Project: Translating Natural Language

Xichuan Wu

International Center for Computational Logic
Technische Universität Dresden

February 22, 2012

Outline

1. **Statistical Machine Translation (SMT)**
2. Synchronous Context-free Grammar
3. Training Procedure for SCFG

Motivation

- How can I reason about these text?
  "*Ein weiteres Angebot für "Vielfahrer" inklusive Familie, Freunde und Kollegen. Mit der "kleinen Übertragbaren" können beliebig viele Personen nacheinander eine ganze Woche lang mit den Wiener Linien unterwegs sein.*"

Motivation

- How can I reason about these text?
  "*Ein weiteres Angebot für "Vielfahrer" inklusive Familie, Freunde und Kollegen. Mit der "kleinen Übertragbaren" können beliebig viele Personen nacheinander eine ganze Woche lang mit den Wiener Linien unterwegs sein.*"
- Life is too short to learn German! Let machine do it. :)
- The Rosetta stone(1799), decoding foreign language with corresponding translation using statistical features such as co-occurrence, distortion, fertility and etc.
- machine can "read" much more than human and can learn how to translate foreign language

Basics of SMT

- given a foreign sentence f, there may be many candidate translations e's
  - conditional probability $P(e \mid f)$
  - better translations should have higher probabilities

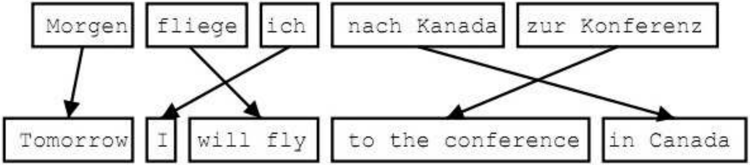
Basics of SMT

- given a foreign sentence f, there may be many candidate translations e's
  - conditional probability $P(e \mid f)$
  - better translations should have higher probabilities
- best candidate translation $\hat{e}$

$$\begin{aligned}\hat{e} &= \arg\max_{e} P(e \mid f) \\ &= \arg\max_{e} \frac{P(f \mid e) \cdot P(e)}{P(f)} \\ &= \arg\max_{e} P(f \mid e) \cdot P(e)\end{aligned}$$

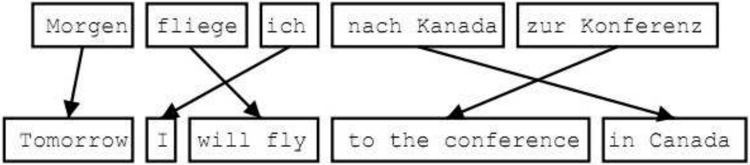
  - language probability $P(e)$ helps translation to be natural and grammatically correct
  - translation probability $P(f \mid e)$ ensures that e is a translation of f

Phrase-based Translation Model



- $P(f \mid e)$ "encodes" e into f as follows Chiang (2007):
  - segment e into phrases $\overline{e}_1 \cdots \overline{e}_I$, typically with a uniform distribution over segmentation
  - reorder the $\overline{e}_i$ according to some distortion model
  - translate each of the $\overline{e}_i$ into foreign phrases according to a model $P(\overline{f} \mid \overline{e})$ estimated from the training data

Phrase-based Translation Model



- $P(f \mid e)$ "encodes" e into f as follows Chiang (2007):
  - segment e into phrases $\overline{e}_1 \cdots \overline{e}_I$, typically with a uniform distribution over segmentation
  - reorder the $\overline{e}_i$ according to some distortion model
  - translate each of the $\overline{e}_i$ into foreign phrases according to a model $P(\overline{f} \mid \overline{e})$ estimated from the training data
- segment phrases further to capture the reordering of (sub-) phrases

Outline

1. Statistical Machine Translation
2. **Synchronous Context-free Grammar**
3. Training Procedure for SCFG

Synchronous Context-free Grammar (SCFG)

- An SCFG is a tuple $\mathcal{G} = (Q, \Sigma, q_0, R)$
  - Q finite set of states/nonterminals $(q_i, q', \ldots)$
  - Σ finite set of terminals $(a, b, c, \ldots)$
  - $q_0 \in Q$ initial state
  - R finite set of rules; each rule of the form

    $$\rho = q \to \langle \zeta, \zeta', \varphi \rangle$$

    where $\varphi : V \to Q$ for some finite set of variables $V \subset X$ and ζ, ζ' strings over $\Sigma \cup V$ s.t. every variable of V occurs exactly once in ζ and ζ'.
  - ζ **input string**, ζ' **output string**

Synchronous Context-free Grammar (SCFG)

- An SCFG is a tuple $\mathcal{G} = (Q, \Sigma, q_0, R)$
  - Q finite set of states/nonterminals $(q_i, q', \ldots)$
  - Σ finite set of terminals $(a, b, c, \ldots)$
  - $q_0 \in Q$ initial state
  - R finite set of rules; each rule of the form

    $$\rho = q \to \langle \zeta, \zeta', \varphi \rangle$$

    where $\varphi : V \to Q$ for some finite set of variables $V \subset X$ and ζ, ζ' strings over $\Sigma \cup V$ s.t. every variable of V occurs exactly once in ζ and ζ'.
  - ζ **input string**, ζ' **output string**
- probabilistic SCFG (pSCFG) $\mathcal{G}' = (\mathcal{G}, p)$ where $\mathcal{G}$ a SCFG and $p : R \to [0, 1]$ prob. assignment s.t. $q \in Q : \sum_{\rho \in R_q} p(\rho) = 1$, $R_q \subseteq R$ rules starting with q.

SCFG Example

Let $\Sigma = \{a, b, \#, \$\}$ and $Q = \{q_0, q_1\}$

$$\begin{aligned}
\rho_1 &= q_0 \to \langle a x_1 x_2, x_2 x_1 a, [x_1/q_0, x_2/q_1]\rangle \\
\rho_2 &= q_0 \to \langle \#, \#, \epsilon\rangle \\
\rho_3 &= q_1 \to \langle b x_1, x_1 b, [x_1/q_1]\rangle \\
\rho_4 &= q_1 \to \langle \$, \$, \epsilon\rangle
\end{aligned}$$

SCFG Example

Let $\Sigma = \{a, b, \#, \$\}$ and $Q = \{q_0, q_1\}$

$$\begin{aligned}
\rho_1 &= q_0 \rightarrow \langle a x_1 x_2, x_2 x_1 a, [x_1/q_0, x_2/q_1]\rangle \\
\rho_2 &= q_0 \rightarrow \langle \#, \#, \epsilon\rangle \\
\rho_3 &= q_1 \rightarrow \langle b x_1, x_1 b, [x_1/q_1]\rangle \\
\rho_4 &= q_1 \rightarrow \langle \$, \$, \epsilon\rangle
\end{aligned}$$

A complete derivation

$$(x_1, x_1, [x_1/q_0])$$

$$(aa\#b\$\$, \$\$b\#aa, \epsilon)$$

SCFG Example

Let $\Sigma = \{a, b, \#, \$\}$ and $Q = \{q_0, q_1\}$

$$\begin{aligned}
\rho_1 &= q_0 \rightarrow \langle a x_1 x_2, x_2 x_1 a, [x_1/q_0, x_2/q_1]\rangle \\
\rho_2 &= q_0 \rightarrow \langle \#, \#, \epsilon\rangle \\
\rho_3 &= q_1 \rightarrow \langle b x_1, x_1 b, [x_1/q_1]\rangle \\
\rho_4 &= q_1 \rightarrow \langle \$, \$, \epsilon\rangle
\end{aligned}$$

A complete derivation

$$\begin{aligned}
& (x_1, x_1, [x_1/q_0]) \\
\Rightarrow_{\rho_1} \quad & (a\underline{x_3}x_2, x_2\underline{x_3}a, [x_3/q_0, x_2/q_1])
\end{aligned}$$

$$(aa\#b\$\$, \$\$b\#aa, \epsilon)$$

SCFG Example

Let $\Sigma = \{a, b, \#, \$\}$ and $Q = \{q_0, q_1\}$

$$
\begin{aligned}
\rho_1 &= q_0 \rightarrow \langle a x_1 x_2, x_2 x_1 a, [x_1/q_0, x_2/q_1] \rangle \\
\rho_2 &= q_0 \rightarrow \langle \#, \#, \epsilon \rangle \\
\rho_3 &= q_1 \rightarrow \langle b x_1, x_1 b, [x_1/q_1] \rangle \\
\rho_4 &= q_1 \rightarrow \langle \$, \$, \epsilon \rangle
\end{aligned}
$$

A complete derivation

$$
\begin{aligned}
& (x_1, x_1, [x_1/q_0]) \\
\Rightarrow_{\rho_1} \quad & (a\underline{x_3}x_2, x_2\underline{x_3}a, [x_3/q_0, x_2/q_1]) \\
\Rightarrow_{\rho_1} \quad & (aa\underline{x_1}x_4x_2, x_2x_4\underline{x_1}aa, [x_2/q_1, x_1/q_0, x_4/q_1]) \\
\Rightarrow_{\rho_2} \quad & (aa\#\underline{x_4}x_2, x_2\underline{x_4}\#aa, [x_2/q_1, x_4/q_1]) \\
\Rightarrow_{\rho_3} \quad & (aa\#b\underline{x_1}x_2, x_2\underline{x_1}b\#aa, [x_2/q_1, x_1/q_1]) \\
\Rightarrow_{\rho_4} \quad & (aa\#b\$\underline{x_2}, \underline{x_2}\$b\#aa, [x_2/q_1]) \\
\Rightarrow_{\rho_4} \quad & (aa\#b\$\$, \$\$b\#aa, \epsilon)
\end{aligned}
$$

Outline

1. Statistical Machine Translation
2. Synchronous Context-free Grammar
3. **Training Procedure for SCFG**

Extracting SCFG

- **Joshua** Li et al. (2009), open source toolkit for parsing-based MT, SCFG-based, state of the art translation performance
- Canadian **Hansard** corpus, sentence-aligned French and English parallel corpus

Extracting SCFG

- **Joshua** Li et al. (2009), open source toolkit for parsing-based MT, SCFG-based, state of the art translation performance
- Canadian **Hansard** corpus, sentence-aligned French and English parallel corpus
- Grammar excerpt

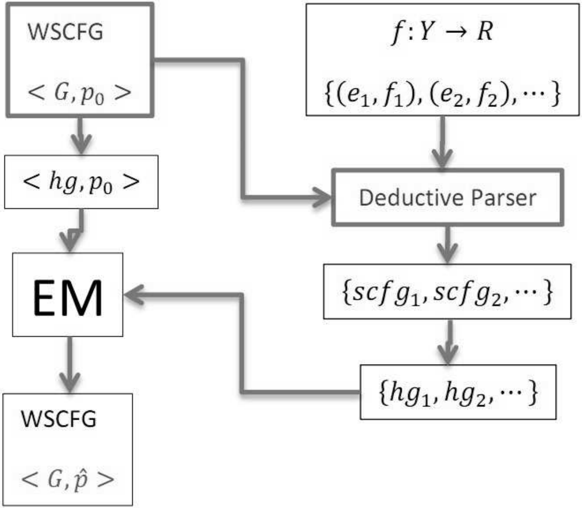
```
...
[X] ||| ajourne ||| until ||| 0.60206 1.6627579 0.845098
[X] ||| ajourne ||| summer ||| 0.60206 0.60206 0.845098
[X] ||| ajourne [X,1] ||| until [X,1] ||| -0.0 1.6627579
    0.845098
[X] ||| [X,1] ajourne ||| [X,1] until ||| 0.30103 1.6627579
    0.845098
[X] ||| [X,1] ajourne ||| [X,1] summer ||| 0.30103 0.60206
    0.845098
[X] ||| [X,1] ajourne [X,2] ||| [X,1] until [X,2] ||| -0.0
    1.6627579 0.845098
...
```

- ▸ $prob(\mathbf{e}|\mathbf{f})$, $lexprob(e|f)$, $lexprob(f|e)$

Training Procedure for SCFG

- **Idea**: given an SCFG $\mathcal{G}$ with p_0 (usually uniform prob. distri.) and a data set $f : \mathcal{Y} \rightarrow R$, train a probability distribution $\hat{p}$ on $\mathcal{G}$, i.e. pSCFG $\langle \mathcal{G}, \hat{p} \rangle$
- **Expectation-Maximization algorithm Prescher (2004)**
  - input: pSCFG $\langle \mathcal{G}, p_0 \rangle$ and incomplete-data corpus $f : \mathcal{Y} \rightarrow R$
  - output: pSCFG $\langle \mathcal{G}, \hat{p} \rangle$

Training Procedure for SCFG



Deductive Parser

- input: $\mathcal{G}$ and $(f, e) \in \Sigma_f^* \times \Sigma_e^*$
- output: $\mathcal{G}'$ s.t. $\mathcal{L}(\mathcal{G}') = \{(f, e)\}$ and $\mathcal{G}'$ derived from $\mathcal{G}$
- idea: use
  - output product of $\mathcal{G}$ and e, compute SCFG $\mathcal{G}^\circ$ s.t. $\mathcal{L}(\mathcal{G}^\circ) = \{(f', e) \mid (f', e) \in \mathcal{L}(\mathcal{G})\}$
  - input product of $\mathcal{G}^\circ$ and f, compute SCFG $\mathcal{G}^+$ s.t. $\mathcal{L}(\mathcal{G}^+) = \{(f, e') \mid (f', e') \in \mathcal{L}(\mathcal{G}^\circ)\} = \{(f, e)\} = \mathcal{L}(\mathcal{G}')$

Deductive Parser

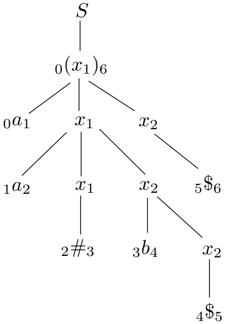
- input: $\mathcal{G}$ and $(f, e) \in \Sigma_f^* \times \Sigma_e^*$
- output: $\mathcal{G}'$ s.t. $\mathcal{L}(\mathcal{G}') = \{(f, e)\}$ and $\mathcal{G}'$ derived from $\mathcal{G}$
- idea: use
  - output product of $\mathcal{G}$ and e, compute SCFG $\mathcal{G}^\circ$ s.t. $\mathcal{L}(\mathcal{G}^\circ) = \{(f', e) \mid (f', e) \in \mathcal{L}(\mathcal{G})\}$
  - input product of $\mathcal{G}^\circ$ and f, compute SCFG $\mathcal{G}^+$ s.t. $\mathcal{L}(\mathcal{G}^+) = \{(f, e') \mid (f', e') \in \mathcal{L}(\mathcal{G}^\circ)\} = \{(f, e)\} = \mathcal{L}(\mathcal{G}')$
- output product $\mathcal{G} \triangleright e$ of $\mathcal{G}$ and e
  - $Q' = \{(q, v) \mid q \in Q, v$ subword of $e\}$, $q_0' = (q_0, e)$
  - R': let $\rho = (q \to \langle \zeta, \zeta', \varphi \rangle) \in R$ and v subword of e and $\nu \in \text{match}(\zeta', v)$. Then construct the following rule for R':

$$\begin{aligned} (\rho, v, \nu) &= ((q_0, v) \to \langle \zeta, \zeta', \varphi' \rangle) \text{ where} \\ \varphi'(x_i) &= (\varphi(x_i), \nu(x_i)) \\ p'((\rho, v, \nu)) &= p(\rho) \end{aligned}$$

  - example
    $\text{match}(x_1 x_2, aa) = \{[x_1/\epsilon, x_2/aa], [x_1/a, x_2/a], [x_1/aa, x_2/\epsilon]\}$

Generating New SCFG

- parsing as deduction, i.e. using Earley's algorithm Shieber et al. (1995) for **input string** "*aa#b$$*"; analogously for the **output string**



- new SCFG $\mathcal{G}'$ deriving exactly "*aa#b$$*"

$$\mathcal{G}' \left| \begin{array}{l} (0, S, 6) \to (0, x_1, 6) \\ (0, S, 6) \to (1, x_1, 5)(5, x_2, 6) \\ \ldots \\ (1, x_1, 5) \to (2, x_1, 3)(3, x_2, 5) \\ \ldots \end{array} \right.$$

Implementation

- recap $\rho_1 = q_0 \rightarrow \langle ax_1x_2, x_2x_1a, [x_1/q_0, x_2/q_1]\rangle$
- represent WSCFG, *WSCFG.hs*

```haskell
module Data.WSCFG (
  Production()
, WSCFG()
-- ...
) where
data RHS v t x = RHS
  { src :: [Either x t]
  , tgt :: [Either x t]
  , mps :: [(x, v)]
  } deriving (Read, Show)
-- ...
```

Implementation

- recap $\rho_1 = q_0 \rightarrow \langle a x_1 x_2, x_2 x_1 a, [x_1/q_0, x_2/q_1] \rangle$
- represent WSCFG, *WSCFG.hs*

```
-- ...
data Production v t w i x = Production
  { pLeft   :: v
  , pRight  :: RHS v t x
  , pWeight :: w
  , pId     :: i
  } deriving (Read, Show)
data WSCFG v t w i x = WSCFG
  { initial              :: v
  , productionsHG :: Hypergraph v (RHS v t x) w i
  } deriving (Read, Show)
-- ...
```

- deductive parser (in progress)

Summary of this talk

- phrased-based translation model
  - SCFG captures the reordering of (sub-)phrases and words
- implementation
  - data structure WSCFG
  - deductive parser based on Earley's algorithm

Questions and Feedback

Thank you for your attention!
Questions & Feedback?

Reference

Chiang, D. (2007). Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

Li, Z., Callison-Burch, C., Dyer, C., Ganitkevitch, J., Khudanpur, S., Schwartz, L., Thornton, W. N. G., Weese, J., and Zaidan, O. (2009). Demonstration of joshua: An open source toolkit for parsing-based machine translation. In *ACL/AFNLP (Software Demonstrations)*, pages 25–28.

Prescher, D. (2004). A tutorial on the expectation-maximization algorithm including maximum-likelihood estimation and em training of probabilistic context-free grammars. *CoRR*, abs/cs/0412015.

Shieber, S. M., Schabes, Y., and Pereira, F. C. N. (1995). Principles and implementation of deductive parsing. *J. Log. Program.*, 24(1&2):3–36.