Sequence-to-Sequence Architectures

Kapil Thadani
kapil@cs.columbia.edu

YAHOO!
RESEARCH

Previously: processing text with RNNs

Inputs

- One-hot vectors for words/characters/previous output
- Embeddings for words/sentences/context
- CNN over characters/words/sentences

$\vdots$

Recurrent layers

- Forward, backward, bidirectional, deep
- Activations: σ, tanh, gated (LSTM, GRU), ReLU initialized with identity

$\vdots$

Outputs

- Softmax over words/characters/labels
- Absent (i.e., pure encoders)

$\vdots$

Outline

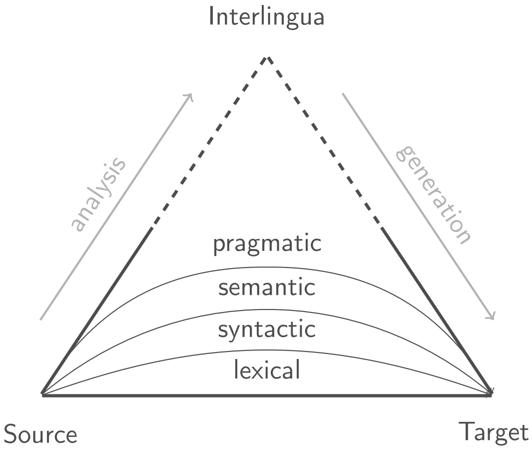
- Machine translation
  - Phrase-based MT
  - Encoder-decoder architecture
- Attention
  - Mechanism
  - Visualizations
  - Variants
  - Transformers
- Decoding large vocabularies
  - Alternatives
  - Copying
- Autoencoders
  - Denoising autoencoders
  - Variational autoencoders (VAEs)

Machine Translation

"*One naturally wonders if the problem of translation could conceivably be treated as a problem in cryptography. When I look at an article in Russian, I say: 'This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode.*"

— Warren Weaver
Translation (1955)

The MT Pyramid

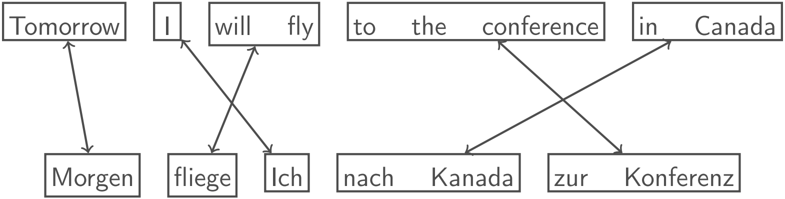


Phrase-based MT

Tomorrow I will fly to the conference in Canada

Morgen fliege Ich nach Kanada zur Konferenz

Phrase-based MT



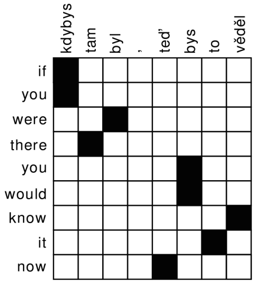
Phrase-based MT

1. Collect bilingual dataset $\langle S_i, T_i \rangle \in \mathcal{D}$

2. Unsupervised phrase-based alignment
   - phrase table π

3. Unsupervised n-gram language modeling
   - language model ψ

4. Supervised decoder
   - parameters θ



$$\widehat{T} = \arg\max_{T} \; p(T|S)$$
$$= \arg\max_{T} \; p(S|T,\pi,\theta) \cdot p(T|\psi)$$

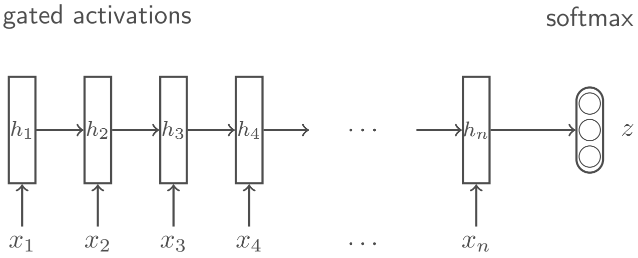
Neural MT

1. Collect bilingual dataset $\langle S_i, T_i \rangle \in \mathcal{D}$
2. Unsupervised phrase-based alignment
   - phrase table π
3. Unsupervised n-gram language modeling
   - language model ψ
4. Supervised encoder-decoder framework
   - parameters θ

RNN

Input words $x_1, \ldots, x_n$

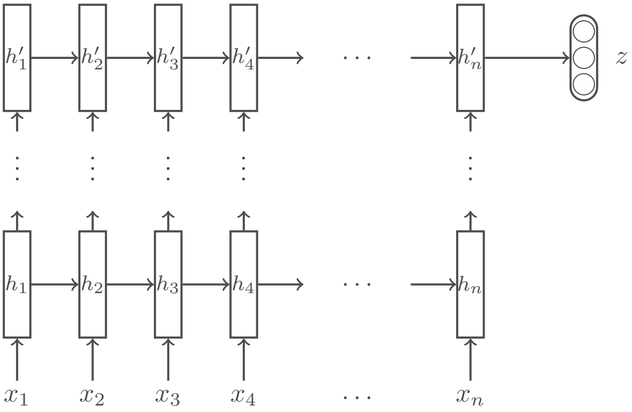
Output label z



Deep RNN

Input words $x_1, \ldots, x_n$

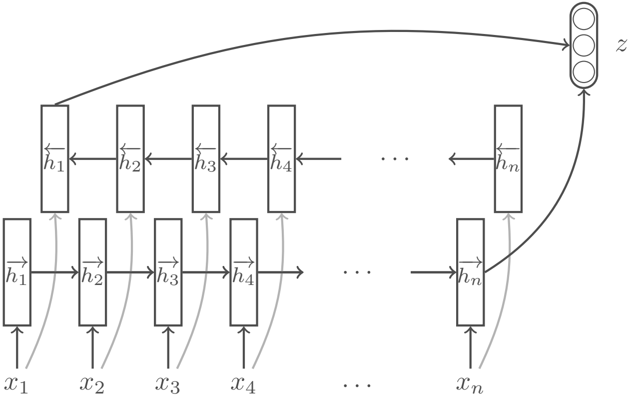
Output label z



Bidirectional RNN

Input words $x_1, \ldots, x_n$

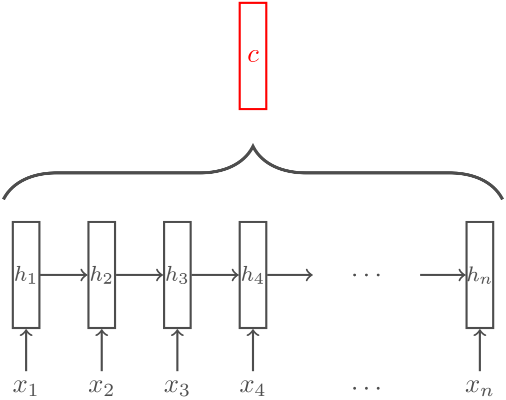
Output label z



RNN encoder

Input words $x_1, \ldots, x_n$

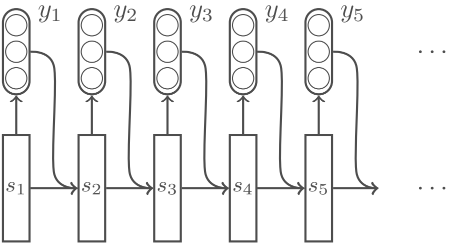
Output encoding c



RNN language model

Input words $y_1, \ldots, y_k$

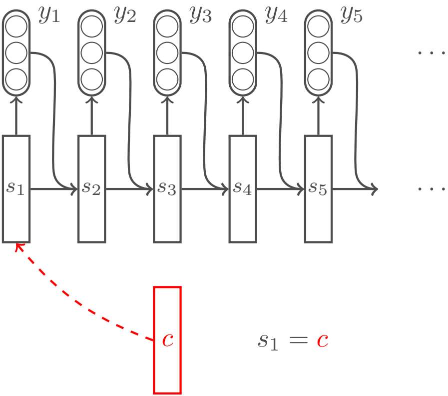
Output following words $y_k, \ldots, y_m$



RNN decoder

Input context c

Output words $y_1, \ldots, y_m$

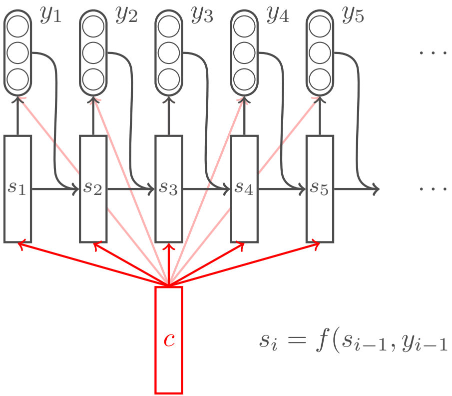


$$s_1 = c$$

RNN decoder

Input context c

Output words $y_1, \ldots, y_m$



$$s_i = f(s_{i-1}, y_{i-1}, c)$$

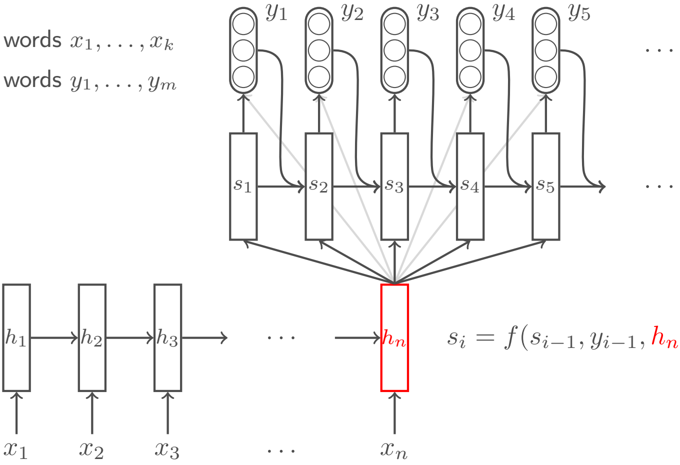
Sequence-to-sequence learning

Sutskever, Vinyals & Le (2014)

Sequence to Sequence Learning with Neural Networks

Input words $x_1, \ldots, x_k$

Output words $y_1, \ldots, y_m$



$$s_i = f(s_{i-1}, y_{i-1}, h_n)$$

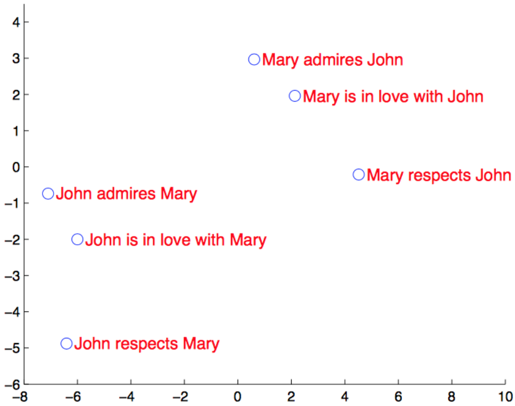
Sequence-to-sequence learning

Sutskever, Vinyals & Le (2014)

Sequence to Sequence Learning with Neural Networks

Produces a fixed length representation of input

- "sentence embedding" or "thought vector"



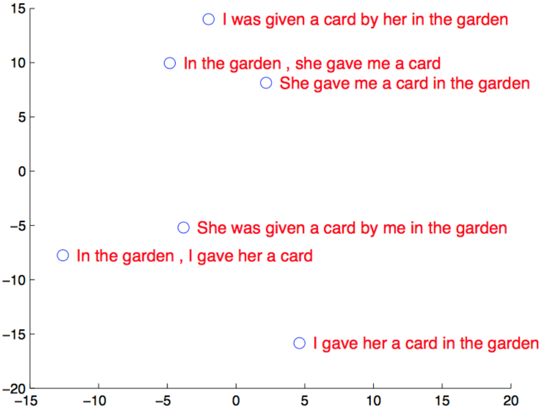
Sequence-to-sequence learning

Sutskever, Vinyals & Le (2014)

Sequence to Sequence Learning with Neural Networks

Produces a fixed length representation of input

- "sentence embedding" or "thought vector"



Sequence-to-sequence learning

Sutskever, Vinyals & Le (2014)

Sequence to Sequence Learning with Neural Networks

LSTM units do not solve vanishing gradients

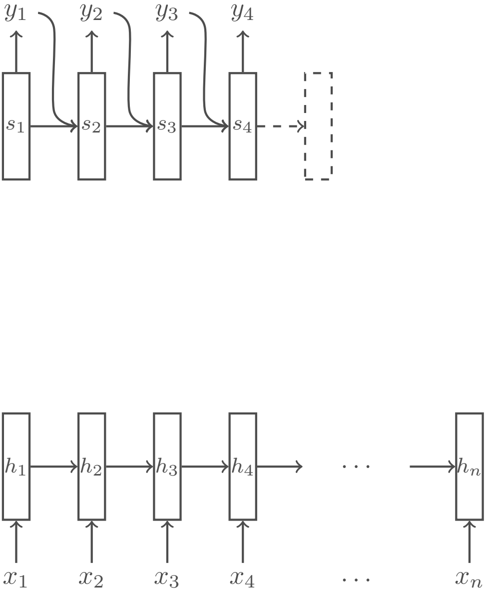
- Poor performance on long sentences
- Need to reverse the input

Method	test BLEU score (ntst14)
Bahdanau et al. [2]	28.45
Baseline System [29]	33.30
Single forward LSTM, beam size 12	26.17
Single reversed LSTM, beam size 12	30.59
[illegible]	[illegible]

Attention-based translation

Bahdanau et al (2015)

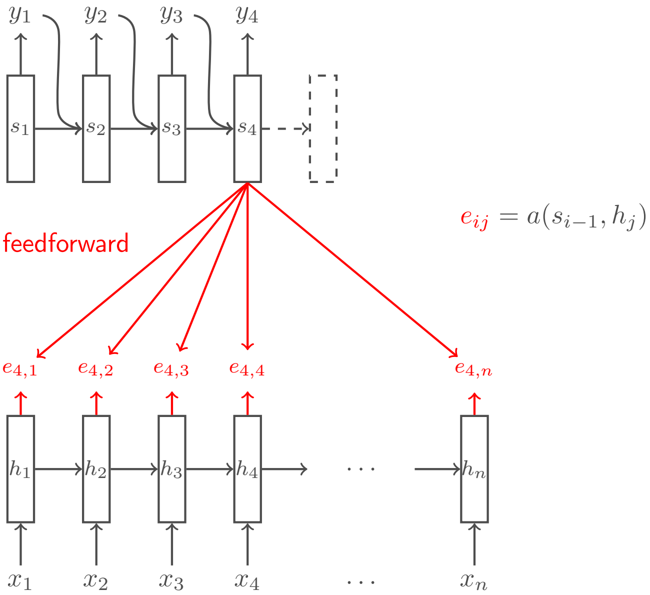
Neural Machine Translation by Jointly Learning to Align and Translate



Attention-based translation

Bahdanau et al (2015)

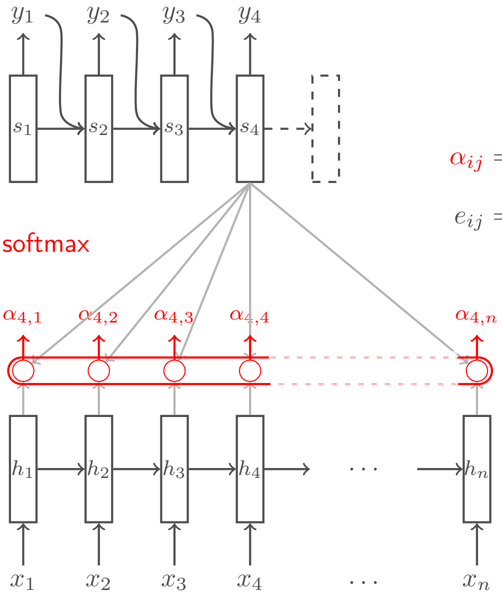
Neural Machine Translation by Jointly Learning to Align and Translate



Attention-based translation

Bahdanau et al (2015)

Neural Machine Translation by Jointly Learning to Align and Translate



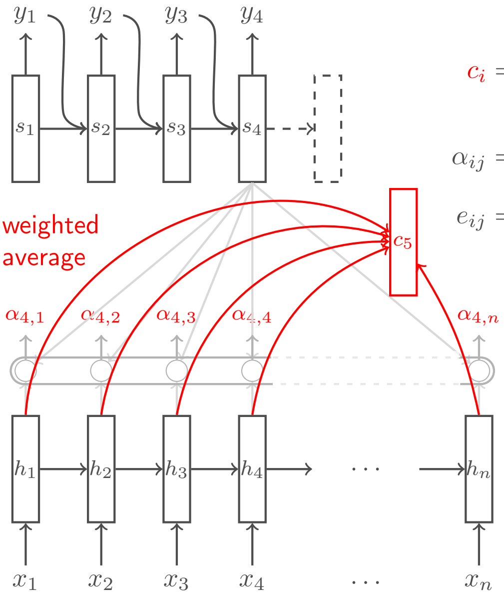
$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_k \exp(e_{ik})}$$

$$e_{ij} = a(s_{i-1}, h_j)$$

Attention-based translation

Bahdanau et al (2015)

Neural Machine Translation by Jointly Learning to Align and Translate



$$c_i = \sum_j \alpha_{ij} h_j$$

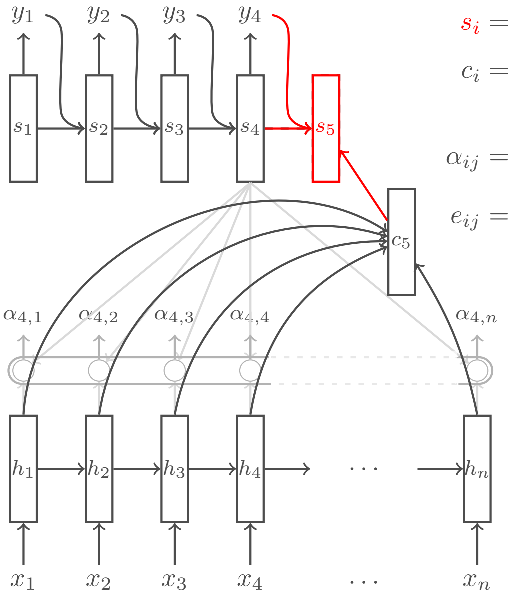
$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_k \exp(e_{ik})}$$

$$e_{ij} = a(s_{i-1}, h_j)$$

Attention-based translation

Bahdanau et al (2015)

Neural Machine Translation by Jointly Learning to Align and Translate



$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

$$c_i = \sum_j \alpha_{ij} h_j$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_k \exp(e_{ik})}$$

$$e_{ij} = a(s_{i-1}, h_j)$$

Attention-based translation

Bahdanau et al (2015)

Neural Machine Translation by Jointly Learning to Align and Translate

- Bidirectional encoder, GRU activations
- Softmax for y_i depends on y_{i-1} and an additional hidden layer

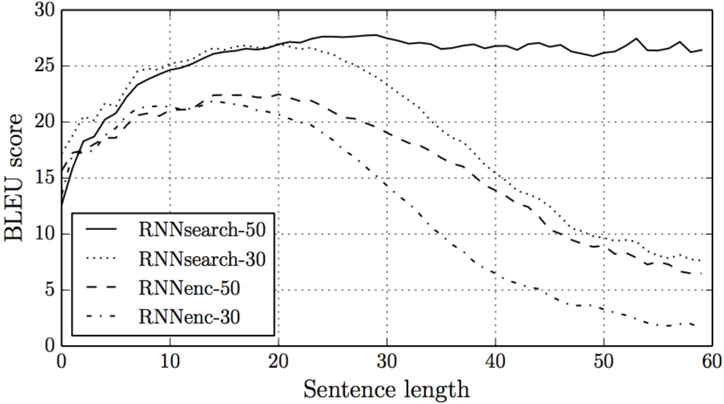
+ Backprop directly to attended regions, avoiding vanishing gradients
+ Can visualize attention weights α_{ij} to interpret prediction

– Inference is $\mathcal{O}(mn)$ instead of $\mathcal{O}(m)$ for seq-to-seq

Attention-based translation

Bahdanau et al (2015)

Neural Machine Translation by Jointly Learning to Align and Translate

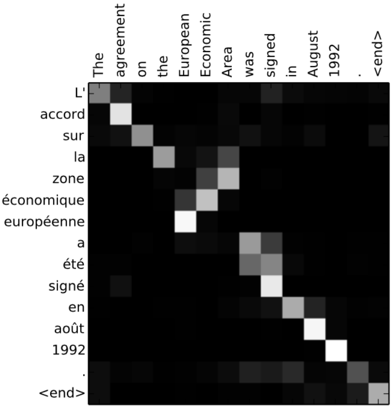


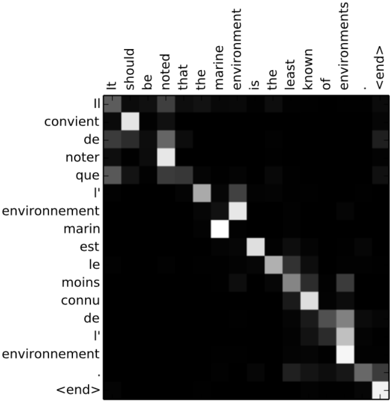
Improved results on long sentences

Attention-based translation

Bahdanau et al (2015)

Neural Machine Translation by Jointly Learning to Align and Translate



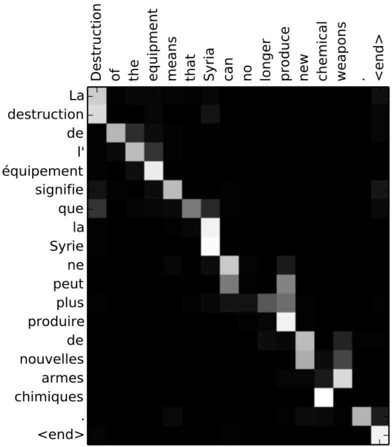


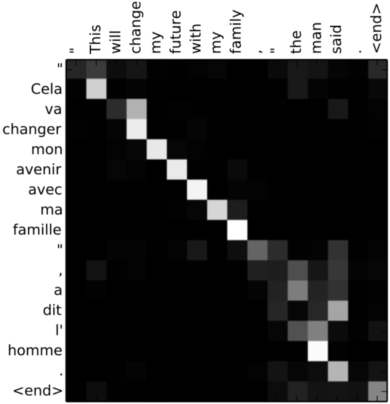
Sensible induced alignments

Attention-based translation

Bahdanau et al (2015)

Neural Machine Translation by Jointly Learning to Align and Translate





Sensible induced alignments

Natural language inference

Given a premise, e.g.,

> *The purchase of Houston-based LexCorp by BMI for $2Bn prompted widespread sell-offs by traders as they sought to minimize exposure. LexCorp had been an employee-owned concern since 2008.*

and a hypothesis, e.g.,

> *BMI acquired an American company.* (1)

predict whether the premise

- entails the hypothesis
- contradicts the hypothesis
- or remains neutral

Natural language inference

Given a premise, e.g.,

The purchase of Houston-based LexCorp by BMI for $2Bn prompted widespread sell-offs by traders as they sought to minimize exposure. LexCorp had been an employee-owned concern since 2008.

and a hypothesis, e.g.,

BMI bought employee-owned LexCorp for $3.4Bn. (2)

predict whether the premise

- entails the hypothesis
- contradicts the hypothesis
- or remains neutral

Natural language inference

Given a premise, e.g.,

The purchase of Houston-based LexCorp by BMI for $2Bn prompted widespread sell-offs by traders as they sought to minimize exposure. LexCorp had been an employee-owned concern since 2008.

and a hypothesis, e.g.,

BMI is an employee-owned concern. (3)

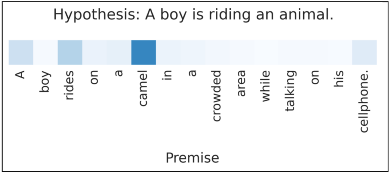
predict whether the premise

- entails the hypothesis
- contradicts the hypothesis
- or remains neutral

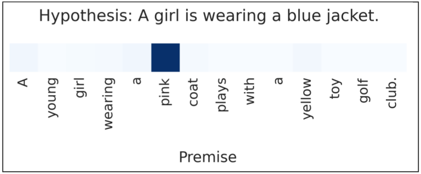
Natural language inference

Rocktäschel et al (2016)

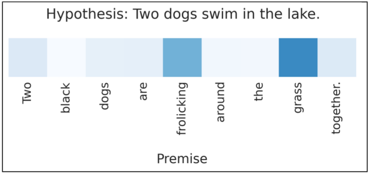
Reasoning about Entailment with Neural Attention



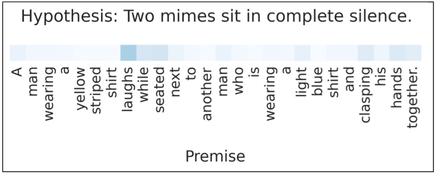
(a)



(b)



(c)



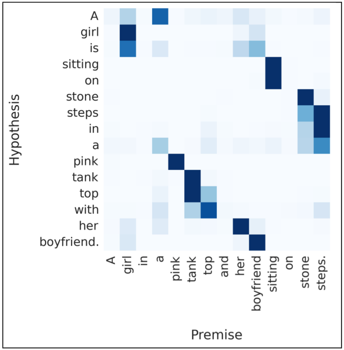
(d)

Attention conditioned on h_T

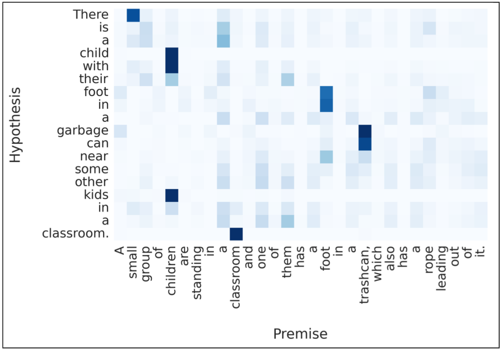
Natural language inference

Rocktäschel et al (2016)

Reasoning about Entailment with Neural Attention



(a)



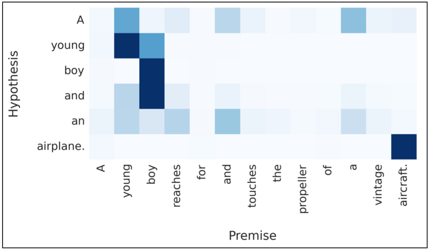
(b)

Attention conditioned on $h_1, \ldots, h_T$: Synonymy, importance

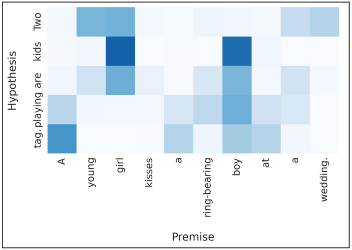
Natural language inference

Rocktäschel et al (2016)

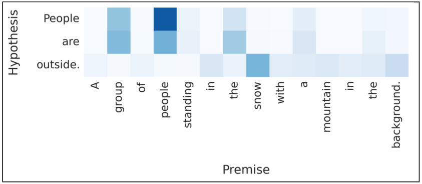
Reasoning about Entailment with Neural Attention



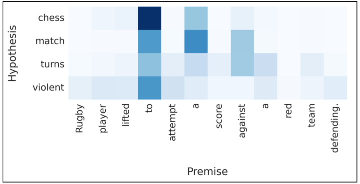
(c)



(d)



(e)



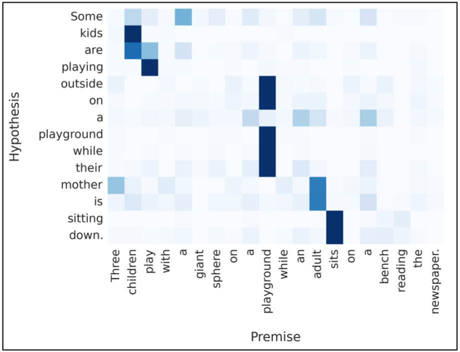
(f)

Attention conditioned on $h_1, \ldots, h_T$: Relatedness

Natural language inference

Rocktäschel et al (2016)

Reasoning about Entailment with Neural Attention



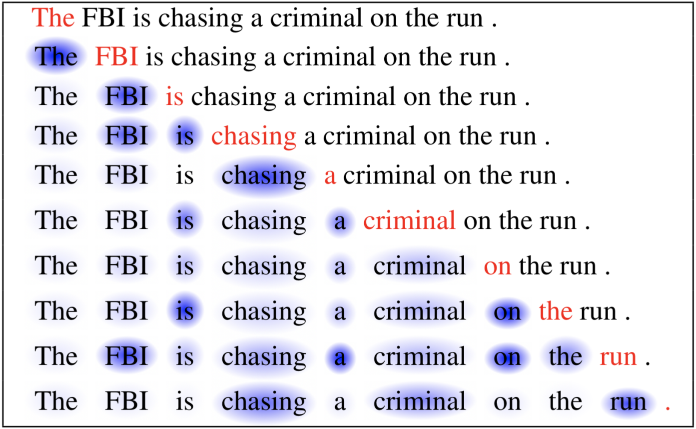
(g)

Attention conditioned on $h_1, \ldots, h_T$: Many:one

Self-attention

Cheng et al (2016)

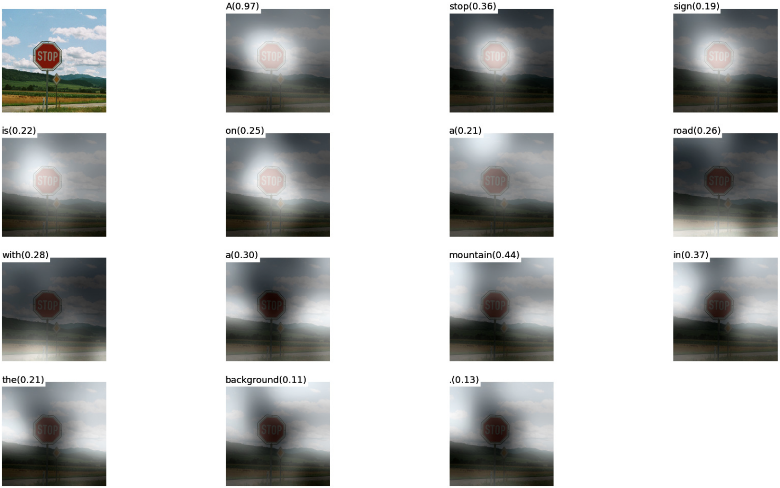
Long Short-Term Memory-Networks for Machine Reading



Attention over images

Xu et al (2015)

Show, Attend & Tell: Neural Image Caption Generation with Visual Attention



(b) A stop sign is on a road with a mountain in the background.

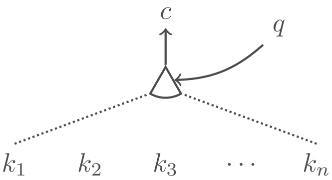
Attention over videos

Yao et al (2015)

Describing Videos by Exploiting Temporal Structure

Attention variants

$c = \textsc{Attention}(\text{query } q, \text{ keys } k_1 \ldots k_n)$

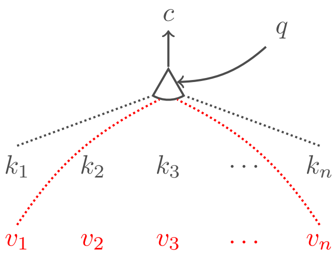


$$\alpha_i = \text{softmax}(\text{score}(q, k_i))$$
$$c = \sum_i \alpha_i, k_i$$

Attention variants

$c = \textsc{Attention}(\text{query } q, \text{ keys } k_1 \ldots k_n, \text{ values } v_1 \ldots v_n)$

e.g., memory networks (Weston et al, 2015; Sukhbataar et al, 2015)



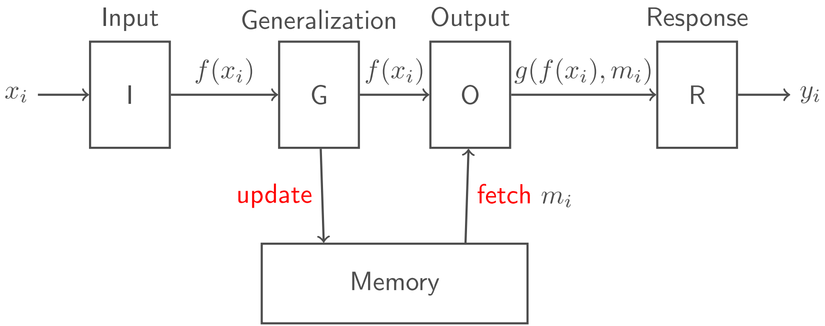
$$\alpha_i = \text{softmax}(\text{score}(q, k_i))$$

$$c = \sum_i \alpha_i, v_i$$

Attention variants

Weston et al (2015)

Memory Networks



MemN2N (Sukhbataar et al, 2015)

+ Soft attention over memories
+ Multiple memory lookups (hops)
+ End-to-end training

Attention scoring functions

- Additive (Bahdanau et al, 2015)

$$\text{score}(q, k) = u^\top \text{tanh}(W[q; k])$$

- Multiplicative (Luong et al, 2015)

$$\text{score}(q, k) = q^\top W k$$

- Scaled dot-product (Vaswani et al, 2017)

$$\text{score}(q, k) = \frac{q^\top k}{\sqrt{d_k}}$$

Attention variants

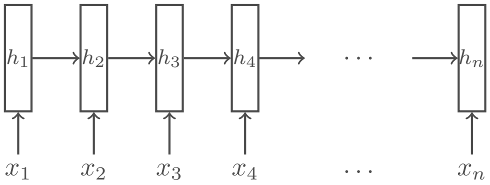
- Stochastic hard attention (Xu et al, 2015)
- Local attention (Luong et al, 2015)
- Monotonic attention (Yu et al, 2016; Raffel et al, 2017)
- Self attention (Cheng et al, 2016; Vaswani et al, 2017)
- Convolutional attention (Allamanis et al, 2016)
- Structured attention (Kim et al, 2017)
- Multi-headed attention (Vaswani et al, 2017)

Transformer

Vaswani et al (2017)

Attention is All You Need

RNN encoder

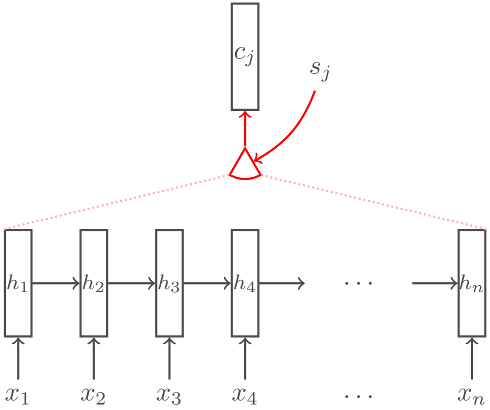


Transformer

Vaswani et al (2017)

Attention is All You Need

RNN encoder with attention

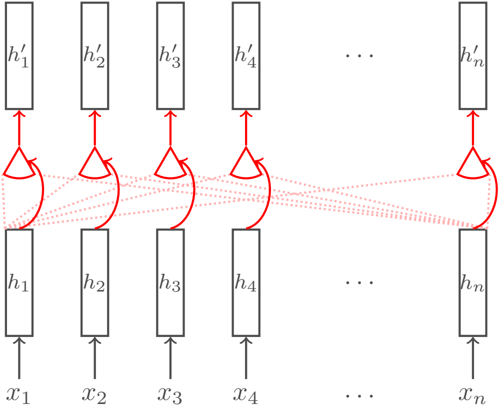


Transformer

Vaswani et al (2017)

Attention is All You Need

Deep encoder with self-attention

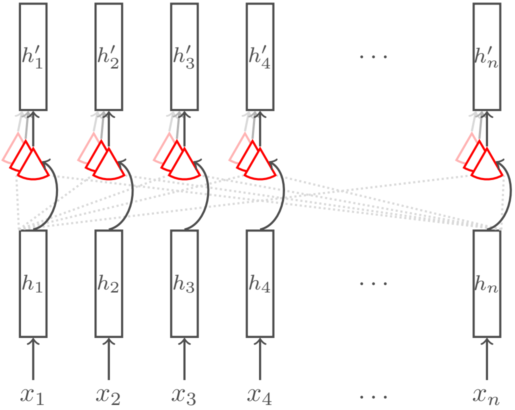


Transformer

Vaswani et al (2017)

Attention is All You Need

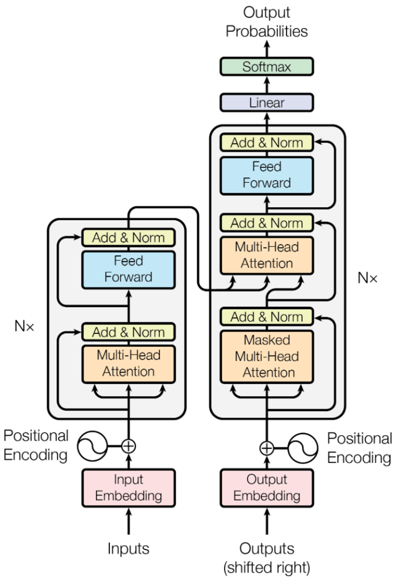
Deep encoder with multi-head self-attention



Transformer

Vaswani et al (2017)

Attention is All You Need



Transformer

Vaswani et al (2017)

Attention is All You Need

- Self-attention at every layer instead of recurrence
  - – Quadratic increase in computation for each hidden state
  - + Inference can be parallelized

- No sensitivity to input position
  - – Positional embeddings required
  - + Can apply to sets

- Deep architecture (6 layers) with multi-head attention
  - + Higher layers appear to learn linguistic structure

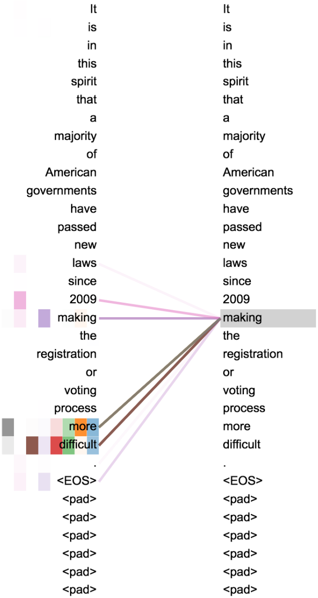
- Scaled dot-product attention with masking
  - + Avoids bias in simple dot-product attention
  - + Fewer parameters needed for rich model

- Improved runtime and performance on translation, parsing, etc

Transformer

Vaswani et al (2017)

Attention is All You Need



Transformer

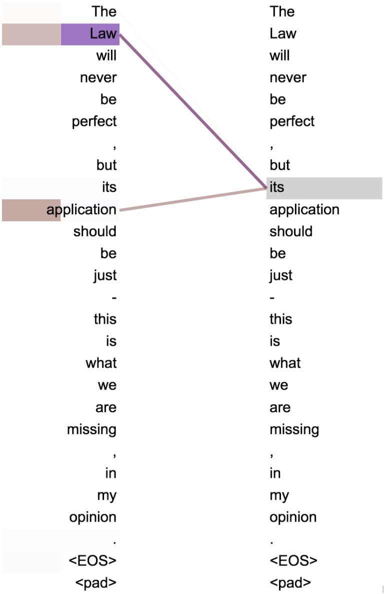
Vaswani et al (2017)

Attention is All You Need

Transformer

Vaswani et al (2017)

Attention is All You Need



Transformer

Vaswani et al (2017)

Attention is All You Need

Transformer

Vaswani et al (2017)

Attention is All You Need

Large vocabularies

Sequence-to-sequence models can typically scale to 30K-50K words

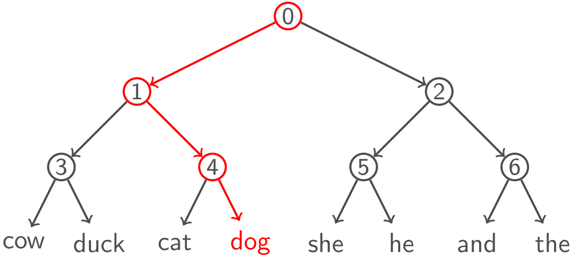
But real-world applications need at least 500K-1M words

Large vocabularies

Alternative 1: Hierarchical softmax

- Predict path in binary tree representation of output layer
- Reduces to $\log_2(V)$ binary decisions

$$p(w_t = \text{“dog”} | \cdots) = (1 - \sigma(U_0 h_t)) \times \sigma(U_1 h_t) \times \sigma(U_4 h_t)$$



Large vocabularies

Jean et al (2015)

On Using Very Large Target Vocabulary for Neural Machine Translation

Alternative 2: Importance sampling

- Expensive to compute the softmax normalization term over V

$$p(y_i = w_j | y_{<i}, x) = \frac{\exp\left(W_j^\top f(s_i, y_{i-1}, c_i)\right)}{\sum_{k=1}^{|V|} \exp\left(W_k^\top f(s_i, y_{i-1}, c_i)\right)}$$

- Use a small subset of the target vocabulary for each update
- Approximate expectation over gradient of loss with fewer samples
- Partition the training corpus and maintain local vocabularies in each partition to use GPUs efficiently

Large vocabularies

Sennrich et al (2016)

Neural Machine Translation of Rare Words with Subword Units

Alternative 3: Subword units

- Reduce vocabulary by replacing infrequent words with sub-words

Jet makers feud over seat width with big orders at stake

$\Downarrow$

_J et _makers _fe ud _over _seat _width _with _big _orders _at _stake

- Code for byte-pair encoding (BPE): `https://github.com/rsennrich/subword-nmt`

Copying

Gu et al (2016)

Incorporating Copying Mechanism in Sequence-to-Sequence Learning

In monolingual tasks, copy rare words directly from the input

- Generation via standard attention-based decoder

$$\psi_g(y_i = w_j) = W_j^\top f(s_i, y_{i-1}, c_i) \qquad w_j \in V$$

- Copying via a non-linear projection of input hidden states

$$\psi_c(y_i = x_j) = \mathsf{tanh}(h_j^\top U) f(s_i, y_{i-1}, c_i) \qquad x_j \in X$$

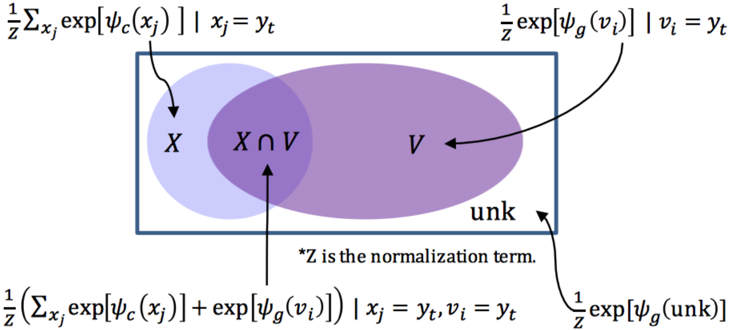
- Both modes compete via the softmax

$$p(y_i = w_j | y_{<i}, x) = \frac{1}{Z}\left(\mathsf{exp}\left(\psi_g(w_j)\right) + \sum_{k:x_k = w_j} \mathsf{exp}\left(\psi_c(x_k)\right)\right)$$

Copying

Gu et al (2016)

Incorporating Copying Mechanism in Sequence-to-Sequence Learning

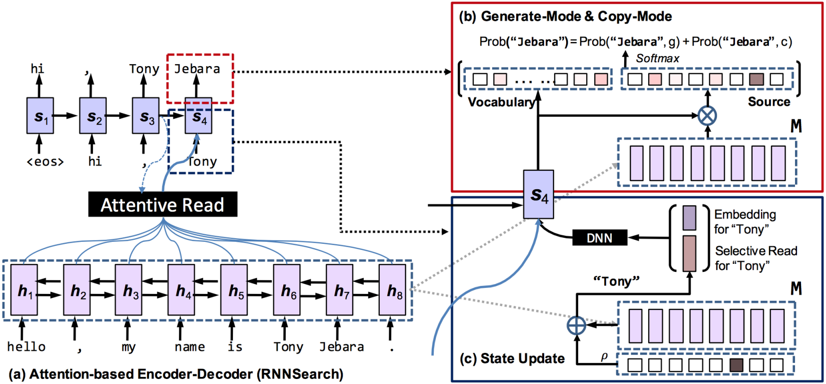


Decoding probability $p(y_t|\cdots)$

Copying

Gu et al (2016)

Incorporating Copying Mechanism in Sequence-to-Sequence Learning

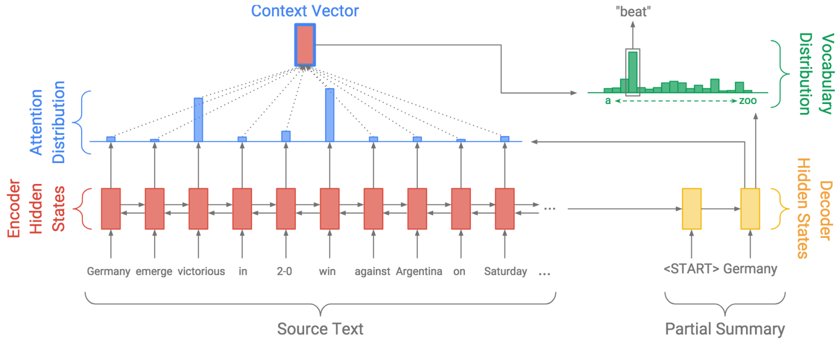


(a) Attention-based Encoder-Decoder (RNNSearch)

Copying

See et al (2017)

Get to the Point: Summarization with Pointer Generator Networks

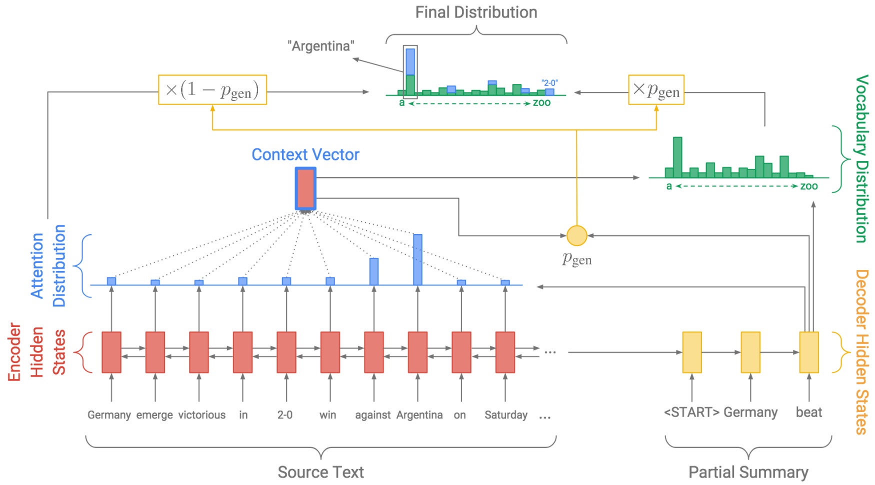


Attention for common words

Copying

See et al (2017)

Get to the Point: Summarization with Pointer Generator Networks



Copying from input for rarer words

Autoencoders

Given input x, learn an encoding z that can be decoded to reconstruct x

For sequence input $x_1, \ldots, x_n$, can use standard MT models

- Is attention viable?

+ Useful for pre-training text classifiers (Dai et al, 2015)

Denoising autoencoders

Hill et al (2016)

Learning Distributed Representations of Sentences from Unlabeled Data

Given noisy input $\tilde{x}$, learn an encoding z that can be decoded to reconstruct x

Noise: drop words or swap two words with some probability

+ Helpful as features for a linear classifier
+ Can learn sentence representations without sentence order

Query	***If he had a weapon, he could maybe take out their last imp, and then beat up Errol and Vanessa.***	***An annoying buzz started to ring in my ears, becoming louder and louder as my vision began to swim.***
CBOW	*Then Rob and I would duke it out, and every once in a while, he would actually beat me.*	*Louder.*
Skip Thought	*If he could ram them from behind, send them saling over the far side of the levee, he had a chance of stopping them.*	*A weighty pressure landed on my lungs and my vision blurred at the edges, threatening my consciousness altogether.*
FastSent	*Isak's close enough to pick off any one of them, maybe all of them, if he had his rifle and a mind to.*	*The noise grew louder, the quaking increased as the sidewalk beneath my feet began to tremble even more.*
SDAE	*He'd even killed some of the most dangerous criminals in the galaxy, but none of those men had gotten to him like Vitktis.*	*I smile because I'm familiar with the knock, pausing to take a deep breath before dashing down the stairs.*
DictRep (FF+embs.)	*Kevin put a gun to the man's head, but even though he cried, he couldn't tell Kevin anything more.*	*Then gradually I began to hear a ringing in my ears.*
Paragraph Vector (DM)	*I take a deep breath and open the doors.*	*They listened as the motorcycle-like roar of an engine got louder and louder then stopped.*

Table 5: Sample nearest neighbour queries selected from a randomly sampled 0.5m sentences of the Toronto Books Corpus.

Variational autoencoders (VAEs)

Kingma & Welling (2014)
Auto-encoding Variational Bayes

Autoencoders often don't generalize well to new data, noisy representations

Approximate the posterior $p(z|x)$ with variational inference

- Encoder: induce $q(z|x)$ with parameters θ
- Decoder: sample z and reconstruct x with parameters ϕ
- Loss:

$$\ell_i = -\mathbb{E}_{z \sim q_\theta(z|x_i)} \log p_\phi(x_i|z) + \mathrm{KL}\left(q_\theta(z|x_i)||p(z)\right)$$

Estimate gradients using *reparameterization trick* for Gaussians

$$z \sim \mathcal{N}(\mu, \sigma^2) = \mu + \sigma \times [z' \sim \mathcal{N}(0,1)]$$

Variational autoencoders (VAEs)

Bowman et al (2016)

Generating Sentences from a Continuous Space

+ Better at word imputation than RNNs
+ Can interpolate smoothly between representations in the latent space

" i want to talk to you . "
"i want to be with you . "
"i do n't want to be with you . "
i do n't want to be with you .
she did n't want to be with him .

he was silent for a long moment .
he was silent for a moment .
it was quiet for a moment .
it was dark and cold .
there was a pause .
it was my turn .

i dont like it , he said .
i waited for what had happened .
it was almost thirty years ago .
it was over thirty years ago .
that was six years ago .
he had died two years ago .
ten , thirty years ago .
" it 's all right here .
" everything is all right here .
" it 's all right here .
it 's all right here .
we are all right here .
come here in five minutes .

this was the only way .
it was the only way .
it was her turn to blink .
it was hard to tell .
it was time to move on .
he had to do it again .
they all looked at each other .
they all turned to look back .
they both turned to face him .
they both turned and walked away .

there is no one else in the world .
there is no one else in sight .
they were the only ones who mattered .
they were the only ones left .
he had to be with me .
she had to be with him .
i had to do this .
i wanted to kill him .
i started to cry .
i turned to him .