Hidden Markov Model (HMM) Factorization in Neural Machine Translation

Parnia Bahar and Hermann Ney

<surname>@i6.informatik.rwth-aachen.de

OpenNMT Workshop
Paris, France, March 02, 2018

Human Language Technology and Pattern Recognition
Computer Science Department, RWTH Aachen University

Introduction

- **Machine translation (MT) aims to translate sentences from one language into another without any human interaction**
- **Given a source sequence $f_1^J = f_1, \cdots, f_J$ of length J and a target sequence $e_1^I = e_1, \cdots, e_I$ of length I, the posterior probability distribution is:**

$$p(e_1^I | f_1^J)$$

- **Word alignment between source and target sequences is a crucial component in MT**
- **Alignment information is modeled either implicitly or explicitly**

Introduction

- **Traditional phrase-based system:**
  - hidden Markov model (HMM) [Vogel & Ney$^+$ 96] and IBM models [Brown & Pietra$^+$ 93] have been widely used to build word alignments explicitly
- **Attention-based system:**
  - attention weights select positions of the source sentence while decoding a target word
  - an implicit probabilistic notion of alignment as an intermediate step of the translation model
- It does not work the same way as its analogy of alignment models in the phrase-based system
- The attention weights do not directly influence the final translation score of a sentence

Motivation

- An approach to mimic the behavior of the HMM-based alignment
- Follow the direct HMM factorization to separate the alignment and the lexicon model for more structured alignment scores
- A separate translation score for every target-source word pair

Attention Model

- Translation from one language to another can be done by **a single neural network**
- The input and output are both variable-length sequences
- Encoder reads the source sentence, encodes it into a set of vectors, h_1^J
- Position-dependent weighted sum of these vectors, c_i where the most relevant information is concentrated
- Decoder generates an output sequence conditioned on encoder representations
- Learn to align and translate simultaneously

Attention Model

[Bahdanau & Cho+ 15]

- **The attention model is based on the encoder-decoder architecture which consists of two long short-term memories (LSTMs)**

$$\overrightarrow{h_j} = LSTM\big(f_j, h_{j-1}\big)$$
$$\overleftarrow{h_j} = LSTM\big(f_j, h_{j+1}\big)$$
$$h_j = \big[\overrightarrow{h_j}; \overleftarrow{h_j}\big]$$

- **While computing e_i at each time step, an attention function, a is used**

- **Consider as alignment probabilities that scores how likely the source word, f_j, is aligned to the current target word, e_i**

$$\alpha'_{i,j} = a\big(s_{i-1}, h_j\big)$$
$$\alpha_{i,j} = softmax(\alpha'_{i,j'})$$

Attention Model

- **The context vector c_i is then computed as a weighed sum of encoder representations**
- **The decoder state is updated to s_i**

$$
\begin{aligned}
c_i &= \sum_{j=1}^{J} \alpha_{i,j} h_j \\
e_i &= softmax(e_{i-1}, s_{i-1}, c_i) \\
s_i &= LSTM(e_i, s_{i-1}, c_i)
\end{aligned}
$$

- **Using the chain rule, the posterior probability distribution of the target sequence:**

$$
p(e_1^I | f_1^J) = \prod_{i=1}^{I} p(e_i | e_1^{i-1}, f_1^J)
$$

HMM-based Factorization Model

- Similar to the direct HMM-based approach, the word alignment as hidden variable is defined from target, i to source, j i.e. $i \rightarrow b_i = j$
- Decompose the posterior probability distribution of the target sequence into two parts: alignment model and lexicon model
- Introduce variable j as an hidden variable

$$p(e_1^I|f_1^J) = \prod_{i=1}^{I} \sum_{j=1}^{J} p(e_i, j|e_1^{i-1}, f_1^J)$$

- Using Markov assumption w.r.t the alignments:

$$p(e_1^I|f_1^J) = \prod_{i=1}^{I} \sum_{j=1}^{J} \underbrace{p(j|e_1^{i-1}, f_1^J)}_{\text{alignment model}} \cdot \underbrace{p(e_i|e_1^{i-1}, j, f_1^J)}_{\text{lexicon model}}$$

HMM-based Model

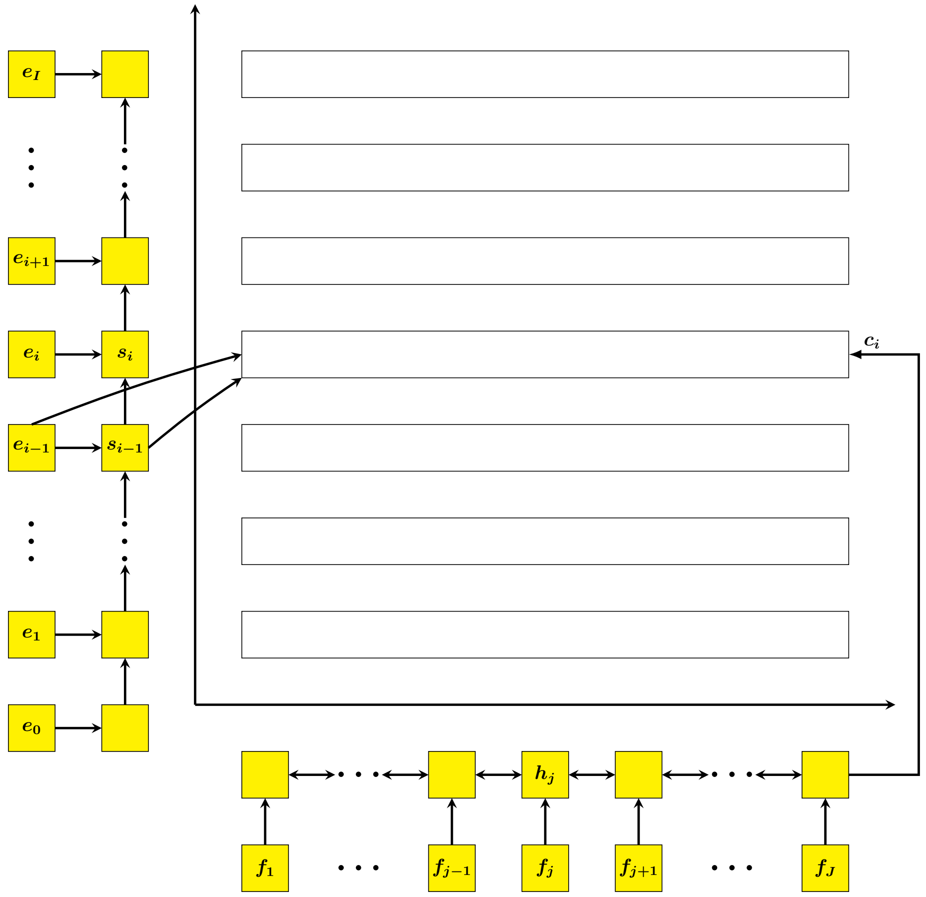
- **Using LSTM representation:**
  - **LSTM in the decoder encodes target histories** $e_1^{i-1} \rightarrow s_{i-1}$
  - **LSTMs in the encoder represent the source word** $f_1^J \rightarrow h_1^J$

$$P(e_1^I|f_1^J) = \prod_{i=1}^{I} \sum_{j=1}^{J} \underbrace{p(j|s_{i-1}, h_j)}_{\text{alignment model} = \alpha_{i,j}} \cdot \underbrace{p(e_i|e_{i-1}, s_{i-1}, h_j)}_{\text{lexicon model}}$$

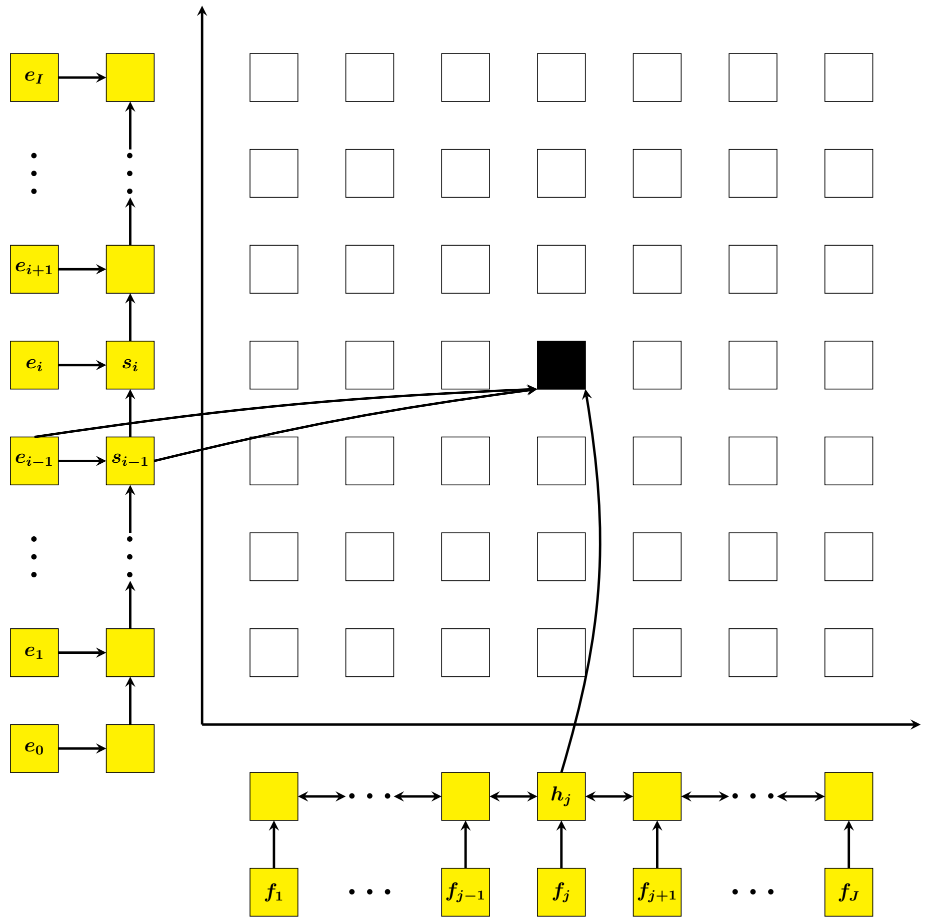
- **Calculate the score of lexicon for each source representation** h_j
- **This lexicon probability is computed by a softmax operation** J **times instead of once for the context vector,** c_i
- **Attention weights serve as alignment probabilities**

HMM-based Model vs. Attention Model

- The main difference between two models is the level of averaging



Attention model



HMM-based model

Experiments

- **Setup:**
  - in-house implementation of NMT approach which relies on the Blocks framework [Merriënboer & Bahdanau$^+$ 15] and Theano [Bastien & Lamblin$^+$ 12]
  - German→English and English→German consisting of 4.6M samples
  - Chinese→English consisting of 23M samples
  - byte pair encoding with 20k operations
  - 620-dimensional embedding both on the source and on the target
  - LSTM nodes with peephole connections using 1000 cells
  - Adam optimizer and dropout of 30%
  - the final model is the average of 4 best snapshots
- **Evaluation:**
  - case-sensitive BLEU computed by `mteval-v13a`
  - case-sensitive TER computed by `tercom`

Translation Results

Models	De-En				En-De				Zh-En	
	newstest2016		newstest2017		newstest2016		newstest2017		newstest2017	
	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER
attention model	33.1	47.6	28.6	52.8	28.2	52.9	23.2	59.6	19.0	67.0
HMM-based model	33.5	47.1	29.1	51.7	29.1	51.7	23.6	58.6	20.0	64.9

Table: Results measured in BLEU [%] and TER [%] on the test sets.

- HMM-based model outperforms a well-tuned attention mechanism on average by:
  - 0.5% BLEU and 0.8% TER on De→En
  - 0.6% BLEU and 1.1% TER on En→De
  - 1.0% BLEU and 2.1% TER on Zh→En
- Lexicalized alignment model assigns properer scores for the target-source word pairs

Speed

- **Employ a softmax layer over target vocabulary to calculate lexicon scores J times**
- **HMM-based model is slower than the standard attention model in training**
  - **attention model:** $0.26\frac{sec}{batch}$
  - **HMM-based model:** $0.47\frac{sec}{batch}$

Future Work

- Incorporate the approach in other architectures like Transformer [Vaswani & Shazeer^{+} 17] and CNN-NMT [Gehring & Auli^{+} 17]
- Extend both lexicon and alignment models with more elaborate dependencies
- In order to speed up, one can sample from the alignment distribution, take the k best source positions and compute only the corresponding lexicon probabilities

Thank you for your attention

Parnia Bahar and Hermann Ney

`<surname>@cs.rwth-aachen.de`

Reference

- **D. Bahdanau, K. Cho, Y. Bengio.**
  Neural machine translation by jointly learning to align and translate.
  ***CoRR*, Vol. abs/1409.0473, 2015.**

- **F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, Y. Bengio.**
  Theano: new features and speed improvements.
  Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

- **P. F. Brown, S. D. Pietra, V. J. D. Pietra, R. L. Mercer.**
  The mathematics of statistical machine translation: Parameter estimation.
  ***Computational Linguistics*, Vol. 19, No. 2, pp. 263–311, 1993.**

- **J. Gehring, M. Auli, D. Grangier, D. Yarats, Y. N. Dauphin.**
  Convolutional sequence to sequence learning.
  In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pp. 1243–1252, 2017.

- **B. Merriënboer, D. Bahdanau, V. Dumoulin, D. Serdyuk, D. Warde-Farley, J. Chorowski, Y. Bengio.**
  Blocks and fuel: Frameworks for deep learning.
  Vol., 2015.

- **R. Sennrich, B. Haddow, A. Birch.**
  Neural machine translation of rare words with subword units.
  In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, 2016.

- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin.
  Attention is all you need.
  In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pp. 6000–6010, 2017.

- S. Vogel, H. Ney, C. Tillmann.
  Hmm-based word alignment in statistical translation.
  In *16th International Conference on Computational Linguistics, Proceedings of the Conference, COLING 1996, Center for Sprogteknologi, Copenhagen, Denmark, August 5-9, 1996*, pp. 836–841, 1996.