Deep Learning Applications in Natural Language Processing

Jindřich Libovický

December 5, 2018

Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics

Outline

Information Search

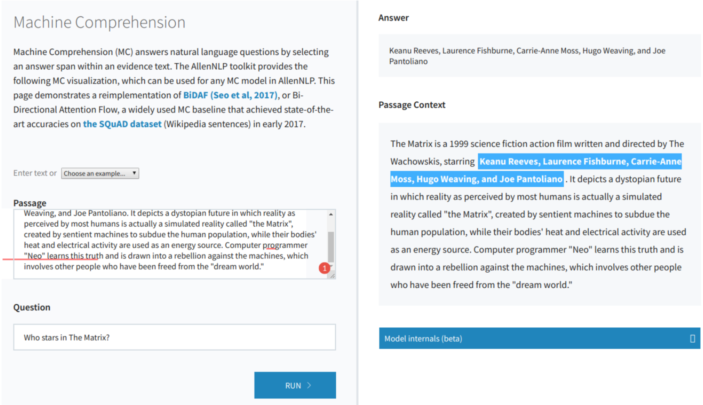
Unsupervised Dictionary Induction

Image Captioning

Information Search

Answer Span Selection

Task: Find an answer for a question given question in a coherent text.



`http://demo.allennlp.org/machine-comprehension`

Standard Dataset: SQuAD

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

- best articles from Wikipedia, of reasonable size (23k paragraphs, 500 articles)
- crowd-sourced more than 100k question-answer pairs
- complex quality testing (which got estimate of single human doing the task)

`https://rajpurkar.github.io/SQuAD-explorer/explore/1.1/dev/`

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. URL `https://aclweb.org/anthology/D16-1264`

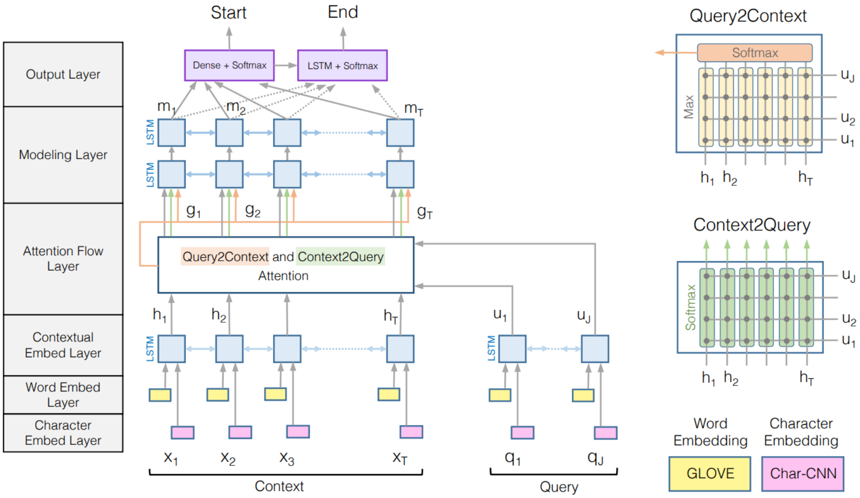
Method Overview

1. Get text and question representation from
    - pre-trained word embeddings
    - character-level CNN

   ...using your favourite architecture.d
2. Compute a similarity between all pairs of words in the text and in the question.
3. Collect all informations we have for each token.
4. Classify where the span is.

Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *CoRR*, abs/1611.01603, 2016. URL `http://arxiv.org/abs/1611.01603`

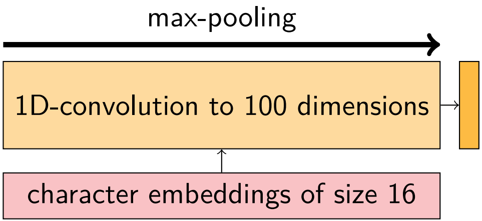
Method Overview: Image



Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *CoRR*, abs/1611.01603, 2016. URL `http://arxiv.org/abs/1611.01603`

Representing Words

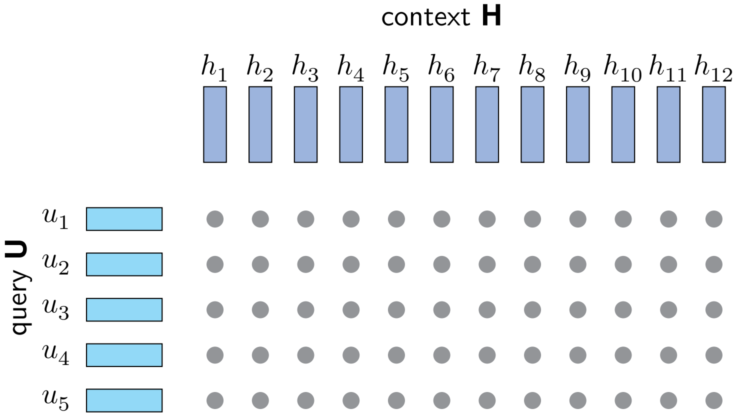
- pre-trained word embeddings
- concatenate with trained character-level representations
- character-level representaions allows searching for out-of-vocabulary structured informations (numbers, addresses)



Contextual Embeddings Layer

- process both question and context with bidirectional LSTM layer
  $\rightarrow$ one state per word
- parameters are shared $\rightarrow$ representaions share the space

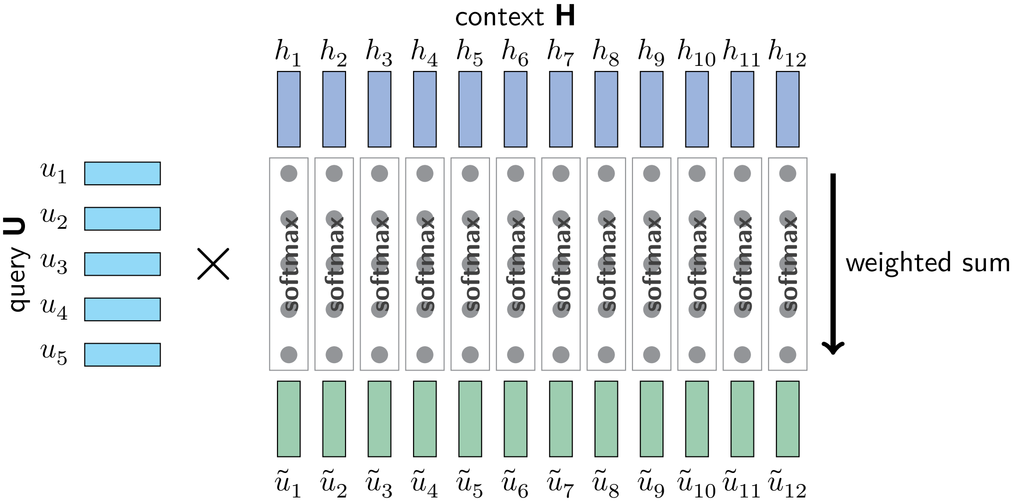
Attention Flow



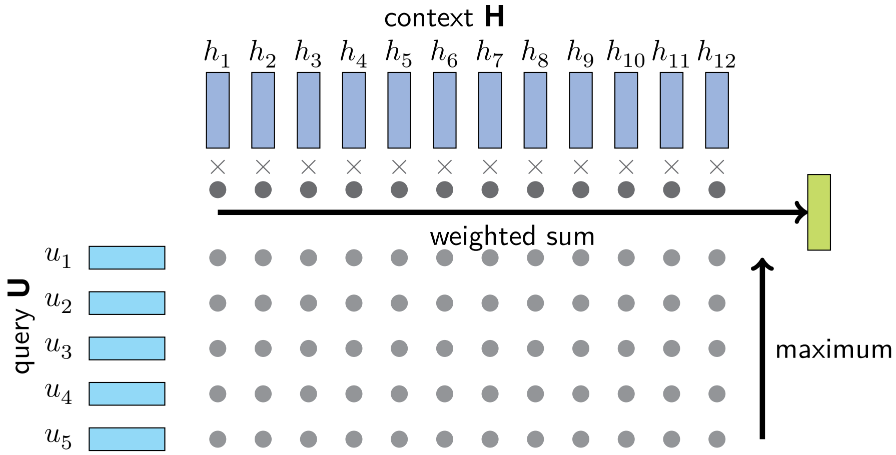
$$\mathbf{S}_{ij} = \mathbf{w}^T \left[h_i, c_j, h_i \odot c_j\right]$$

Captures affinity / similarity between pairs of question and context words.

Context-to-query Attention



Query-to-Context Attention



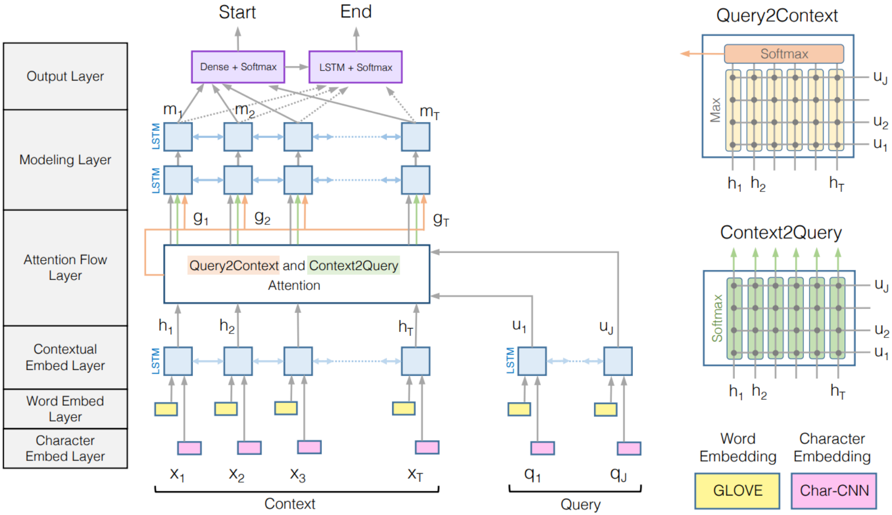
Modeling Layer

- concatenate: LSTM outputs for each context word, context-to-query-vectors
- copy query-to-context vector to each of them
- apply one non-linear layer and bidirectional LSTM

Output Layer

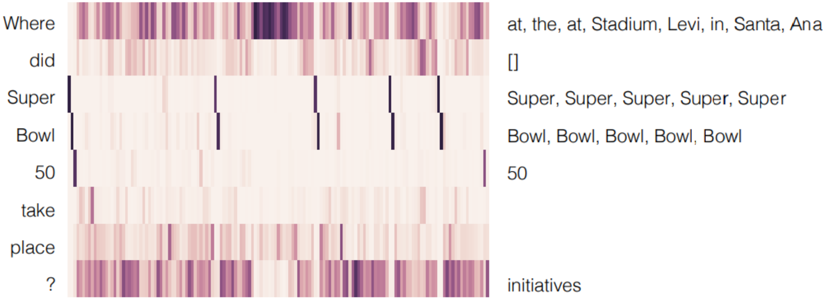
1. Start-token probabilities: project each state to scalar $\rightarrow$ apply softmax over the context
2. End-token probabilities:
   - Compute weighted averate using the start-token probablities $\rightarrow$ single vector
   - Concatenate the vector to each state
   - Project states to scalar, renormalize with softmax
3. At the end select the most probable span

Method Overview: Recap

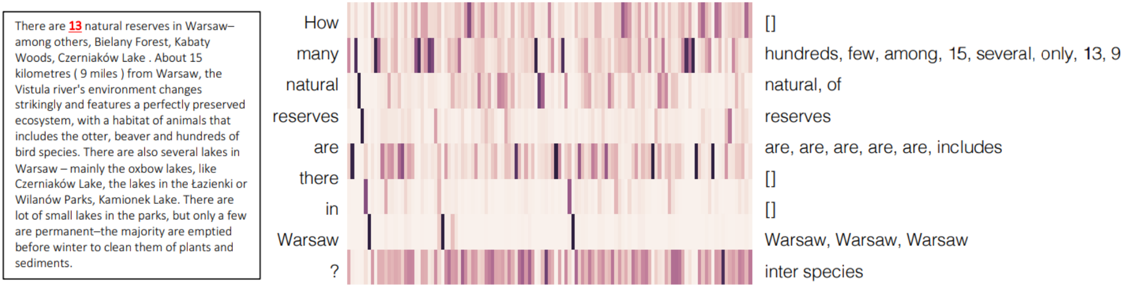


Attention Analysis (1)

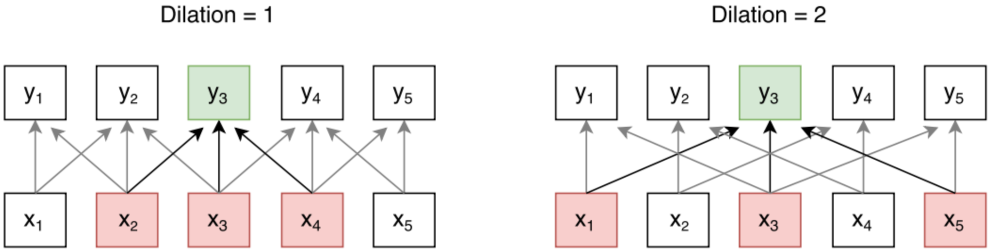
Super Bowl 50 was an American football game to determine the champion of the National Football League (NFL) for the 2015 season. The American Football Conference (AFC) champion Denver Broncos defeated the National Football Conference (NFC) champion Carolina Panthers 24–10 to earn their third Super Bowl title. The game was played on February 7, 2016, **at Levi's Stadium in the San Francisco Bay Area at Santa Clara, California**. As this was the 50th Super Bowl, the league emphasized the "golden anniversary" with various gold-themed initiatives, as well as temporarily suspending the tradition of naming each Super Bowl game with Roman numerals (under which the game would have been known as "Super Bowl L"), so that the logo could prominently feature the Arabic numerals 50.



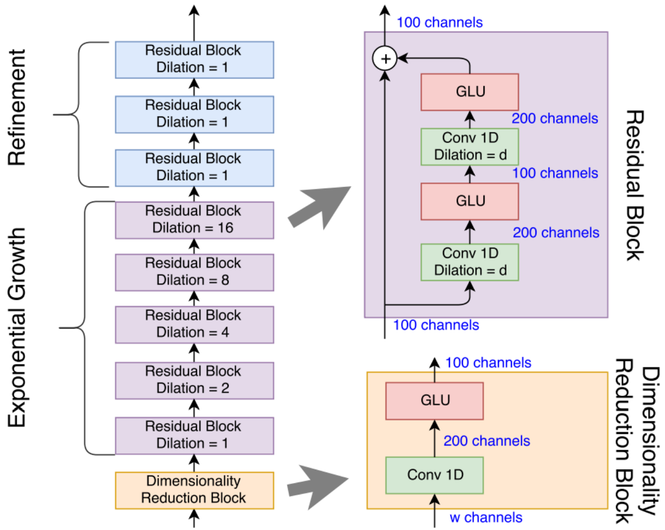
Attention Analysis (2)



Replace LSTMs by dilated convolutions.



Convolutional Blocks



Using Pre-Trained Representations

Just replace the contextual embeddings wiht ELMo or BERT...

SQuAD Leaderboard

method	Exact Match	F1 Score
Human performacne	82.304	91.221
BiDAF with BERT	87.433	93.160
BiDAF with ELMo	81.003	87.432
BiDAF trained from scratch	73.744	81.525

Unsupervised Dictionary Induction

Unsupervised Bilingual Dictionary

Task: Get a translation dictionary between two languages using monolignual data only.

- makes NLP accessible for low-resourced languages
- basic for unsupervised machine translation
- hot research topic (at least 10 research papers on this topic this year)

We will approach: Mikel Artetxe, Gorka Labaka, and Eneko Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia, July 2018. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P18-1073`

How it is done

1. Train word embeddings on large monolignual corpora.
2. Find a mapping between the two languages.

So far looks simple...

Dictionary and Common Projection

X, Z embedding matrices for 2 languages.
Dictionary matrix $D_{ij} = 1$ if X_i is translation of Z_j.

Supervised projection between embeddings

Given existing dictionary D (small seed dictionary):

$$\underset{W_Z, W_X}{\text{argmax}} \sum_i \sum_j D_{ij} \cdot \text{similarity}\left(X_{i:}W_X, Z_{j:}W_Z\right) \left(X_{:i}W_X \left(Z_{j:}W_Z\right)^T\right)$$

...but we need to find all D, W_X, and W_Z.

A Tiny Observation

$$XX^T$$

Question: How would you interpret this matrix?

It is a table of similarities between pairs of words.

If the Vocabularies were Isometric...

- $M_X = XX^T$ and $M_Z = ZZ^T$ would only have permuted rows and columns
- if we sorted values in each row of M_X and M_Z, corresponding words would have the same vectors

Let's assume, it is true (at least approximately)

$$D_{i,:} \leftarrow \mathbf{1}\left[\operatorname*{argmin}_j (M_X)_{i,:}(M_Z)^T_{j,:}\right]$$

Assign nearest neighbor from the other language.

……in practice tragically bad but at least good initialization.

Self-Learning

Iterate until convergence:

1. Optimize W_Z and W_X, w.r.t to current dictionary

$$\underset{W_Z, W_X}{\text{argmax}} \sum_i \sum_j D_{ij} \cdot \left(X_{:i} W_X \left(Z_{j:} W_Z \right)^T \right)$$

2. Update dictionary matrix D

$$D_{ij} = \begin{cases} 1, & \text{if } i \text{ is nearest neighbor of } j \text{ or wise versa} \\ 0, & \text{otherwise} \end{cases}$$

Accuracy on Large Dictionary

Supervision	Method	EN-IT	EN-DE	EN-FI	EN-ES
5k dict.	Mikolov et al. (2013)	34.93[†]	35.00[†]	25.91[†]	27.73[†]
	Faruqui and Dyer (2014)	38.40[*]	37.13[*]	27.60[*]	26.80[*]
	Shigeto et al. (2015)	41.53[†]	43.07[†]	31.04[†]	33.73[†]
	Dinu et al. (2015)	37.7	38.93[*]	29.14[*]	30.40[*]
	Lazaridou et al. (2015)	40.2	-	-	-
	Xing et al. (2015)	36.87[†]	41.27[†]	28.23[†]	31.20[†]
	Zhang et al. (2016)	36.73[†]	40.80[†]	28.16[†]	31.07[†]
	Artetxe et al. (2016)	39.27	41.87[*]	30.62[*]	31.40[*]
	Artetxe et al. (2017)	39.67	40.87	28.72	-
	Smith et al. (2017)	43.1	43.33[†]	29.42[†]	35.13[†]
	Artetxe et al. (2018a)	45.27	44.13	**32.94**	36.60
25 dict.	Artetxe et al. (2017)	37.27	39.60	28.16	-
Init. heurist.	Smith et al. (2017), cognates	39.9	-	-	-
	Artetxe et al. (2017), num.	39.40	40.27	26.47	-
None	Zhang et al. (2017a), $\lambda = 1$	0.00[*]	0.00[*]	0.00[*]	0.00[*]
	Zhang et al. (2017a), $\lambda = 10$	0.00[*]	0.00[*]	0.01[*]	0.01[*]
	Conneau et al. (2018), code[‡]	45.15[*]	46.83[*]	0.38[*]	35.38[*]
	Conneau et al. (2018), paper[‡]	45.1	0.01[*]	0.01[*]	35.44[*]
	Proposed method	**48.13**	**48.19**	32.63	**37.33**

Try it yourself!

- Pre-train monolingual word embeddings using FastText / Word2Vec
- Install VecMap
  https://github.com/artetxem/vecmap

```
python3 map_embeddings.py --unsupervised SRC.EMB TRG.EMB SRC_MAPPED.EMB
    TRG_MAPPED.EMB
```

Image Captioning

Image Captioning

Task: Generate a caption in natural language given an image.

Example:

A group of people wearing snowshoes, and dressed for winter hiking, is standing in front of a building that looks like it's made of blocks of ice.

The people are quietly listening while the story of the ice cabin was explained to them.

A group of people standing in front of an igloo.

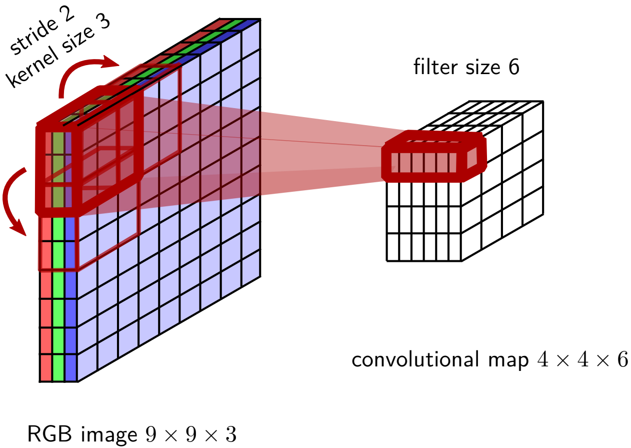
Several students waiting outside an igloo.

Deep Learning Solution

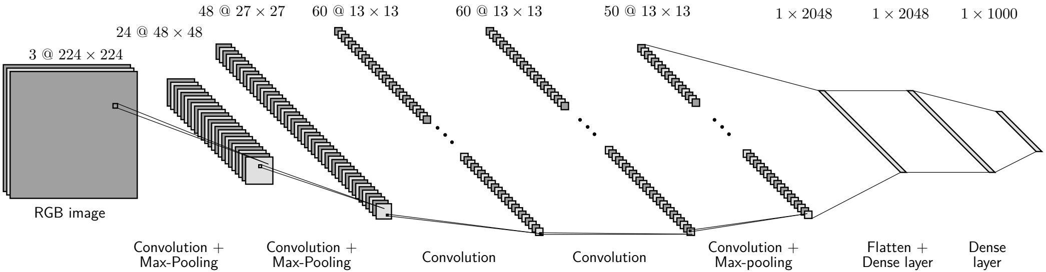
1. Obtain pre-trained image representation.
2. Use autoregressive decoder to generate the caption using the image representation.

2D Convolution over an Image

Basic method in deep learning for computer vision.

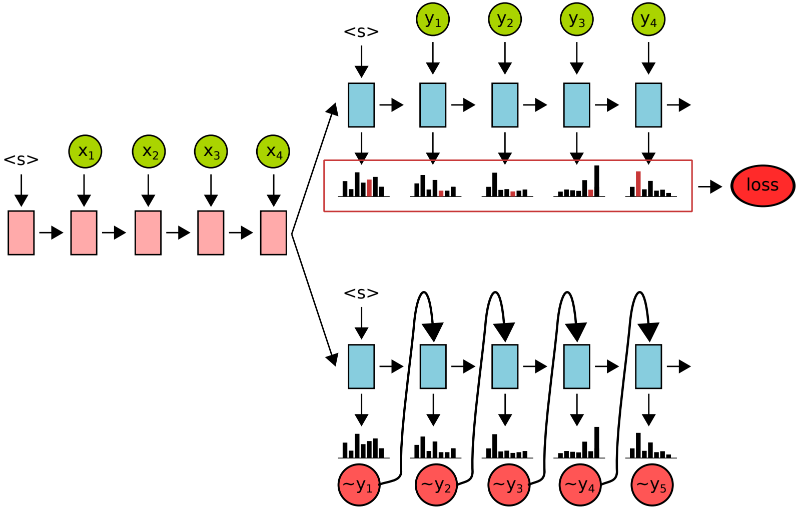


Convolutional Network for Image Classification



- Trained for 1k classes classification, millions of training examples
- Architecture: convolutions, max-pooling, residual connections, batch normalization, 50–150 layers

Reminder: Autoregressive Decoder



Attention Model in Equations (1)

Inputs:

decoder state s_i

encoder states $h_j = \left[\overrightarrow{h_j}; \overleftarrow{h_j}\right] \quad \forall i = 1 \ldots T_x$

Attention energies:

$$e_{ij} = v_a^\top \operatorname{tanh}\left(W_a s_{i-1} + U_a h_j + b_a\right)$$

Attention distribution:

$$\alpha_{ij} = \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{T_x} \exp\left(e_{ik}\right)}$$

Context vector:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

Attention Model in Equations (2)

Output projection:

$$t_i = \mathsf{MLP}\left(U_o s_{i-1} + V_o E y_{i-1} + C_o c_i + b_o\right)$$

...attention is mixed with the hidden state

Output distribution:

$$p\left(y_i = k | s_i, y_{i-1}, c_i\right) \propto \mathsf{exp}\left(W_o t_i\right)_k + b_k$$

Example Outputs: Correct

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

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

A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

Example Outputs: Incorrect


A large white bird standing in a forest.


A woman holding a clock in her hand.

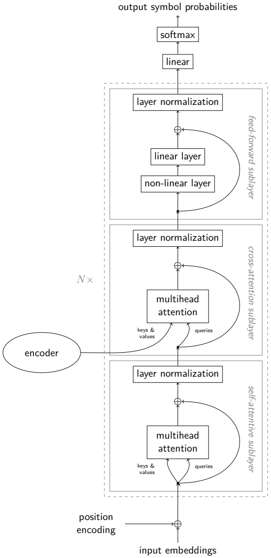

A man wearing a hat and a hat on a skateboard.

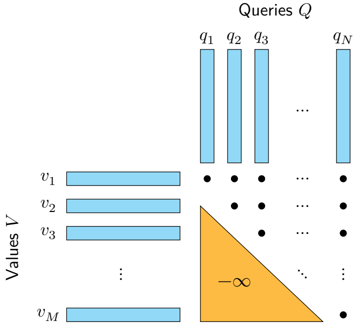

A person is standing on a beach with a surfboard.


A woman is sitting at a table with a large pizza.


A man is talking on his cell phone while another man watches.

Employing Transformer Decoder





Quantitative Results

model	BLEU score
RNN + attention (original)	24.3
RNN + attention (with better image representation)	32.6
Transformer (with better image representation)	33.3