Unsupervised Tokenization for Machine Translation

Tagyoung Chung   Daniel Gildea

Computer Science Department
University of Rochester
Tokenization

• Usually the first step of the SMT

• Two different problems for different languages
  – Finding unknown word boundaries
    (isolating languages such as Chinese)
  – Finding morpheme groupings of right granularity
    (agglutinative languages such as Korean)

• Supervised methods require training data or set of rules

• We present two unsupervised methods for MT
Problem of Granularity - Examples

- Korean word *meok-eoss-da* consists of three morphemes
  - *eat-past-indicative* (*TRANSLATION*: *ate*)
  - No reason to separate morphemes

- Korean word *hakgyo-e* consists of two morphemes
  - *school-locative* (*TRANSLATION*: *at school*)
  - It is desirable to separate morphemes in this case

- Bilingual model may help with this issue
Overview

- Monolingual model
  - Model description
  - Handling overfitting

- Bilingual model
  - Model description
  - Inference
  - Handling overfitting

- Results
Monolingual Model

• A quick way to learn tokenization

• Easiest solution: substring counts
  \[ P(w_i) = \frac{\text{count}(w_i)}{\sum_k \text{count}(w_k)} \]
  - Single pass through corpus
  - Learning probability with EM overfits
  - Simple substring counts overfits as well
Overfitting

• Problem: Longer substrings are preferred under this model

• However, shorter tokens are more frequent in the real world
Solution to Overfitting

• Control token size with length factor

• \( P(w_i) \propto \text{count}(w_i) \phi(|w_i|) \)

• Geometric distribution would be a natural choice:
  \[ \phi_1(\ell) = P(s)(1 - P(s))^{\ell-1} \]

• Observation: Heavier penalty for longer tokens is desired

• Doubly exponential length factor:
  \[ \phi_2(\ell) = 2^{-\ell^\lambda} \]
Length Factor

- Length factor vs. empirical token length distribution

\[
P(s) = 0.58
\]

\[
\lambda = 2.13
\]
Setting the Parameter

• Both $\phi_1$ and $\phi_2$ have a single parameter

• We set the parameter such that number of tokens in the half of the parallel corpus match the other half

• Justification:
  – Hypothesis: Tokenizing this way will produce tokens that are closer to an ideal situation thus result in better MT system
  – Ideal case: one-to-one correspondence between tokens of two languages
Related Work

- Goldwater et al. (2006) use geometric distribution as base distribution for Dirichlet process in their Bayesian segmentation model to model word acquisition in infants.

- Liang and Klein (2009) use doubly exponential length factor in their word segmentation model to test an online EM algorithm.

- Chang et al. (2008) use a feature in their CRF Chinese segmenter to tweak average size of tokens to improve MT performance.
Bilingual Model

- Can we learn segmentation of one language from the other language in parallel corpus?

- Our generative Model:

  ![Diagram]

  The model learns alignments and segmentation is by-product of alignments.
Inference

• The model uses IBM word alignment model 1
  \[ P(f | e) = \prod_i \sum_j P(f_i | e_j) P(a_i = j) \]

• \( f \) is unknown
  \[ f = s \circ c \]

• Apply dynamic programming over hidden segmentation \( s \)
  – Analogous to HMM’s forward-backward algorithm
    – Transition: segmentation
    – Emission: alignment
Forward-Backward Algorithm

\[ s = 0 \]

\[ s = 1 \]

\[ c_1 \quad c_2 \quad \ldots \quad c_i \quad \ldots \quad c_j \quad \ldots \quad c_m \]

\[ e_1 \quad \ldots \quad e_k \quad \ldots \quad e_n \]

\[
P(c_{i+1}^j, a = k | e) = \frac{\alpha(i)P(c_{i+1}^j | e_k)P(a = k)\beta(j)}{P(c | e)}
\]

\[
\alpha(i) = P(c_1^i, s_i = 1 | e)
\]

\[
\beta(j) = P(c_{j+1}^m, s_j = 1 | e)
\]
Overfitting

- We know the solution has to be **very** sparse
  - Solution: use a sparse prior
    \[
    \theta_e \mid \alpha \sim \text{Dir}(\alpha),
    \]
    \[
    f_i \mid e_i = e \sim \text{Multi}(\theta_e).
    \]
  - Use VB: minor change to inference (Johnson 2007)

- Further controlling overfitting with length factor
  - \( \phi_1 \) can be embedded in the model
    (the parameter can be learned)
  - \( \phi_2 \) can be used in the same manner as the first model
Related Work

• Kikui and Yamamoto (2002) use similar word alignment-based unsupervised segmentation to find new translation pairs from untokenized corpus.

• Xu et al. (2008) use similar word alignment-based segmentation model (using Gibbs sampling for inference) as part of their Chinese word segmenter.
Summary of Models

• Both models are unigram segmentation model

• Both models have explicit means to control size of tokens

• Monolingual model uses substring count to estimate \( P(f) \)

• Bilingual model uses word alignment to estimate

\[
P(f) = \sum_e P(f \mid e)P(e)
\]

• Both models use the Viterbi algorithm to find the best segmentation according to \( P(f) \)

• Both models limit maximum size of \( f \) for practical reasons
Experiments

- MT Systems for Chi-Eng, and Kor-Eng language pairs
  - 2M words on English side for both language pairs
  - monolingual/bilingual models with length factors
  - Moses (Koehn et al., 2007)

- Three Questions:
  - How do the models compare to other tokenization?
  - What are the effects of length factors?
  - Does bilingual model learn to segment better?
## Comparison to Supervised Segmentation

| Supervised                                      | Chinese  | Korean |
|------------------------------------------------|----------|--------|
| Rule-based morphological analyzer              |          | 7.27   |
| LDC segmenter                                   | 20.03    |        |
| Xue’s segmenter                                 | 23.02    |        |
| Stanford segmenter (pku)                        | 21.69    |        |
| Stanford segmenter (ctb)                        | 22.45    |        |

| Unsupervised                                    | Chinese  | Korean |
|------------------------------------------------|----------|--------|
| Bilingual model with $\phi_1 \ P(s) = 0.9$     | 20.75    | 7.46   |
| Bilingual model with $\phi_2$                  | 22.31    | 7.35   |
Effect of length factor

Chinese
## Monolingual vs. Bilingual

| Model Description | Chinese | Korean |
|-------------------|---------|--------|
| Bilingual model with $\phi_1$, $P(s) = learned$ | 20.04 | 7.06 |
| Bilingual model with $\phi_1$, $P(s) = 0.9$ | 20.75 | **7.46** |
| Bilingual model with $\phi_1$, $P(s) = 0.7$ | 20.59 | 7.31 |
| Bilingual model with $\phi_1$, $P(s) = 0.5$ | 19.68 | 7.18 |
| Bilingual model with $\phi_1$, $P(s) = 0.3$ | 20.02 | 7.38 |
| Bilingual model with $\phi_2$ | **22.31** | 7.35 |
| Monolingual model with $\phi_1$ | 20.93 | 6.76 |
| Monolingual model with $\phi_2$ | 20.72 | 7.02 |
Summary

• Unsupervised tokenization methods are comparable to supervised ones in use for MT

• Bilingual model does learn better token probability for MT

• Heavier penalty for longer tokens is a useful means to prevent overfitting in segmentation

• Need to optimize parameters for end-to-end translation quality
## Additional Results

| Model                                                                 | Chinese Score |
|----------------------------------------------------------------------|---------------|
| Monolingual model with $\phi_1$ (EM)                                 | 15.70         |
| Monolingual model with $\phi_2$ (EM)                                 | 21.30         |
| Monolingual model with $\phi_1$                                      | 20.93         |
| Monolingual model with $\phi_2$                                      | 20.72         |