Natural Language Processing (CSEP 517): Machine Translation (Continued), Summarization, & Finale

Noah Smith

University of Washington
nasmith@cs.washington.edu

May 22, 2017

To-Do List

- Online quiz: due Sunday
- A5 due May 28 (Sunday)
- Watch for final exam instrutions around May 29 (Monday)

Neural Machine Translation

Original idea proposed by Forcada and Ñeco (1997); resurgence in interest starting around 2013.

Strong starting point for current work: Bahdanau et al. (2014). (My exposition is borrowed with gratitude from a lecture by Chris Dyer.)

This approach eliminates (hard) alignment and phrases.

Take care: here, the terminology "encoder" and "decoder" are used differently than in the noisy-channel pattern.

High-Level Model

$$\begin{aligned} p(\boldsymbol{E} = \boldsymbol{e} \mid \boldsymbol{f}) &= p(\boldsymbol{E} = \boldsymbol{e} \mid \mathsf{encode}(\boldsymbol{f})) \\ &= \prod_{j=1}^{\ell} p(e_j \mid e_0, \ldots, e_{j-1}, \mathsf{encode}(\boldsymbol{f})) \end{aligned}$$

The encoding of the source sentence is a *deterministic* function of the words in that sentence.

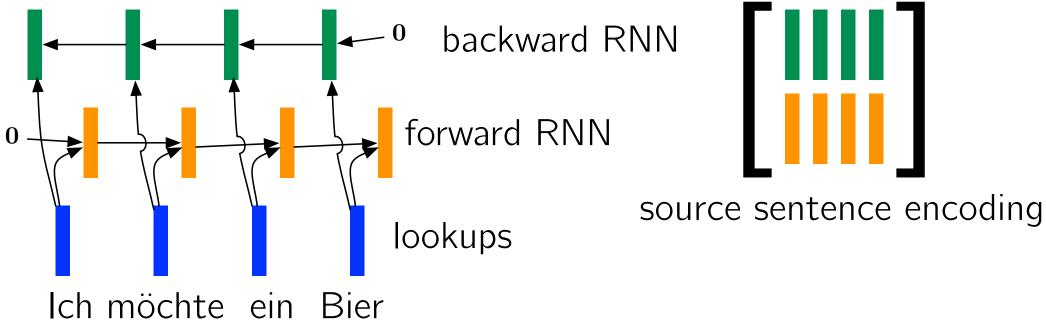
Building Block: Recurrent Neural Network

Review from lecture 2!

- Each input element is understood to be an element of a sequence: $\langle \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_\ell \rangle$
- At each timestep t:
    - The tth input element $\mathbf{x}_t$ is processed alongside the previous state $\mathbf{s}_{t-1}$ to calculate the new **state** $(\mathbf{s}_t)$.
    - The tth output is a function of the state $\mathbf{s}_t$.
    - The *same functions* are applied at each iteration:

$$\begin{aligned} \mathbf{s}_t &= g_{\text{recurrent}}(\mathbf{x}_t, \mathbf{s}_{t-1}) \\ \mathbf{y}_t &= g_{\text{output}}(\mathbf{s}_t) \end{aligned}$$

Neural MT Source-Sentence Encoder



$\mathbf{0}$

backward RNN

$\mathbf{0}$

forward RNN

lookups

Ich möchte ein Bier

source sentence encoding

$\mathbf{F}$ is a $d \times m$ matrix encoding the source sentence $\boldsymbol{f}$ (length m).

Decoder: Contextual Language Model

Two inputs, the previous word and the source sentence context.

$$\mathbf{s}_t = g_{\text{recurrent}}(\mathbf{e}_{e_{t-1}}, \underbrace{\mathbf{F}\mathbf{a}_t}_{\text{“context”}}, \mathbf{s}_{t-1})$$

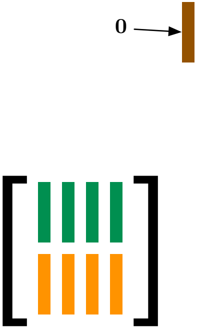
$$\mathbf{y}_t = g_{\text{output}}(\mathbf{s}_t)$$

$$p(E_t = v \mid e_1, \ldots, e_{t-1}, \boldsymbol{f}) = [\mathbf{y}_t]_v$$

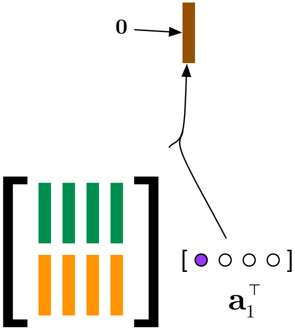
(The forms of the two component gs are suppressed; just remember that they (i) have parameters and (ii) are differentiable with respect to those parameters.)

The neural language model we discussed earlier (Mikolov et al., 2010) didn't have the context as an input to $g_{\text{recurrent}}$.

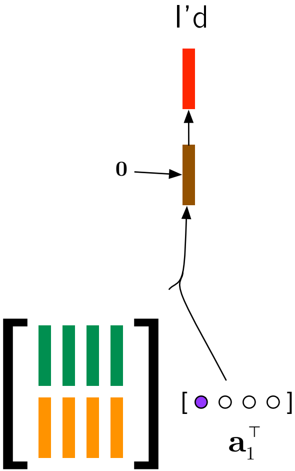
Neural MT Decoder



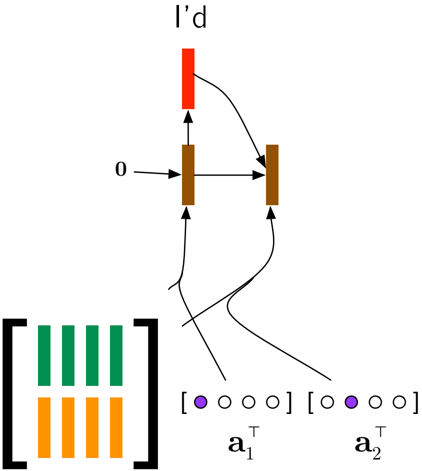
Neural MT Decoder



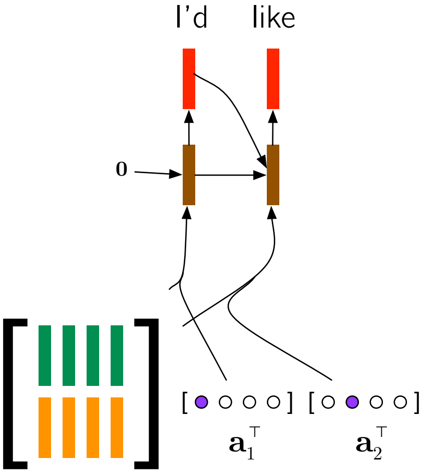
Neural MT Decoder



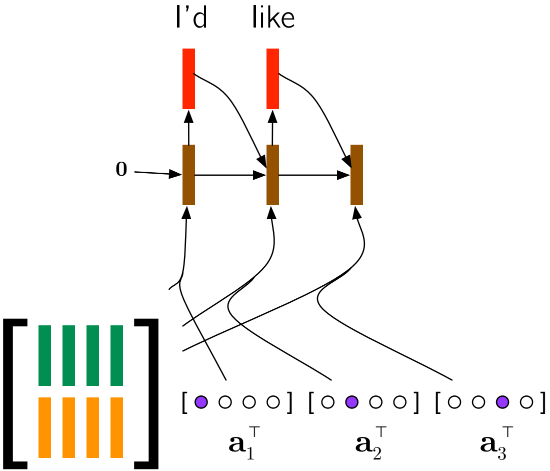
Neural MT Decoder



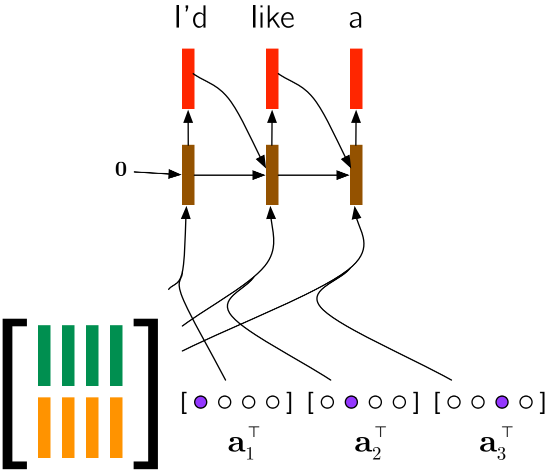
Neural MT Decoder



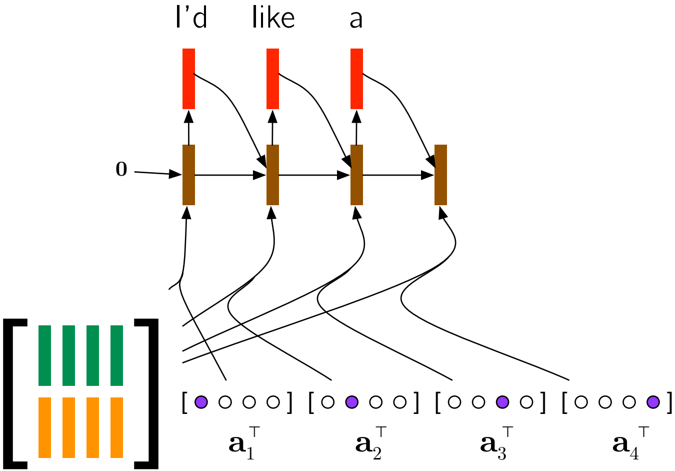
Neural MT Decoder



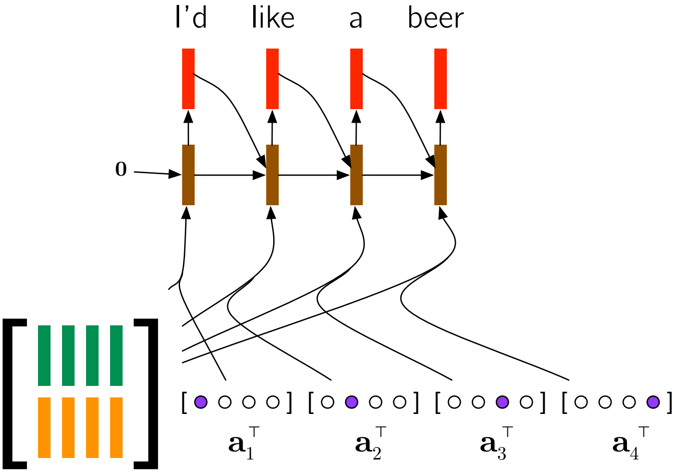
Neural MT Decoder



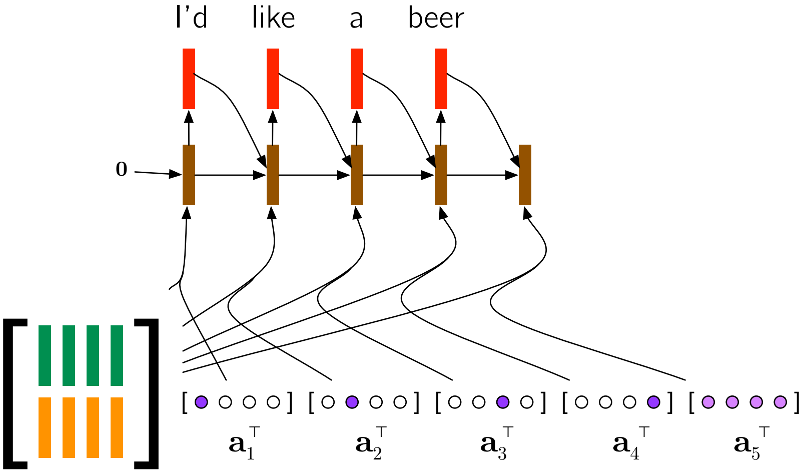
Neural MT Decoder



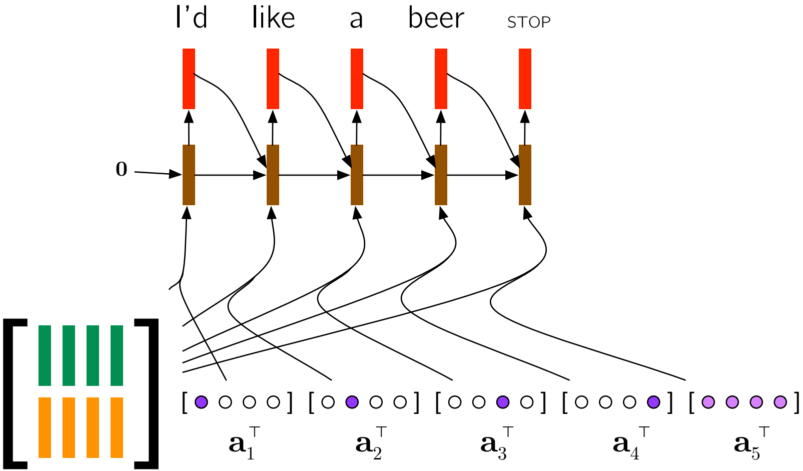
Neural MT Decoder



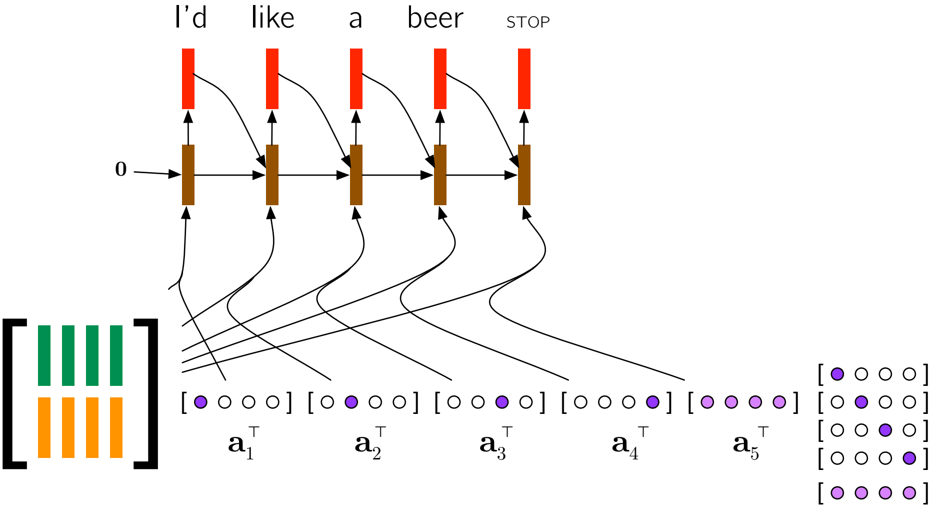
Neural MT Decoder



Neural MT Decoder



Neural MT Decoder



Computing "Attention"

Let $\mathbf{V}\mathbf{s}_{t-1}$ be the "expected" input embedding for timestep t.
(Parameters: $\mathbf{V}$.)

Attention is $\mathbf{a}_t = \text{softmax}\left(\mathbf{F}^\top \mathbf{V}\mathbf{s}_{t-1}\right)$.

Context is $\mathbf{F}\mathbf{a}_t$, i.e., a weighted sum of the source words' in-context representations.

Learning and Decoding

$$\log p(\boldsymbol{e} \mid \mathsf{encode}(\boldsymbol{f})) = \sum_{i=1}^{m} \log p(e_i \mid \boldsymbol{e}_{0:i-1}, \mathsf{encode}(\boldsymbol{f}))$$

is differentiable with respect to all parameters of the neural network, allowing "end-to-end" training.

Trick: train on shorter sentences first, then add in longer ones.

Decoding typically uses beam search.

Remarks

We covered two approaches to machine translation:

- Phrase-based statistical MT following Koehn et al. (2003), including probabilistic noisy-channel models for alignment (a key preprocessing step; Brown et al., 1993), and
- Neural MT with attention, following Bahdanau et al. (2014).

Note two key differences:

- Noisy channel $p(\boldsymbol{e}) \times p(\boldsymbol{f} \mid \boldsymbol{e})$ vs. “direct” model $p(\boldsymbol{e} \mid \boldsymbol{f})$
- Alignment as a discrete random variable vs. attention as a deterministic, differentiable function

At the moment, neural MT is winning when you have enough data; if not, phrase-based MT dominates.

When monolingual target-language data is plentiful, we’d like to use it! Recent neural models try (Sennrich et al., 2016; Xia et al., 2016; Yu et al., 2017).

Summarization

Automatic Text Summarization

Mani (2001) provides a survey from before statistical methods came to dominate; more recent survey by Das and Martins (2008).

Parallel history to machine translation:

- Noisy channel view (Knight and Marcu, 2002)
- Automatic evaluation (Lin, 2004)

Differences:

- Natural data sources are less obvious
- Human information needs are less obvious

We'll briefly consider two subtasks: **compression** and **selection**

Sentence Compression as Structured Prediction

(McDonald, 2006)

Input: a sentence

Output: the same sentence, with some words deleted

McDonald's approach:

- Define a scoring function for compressed sentences that factors locally in the output.
  - He factored into *bigrams* but considered input parse tree features.
- Decoding is dynamic programming (not unlike Viterbi).
- Learn feature weights from a corpus of compressed sentences, using structured perceptron or similar.

Sentence Selection

Input: one or more documents and a "budget"

Output: a within-budget subset of sentences (or passages) from the input

Challenge: **diminishing returns** as more sentences are added to the summary.

Classical greedy method: "maximum marginal relevance" (Carbonell and Goldstein, 1998)

Casting the problem as **submodular optimization**: Lin and Bilmes (2009)

Joint selection and compression: Martins and Smith (2009)

Finale

Mental Health for Exam Preparation (and Beyond)

Most lectures included discussion of:

- Representations or tasks (input/output)
- Evaluation criteria
- Models (often with variations)
- Learning/estimation algorithms
- NLP algorithms
- Practical advice

For each task, keep these elements separate in your mind, and reuse them where possible.

References I

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proc. of ICLR*, 2014. URL `https://arxiv.org/abs/1409.0473`.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

Jaime Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proc. of SIGIR*, 1998.

Dipanjan Das and André F. T. Martins. A survey of methods for automatic text summarization, 2008.

Mikel L. Forcada and Ramón P. Ñeco. Recursive hetero-associative memories for translation. In *International Work-Conference on Artificial Neural Networks*, 1997.

Kevin Knight and Daniel Marcu. Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, 139(1):91–107, 2002.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proc. of NAACL*, 2003.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Proc. of ACL Workshop: Text Summarization Branches Out*, 2004.

Hui Lin and Jeff A. Bilmes. How to select a good training-data subset for transcription: Submodular active selection for sequences. In *Proc. of Interspeech*, 2009.

Inderjeet Mani. *Automatic Summarization*. John Benjamins Publishing, 2001.

References II

André F. T. Martins and Noah A. Smith. Summarization with a joint model for sentence extraction and compression. In *Proc. of the ACL Workshop on Integer Linear Programming for Natural Langauge Processing*, 2009.

Ryan T. McDonald. Discriminative sentence compression with soft syntactic evidence. In *Proc. of EACL*, 2006.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Proc. of Interspeech*, 2010. URL `http://www.fit.vutbr.cz/research/groups/speech/publi/2010/mikolov_interspeech2010_IS100722.pdf`.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. In *Proc. of ACL*, 2016. URL `http://www.aclweb.org/anthology/P16-1009`.

Yingce Xia, Di He, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. Dual learning for machine translation. In *NIPS*, 2016.

Lei Yu, Phil Blunsom, Chris Dyer, Edward Grefenstette, and Tomas Kocisky. The neural noisy channel. In *Proc. of ICLR*, 2017.