_{\text{FA}} & \quad \text{NP} & \quad \rightarrow_{\text{FA}} & \quad \text{NP}
\end{align*}
\]
The purchase price includes taxes.
The purchase price includes taxes

\[
\frac{\text{NP/NP} \ (\text{NP}) \ NP \ (\text{S/NP})/\text{NP} \ NP}{\text{NP}} \xrightarrow{\text{FA}} \frac{\text{NP}}{\text{S/NP}} \xrightarrow{\text{FA}} \frac{\text{NP}}{} \xrightarrow{\text{FA}} \text{NP} \xrightarrow{\text{FA}} \text{S} \xrightarrow{\text{BA}}
\]
Lexical Syntax for SMT

Two levels of support:

- Supertagged TM & LM
- Fully incremental parsing
Outline

Introduction

Syntax for Phrase-based SMT

Supertagged Phrase-based SMT

From Supertagged to Dependency-based Language Models

Incremental Dependency-based Language Model (IDLM)

DTM2

Dependency-based SMT

Future Work

Conclusion and Discussion
Can linguistic syntax improve the output of Phrase-based SMT systems?

- Which syntax could fit with PBSMT?
- Lexical Syntax: LTAG/CCG Supertags
- Supertags improve the performance of state-of-the-art PBSMT system on large data sets:
  - Arabic-to-English NIST’05
  - German-to-English shared task 07
Baseline PBSMT vs Supertags PBSMT

Baseline PBSMT
- Many candidate phrases
- Not constrained enough
- N-gram LM can not choose best candidates

Supertags PBSMT
- Many candidate phrases
- Syntactically Constrained Phrases
- Further sophisticated techniques could choose best candidates
Supertagged PBSMT Model

Supertags PBSMT: Noisy Channel Model

\[
\arg \max_{t} \sum_{ST} P(s \mid t, ST) P_{ST}(t, ST) \approx \\
\arg \max_{t,ST} P(s \mid t, ST) P_{ST}(t, ST) \approx \\
\arg \max_{\sigma,t,ST} P(\phi_s \mid \phi_t, ST) P(O_s \mid O_t)^{\lambda_o} P_{ST}(t, ST)
\]
Supertagged PBSMT Model

Supertags PBSMT: Log-Linear Model

\[ t^* = \arg \max_{t, \sigma, ST} \prod_{f \in F} H_f(s, t, \sigma, ST)^{\lambda_f} \]

- Log-linear model representation
- Added features for supertags
Supertagged PBSMT Model

Supertags Language Model

$$P(t, ST) = \prod_{i=1}^{n} P(st_i | st_{i-1})P(t_i | st_i)$$

- Log-linear Language Model for Supertags
- 5-gram Markov Language Model over supertags sequence
Supertagged PBSMT Model

Supertagged Phrase Translation Probability

\[
P(\phi_s \mid \phi_t, ST) \approx \prod_{\langle s_i, t_i ST_i \rangle \in (\phi_s \times \phi_t, ST)} P(s_i \mid t_i, ST_i)
\]

\[
P(\phi_t, ST \mid \phi_s) \approx \prod_{\langle s_i, t_i ST_i \rangle \in (\phi_s \times \phi_t, ST)} P(t_i, ST_i \mid s_i)
\]

- Phrase translation probability and its reverse
- Generate target words and supertags simultaneously
LMs with Global Grammaticality Measures

- Log-linear feature
- Smoothing factor for supertags LM
- Number of operator violations

John bought quickly shares
NNP_NN VBD_(S[dcl]\NP)/NP RB|(S\NP)(S\NP) NNS_N
2 Violations
LMs with Global Grammaticality Measures

He believes in what he said

NP (S_{dcl}\backslash NP)/S_{dcl} PP/NP NP/(S/NP) NP (S\backslash NP)/NP

The supertag of “believes” (in boldface) demands directly to its right (for “in”) an “S_{dcl}” (Forward Application); however, it finds a “(in PP/NP)” instead. This counts as a single violation $V = 1$. Note that the supertag that fits best in the given sequence for “believes” is “(S\backslash NP)/PP”.
Experimental Setup

Two Language Pairs:
- Arabic to English NIST 05
- German to English Shared Task 07

Supertaggers:
- LTAG supertaggers (XTAG and Bangalore’s Maxent tagger)
- CCG supertagger (C&C tools)

Supertags:
- N-gram Language model on supertags sequence for LTAG & CCG
- Grammatical validation for CCG operators
Scalability: Larger Training Corpora

Performance on large training data

<table>
<thead>
<tr>
<th>System</th>
<th>BLEU Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base-LARGE</td>
<td>0.4418</td>
</tr>
<tr>
<td>LTAG-LARGE</td>
<td>0.4600</td>
</tr>
<tr>
<td>CCG-LARGE</td>
<td>0.4609</td>
</tr>
</tbody>
</table>
Adding a grammaticality factor

<table>
<thead>
<tr>
<th>System</th>
<th>BLEU Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base-LARGE</td>
<td>0.4418</td>
</tr>
<tr>
<td>CCG-LARGE</td>
<td>0.4609</td>
</tr>
<tr>
<td>CCG-LARGE-GRAM</td>
<td>0.4688</td>
</tr>
</tbody>
</table>

LM with CCG Grammaticality

Adding a grammaticality factor
## German to English Results

<table>
<thead>
<tr>
<th>System</th>
<th>BLEU Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base-Line</td>
<td>0.2704</td>
</tr>
<tr>
<td>Supertags</td>
<td>0.2755</td>
</tr>
<tr>
<td>Supertags no Brevity</td>
<td><strong>0.2947</strong></td>
</tr>
</tbody>
</table>
## Results Analysis

<table>
<thead>
<tr>
<th>Reason</th>
<th>#</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inserting verb omitted by baseline</td>
<td>11</td>
<td>22</td>
</tr>
<tr>
<td>Better reordering</td>
<td>11</td>
<td>22</td>
</tr>
<tr>
<td>Better word/phrase selection</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>Other reasons</td>
<td>23</td>
<td>46</td>
</tr>
</tbody>
</table>

### Table: How CCG improves over baseline

N=50 test sentences

- Inserting verb omitted by baseline: 11% (22%)
- Better reordering: 11% (22%)
- Better word/phrase selection: 5% (10%)
- Other reasons: 23% (46%)
Arabic to English Examples

Reference: Annan opened an internal investigation in February but cancelled it in March in preparation for a broader, independent investigation.
Baseline: Annan was to internally in February but abolished in March as a prelude to broader and independent.
Supertags: Annan conducted an internal inquiry in February but abolished in March in preparation for broader and independent.
Arabic to English Examples

Reference: Rabat 1-14 (AFP) - A sharp debate is raging in Morocco on the freedom of the press with regard to matters connected personally to King Mohamed VI following the publication of articles criticizing the Moroccan monarch’s income and activities.

Baseline: Rabat 14-1 (AFP) - was a sharp controversy in Morocco on press freedom in terms of topics affecting King Mohamed VI himself after publishing articles critical of the revenues of the Moroccan

Supertags: Rabat 14-1 (AFP) - a sharp controversy in Morocco on press freedom in respect of topics affecting King Mohamed VI personally after the publication of articles criticizing the Moroccan monarch revenues.
German to English Examples

Source: Ich habe nicht für den Bericht Mann gestimmt, denn bei allem tatsächlich notwendigen Streben nach Gleichbehandlung in Beschäftigung und Beruf braucht deswegen noch nicht im bereifer soweit gegangen zu werden, dass der Schutz der Freiheiten und die Achtung des Rechtsstaates dabeivöllig in Vergessenheit geraten.

Reference: I have not voted for the Mann report because, while it is indeed necessary to seek equal treatment for people in employment and occupation, it is also necessary to refrain from pushing zeal to the point of abandoning all protection of freedoms and all respect for the rule of law.

Baseline: I have voted in favour of the report because, in particular, man is actually needed quest for equal treatment in employment and occupation is therefore not yet in excess of zeal went so far as to say, the protection of freedoms and respect for the rule of law is completely forgotten.

Supertags: I have not voted for the Mann report because, in fact, with all the necessary search for equal treatment in employment and occupation is therefore not yet gone so far in excess of zeal, that the protection of freedoms and respect for the rule of law is being completely forgotten.
Outline

Introduction
Syntax for Phrase-based SMT
Supertagged Phrase-based SMT
From Supertagged to Dependency-based Language Models
Incremental Dependency-based Language Model (IDLM)
DTM2
Dependency-based SMT
Future Work
Conclusion and Discussion
Almost parsing for MT:

- A \textit{n-gram} language model over the sequence of supertags (‘almost parsing’).
- ‘almost parsing’ for monolingual parsing
- ‘almost parsing’ for bilingual parsing
What is the parsing mechanism we need for SMT?

- Support long-range dependencies
- Distinguish between different translation candidates based on their role in constructing the parse structure
- Satisfy the syntactic dependencies
- Work in an incremental manner similar to SMT decoders
- Be computationally efficient to be integrated into SMT
What is the parsing mechanism we need for SMT?

Our proposed IDLM differs from the related work in four major respects:

- It is based on incremental parsing that seamlessly matches the incremental nature of SMT decoders.
- It is deterministic, in the sense that it maintains a limited number of parse-states that represent possible parsing decisions at each word position. This characteristic is very important for incorporating IDLM into large-scale MT systems due to its computational efficiency.
- The grammatical representation is based on CCG structures which enable the handling of non-constituent constructions.
- The parser seeks out intermediate connected structures, unlike previous approaches which deployed dependency relations or head words to enable syntax-based probabilities into the language model.
Incremental Parsing Representation
Incremental CCG

Mr. Warren will remain on the company's board

<table>
<thead>
<tr>
<th>Supertag</th>
<th>NP/NP</th>
<th>NP</th>
<th>(S\NP)/ (S\NP)</th>
<th>(S\NP) /PP</th>
<th>PP/NP</th>
<th>NP/NP</th>
<th>NP</th>
<th>(NP/NP)</th>
<th>NP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operator</td>
<td>NOP</td>
<td>FA</td>
<td>TRFC</td>
<td>FC</td>
<td>FC</td>
<td>TRFC</td>
<td>FA</td>
<td>FC</td>
<td>FA</td>
</tr>
<tr>
<td>State Cat.</td>
<td>NP/NP</td>
<td>NP</td>
<td>S/(S\NP)</td>
<td>(S/PP)</td>
<td>(S/NP)</td>
<td>(S/(NP\NP)) /NP</td>
<td>(S/(NP \NP))</td>
<td>(S/NP)</td>
<td>S</td>
</tr>
</tbody>
</table>
Incremental Parsing Training

CCGBank

Transformation to Incremental representation

Supertags
Operators

MaxEnt Framework

Supertagger
Operator Tagger
Incremental Parsing Runtime

Sentence to parse

Supertagger

Operator Tagger

State Realizer

Dependency Structures
Incremental Parsing Features: Apposition Handling

The man, who plays tennis, likes football.

\[ S_0 : NP/NP, \quad APSV, \quad (NP/NP)/(S/NP), \quad (S/NP_1)/NP, \quad NP, \quad APSV, \quad (S/NP)/NP \]

\[ S_1 : NP \quad \rightarrow_{FA} \]

\[ S_2 : NULL \quad \rightarrow_{INTR} \]

\[ S_3 : NP/(S/NP) \quad \rightarrow_{NOP} \]

\[ S_4 : NP/NP \quad \rightarrow_{FC} \]

\[ S_5 : NP \quad \rightarrow_{FA} \]

\[ S_6 : NP \quad \rightarrow_{INTR} \]

\[ S_7 : S/NP \quad \rightarrow_{TRFC} \]

\[ S_8 : S \quad \rightarrow_{FA} \]

**Figure:** Apposition Handling.
Incremental Parsing Features: Coordination Handling

He plays football and tennis

\[
S_1 : \text{NP} (S\backslash\text{NP})/\text{NP} \quad \text{NP}_2 \quad (\text{NP}_1 \backslash \text{NP}_2)/\text{NP}_3 \quad \text{NP}_3
\]

\[
S_2 : S/\text{NP} \quad \text{TRFC} \quad S_3 : S \quad \text{FA} \quad S_4 : S/\text{NP} \quad \text{COORD} \quad S_5 : S \quad \text{FA}
\]

**Figure**: Coordination Handling.
Incremental Parsing Features: WH-movement Handling

He bought what she sold

\[ S_0 : \text{NP} \begin{array}{c} (\text{S}/\text{NP})/\text{NP} \\ \text{NP}/(\text{S}/\text{NP}) \end{array} \]

\[ \text{NP}_1 \begin{array}{c} (\text{S}/\text{NP}_1)/\text{NP}_2 \end{array} \]

\[ S_1 : \text{S}/\text{NP} \begin{array}{c} \rightarrow \text{TRFC} \end{array} \]

\[ S_2 : \text{S}/(\text{S}/\text{NP}) \begin{array}{c} \rightarrow \text{FC} \end{array} \]

\[ S_3 : \text{S}/((\text{S}/\text{NP}) \text{NP}) \begin{array}{c} \rightarrow \text{TRFC} \end{array} \]

\[ S_4 : \text{S} \begin{array}{c} < \text{WHMV} \end{array} \]

Figure: WH-movement Handling.
Incremental Parsing Evaluation

<table>
<thead>
<tr>
<th>Input/Features</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold standard POS and Supertags</td>
<td>96.73</td>
</tr>
<tr>
<td>System POS and Supertags</td>
<td>90.90</td>
</tr>
<tr>
<td>Preceding correct state as feature</td>
<td>99.22</td>
</tr>
</tbody>
</table>

**Table:** Operator Tagger Results.

<table>
<thead>
<tr>
<th>Input</th>
<th>F-Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold standard POS and Supertags</td>
<td>87.5</td>
</tr>
<tr>
<td>System POS and Supertags</td>
<td>86.7</td>
</tr>
</tbody>
</table>

**Table:** Unlabeled dependency results for section 23.
Outline

Introduction

Syntax for Phrase-based SMT

Supertagged Phrase-based SMT

From Supertagged to Dependency-based Language Models

Incremental Dependency-based Language Model (IDLM)

DTM2

Dependency-based SMT

Future Work

Conclusion and Discussion
MERT Estimation for log-linear Models:

- Approximation for Maximum Entropy log-linear models
- Can handle a few number of parameters (in order of ten)
- A bottleneck to further serious development of features-rich SMT systems
- Parameters of different components are not related
DTM2 Phrase Structures

<table>
<thead>
<tr>
<th>Algnp</th>
<th>of the X committee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Almrkzyp</td>
<td>central</td>
</tr>
<tr>
<td>llhzb</td>
<td>of the X Party</td>
</tr>
</tbody>
</table>

**Figure:** Phrase structures in DTM2. X represents a variable in the target phrase
Outline

Introduction
Syntax for Phrase-based SMT
Supertagged Phrase-based SMT
From Supertagged to Dependency-based Language Models
Incremental Dependency-based Language Model (IDLM)
DTM2

Dependency-based SMT
Future Work
Conclusion and Discussion
Dependency Direct Translation Model (DDTM)

\[ P(T|S) = P_0(T, J|S)/Z \exp \sum_i \lambda_i \phi_i(T, J, S) \]  (1)

- \( P_0 \) is the prior distribution for the phrase probability
- \( J \) is the skip reordering factor for this phrase pair
DDTM Features

In our DDTM, we have implemented many features along with the baseline DTM2 features:

- **Supertag-Word features**: these features examine the target phrase words with their associated supertags.
- **Supertag sequence features**: these features encode $n$-gram supertags (equivalent to the $n$-gram supertags Language Model).
- **Supertag-Operator features**: these features encode supertags and their associated operators.
- **Supertag-State features**: these features encode states and supertags co-occurrence.
- **State sequence features**: these features encode $n$-gram states features and are equivalent to an $n$-gram states Language Model.
- **Word-State sequence features**: these features encode words and states co-occurrence.
DDTM Decoder

- A beam search decoder similar to decoders used in standard phrase-based log-linear systems
- Performs incremental dependency parsing during decoding
- Supports new pruning strategies to handle the large search space
DDTM Decoder

Introduction
Syntax for Phrase-based SMT
Supertagged Phrase-based SMT
From Supertagged to Dependency-based Language Models
Incremental Dependency-based Language Model (IDLM)
DTM2
Dependency-based SMT
Future Work
Conclusion and Discussion
DDTM Decoder
Experiments

- Arabic–English with 3.7M parallel sentences.
- 5-gram LM trained on English Gigaword Corpus.
- Testset: Arabic–English MT05
- Baseline is top ranked in two recent MT large scale evaluations
Evaluated Systems

- IBM-PB: IBM Phrase–based SMT baseline system.
- DTM2: the baseline Direct Translation model system.
- D-SW: examines Supertag-Word features.
- D-SLM: examines Supertag-Word features and supertags n-gram features.
- D-SO: examines Supertag-Operator features.
- D-OLM: examines operator n-gram features.
- D-SS: examines supertags and states features with parse-state construction.
- D-WS: examines words and states features with parse-state construction.
- D-SLM: examines n-gram states features with parse-state construction.
- DDTM: fully fledged system with all features that proved useful above.
## Results

<table>
<thead>
<tr>
<th>System</th>
<th>BLEU Score on MT05</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM-PB</td>
<td>50.16</td>
</tr>
<tr>
<td>DTM2-Baseline</td>
<td>52.24</td>
</tr>
<tr>
<td>D-SW</td>
<td>52.28</td>
</tr>
<tr>
<td>D-SLM</td>
<td>52.29</td>
</tr>
<tr>
<td>D-SO</td>
<td>52.01</td>
</tr>
<tr>
<td>D-OLM</td>
<td>51.87</td>
</tr>
<tr>
<td>D-SS</td>
<td>52.39</td>
</tr>
<tr>
<td>D-WS</td>
<td>52.03</td>
</tr>
<tr>
<td>D-SLM</td>
<td>52.53</td>
</tr>
<tr>
<td>DDTM</td>
<td>52.61</td>
</tr>
</tbody>
</table>

**Table:** DDTM Results with various features.
Examples

Source: وخط ط بعد ذلك لفحوصات إجرائية أحد أطباء الشرطة

Reference: He then underwent medical examinations by a police doctor.

Baseline: He was subjected after that tests conducted by doctors of the police.

DDTM: Then he underwent tests conducted by doctors of the police.
Example

Source: وقد هز الرياض مساء اليوم هجمات بسيارتين مفحختين.

Reference: Riyadh was rocked tonight by two car bomb attacks.

Baseline: Riyadh rocked today night attacks by two booby - trapped cars.

DDTM: Attacks rocked Riyadh today evening in two car bombs.

Figure: DDTM provides better syntactic structure with more concise translation.
Outline

Introduction

Syntax for Phrase-based SMT

Supertagged Phrase-based SMT

From Supertagged to Dependency-based Language Models

Incremental Dependency-based Language Model (IDLM)

DTM2

Dependency-based SMT

Future Work

Conclusion and Discussion
Future Work

- Source Dependency information
- Enhance the Dependency parser accuracy
- Possible implementation of the framework into Moses.
- Extend the approach for logical semantics as well
Outline

Introduction
Syntax for Phrase-based SMT
Supertagged Phrase-based SMT
From Supertagged to Dependency-based Language Models
Incremental Dependency-based Language Model (IDLM)
DTM2
Dependency-based SMT
Future Work

Conclusion and Discussion
We introduced a novel model of supertagged Phrase-based SMT which integrates supertags into the target language model and the target side of the translation.

We introduced a novel dependency-based LM which is deterministic in that it maintains a limited number of parsing decisions at each state which. Furthermore, it is incremental in Markovian fashion similar to Phrase-based SMT decoders and it can naturally handle non-constituent constructions, being based on CCG.

We introduced an extension to direct translation models that integrates incremental dependency parsing while retaining the linear decoding assumed in conventional Phrase-based SMT systems.
Thanks for Listening