Reranking with Multiple Features for Better Transliteration

Yan Song, Chunyu Kit, Hai Zhao

ACL 2010

Min-Hsuan Lai

Department of Computer Science & Information Engineering
National Taiwan Normal University

Outline

- Introduction
- Generation
- Reranking
  - Learning Framework
  - Multiple Features
- Results
- Conclusion

Introduction

- The top-1 accuracy for the generated candidates cannot be good if the right one is not ranked at the top.
- To tackle this issue, they propose to rerank the output candidates for a better order using the averaged perceptron with multiple features.
- In this paper, they present their recent work on reranking the transliteration candidates via an online discriminative learning framework.
- Multiple features are incorporated into it for performance enhancement.

Generation

- For the generation of transliteration candidates, they follow the work (Song et al., 2009), using a phrase-based SMT procedure with the log-linear model for decoding.

$$P(t \mid s) = \frac{\exp\left[\sum_{i=1}^{n} \lambda_i h_i(s,t)\right]}{\sum_t \exp\left[\sum_{i=1}^{n} \lambda_i h_i(s,t)\right]} \qquad (1)$$

- The parameter for each feature function in log-linear model is optimized by MERT training (Och, 2003).

- Finally, a maximum number of 50 candidates are generated for each source name.

Reranking

Learning Framework

- For reranking training and prediction, they adopt the averaged perceptron (Collins, 2002) as their learning framework, which has a more stable performance than the non-averaged version.
- Where $\vec{w}$ is the vector of parameters they want to optimize, *x*, *y* are the corresponding source (with different syllabification) and target graphemes in the candidate list, and Φ represents the feature vector in the pair of *x* and *y*.
- In this algorithm, reference y_i^* is the most appropriate output in the candidate list according to the true target named entity in the training data.

Reranking

Learning Framework

Algorithm 1 Averaged perceptron training

Input: Candidate list with reference

$\{LIST(x_j, y_j)_{j=1}^{n}, y_i^*\}_{i=1}^{N}$

Output: Averaged parameters

$\vec{\omega} \leftarrow 0$, $\vec{\omega}_a \leftarrow 0$, $c \leftarrow 1$

for $t = 1$ **to** T **do**

  for $i = 1$ **to** N **do**

    $\hat{y}_i \leftarrow argmax_{y \in LIST(x_j, y_j)} \vec{\omega} \cdot \Phi(x_i, y_i)$

    if $\hat{y}_i \neq y_i^*$ **then**

      $\vec{\omega} \leftarrow \vec{\omega} + \Phi(x_i^*, y_i^*) - \Phi(\hat{x}_i, \hat{y}_i)$

      $\vec{\omega}_a \leftarrow \vec{\omega}_a + c \cdot \{\Phi(x_i^*, y_i^*) - \Phi(\hat{x}_i, \hat{y}_i)\}$

    end if

    $c \leftarrow c + 1$

  end for

end for

return $\vec{\omega} - \vec{\omega}_a / c$

Reranking

Multiple Features

- The following features are used in their reranking process:
  - *Transliteration correspondence feature*, $f(s_i, t_i)$;
    - ➢ This feature describes the mapping between source and target graphemes, similar to the transliteration options in the phrase table in their previous generation process.
  - *Source grapheme chain feature*, $f(s_{i-1}^{i})$;
    - ➢ These features on different source grapheme levels can help the system to achieve a more reliable syllabification result from the candidates. They only consider bi-grams when using this feature.
  - *Target grapheme chain feature*, $f(t_{i-2}^{i})$;
    - ➢ It performs in a similar way as the language model for SMT decoding. They use *tri*-gram syllables in this learning framework.

Reranking
Multiple Features

– *Paired source-to-target transition feature*, $f\left(\langle s,t\rangle_{i-1}^{i}\right)$;
  - This type of feature is firstly proposed in (Li et al., 2004), aiming at generating source and target graphemes simultaneously under a suitable constraint.

– *Hidden Markov model (HMM) style features*;
  - There are a group of features with HMM style constraint for evaluating the candidates generated in previous SMT process,

– *Target grapheme position feature*, $f(t_i, p)$;
  - This feature is used to exploit the observation that some characters are more likely to appear at certain positions in the target name.

– *Target tone feature*;
  - This feature is only applied to the transliteration task with Chinese as the target language.

Reranking

Multiple Features

- Besides the above string features, they also have some numeric features, as listed below
  - *Transliteration score*;
    - This score is the joint probabilities of all transliteration options, included in the output candidates generated by our decoder.
  - *Target language model score*;
    - This score is calculated from the probabilistic tri-gram language model.
  - *Source/target Pinyin feature*;
    - It measures how good the output candidates can be in terms of the comparison between English text and Pinyin representation.

Reranking

Multiple Features

- For a task with English as the target language, they add the following two additional features into the learning framework.
  - *Vowel feature*;
    - ➢ This feature is thus used to punish those outputs unqualified to be a valid English word for carrying no vowel.
  - *Syllable consistent feature*;
    - ➢ This feature measures whether an English target name generated in the previous step has the same number of syllables as the source name.

Results

- For NEWS2010, they participated in all two Chinese related transliteration tasks, namely, EnCh (English-to-Chinese) and ChEn (Chineseto-English back transliteration).

- The official evaluation scores for our submissions are presented in Table 1 with recall rate, and the ACC score (ACC_{SMT}) for original SMT outputs.

Table 1: Evaluation results for our NEWS2010 task.

Task	Source	Target	ACC	Mean F	MRR	Map_ref	Recall	ACC_{SMT}
EnCh	English	Chinese	0.477	0.740	0.506	0.455	0.561	0.381
ChEn	Chinese	English	0.227	0.749	0.269	0.226	0.371	0.152

Conclusion

- It NEWS2010 results show that this approach is effective and promising, in the sense that it ranks the best in EnCh and ChEn tasks.

- Though, those features are strongly dependent on the nature of English and Chinese languages, it is thus not an easy task to transplant this model for other language pairs.

- It is an interesting job to turn it into a language independent model that can be applied to other languages.