

Using Supertags as Source Language context in SMT

Rejwanul Haque
Sudip Kumar Naskar
Yanjun Ma
Andy Way

CNGL/NCLT
School of Computing
Dublin City University

Outline

- Motivation
- The Standard Approach
- Supertags as Context-Informed Features
- Memory-Based Disambiguation
- Evaluation & Results
- Related Work
- Conclusions and Future Work

Motivation

- PB-SMT systems can benefit from two enhancements:
 - Using **words** and **POS** tags as context-informed features of the **source side** (Stroppa et al., 2007).

Motivation

- PB-SMT systems can benefit from two enhancements:
 - Using **words** and **POS** tags as context-informed features of the **source side** (Stroppa et al., 2007).
 - Incorporating lexical syntactic descriptions in the form of **supertags** on the **target side** (Hassan et al., 2006, '07, '08).

Motivation

- PB-SMT systems can benefit from two enhancements:
 - Using **words** and **POS** tags as context-informed features of the **source side** (Stroppa et al., 2007).
 - Incorporating lexical syntactic descriptions in the form of **supertags** on the **target side** (Hassan et al., 2006, '07, '08).
 - Aim: A fully-supertagged system.

Phrase Based SMT

- In SMT, translation is modeled as a decision process, in which the translation $f_1^J = f_1 \dots f_j \dots f_J$ of a source sentence $e_1^I = e_1 \dots e_i \dots e_I$ is chosen to maximize:

$$\arg \max_{I, e_1^I} P(e_1^I | f_1^J) = \arg \max_{I, e_1^I} P(f_1^J | e_1^I) \cdot P(e_1^I) \quad (1)$$

Log-linear Phrase Based SMT

- In log-linear phrase-based SMT, the posterior probability $P(e_1^I | f_1^J)$ is directly modeled as a (log-linear) combination of features (Och and Ney, 2002), that usually comprise M translational features, and the **language model**, as in (2):

$$\log P(e_1^I | f_1^J) = \sum_{m=1}^M \lambda_m h_m(f_1^J, e_1^I, s_1^K) + \lambda_{LM} \log P(e_1^I) \quad (2)$$

Log-linear Phrase Based SMT

- In log-linear phrase-based SMT, the posterior probability $P(e_1^I | f_1^J)$ is directly modeled as a (log-linear) combination of features (Och and Ney, 2002), that usually comprise M translational features, and the language model, as in (2):

$$\log P(e_1^I | f_1^J) = \sum_{m=1}^m \lambda_m h_m(f_1^J, e_1^I, s_1^K) + \lambda_{LM} \log P(e_1^I) \quad (2)$$

where $s_1^k = s_1 \dots s_k$ denotes a segmentation of the source and target sentences respectively into the sequences of phrases $(\hat{e}_1, \dots, \hat{e}_k)$ and $(\hat{f}_1, \dots, \hat{f}_k)$.

Log-linear Phrase Based SMT

- Each feature h_m in (2) can be rewritten as:

$$h_m (f_1^J , e_1^I , s_1^K) = \sum_{k=1}^K \hat{h}_m (\hat{f}_k , \hat{e}_k , s_k) \quad (3)$$

where \hat{h}_m is a feature that applies to a single phrase-pair.

Log-linear Phrase Based SMT

- Each feature h_m in (2) can be rewritten as:

$$h_m(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^K \hat{h}_m(\hat{f}_k, \hat{e}_k, s_k) \quad (3)$$

where \hat{h}_m is a feature that applies to a single phrase-pair.

- In theory, log-linear PB-SMT *can* apply to entire sentences. In practice, those features apply to *single phrase pairs*.

Log-linear Phrase Based SMT

- Each feature h_m in (2) can be rewritten as:

$$h_m(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^K \hat{h}_m(\hat{f}_k, \hat{e}_k, s_k) \quad (3)$$

where \hat{h}_m is a feature that applies to a single phrase-pair.

- In theory, log-linear PB-SMT *can* apply to entire sentences. In practice, those features apply to *single phrase pairs*.
- Translational features depend on an individual pair of *source/target phrases*, they do not take into account the *contexts* of those phrases.

Log-linear Phrase Based SMT

- In this context, the translation process amounts to:
 - choosing a segmentation of the source sentence.

Log-linear Phrase Based SMT

- In this context, the translation process amounts to:
 - choosing a segmentation of the source sentence,
 - translating each source segment into target segment.

Log-linear Phrase Based SMT

- In this context, the translation process amounts to:
 - choosing a segmentation of the source sentence,
 - translating each source segment into target segment,
 - re-ordering the target segments obtained.

Log-linear Phrase Based SMT

- In this context, the translation process amounts to:
 - choosing a segmentation of the source sentence,
 - translating each source segment into target segment
 - re-ordering the target segments obtained.
- But translational choices are strongly driven by the target LM.
- In addition to the LM, we try to use the **source context** to resolve ambiguities.

Supertags as Context-Informed Features

- Supertags: Lexical entries consist of *Syntactic constructs*.

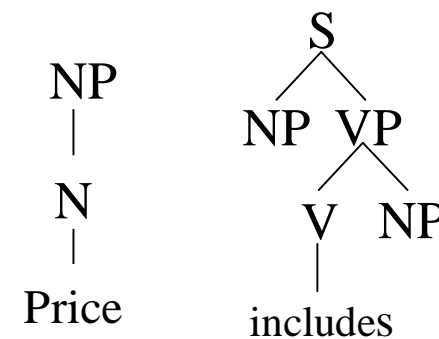
Overview on Supertags

- **Supertags:** Lexical entries consist of *Syntactic constructs*.

CCG

John	cleaned
NNP_NN	(S\NP)/NP

LTAG



Overview on Supertags

- LTAG and CCG assign one or more *syntactic constructs* to each word in a sentence.

Overview on Supertags

- LTAG and CCG assign one or more *syntactic constructs* to each word in a sentence.
- Supertagger chooses the most correct elementary structure (**Supertag**).

Overview on Supertags

- LTAG and CCG assigns one or more *syntactic construct* to each word in a sentence.
- Supertagger chooses most correct elementary structure (**Supertag**).
- Supertags capture long-distance dependencies.

Overview on Supertags

- LTAG and CCG assigns one or more *syntactic constructs* to each words in a sentence.
- Supertagger chooses most correct elementary structure (**Supertag**).
- Supertags capture long-distance dependencies.
- Supertags should have a greater influence than the surface-word or POS tag in **lexical choice** and **reordering** in PB-SMT.

Overview on Supertags

- Supertagging was introduced to reduce no of syntactic structure (elementary tree) for each word in sentence (LTAG: Bangalore and Joshi, 1999)
- Supertagging has most recently been applied to CCG (Clark 2002; Curran and Clark 2004).
- Bangalore and Joshi (1999) used a standard Markov model tagger to assign LTAG elementary trees to words.
- Similarly for CCG, Curran and Clark (2003) used the Maximum Entropy models to assign lexical categories to each word of the sentences

LTAG

- LTAGs are automatically extracted from the Penn Treebank (Chen et al., 2006).
- In LTAG, a lexical item may be associated with more than one elementary structure (tree).
- Supertagger chooses most correct elementary structure (Supertag).

CCG

- CCGbank is a translation of the Penn Treebank into a corpus of Combinatory Categorical Grammar derivations, (Julia Hockenmaier and Mark Steedman, 2002).
- CCGbank pairs lexical categories with sets of word-word dependencies.
- A category may be either atomic (S,NP) or complex (S\S, S\NP, (S\NP)/NP)

Example :

John	plays	football
NP	(S \ NP) / NP	NP

where the derivation proceeds as follows: “plays” is combined with “football” under the operation of forward application. ‘play’ can be thought of as a function that takes an NP to the right and returns a S\NP.

Context-Informed Features

- We have employed two kinds of features in our experiments:
 - Lexical Context Features
 - Syntactic Context Features

Lexical Context Features

- Lexical Context Features

$$CI = \{ f_{i_k - l} \cdots f_{i_k - 1}, f_{j_k + 1} \cdots f_{j_k + l} \}$$

- These include the direct left and right context words of length l ($f_{i_k - l} \cdots f_{i_k - 1}$ and $f_{j_k + 1} \cdots f_{j_k + l}$).
- It forms a window of $2l+1$ features (including the focus phrase).

Syntactic Context Features

- Syntactic information:
 - POS tags.
 - Supertags.
- Syntactic information (*SI*) of both the **focus phrase** and the **context words** considered.
- SI of a multi-word focus phrase.

Syntactic Context Features

- Syntactic Context Features

$$CI = \{SI(f_{i_k-l}) \dots SI(f_{i_k-1}), SI(\hat{f}_k), SI(f_{j_k+1}) \dots SI(f_{j_k+l})\}$$

- Thus a window of $2l+2$ features is formed (includes the focus phrase itself).

Context-Informed Features

- We also tried combining features (LTAG, CCG and POS).
- The combined contextual information is formed by the union of the two Syntactic Context features as:

$$CI = CI_{syn1} \cup CI_{syn2}$$

Context-Informed Features

- We also tried combining features (LTAG, CCG and POS).
- The combined contextual information is formed by the union of the two Syntactic Context features as:

$$CI = CI_{syn1} \cup CI_{syn2}$$

- In some experiments we combined Lexical Context features with Syntactic Context features.

$$CI = CI_{lex} \cup CI_{syn}$$

Context-Informed Features

- Context-informed features are expressed as the conditional probability of the **target phrase** given the **source phrase** and its **context information (CI)**, as in (4):

$$\hat{h}_{mbl}(\hat{f}, CI(\hat{f}_k), \hat{e}_k, s_k) = \log P(\hat{e}_k | \hat{f}_k, CI(\hat{f}_k)) \quad (4)$$

- We use Tilburg Memory Based Learner (**TiMBL**) (Daelemans et al., 2007) to estimate the **context-dependent translation probability**.

An Example: t-table

<i>Source Phrase</i>	<i>Target Phrase</i>	<i>P(e/f)</i>
f_1	e_1	0.002
f_1	e_2	0.811
f_1	e_3	0.021
f_1	e_4	0.106
f_1	e_5	0.003.
f_1	e_6	0.001
f_1	e_7	0.007
f_1	e_8	0.051

Context-informed t-table

<i>Source Phrase</i>	<i>Target Phrase</i>	<i>$P(e f, CI(f))$</i>
$f_1 + CI_1$	e_3	0.2
	e_5	0.8
$f_1 + CI_2$	e_1	0.2
	e_4	0.75
	e_8	0.05

Memory-Based Disambiguation

- Training data for classifier.
- Modified standard phrase-extraction method of (Koehn et al., 2003).

Memory-Based Disambiguation

- Directly estimating **Context dependent phrase translation probability**

$P(\hat{e}_k | \hat{f}_k, CI(\hat{f}_k))$ using relative frequency is problematic.

Memory-Based Disambiguation

- Directly estimating **Context dependent phrase translation probability**

$P(\hat{e}_k | \hat{f}_k, CI(\hat{f}_k))$ using relative frequency is problematic.

- To avoid this problem, we use Tilburg Memory Based Learner (**TiMBL**) (Daelemans et al., 2007). TiMBL includes three different memory-based classifiers:
 - IGTre
 - IB1
 - TRIBL

Memory-Based Disambiguation

- **IGTree**
 - Heuristic approximation of nearest neighbour search.
 - Instance-base in the form of compressed decision tree.
- **IB1**
 - The major difference of TiMBL version of IB1 with original IB1(*K-nn*) [Aha et al. 1991] algorithm is that the value *k* refers to *k-nearest* distance rather than *k-nearest* example.
- **TRIBL**
 - A hybrid combination of IGTree and IB1.
 - A parameter determines the switching point in feature ordering from IGTree to IB1.

Memory-Based Disambiguation

- **Source phrase:** the feature with the highest prediction power (**Information Gain**).
- **Information Gain:** looks at each feature in isolation, and measures how much information it contributes to predict the correct class.

Memory-Based Disambiguation

- MBL classifies the source phrase with Context-Information (CI) :

$$\langle \hat{f}_k, CI(\hat{f}_k) \rangle$$

Memory-Based Disambiguation

- MBL classifies the source phrase with **Context-Information (CI)** :

$$\langle \hat{f}_k, CI(\hat{f}_k) \rangle$$

- The result of this classification is a set of weighted class labels, representing the possible target phrases \hat{e}_k .

Memory-Based Disambiguation

- MBL classifies the source phrase with **Context-Information (CI)** :

$$\langle \hat{f}_k, CI(\hat{f}_k) \rangle$$

- The result of this classification is a set of weighted class labels, representing the possible target phrases \hat{e}_k .
- Normalization: to derive $P(\hat{e}_k | \hat{f}_k, CI(\hat{f}_k))$

Memory-Based Context-informed Feature

- Thus, memory based context-informed feature \hat{h}_{mbl} is derived as :

$$\hat{h}_{mbl} = \log P(\hat{e}_k | \hat{f}_k, CI(\hat{f}_k)) \quad (5)$$

Experimental-Setup

- Baseline Model consists of the following features:
 - Forward/backward phrase translation probabilities
 - Forward/backward Lexical weighting probabilities
 - Phrase Penalty
 - Word Penalty
 - Distance based reordering probability
 - Lexicalised Reordering probabilities
 - Language Model Probability
- Further, we incorporate context informed feature \hat{h}_{mbl} in the log-linear framework of PBSMT.

Experimental-Setup

- Feature weights are optimized using minimum-error-rate training (MERT) [Och, 2003].
- Context size: ± 1 and ± 2 (i.e. $l=1, 2$).

Experimental Data

- English-Chinese IWSLT-06 data.
- Data extracted from the *Basic Travel Expressions Corpus* (BTEC) [Takezawa et al., 02].
- Multilingual speech corpus containing tourist phrase-book type sentences.
 - Training: 40,274 sentences
 - Development: 489 sentences
 - Test: 486 sentences
 - Avg Sentence length: En- 9.74 , Cn- 8.77

Evaluation & Results using IGTre

Experiments	BLEU	
<i>Baseline</i>	<i>20.56</i>	
Context length	± 1	± 2
CCG	21.75	21.52
LTAG	21.92	21.34
POS	21.52	21.70
Word	21.64	21.59

Experiments with single feature on uniform context size

Evaluation & Results using IGTtree

Experiments	BLEU	
<i>Baseline</i>	<i>20.56</i>	
Context length	± 1	± 2
Word + CCG	21.52	21.53
Word + LTAG	21.64	21.37
Word + POS	21.77	21.89

Experiments with Uniform Context Size

Evaluation & Results using IGTre

Experiments	BLEU
<i>Baseline</i>	<i>20.56</i>
Word \pm 2 + CCG \pm 1	22.01
Word \pm 2 + LTAG \pm 1	21.38
Word \pm 2 + POS \pm 1	21.61
POS \pm 2 + CCG \pm 1	21.08

Experiments with Varying Context Size

Evaluation & Results using IGTre

Experiments	BLEU
<i>Baseline</i>	<i>20.56</i>
CCG ± 1 + LTAG ± 1	22.11

Experiments with combining CCG and LTAG

Evaluation & Results using IB1 and TRIBL

Experiments	BLEU
Baseline	20.56
IB1	
CCG±1	22.08
LTAG±1	22.06
CCG±1+LTAG±1	21.72
Supertag-Pair±1	22.03
TRIBL	
CCG±1	22.18 (p=0.02)
LTAG±1	21.39
CCG±1+LTAG±1	22.00
Supertag-Pair±1	22.13

Analysis

- We performed sentence-level evaluation to compare our best system with the baseline system. (Test Sentences=486).

	BestSys>Base	BestSys<Base	BestSys=Base
BLEU			
NIST	76	61	349
WER	50	52	384
PER	50	53	383

Output

- ENG: about **twenty-five** seconds .
- BASE: 大约 **二十** 秒。
- CCG: 大约 **二十五** 秒。

- ENG : do you have any local specialties or **something** ?
- BASE: 有本地 特色菜 和 **一些** 吗 ?
- CCG: 有本地 特色菜 或 **什么** 吗 ?
-
- ENG : i 'd like **it** rare .
- BASE: 我想 **是** 半熟。
- CCG: 我 想要 **它** 半熟。

- ENG: please **show** me to my seat .
- BASE: 请 **给** 我的座位。
- CCG: 请 **带我到** 我的座位。

Related Work

- (Berger et al., 1996) suggested to integrate local contextual information into IBM translation models.

Related Work

- (Berger et al., 1996) suggested to integrate local contextual information into IBM translation models.
- Initial attempts to embed context-rich approaches of WSD methods into SMT to enhance lexical selection did not lead to any improvement (Carpuat and Wu, 2005).

Related Work

- (Berger et al., 1996) suggested to integrate local contextual information into IBM translation models.
- Initial attempts to embed context-rich approaches of WSD methods into SMT to enhance lexical selection did not lead to any improvement (Carpuat and Wu, 2005).
- Recent approaches (Carpuat and Wu, 2007; Chan et al., 2007; Giménez and Màrquez, 2007) of integrating state-of-the-art WSD methods into SMT to improve the overall translation quality have shown some success .

Related Work

- (Stroppa et al., 2007) added source-side contextual features to PB-SMT system by incorporating context-dependent phrasal translation probabilities learned using decision trees.

Related Work

- (Stroppa et al., 2007) added source-side contextual features to PB-SMT system by incorporating context-dependent phrasal translation probabilities learned using decision trees.
- (Max et al., 2008) show modest gains over PB-SMT baseline using broader context and grammatical dependency information.

Related Work

- (Stroppa et al., 2007) added source-side contextual features to PB-SMT system by incorporating context-dependent phrasal translation probabilities learned using decision trees.
- (Max et al., 2008) show modest gains over PB-SMT baseline using broader context and grammatical dependency information.
- (Gimpel and Smith, 2008) considers lexical features, as well as shallow syntactic features and positional features. They shows some improvements on Chinese-English but no improvements on English-German and German-English experiments.

Conclusions

- Successfully incorporated supertags as a new feature into a state-of-the-art log-linear PB-SMT system.
- Any type of Source Language context improves system performance.
- Best result (1.62 BLEU points, 7.88% relative improvement) is obtained on CCG using TRIBL.
- Supertag consistently produces good results.
- Supertag is more powerful feature rather than the POS and word when applying as source language context.

Future Work

- To develop a fully supertagged PB-SMT system, with supertags deployed as SL context, as well as in the TL model and the target side of the t-table.
- To consider all neighbouring content words (Bag-of-words approach) as context.
- To use dependency parse relations of head word in the source phrase with neighbouring words as context.

References

- Stroppa, Nicolas, Antal van den Bosch and Andy Way. 2007. Exploiting Source Similarity for SMT using Context-Informed Features. *TMI-2007, 11th Conference on Theoretical and Methodological Issues in Machine Translation*, Skövde, Sweden, 231—240.
- Hassan, Hany, Khalil Sima'an, and Andy Way. 2008. Syntactically Lexicalized Phrase-Based SMT. *IEEE Transactions on Audio, Speech and Language Processing* **6**(7):1260—1273.

References

- Berger, Adam, Stephen Della Pietra, and Vincent Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, **22**(1):39–68.
- Max, A., R. Makhloufi and P. Langlais. 2008. Explorations in Using Grammatical Dependencies for Contextual Phrase Translation Disambiguation. In *Proceedings of the 12th EAMT Conference*, Hamburg, Germany, pp.112—117.
- Gimpel, Kevin and Noah A. Smith. 2008. Rich source-side context for statistical machine translation. *ACL-08: HLT. Third Workshop on Statistical Machine Translation*, Columbus, OH, pp.9—17.

References

- Giménez, Jesús, and Lluís Màrquez. 2007. Context-aware discriminative phrase selection for statistical machine translation. *ACL 2007: proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, 159—166.
- Chan, Y. S., H. T. Ng., and D. Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, 33—40.
- Carpuat, Marine, and Dekai Wu. 2007. Improving statistical machine translation using word sense disambiguation. *EMNLP-CoNLL-2007, Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic, 61—72.

Thank You!

