Dependency-Based Bilingual Language Models for Reordering in SMT

Ekaterina Garmash

Informatics Institute, University of Amsterdam

University of Amsterdam

Motivation

Background: bilingual language models (BiLMs)
- n-gram model of sequences of elementary translation events
- elementary translation event — a pair of source and target words
- we adopt the definition of Niehues et al. (2011) of a bilingual token: (given word alignment) a target word and all the source words aligned to it

Reordering with BiLMs

How well do various labelings differentiate between correct and incorrect reorderings?

Dependency-based BiLMs
- dependency grammar is commonly used in NLP to formalise role-based relations between words
- to label bilingual tokens, we try out combinations of different properties based on a source dependency parse
- generalized definition of a labeling for a bilingual token sequence $\{t_1, \ldots, t_n\}$

$$c_i = (\text{ContF}(t_i))$$

where $c_i$ is the i-th target word, $A: E \to F$ is an alignment function, $F$ and $E$ — source and target sentences, $\text{ContF}$ and $\text{ContE}$ — contextual functions

Contextual functions

Return a word’s sentential context (source or target).
We focus on $\text{ContE}F$s, since they allow for a richer set of definitions in the MT setting (source side fully given) than $\text{ContE}F$s.

Proposed contextual functions return:
- the word itself (designation: Lex)
- POS tag of the word (designation: Pos)
- POS tag of the word’s parent (see below)
- POS tag of the word’s grandparent (see below)

Notation

We use the proposed contextual functions in combinations to define individual BiLMs.

- $\text{Lex}$ is a BiLM with lexicalized tokens
- $\text{Pos}$ is a BiLM with words in tokens substituted with their POS tags
- $\text{Lex}$ and $\text{Pos}$ connect parent (left) and child (right) from a dependency tree

Example

Sequence of bilingual tokens produced by a $\text{Pos}$ — $\text{Pos}$ — $\text{Pos}$ after translating three words of the source sentence:

Reordering potential of the model

Distortion limit extended to 10 words.

Implementation and Experiments

Implementation and integration into PBSMT decoder
- n-gram model training: 5-gram model with Kneser-Ney smoothing using SRILM (Stolke et al., 2011)
- dependency-based BiLMs are integrated as a feature in a log-linear model
- for each phrase pair, its most likely internal word alignment and target-side POS labeling is stored in the phrase table

Basic experimental setup
- phrase-based decoder
- distortion limit: 5
- lexicalized distortion model included in the log-linear interpolation
- compare performance of the original BiLMs (Niehues et al. 2011) and the dependency-based BiLMs

Statistical significance notation
- significant improvement over $\text{Lex}$ at $p < .01$
- significant improvement over $\text{Lex}$ at $p < .05$

Arabic-English experiments

Chinese-English experiments

Reordering with BiLMs

How well do various labelings differentiate between correct and incorrect reorderings?

contributions
- capture reordering as an order of a sequence of translation events
- characterize translation events with their source and target syntactic features
- simple alternative to tree-based models
- up to +0.98 BLEU improvement for Chinese-English and +0.4 BLEU improvement for Arabic-English over lexicalized BiLM

Approach

Dependency-based BiLMs
- dependency grammar is commonly used in NLP to formalise role-based relations between words
- to label bilingual tokens, we try out combinations of different properties based on a source dependency parse
- generalized definition of a labeling for a bilingual token sequence $t_1, \ldots, t_n$

$$c_i = (\text{ContF}(t_i))$$

where $c_i$ is the i-th target word, $A: E \to F$ is an alignment function, $F$ and $E$ — source and target sentences, $\text{ContF}$ and $\text{ContE}$ — contextual functions

Contextual functions

Return a word’s sentential context (source or target).
We focus on $\text{ContE}F$s, since they allow for a richer set of definitions in the MT setting (source side fully given) than $\text{ContE}F$s.

Proposed contextual functions return:
- the word itself (designation: Lex)
- POS tag of the word (designation: Pos)
- POS tag of the word’s parent (see below)
- POS tag of the word’s grandparent (see below)

Notation

We use the proposed contextual functions in combinations to define individual BiLMs.

- $\text{Lex}$ is a BiLM with lexicalized tokens
- $\text{Pos}$ is a BiLM with words in tokens substituted with their POS tags
- $\text{Lex}$ and $\text{Pos}$ connect parent (left) and child (right) from a dependency tree

Example

Sequence of bilingual tokens produced by a $\text{Pos}$ — $\text{Pos}$ — $\text{Pos}$ after translating three words of the source sentence: