Machine Reading & Reasoning

with Differentiable Interpreters

Sebastian Riedel *UCL, Facebook AI Research // UK*
Pasquale Minervini *UCL // UK*

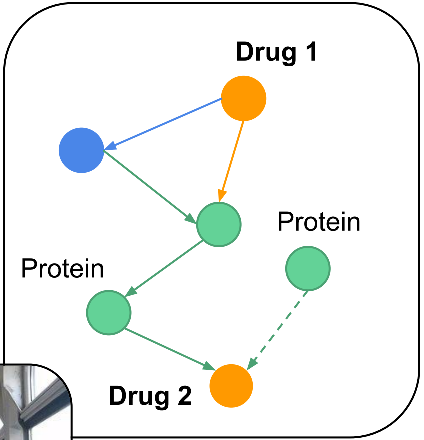
Machine Reading and Reasoning (old-school)

Map Text to Relational Representation, then do Relational Reasoning

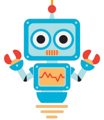
Part 1

Whole-cell patch-clamp recordings were made from CA1 pyramidal neurons of the rat hippocampus to study the modulation of gonadotropin-releasing hormone (GnRH) on synaptic transmission mediated by ionotropic glutamate receptors. Leuprolide (10(-9)-10(-7) M), a specific GnRH analog, concentration-dependently elicited a long-lasting potentiation of excitatory postsynaptic currents (EPSCs) mediated by ionotropic glutamate receptors.

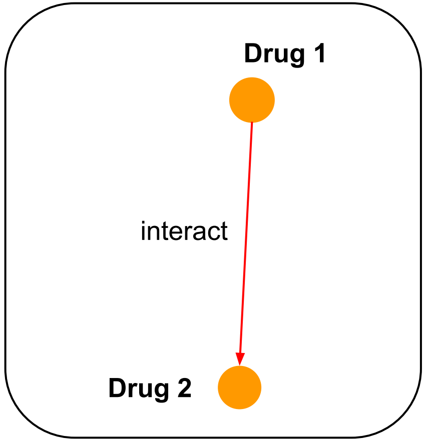
[Text]



[Meaning]



reasons



[New Knowledge]

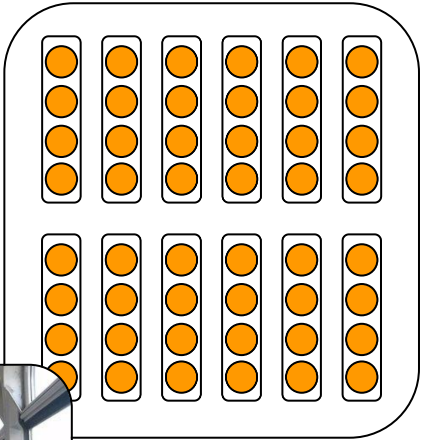
Machine Reading and Reasoning (new-school)

Map Text to Continuous Representation, then what?

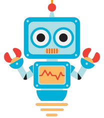
Part 1

Whole-cell patch-clamp recordings were made from CA1 pyramidal neurons of the rat hippocampus to study the modulation of gonadotropin-releasing hormone (GnRH) on synaptic transmission mediated by ionotropic glutamate receptors. Leuprolide (10(-9)-10(-7) M), a specific GnRH analog, concentration-dependently elicited a long-lasting potentiation of excitatory postsynaptic currents (EPSCs) mediated by ionotropic glutamate receptors.

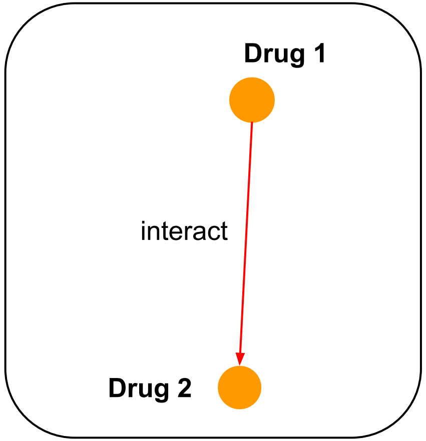
[Text]



[Meaning]



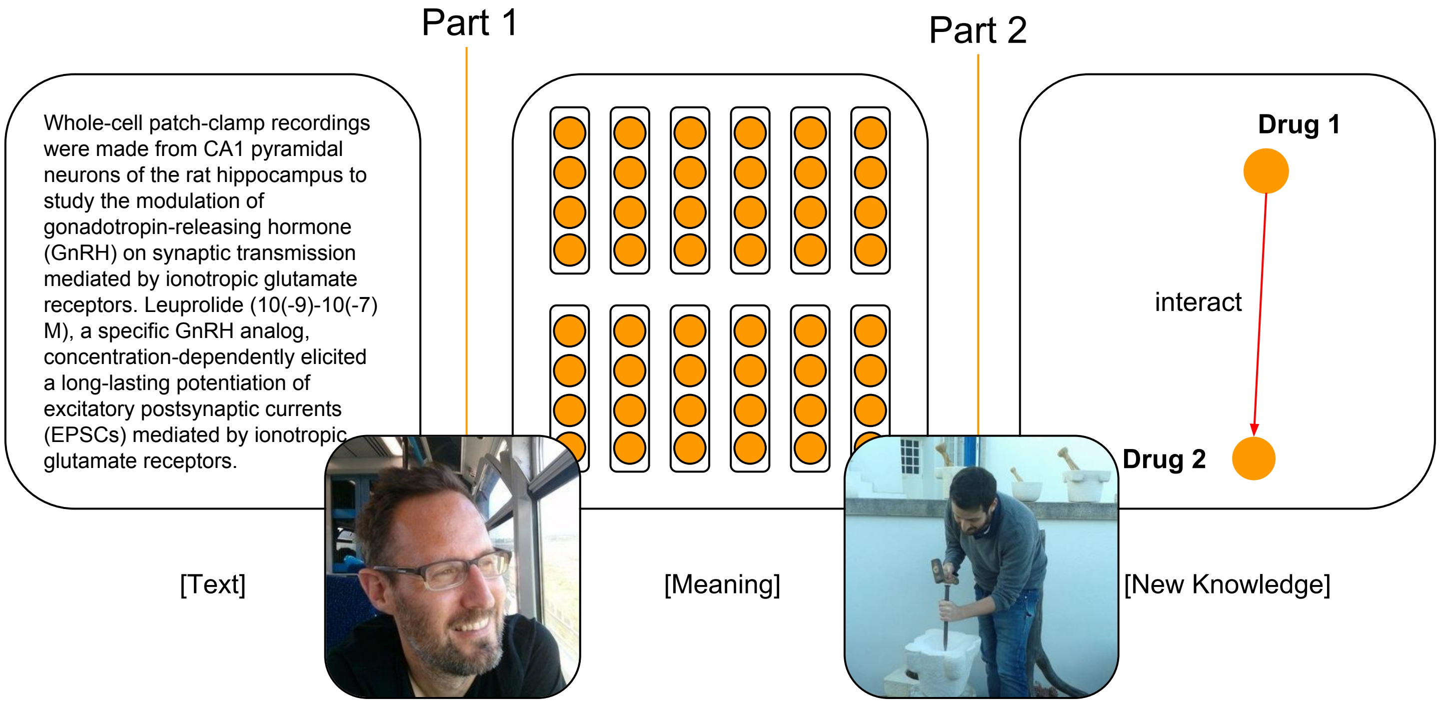
?



[New Knowledge]

Machine Reading and Reasoning (new-school)

Map Text to Continuous Representation, then what?



Overview

- Part 1: Machine Reading
  - Explicit Relational Representations of Meaning
  - End-to-End Machine Reading and Question Answering
  - Open Problems
- Part 2: Differentiable Interpreters (for Machine Reasoning)
  - Learning with External Memory
  - Differentiable Abstract Machines
  - Neural Theorem Proving
  - Open Problems

ROBOTS CAN NOW READ BETTER THAN HUMANS, PUTTING MILLIONS OF JOBS AT RISK

BY ANTHONY CUTHBERTSON ON 1/15/18 AT 8:00 AM

ROBOTS CAN NOW PATTERN MATCH ON A BENCHMARK DATASET BETTER THAN HUMANS

BY ANTHONY CUTHBERTSON ON 1/15/18 AT 8:00 AM

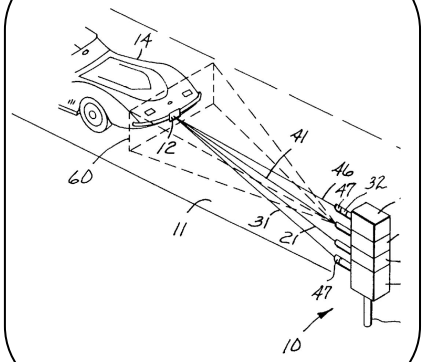
THERE HAS BEEN A LOT OF PROGRESS AND MACHINE READING RESEARCH ACTIVITY HAS SKYROCKETED

BY ANTHONY CUTHBERTSON ON 1/15/18 AT 8:00 AM

What's *Machine Reading*?

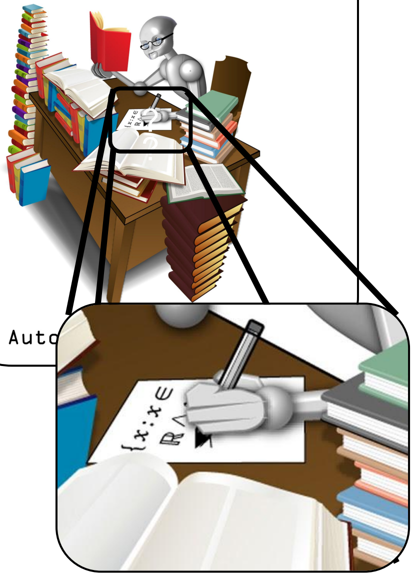
Don't Anthropomorphize Computers, They Hate it When You do That.

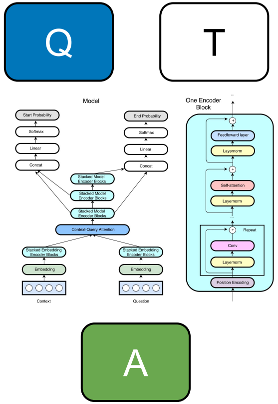
Something else entirely!	Text to Symbolic Representations	End-to-End Question Answering



before 2006

Machine Reading





since 2014

Hermann et al., 2014

What’s this Part of the Tutorial about?

Text to Symbolic Representations

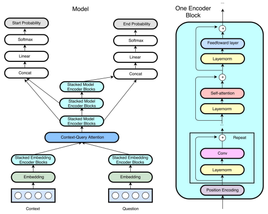
Machine Reading

Auto-Text to Knowledge

End-to-End Question Answering

Q

T



A

aka Information Extraction, Semantics, Question Answering

Machine Reading: Content

- Context
  - What is MR?
  - Why should we care?
- Methods
  - Paradigms
  - Models
- Challenges
  - Why is it hard?
  - strengths & weaknesses
- Tools
  - Datasets
  - (Software)

Text to Symbolic Representations

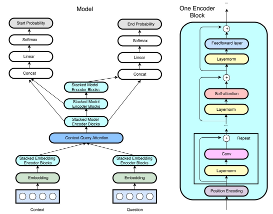
Machine Reading

Auto-Text to Knowledge

End-to-End Question Answering

Q

T



A

aka Information Extraction, Semantics, Question Answering

Machine Reading: Parts

- Context
  - What is MR?
  - Why should we care?
- Methods
  - Paradigms
  - Models
- Challenges
  - Why is it hard?
  - strengths & weaknesses
- Tools
  - Datasets
  - (Software)

Text to Symbolic Representations

Machine Reading

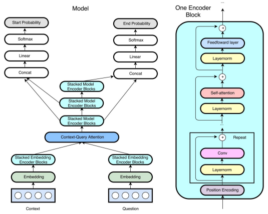
Auto-Text to Knowledge

Part 1

End-to-End Question Answering

Q

T

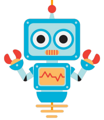


A

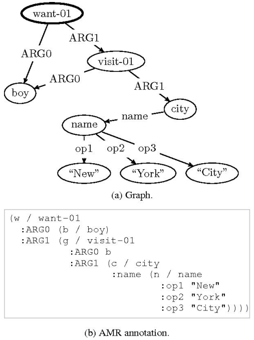
Part 2

Machine Reading

[Text]



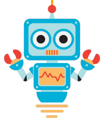
converts into



(a) Graph.

(b) AMR annotation.

[Meaning]

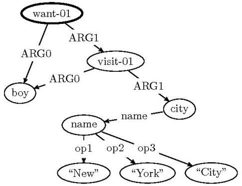


uses for

[Information Need]

Where do we see you?

use machine reading



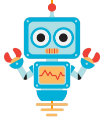
(a) Graph.

```
(w / want-01
  :ARG0 (b / boy)
  :ARG1 (g / visit-01
          :ARG0 b
          :ARG1 (c / city
                  :name (n / name
                          :op1 "New"
                          :op2 "York"
                          :op3 "City"))))
```

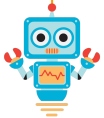
(b) AMR annotation.

I am a representative member of the UAI community

[Text]



[Meaning]



[Information Need]

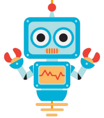
converts into

uses for

Where do we see you?

innovate for machine reading!

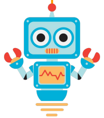
[Text]



converts into

I am a representative member of the UAI community

[Meaning]



uses for

[Information Need]

Relevant Topics

[Text]

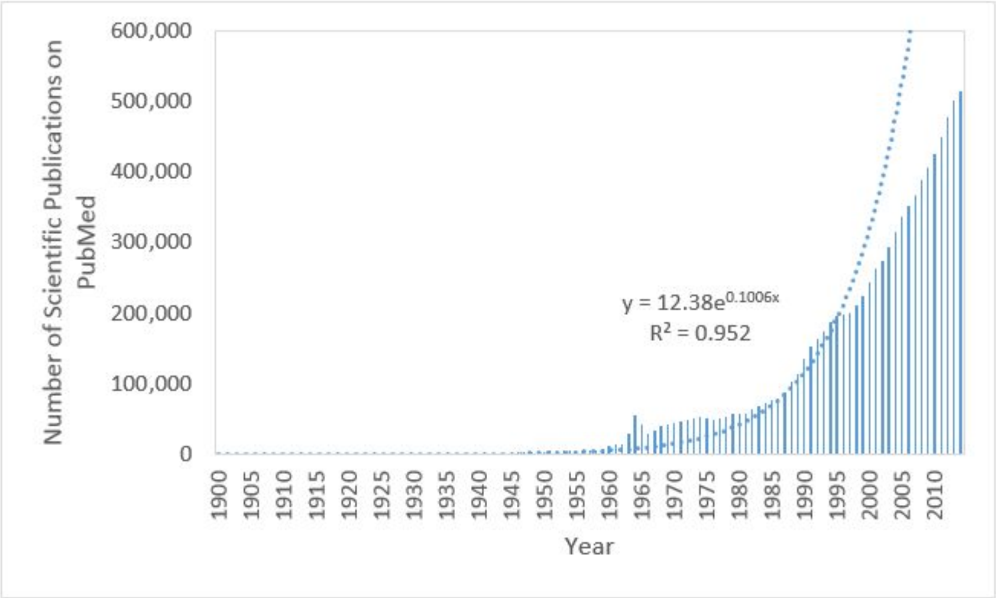
- Deep Learning
- Relational Learning
- Unsupervised Learning
- Multitask Learning
- Domain Adaptation
- Scalable ML
- Reasoning (+Logic)
- Reinforcement Learning
- Adversarial and Robust ML

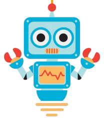
[Information Need]

What do we mean by Machine Reading?

A **machine** converts **text** into a representation of **meaning** that can satisfy (a broad set of) **information needs**

Motivation 1: Information Overload



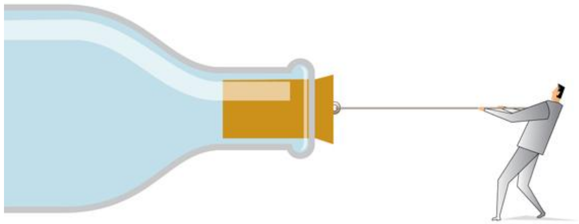


[Information Need]

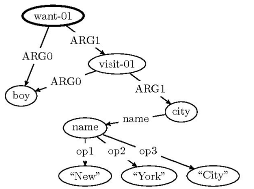
uses for

Motivation 2: The Knowledge Acquisition Bottleneck

"The problem of knowledge acquisition is the critical bottleneck problem in artificial intelligence."
EDWARD A. FEIGENBAUM 1984



[Knowledge]

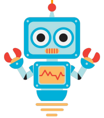


(a) Graph.

```
(w / want-01
  :ARG0 (b / boy)
  :ARG1 (g / visit-01
          :ARG0 b
          :ARG1 (c / city
                  :name (n / name
                          :op1 "New"
                          :op2 "York"
                          :op3 "City"))))
```

(b) AMR annotation.

[Meaning]



uses for

[Information Need]

Applications: Question Answering

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for **Prague** where he was to study.

?

What city did Tesla move to in 1880?

Prague

[Text]

[Meaning]

[Information Need]

Applications: Helping Agents to learn Faster

Branavan et al., 2012

The natural resources available where a population settles affects its ability to produce food and goods. Build your city on a plains or grassland square with a river running through it if possible.

?

[Text]

[Meaning]

[Information Need]

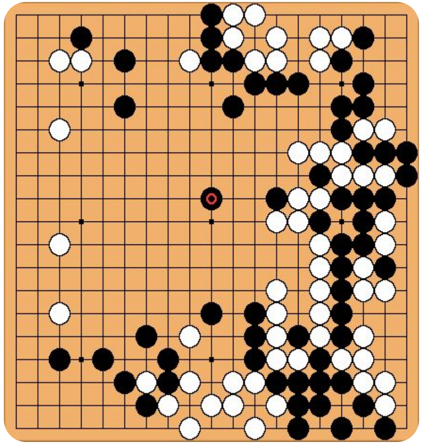
Applications: Helping Agents to learn Faster

A fundamental Go strategy involves keeping stones connected. Connecting a group with one eye to another one-eyed group makes them live together. Connecting individual stones into a single group results in an increase of liberties ...

[Text]

?

[Meaning]



[Information Need]

Applications: Precision Medicine

Poon et. al, 2017

Medical papers

The deletion mutation on exon-19 of EGFR gene was present in 16 patients, while the L858E point mutation on exon-21 was noted in 10. All patients were treated with gefitinib and showed a partial response.

[Text]

?

[Meaning]

Molecular Tumor Board

[Information Need]

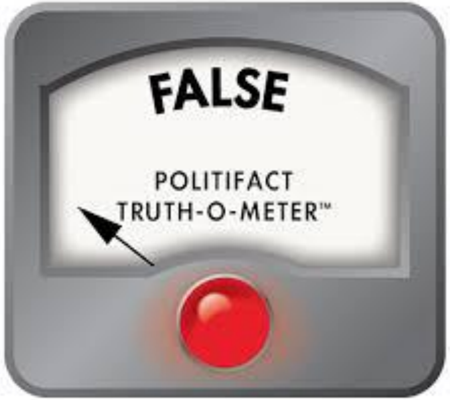
Applications: Misinformation

Vlachos & Riedel, 2016

"Once we have settled our accounts, we will take back control of roughly **£350m** per week." *Boris Johnson*

"...When those are taken into account the figure is **£250m**." *Independent*

?



[Text]

[Meaning]

[Information Need]

Machine Reading Approaches

[Text]

?

[Meaning]

[Information Need]

Semantic Parsing

Ewan forgot the mozarella in his car

$$
\begin{aligned}
&\exists x0\ \mathrm{named}(x0, \mathrm{ewan}, \mathrm{person}) \wedge \\
&\exists x1\ \mathrm{mozzarella}(x1) \wedge \\
&\exists x2\ \mathrm{car}(x2) \wedge \mathrm{of}(x2,x0) \wedge \mathrm{in}(x1, x2) \wedge \\
&\exists e\ \ \mathrm{event}(e) \wedge \mathrm{forget}(e) \wedge \mathrm{agent}(e, x0) \wedge \\
&\quad \mathrm{patient}(e, x1)
\end{aligned}
$$

[Text]

[Meaning]

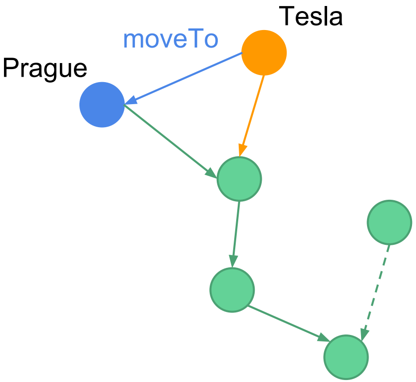
[Information Need]

Automatic Knowledge Base Construction

Banko et al. 2007, Carlson et al. 2010

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study. Unfortunately, he arrived too late to enrol at Charles-Ferdinand University; he never studied Greek, a required subject; and he was illiterate in Czech, another required subject. Tesla did, however, attend lectures at the university, although, as an auditor, he did not receive grades for the courses.

[Text]

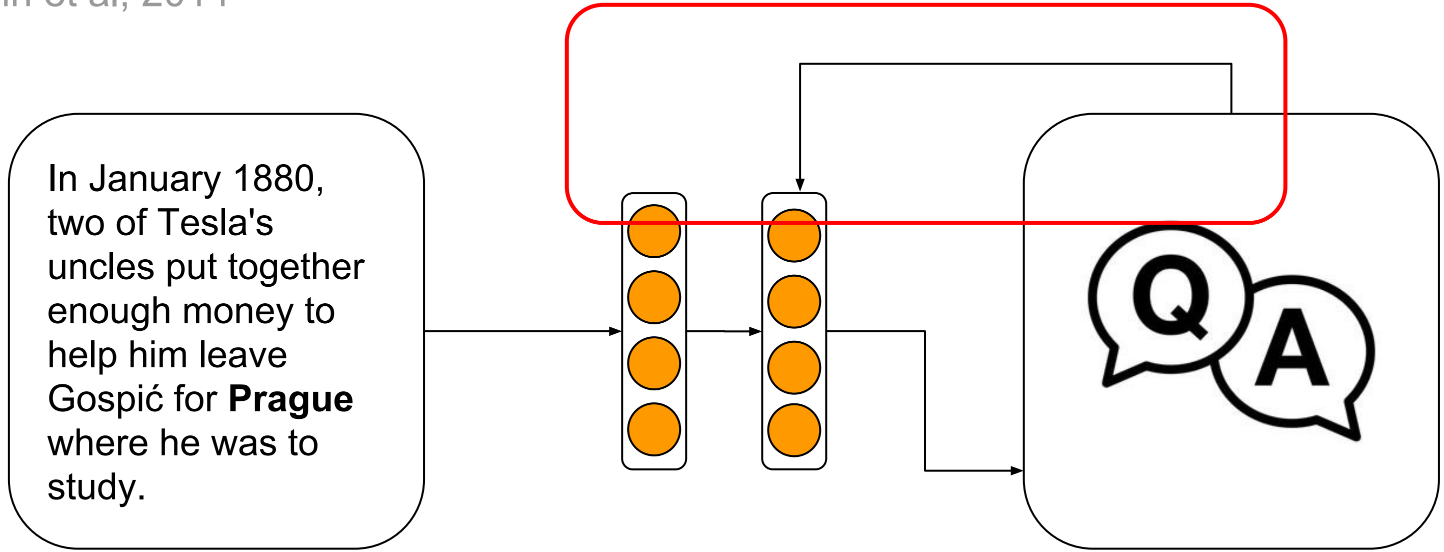


[Meaning]

[Information Need]

End-to-End Machine Comprehension

Hermann et al, 2014



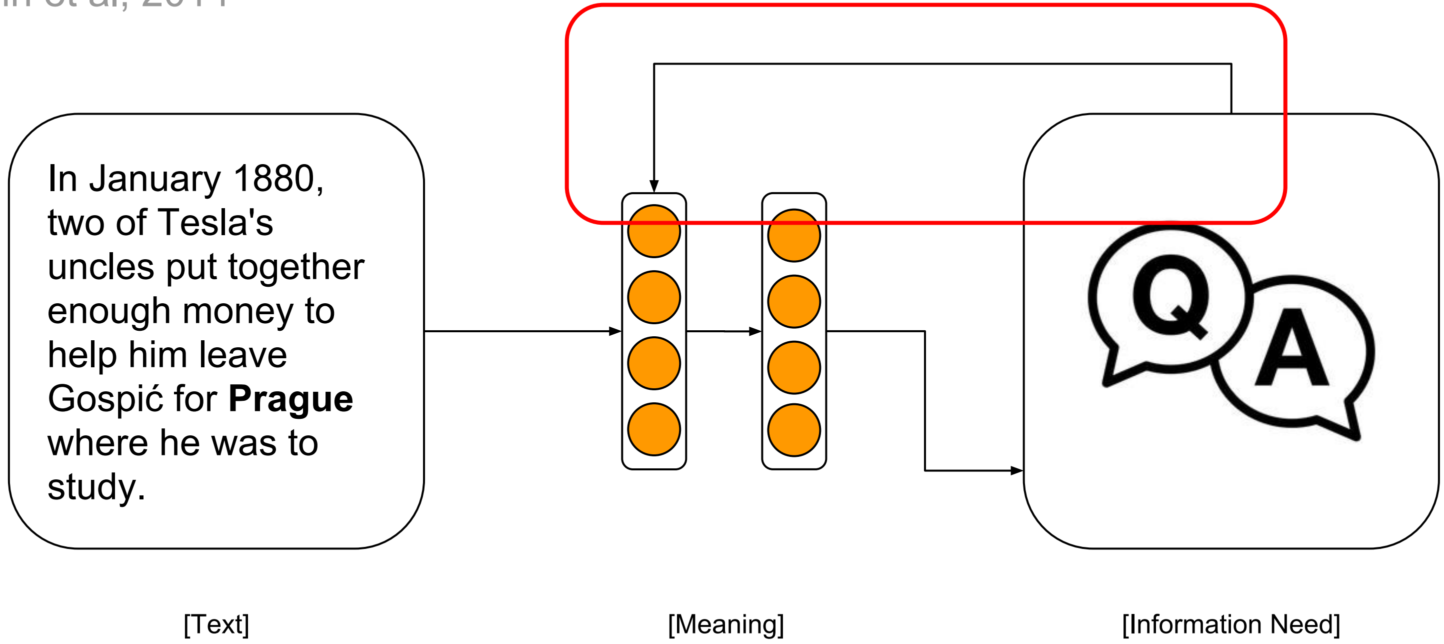
[Text]

[Meaning]

[Information Need]

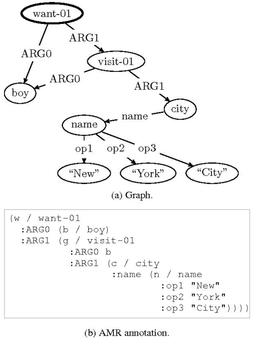
End-to-End Machine Comprehension

Hermann et al, 2014



What do we need from a representation?

[Text]



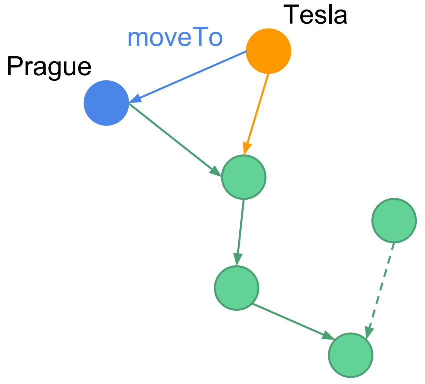
[Meaning]

- *Resolve* **Ambiguity**
- *Unify* **Variation**
- *Integrate* **Distributed Information**

Automatic
Knowledge Base Construction

Automated Knowledge Base Construction

Two of Tesla's uncles put together enough money to help him leave Gospić for Prague



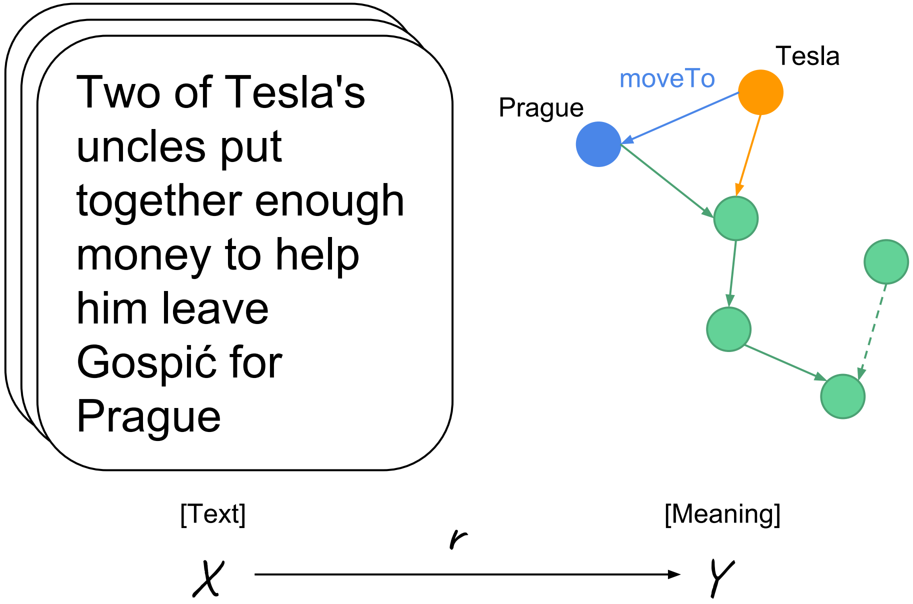
[Text] [Meaning]

$$X \xrightarrow{r} Y$$

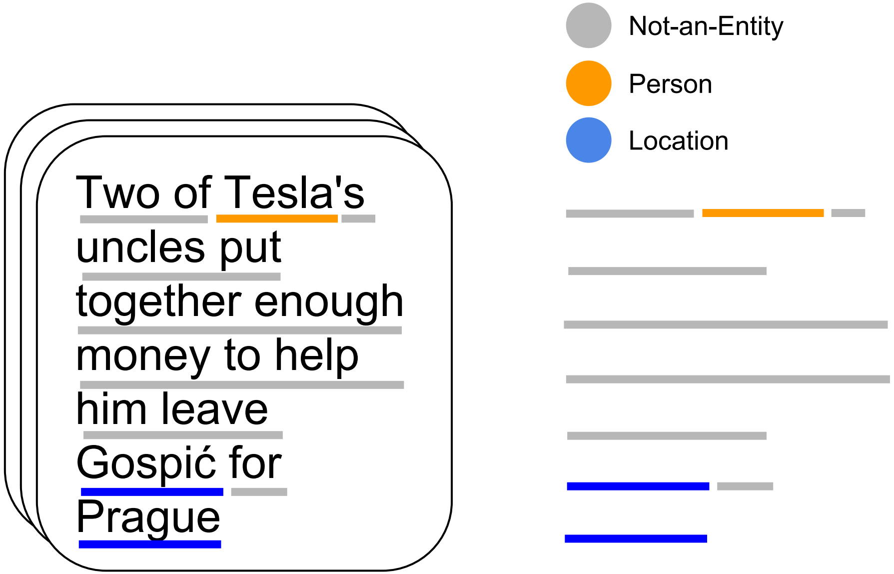
What city did Tesla move to in 1880?

Prague

Knowledge Graph Construction

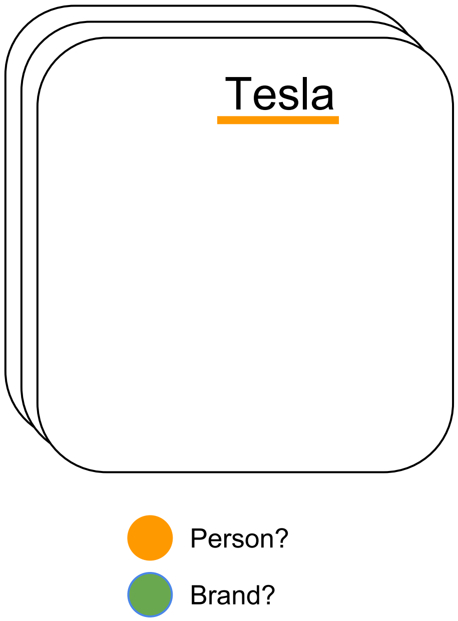


Entity Extraction and Typing as Sequence Labelling



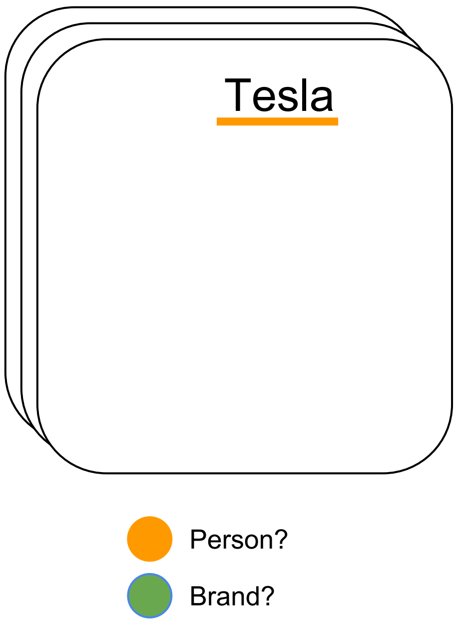
- Linear Chain CRF
- Bi-directional RNNs
- Hybrid RNN & CRFs

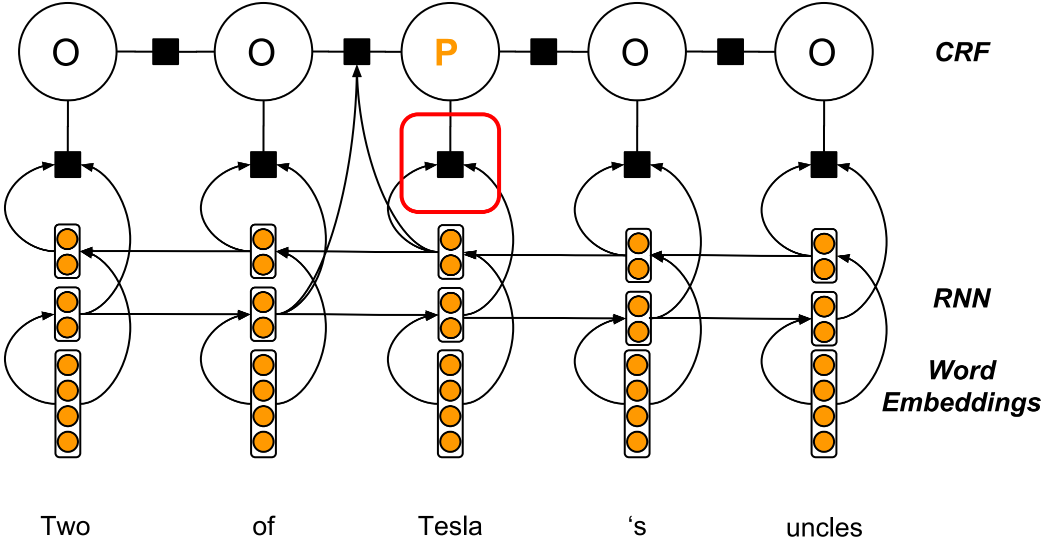
Challenge: Ambiguity



Conditional Random Fields with RNN Potentials

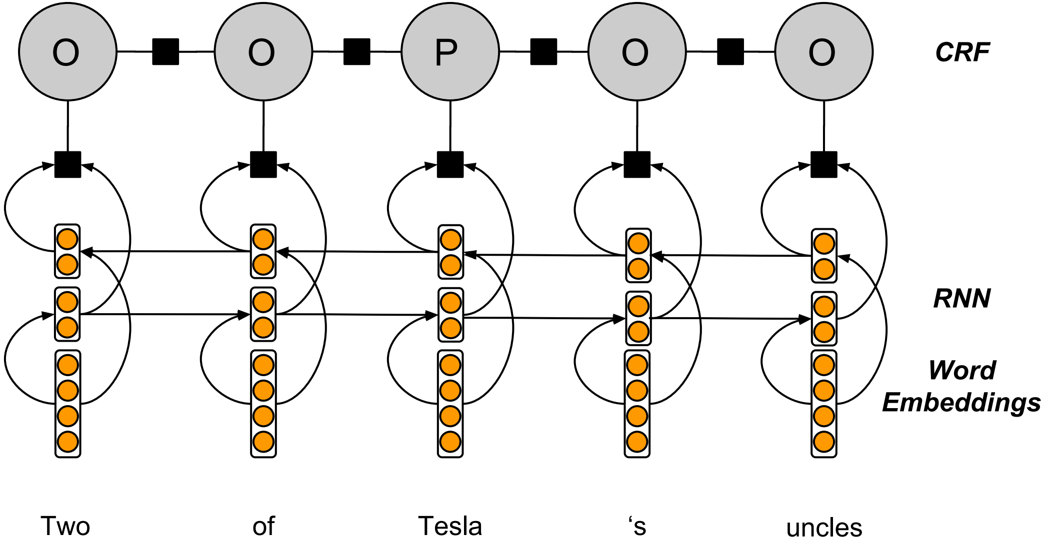
Huang et al., 2015





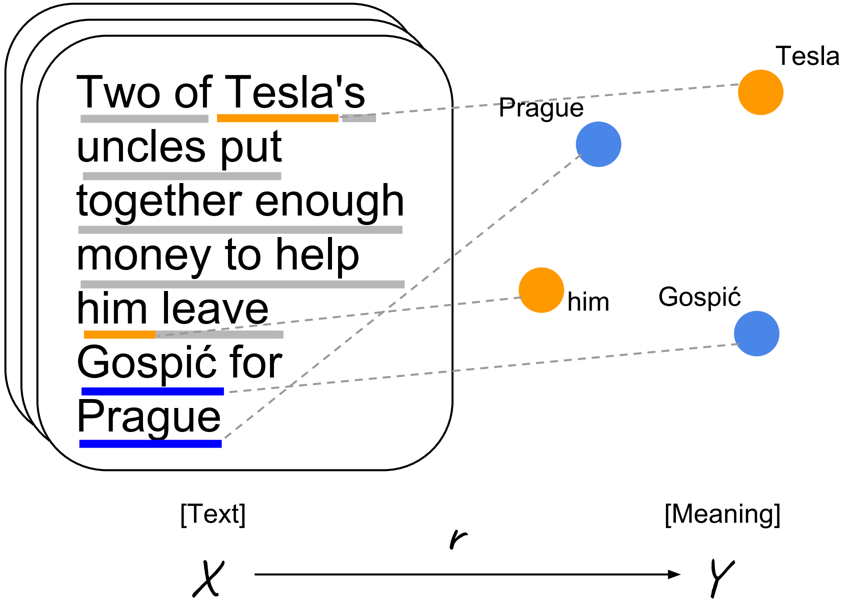
$p(\mathbf{y}|\mathbf{x})$

Direct Supervision

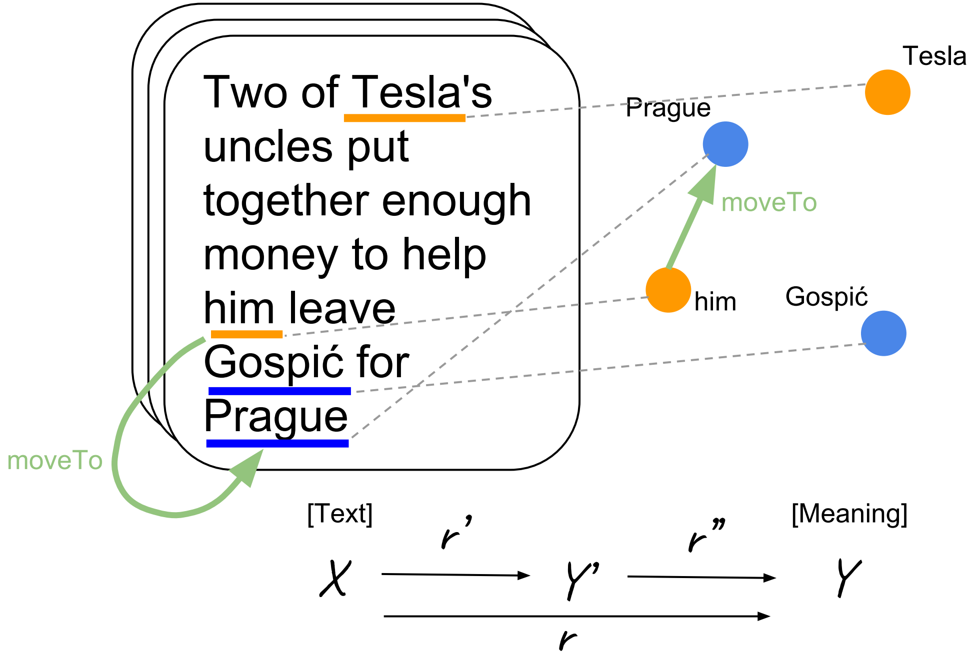


per token labels at training time

Instantiate Nodes



Relation Extraction



- Neural Classification
- Distant Supervision

Challenge: Variation

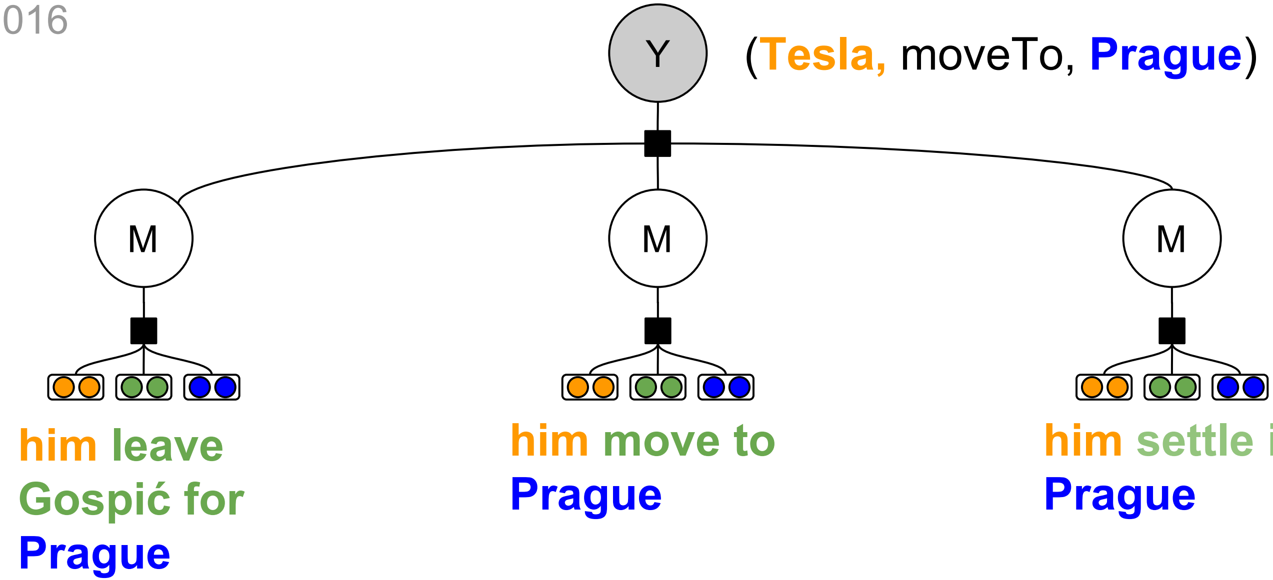
Two of Tesla's uncles put together enough money to help **him leave Gospić for Prague**

Two of Tesla's uncles put together enough money to help **him move to Prague**

Two of Tesla's uncles put together enough money to help **him settle in Prague**

Relation Classification

Lin et al., 2016

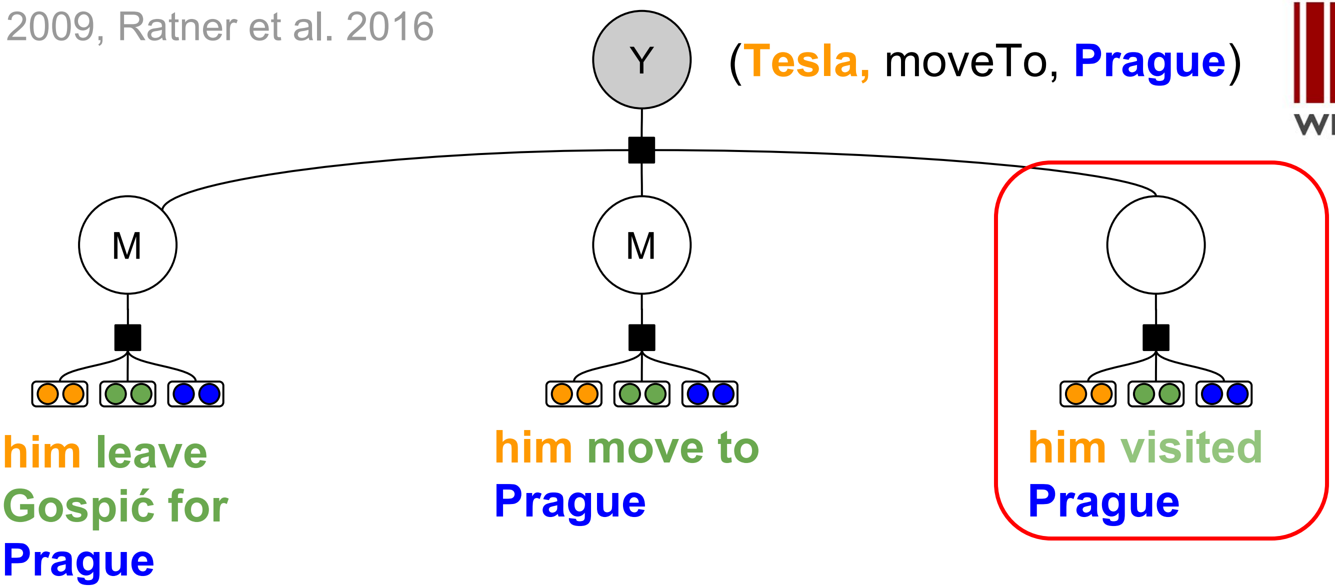


no **per-mention labels** for training

but **per entity-pair labels** in existing KBs

Distant Supervision & Multiple Instance Learning

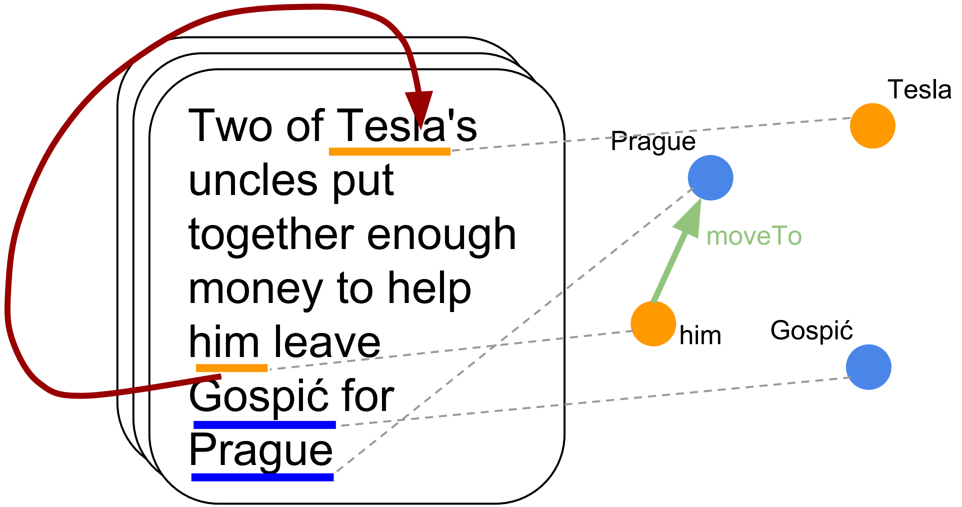
Mintz et al., 2009, Ratner et al. 2016



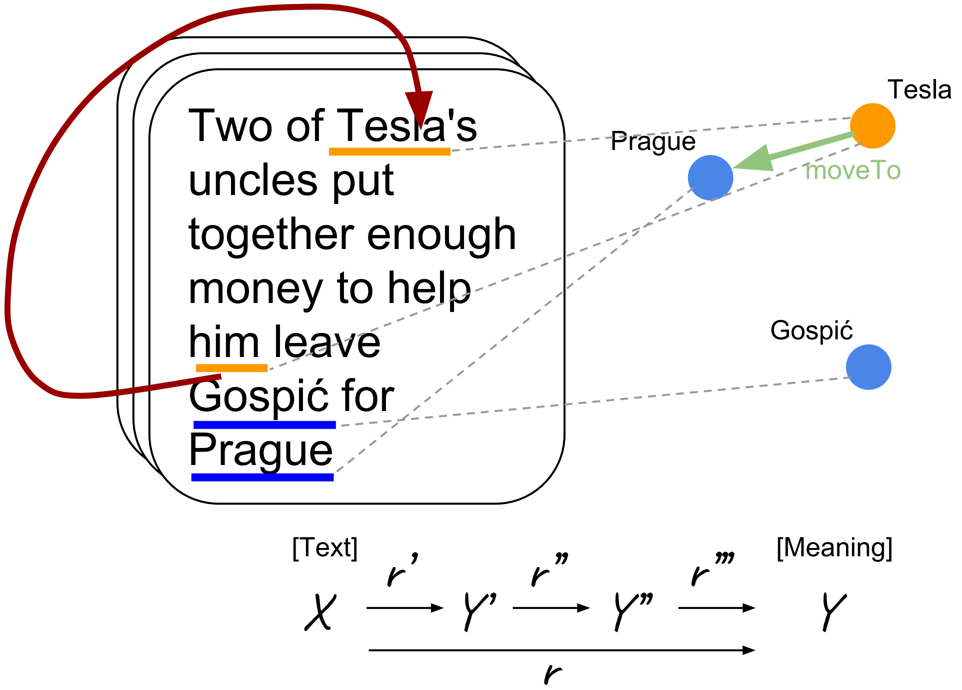
WIKIDATA

Not all mentions express the relation

Coreference Resolution



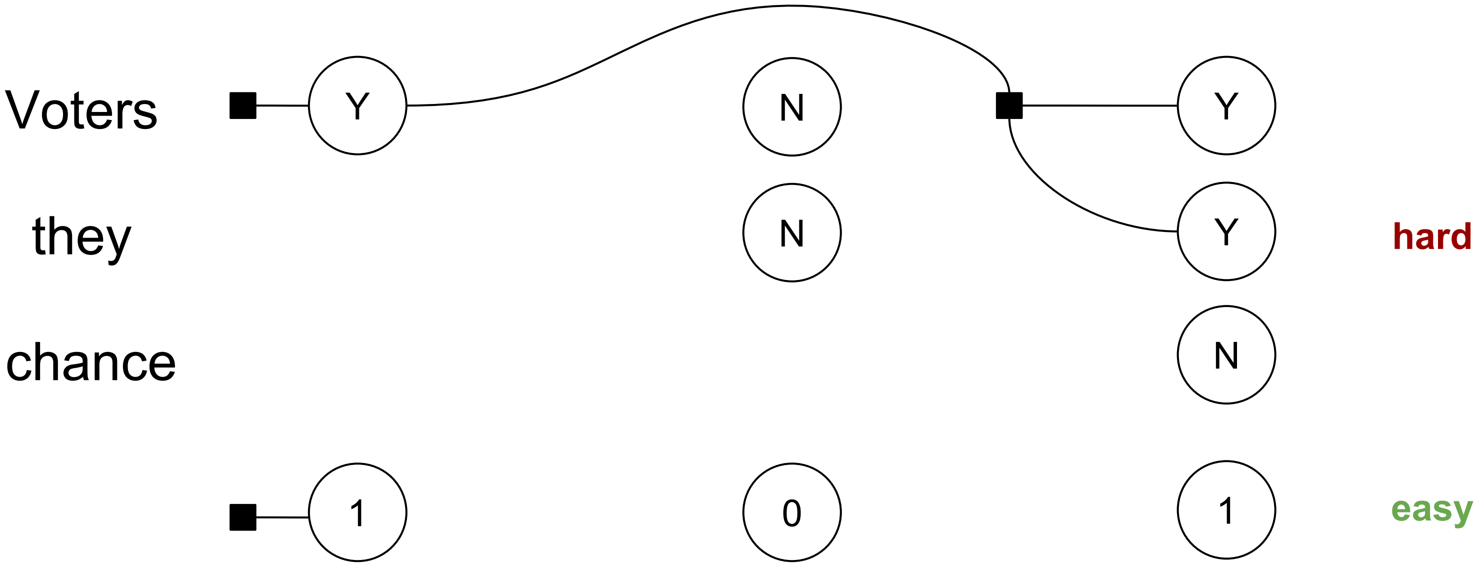
Collapsing Nodes



- Neural Classification
- Latent Variable Modelling

Coreference Resolution

Inference



Voters agree when **they** are given a **chance** to decide if **they** ...

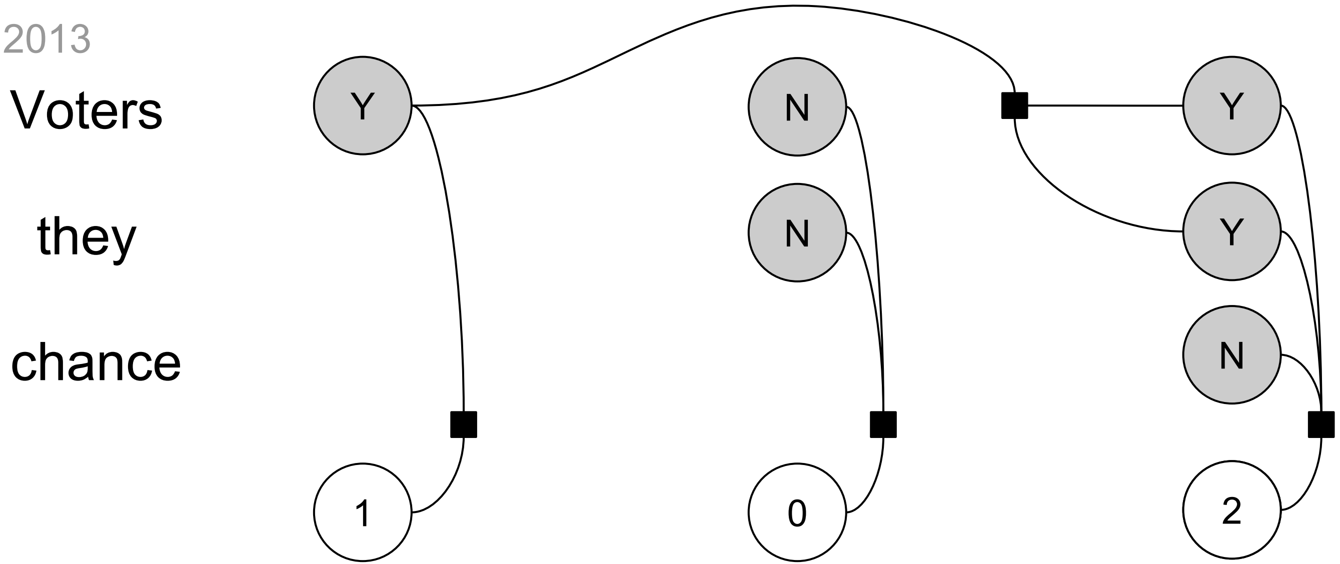
1 2 3 4

what is my **“best antecedent”?**

Latent Variables

Durrett & Klein, 2013

Only clusters are given at training time



Voters agree when they are given a chance to decide if they ...

1 2 3 4

marginalize out at training time

Challenge: Common Sense

Levesque, 2011

Two of Tesla's uncles put together enough money to help **him** leave Gospić for Prague

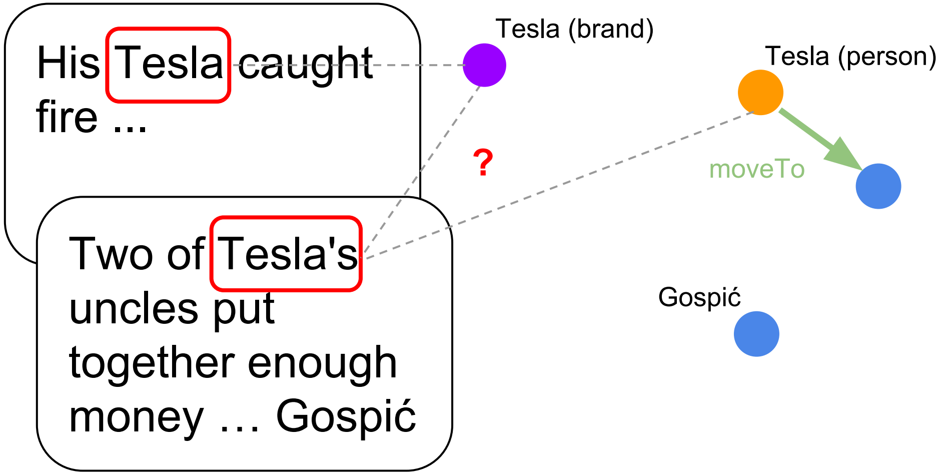
The **trophy** would not fit in the brown **suitcase** because **it** was too ***big***.

The **trophy** would not fit in the brown **suitcase** because **it** was too ***small***.

Surface

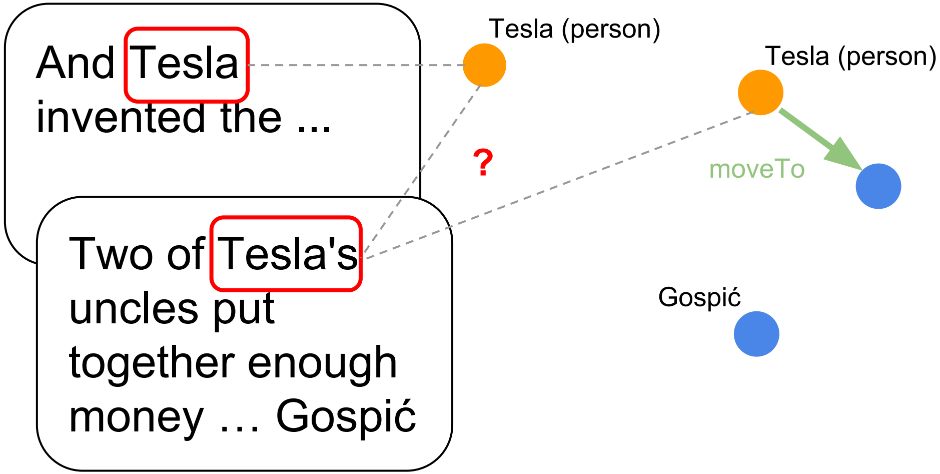
Common Sense

Entity Linking



Entity Linking

Le and Titov, 2018



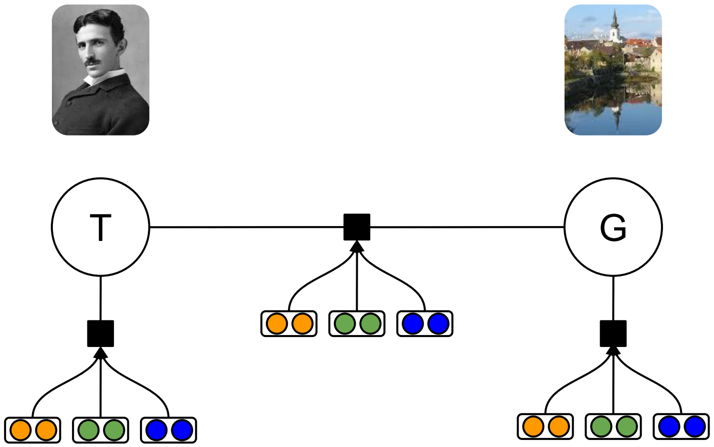
- Neural Potentials
- Belief Propagation

Entity Linking

Le and Titov, 2018

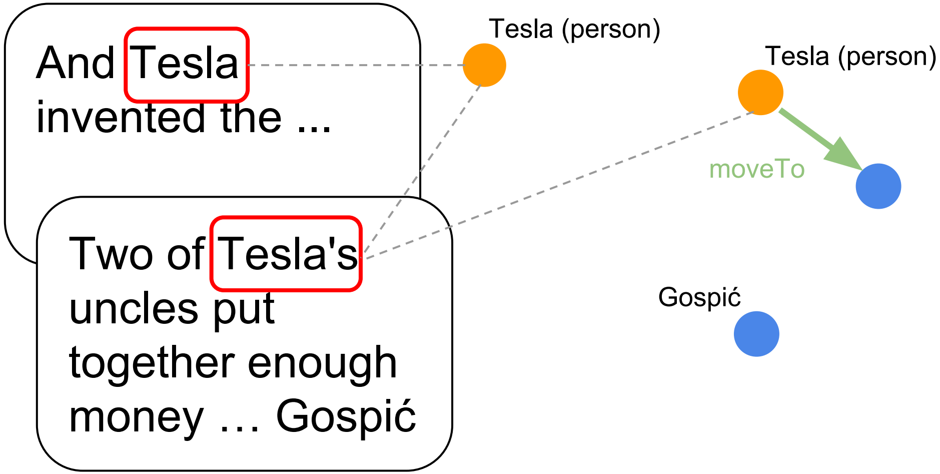
And Tesla invented the ...

Two of Tesla's uncles put together enough money … Gospić

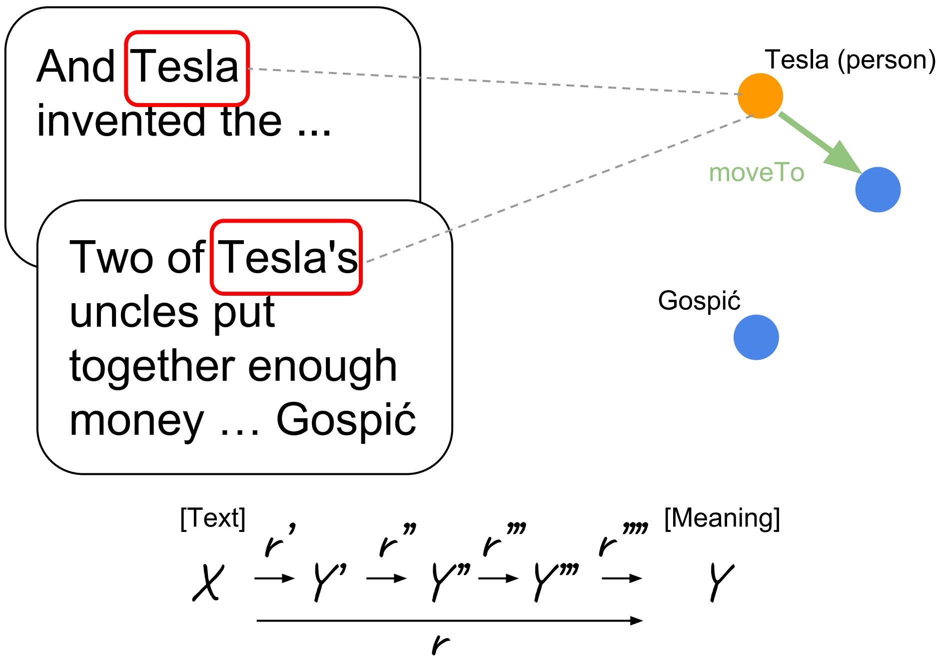


per-document graphical model

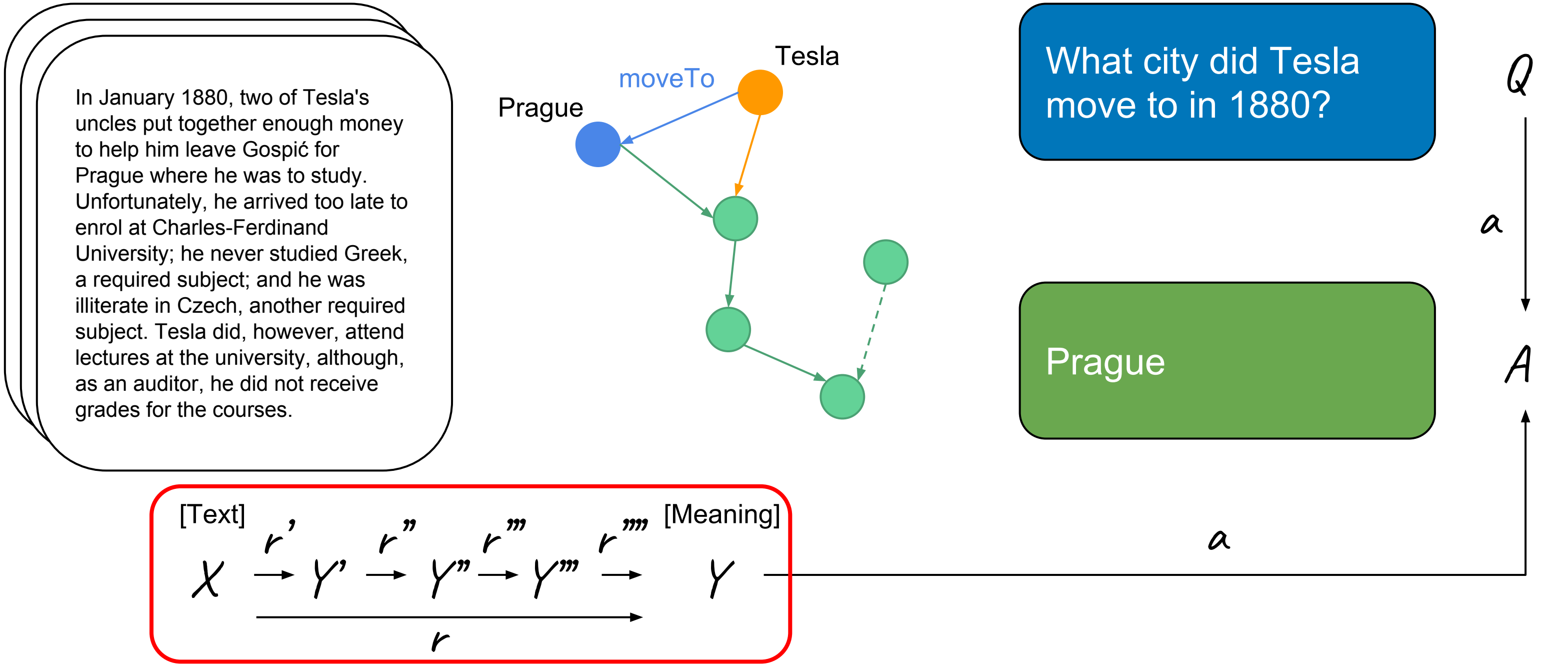
Collapsing Nodes



Collapsing Nodes

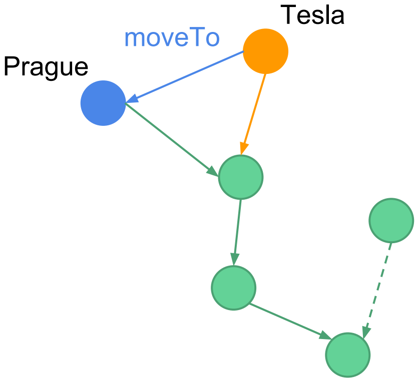


Weakness: Cascading errors



Weakness: Cascading errors

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study. Unfortunately, he arrived too late to enrol at Charles-Ferdinand University; he never studied Greek, a required subject; and he was illiterate in Czech, another required subject. Tesla did, however, attend lectures at the university, although, as an auditor, he did not receive grades for the courses.



What city did Tesla move to in 1880?

moveTo(Tesla,X)?

Prague

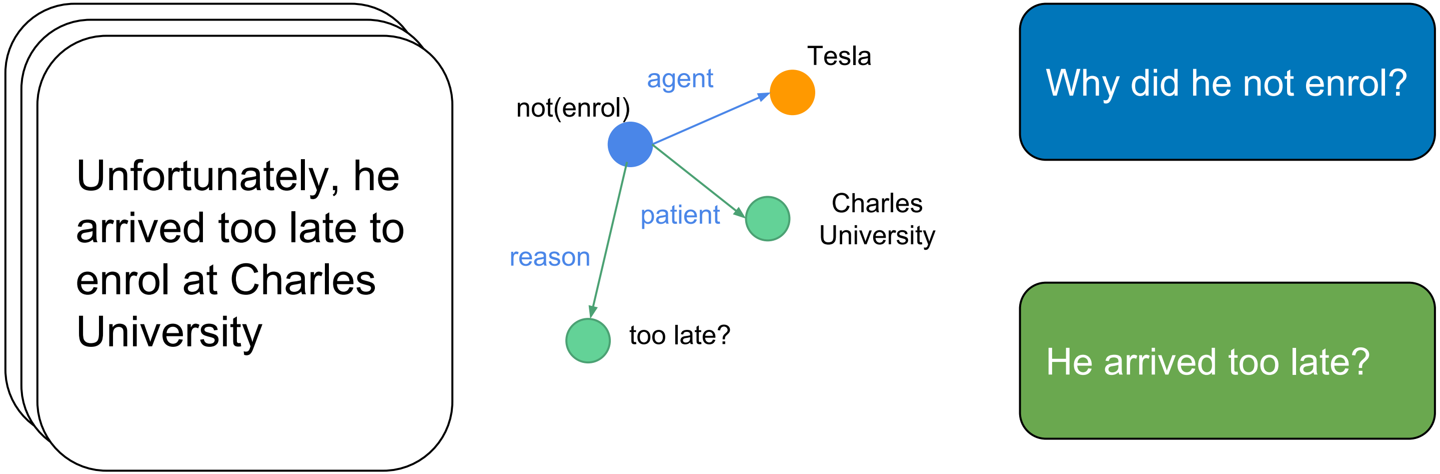
$$Q \xrightarrow{a'} Q' \xrightarrow{a''} A \qquad Q \xrightarrow{a} A$$

[Text] [Meaning]

$$X \xrightarrow{r'} Y' \xrightarrow{r''} Y'' \xrightarrow{r'''} Y''' \xrightarrow{r''''} Y \qquad X \xrightarrow{r} Y$$

$$Y \xrightarrow{a} A$$

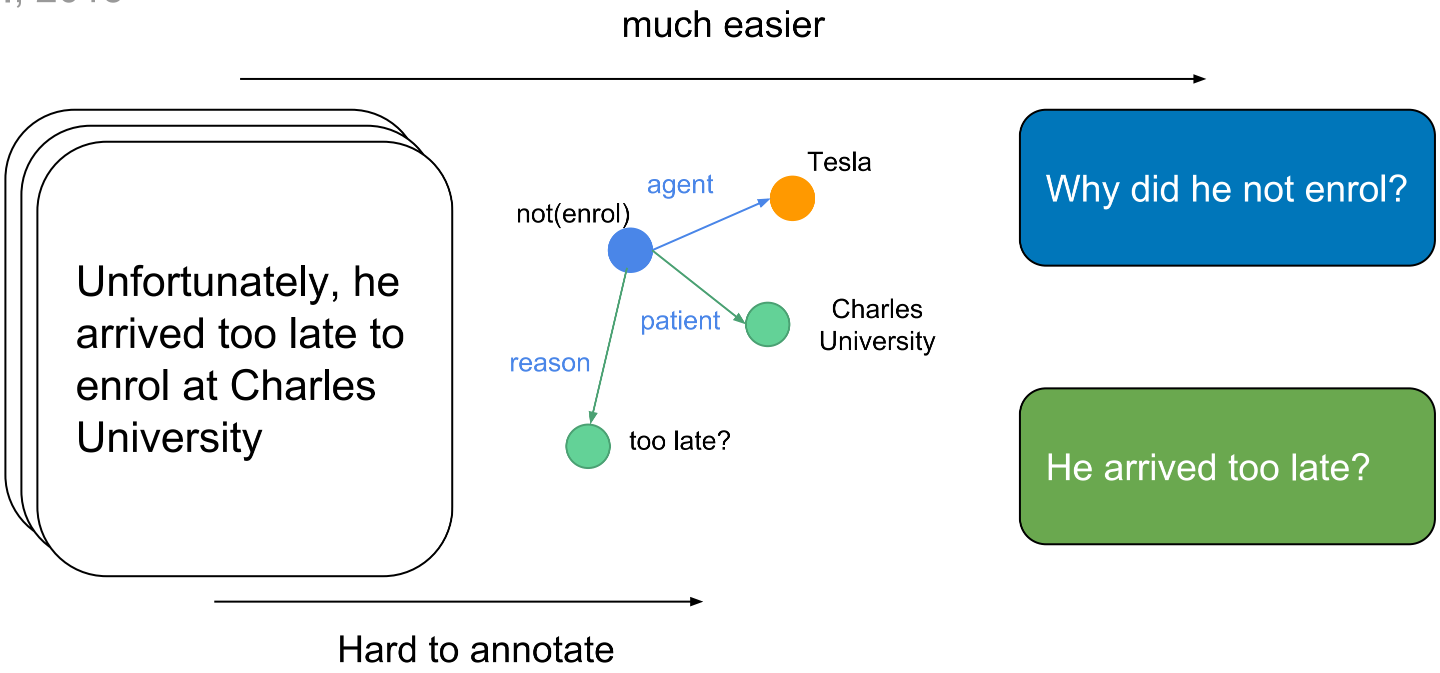
Weakness: Engineering Schemas and Formalisms



getting this right is hard

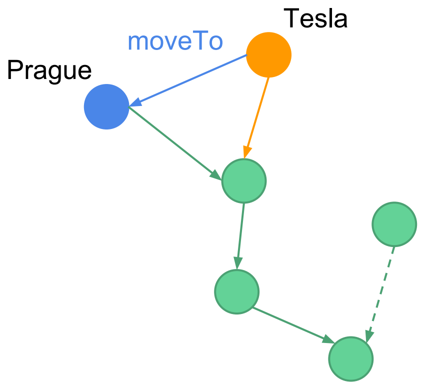
Weakness: Annotation

He et al., 2015



Is there another way?

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study. Unfortunately, he arrived too late to enrol at Charles-Ferdinand University; he never studied Greek, a required subject; and he was illiterate in Czech, another required subject. Tesla did, however, attend lectures at the university, although, as an auditor, he did not receive grades for the courses.



What city did Tesla move to in 1880?

Prague

$Q \xrightarrow{a} A$

[Text] [Meaning]

$X \xrightarrow{r} Y \xrightarrow{a} A$

Learn the Mapping End-To-End

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study. Unfortunately, he arrived too late to enrol at Charles-Ferdinand University; he never studied Greek, a required subject; and he was illiterate in Czech, another required subject. Tesla did, however, attend lectures at the university, although, as an auditor, he did not receive grades for the courses.

[Text]

X

What city did Tesla move to in 1880?

Q

a

Prague

A

a

End-to-End QA

Stanford Question Answering Dataset (SQuAD)

Rajpurkar et. al. 2016

Text Passage

[...] 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".

Question + Answer

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

within a cloud

Task: Given a paragraph and a question about it, predict the text span that states the correct answer.

Stanford Question Answering Dataset (SQuAD)

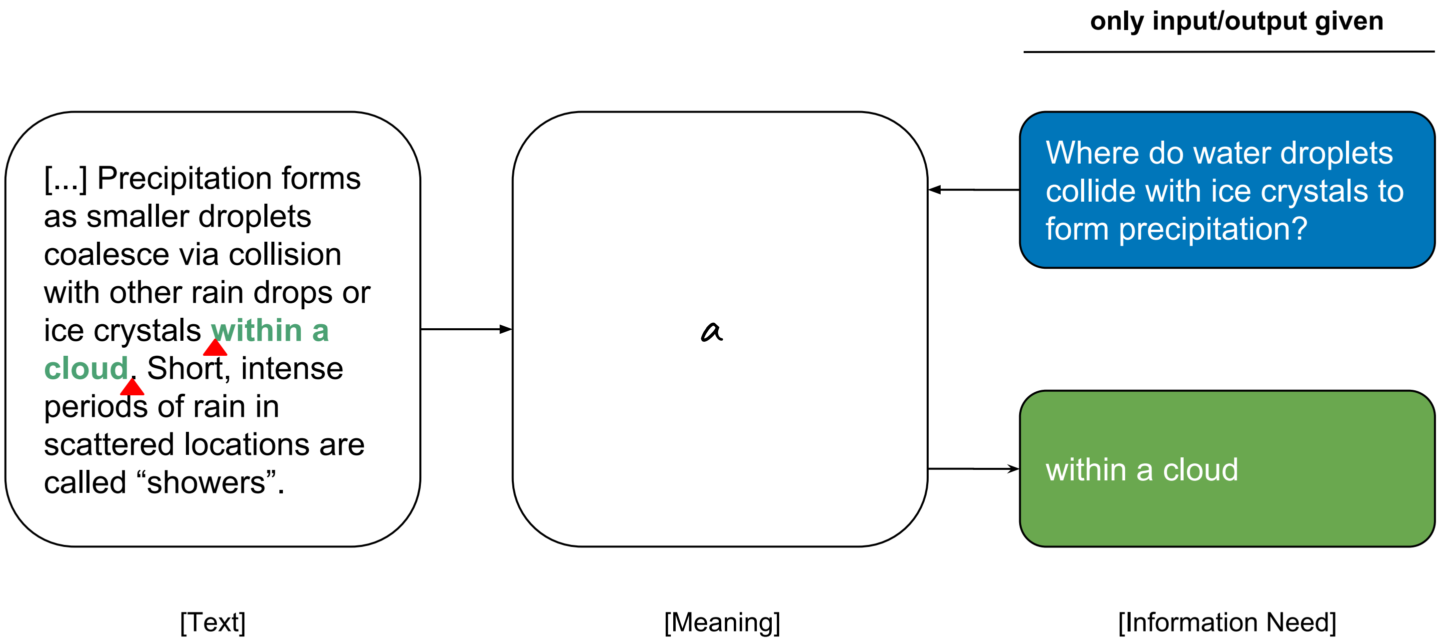
Rajpurkar et. al. 2016

- **Dataset size**: 107,702 samples
- Widely used benchmark dataset
- **Task:** *Extractive* Question Answering
    - Other forms of QA exist, e.g. free-form answer generation, multiple choice

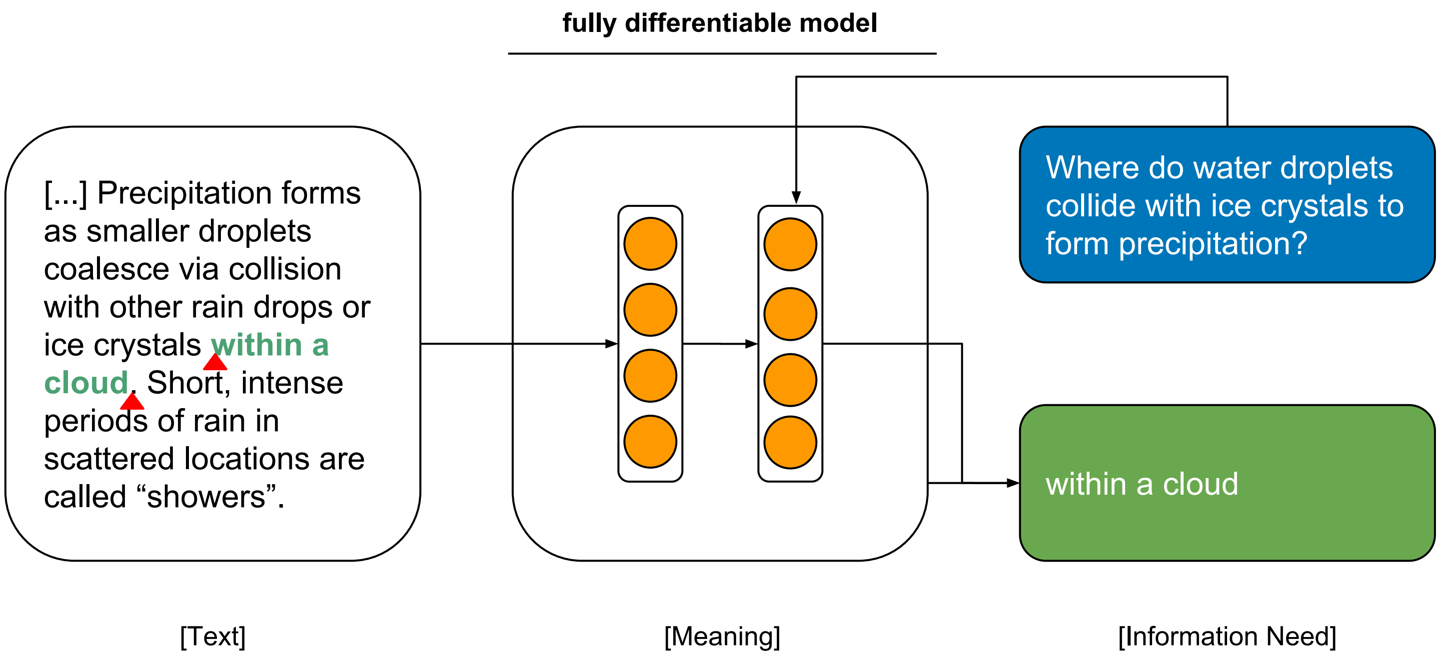
List of Other QA Datasets

Dataset Name	Task Format	Supervision type	Total Size	Authors / Reference
TREC-QA	Query log, IR + free form	Human verification	1,479	Voorhees and Tice (2000)
QuizBowl	Trivia Question Answering	Expert Creation	37,225	Boyd-Graber et al (2012)
WebQuestions	NL question + KB	Google Search API & Human verification	5,810	Berant et al. (2013)
MCTest	Multiple Choice QA	crowdsourced	2640	Richardson et al. (2013)
CNN & Daily Mail	Cloze, Multiple Choice QA	Distant Supervision	387,420 + 997,467	Hermann et al. (2015)
WikiQA	Extractive QA/ sentence selection Â with Bing queries	crowdsourced	3,047	Yang et al. (2015)
SimpleQuestions	NL question + KB	KB + crowdsourced questions	108,442	Bordes et al (2015)
Children Book Test	Multiple Choice Cloze QA	Automatic (fill-the-blank)	687,343	Hill et al. (2016)
SQuAD (1.0 + 2.0)	**Extractive QA**	**Crowdsourced**	**107,702**	**Rajpurkar et al (2016), Rajpurkar and Jia et al (2018)**
bAbI	20 complex reasoning tasks with controlled language	Automatically Generated	20,000	Weston et al. (2016)
ComplexQuestions	NL question + KB	Search API & Human verification	2,100	Bao et al. (2016)
MovieQA	Multiple choice QA, text & video.	crowdsourced	14,944	Tapawasi et al. (2016)
WhoDidWhat	Cloze, Multiple Choice QA	Distant Supervision	205,978	Onishi et al. (2016)
MS MARCO	Bing queries and NL answers	crowdsourced	100,000	Nguyen et al (2016)
Lambada	Cloze QA	Automatic (human verification)	10,022	Paperno et al. (2016)
WikiReading	KB query, NL text	Distant Supervision	18.58M	Hewlett et al. (2016)
TriviaQA	Trivia Question Answering	Expert Creation + Distant Supervision	662,659	Joshi et al. (2017)
SciQ	Multiple choice QA	crowdsourced	13,679	Welbl et al. (2017)
RACE	Multiple choice Exam questions	Expert Creation	97,687	Lai et al. (2017)
NewsQA	Extractive QA	crowdsourced	119,633	Trischler et al. (2017)
AI2 Science Questions	Multiple Choice Science Exam QA	Expert Creation	5,059	Allen Institute for AI (2017 release)
SearchQA	Trivia questions + Search Engine Results	Expert Creation + distant supervision	140,461	Dunn et al. (2017)
QUASAR-S & QUASAR-T	Cloze & free-form trivia questions	Distant supervision	37,362 + 43,013	Dhingra et al. (2017)
Wikihop & Medhop	KB query, NL text, multiple Choice	Distant Supervision	51,318+2,508	Welbl et al. (2018)
NarrativeQA	free-form answer generation	crowdsourced	46,765	Kocisky□ et al. (2018)

End-to-end Machine Reading for Question Answering



End-to-end Machine Reading for Question Answering

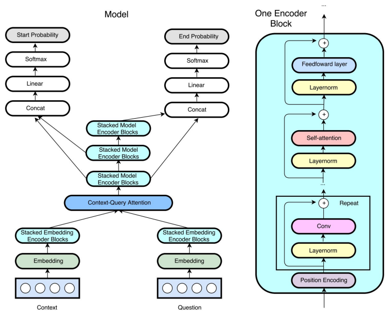


End-to-end Machine Reading for Question Answering

QANet, Yu et. al. 2018

State-of-the-Art Architecture

[...] 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”.



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

within a cloud

[Text]

[Meaning]

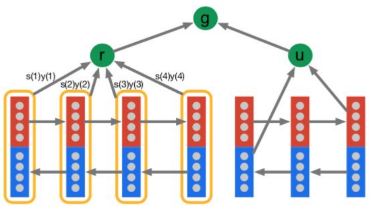
[Information Need]

End-to-end Machine Reading for Question Answering

Hermann et. al. 2015

Simpler Architecture

[...] 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”.



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

within a cloud

[Text]

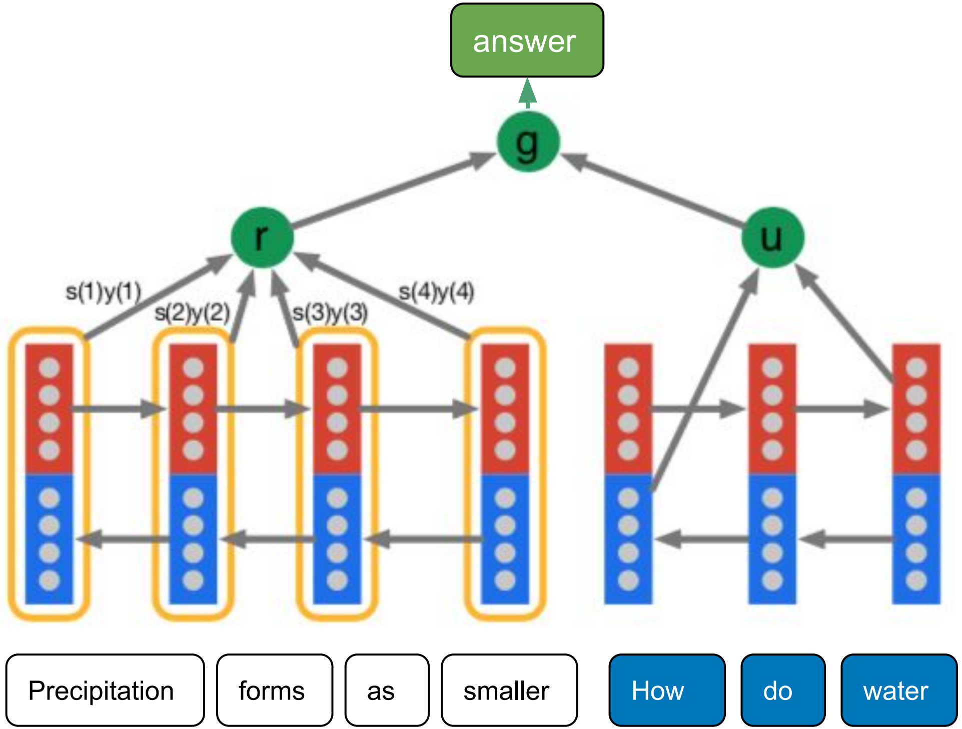
[Meaning]

[Information Need]

The Attentive Reader Model: Overview

Hermann et. al. 2015

- ‘early’ neural model for Machine Reading
- main components reused in many other models



Modified visualization from Hermann et. al. 2015

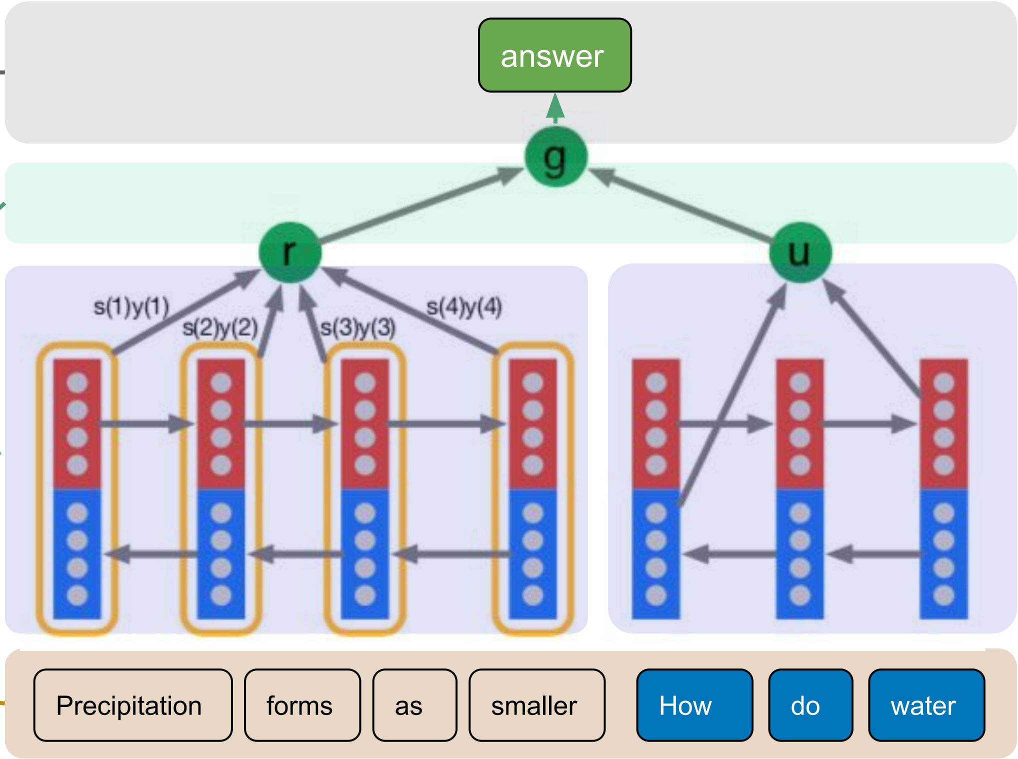
The Attentive Reader Model: Overview

Answer Selection: generate an answer or span

Sequence Interaction: Matching text with question

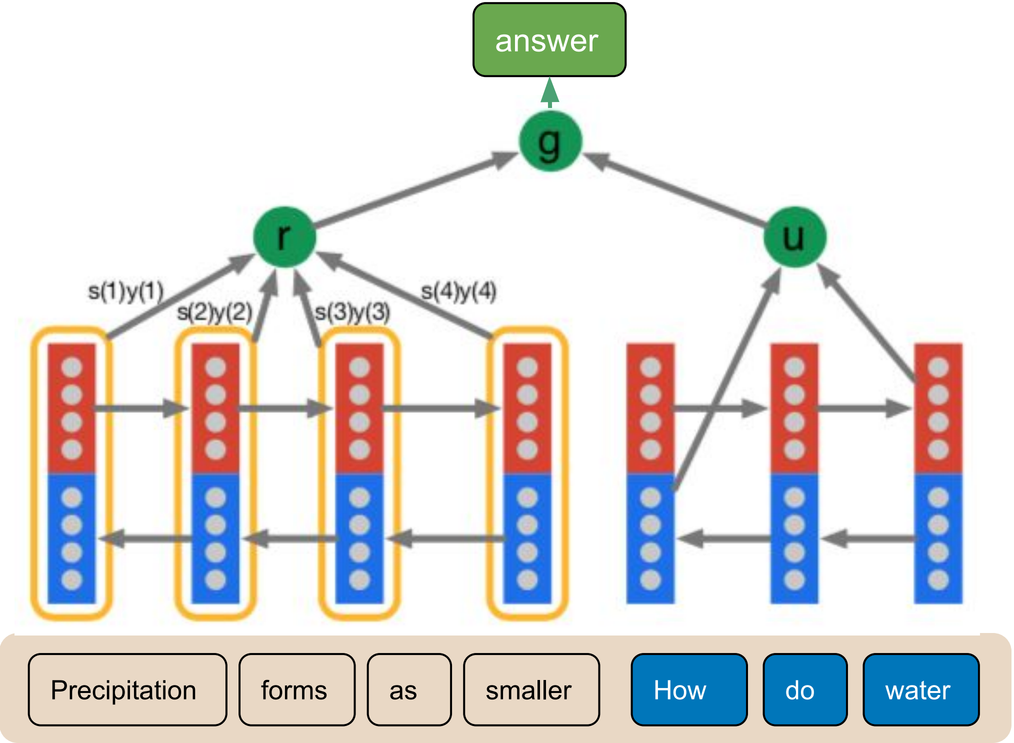
Composition: incorporating context around words

Input: Representing symbols as vectors



Modified visualization from Hermann et. al. 2015

The Attentive Reader Model: Overview

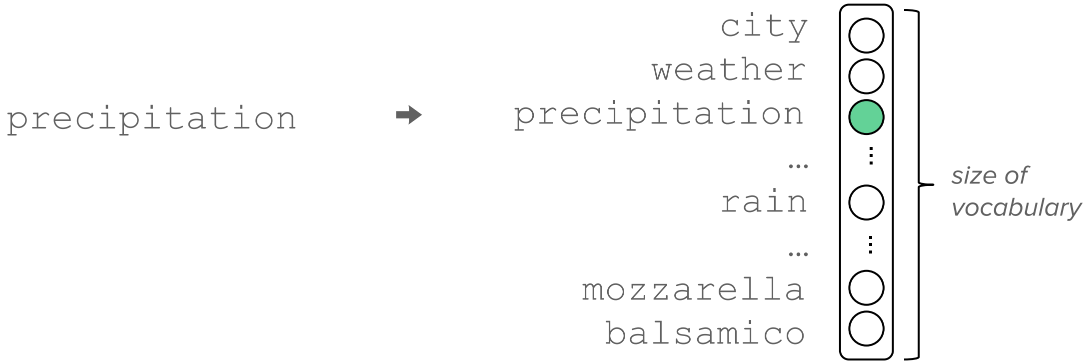


Input: Representing symbols as vectors

Modified visualization from Hermann et. al. 2015

Representing Symbols as Vectors

- **Problem:** Words / characters are discrete symbols, but neural nets work with vector inputs
- **Naive solution:** construct one-hot vector for each word



Representing Symbols as Vectors

Problem with naive solution:

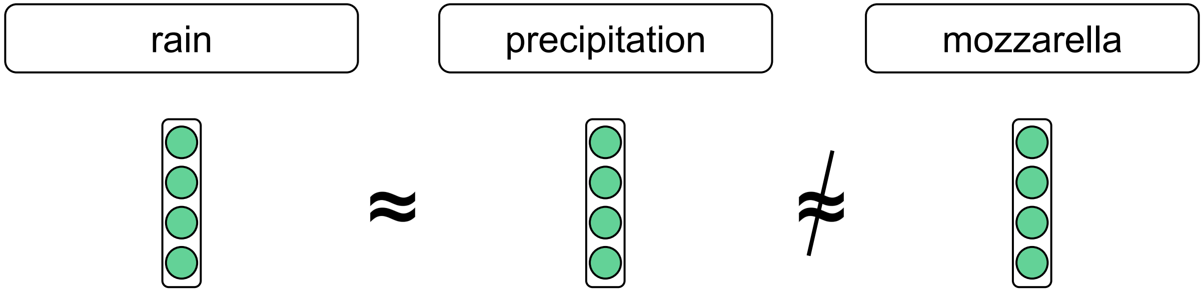
- one-hot vectors do not represent relationships between words
  - all one-hot vectors are orthonormal
  - hard to train model which generalizes across similar words
    - e.g. rain vs. precipitation
- high-dimensional, extremely sparse input -> computational issues

Representing Symbols as Vectors

Problem with naive solution:

- one-hot vectors do not represent relationships between words
  - all one-hot vectors are orthonormal
- high-dimensional, extremely sparse input
- hard to train model which generalizes across similar words
  - e.g. rain vs. precipitation

Ideal Vector Representations for Words



Similar meaning of words → similar vector representations

?

Word Similarity

?

We found a little, hairy wampimuk
sleeping behind the tree.

after Marco Baroni

use context to
infer meaning!

Distributional Hypothesis: *"Words that are used and occur in the same contexts tend to purport similar meanings."* (Harris, 1954)

Short Version:
"You shall know a word by the company it keeps." (Firth, 1957)

Word Similarity

"You shall know a word by the company it keeps."

➜ Two words are similar if they appear in the same documents.

Term-Document matrix:

	d1	d2	d3	d4	...	d*M*
city	2	0	0	0	...	1
weather	0	1	0	1	...	1
precipitation	4	2	0	1	...	1
...	...	...	...	...	...	...
rain	1	1	0	1	...	1
mozzarella	0	0	3	0	...	0
balsamico	0	0	1	0	...	0

Vector for "rain" is similar to "precipitation", not to "mozzarella".

Word Similarity

"You shall know a word by the company it keeps."

➔ Two words are similar if they appear in the same documents.

Term-Document matrix:

	d1	d2	d3	d4	…	d*M*
city	2	0	0	0	…	1
weather	0	1	0	1	…	1
precipitation	4	2	0	1	…	1
…	…	…	…	…	…	…
rain	1	1	0	1	…	1
mozzarella	0	0	3	0	…	0
balsamico	0	0	1	0	…	0

Somewhat collinear, but very sparse

Combatting Sparsity

- **Key Idea:** Approximate Sparse matrix using low-rank matrix factorization

  ➜ Dense Factor matrices for words, and for documents

	d1	d2	d3	d4	...	d*M*
city	2	0	0	0	...	1
weather	0	1	0	1	...	1
precipitation	4	2	0	1	...	1
...	...	...	...	...	...	...
rain	1	1	0	1	...	1
mozzarella	0	0	3	0	...	0
balsamico	0	0	1	0	...	0

$$\approx \; U \times \Sigma \times V$$

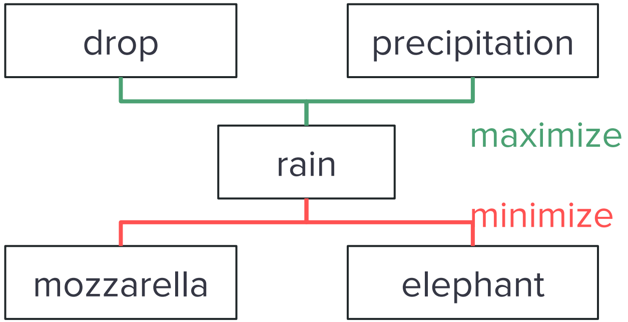
Row vectors: dense representations for each word

Word Embeddings

- **word embeddings:**
  dense vector representations for words of low dimensionality (e.g. 300)
- can capture word similarity (to a degree)
- usually pretrained on large text corpus
- e.g. **word2vec** (Mikolov et al., 2013)
- Different approach: character-based word embeddings, *e.g., Kim et al. 2016*

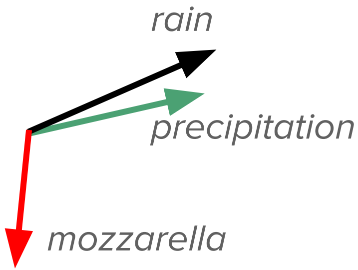
Word2Vec - (SkipGram with Negative Sampling)

1. *Maximize similarity between co-occurring words*
2. *minimize similarity between non co-occurring words*



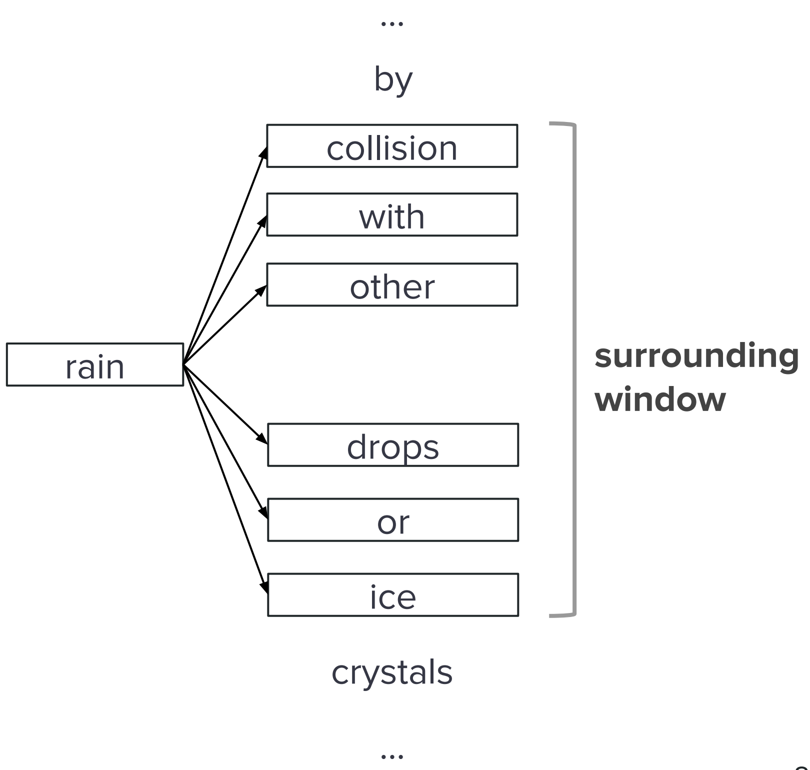
similarity = collinearity

$$s(a, b) = \sigma(\mathbf{v}_a \cdot \mathbf{v}_b)$$

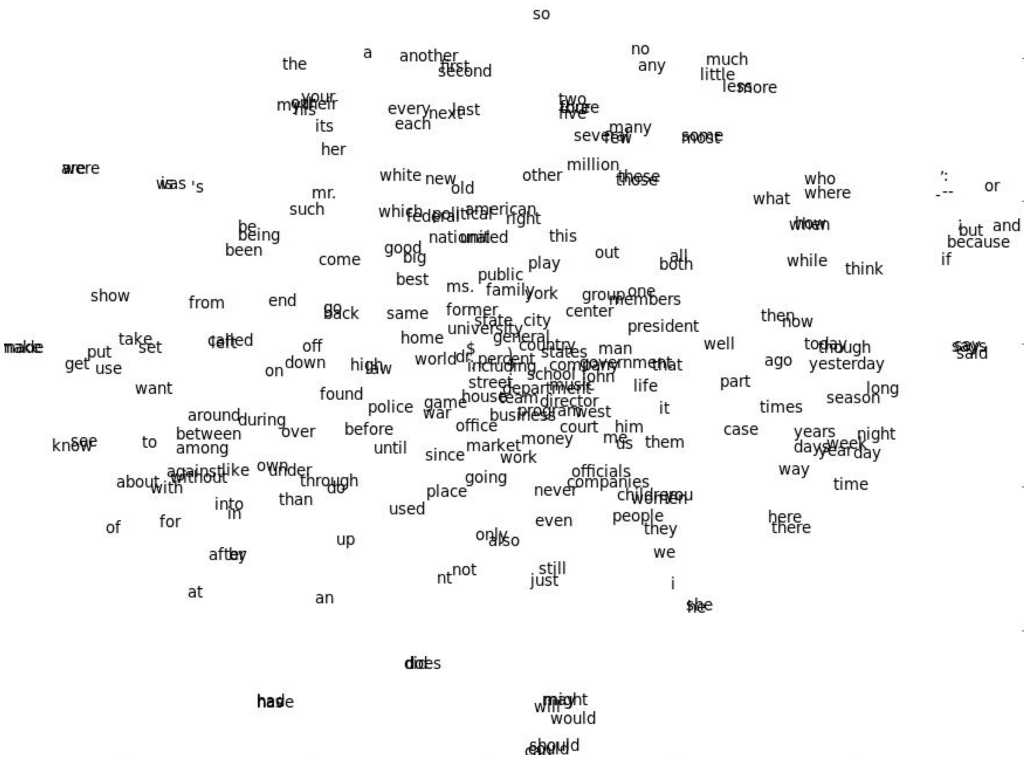


Word2Vec - (SkipGram with Negative Sampling)

- Training: use vectors to predict words in surrounding window
- Implicitly related to factorization of word-context PMI matrix *(Levy and Goldberg, 2014)*

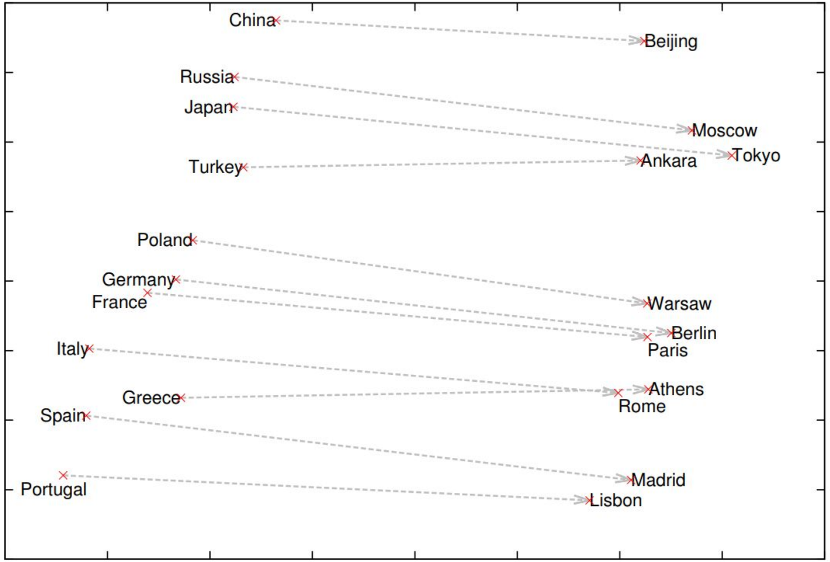


Visualizing Word Embeddings



Turian et al. 2010

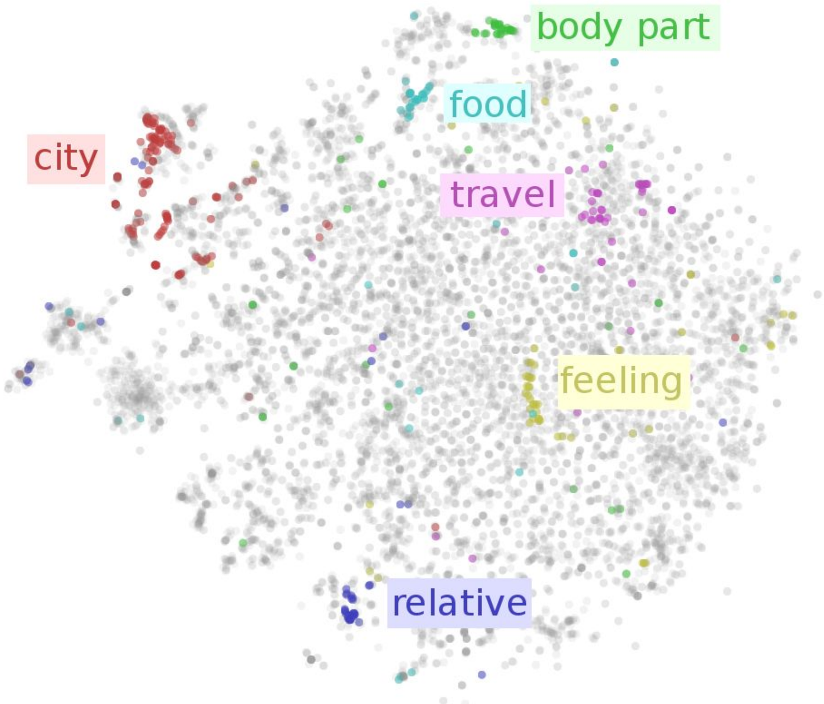
https://cdn-images-1.medium.com/max/2000/1*xsjuepBTKkBG1hr-ECpGKg.png



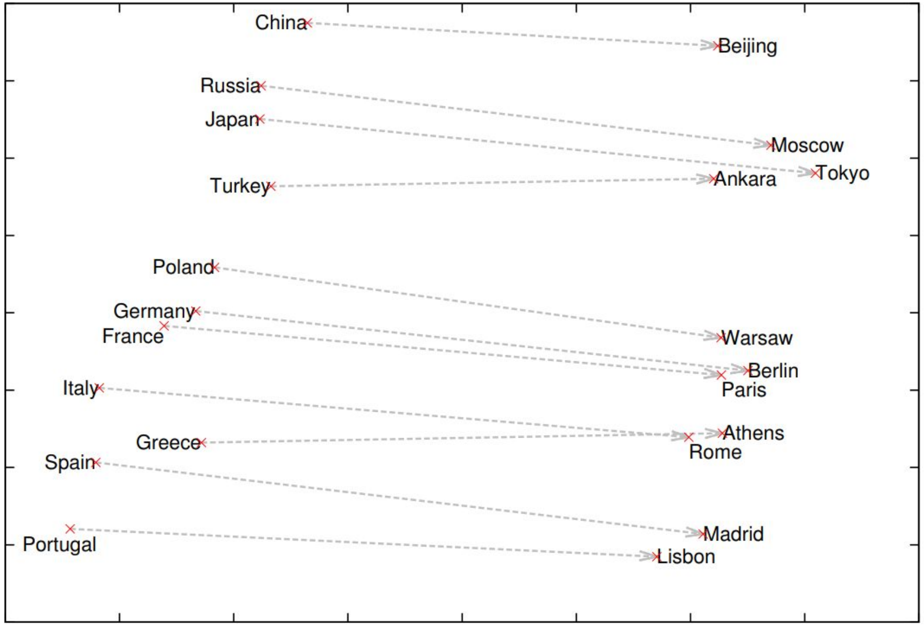
PCA Plot of Country Capital

Mikolov et al. (2013)

Visualizing Word Embeddings



T-SNE visualization of word embeddings
http://colah.github.io/posts/2015-01-Visualizing-Representations/



PCA Plot of Country Capital
Mikolov et al. (2013)

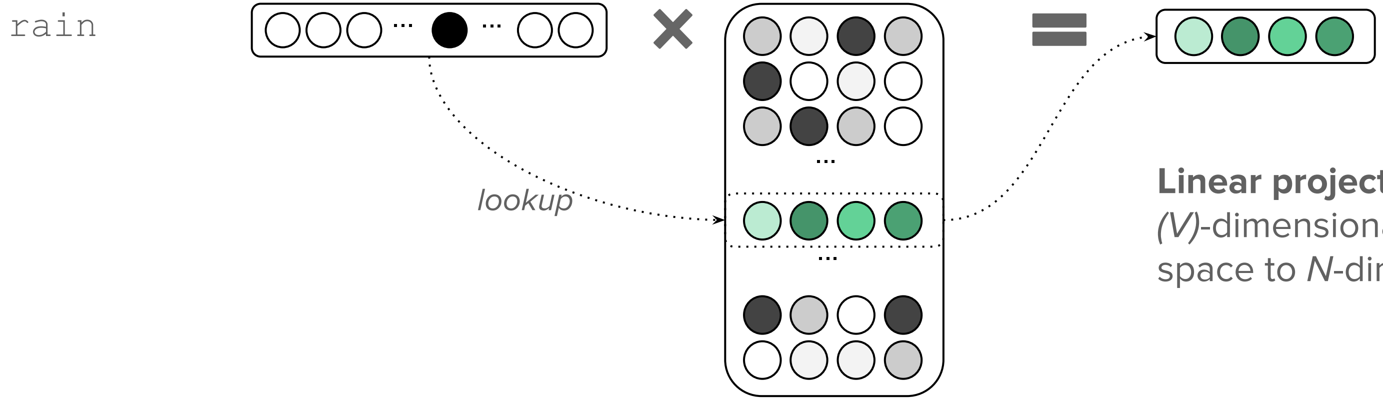
Interpretation as Linear Projection

Symbol →
word,
character, ...

One-hot vector
High-dimensional *(V)*
Sparse vector

Linear projection →

Word embedding
Low-dimensional *(N)*
dense vector



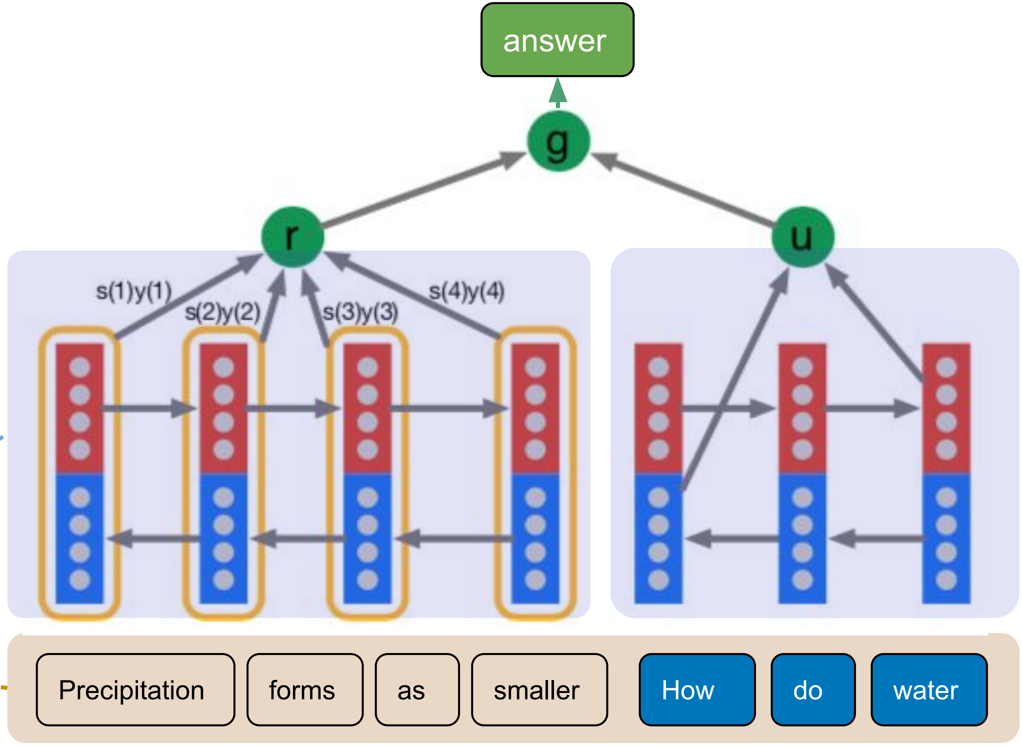
Word Embedding Matrix
[projection matrix]

Linear projections from *(V)*-dimensional discrete symbol space to *N*-dimensional

The Attentive Reader Model: Overview

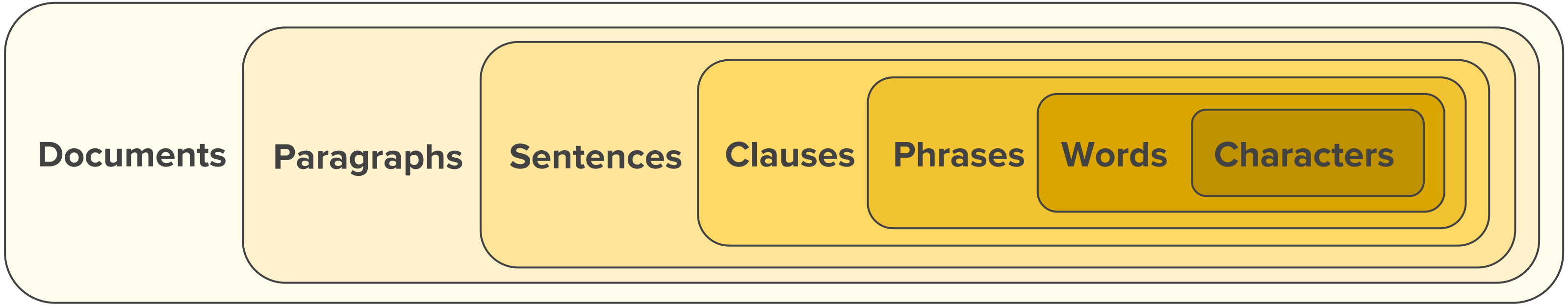
Composition: incorporating context around words

Input: Representing symbols as vectors



Modified visualization from Hermann et. al. 2015

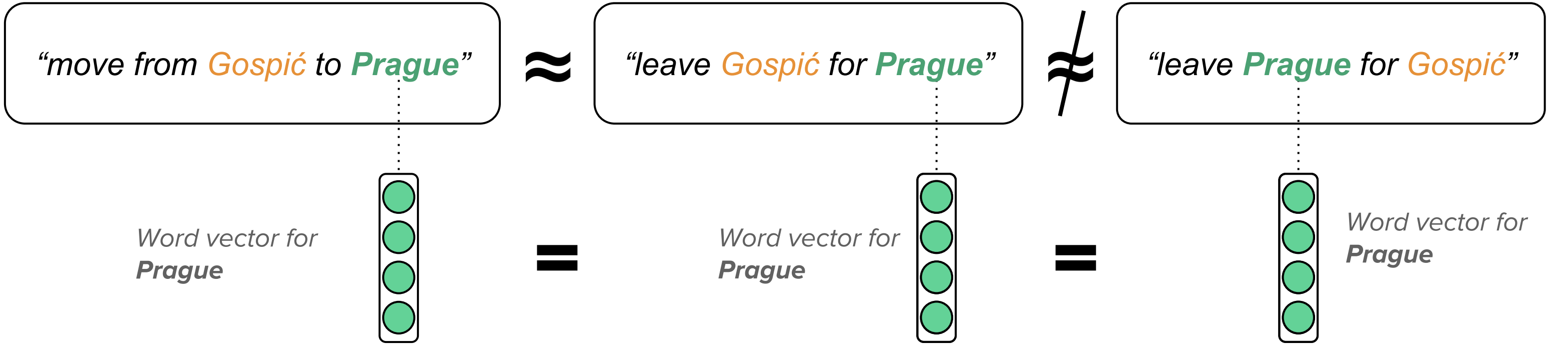
Language is Compositional



Challenges

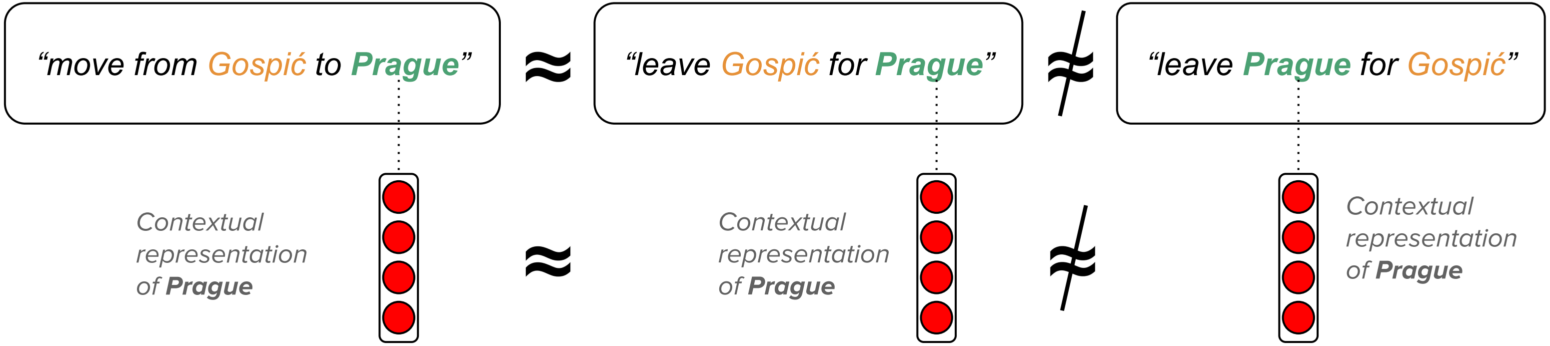
- Inductive bias: which composition function to use?
  - sequence, tree or more general graph structures?
  - Varies for different levels
- capturing long-range dependencies
  - co-reference (tracking entities)
  - effective information flow: ease of learning

Representing Words in Context



- Word representations should vary depending on context

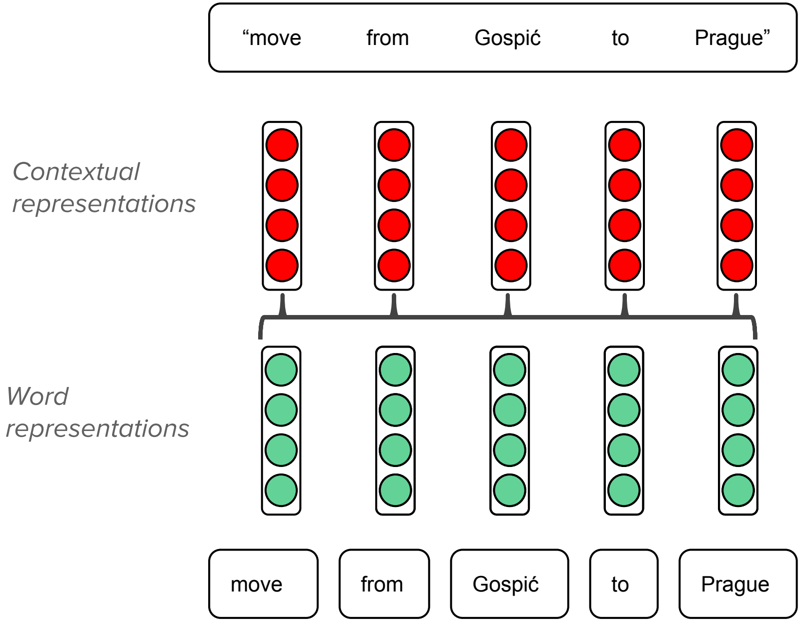
Representing Words in Context



- Word representations should vary depending on context
- **Contextual word representation:**
  - a word representation, computed conditionally on the given context

Representing Words in Context

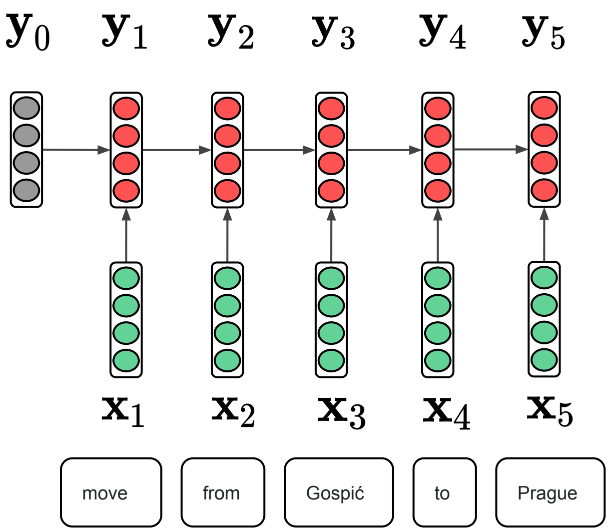
- composition of word vectors into contextualized word representations
- use vector composition function
    - different options



Recurrent Neural Network Layers

- **Idea**: text as sequence
- Prominent types: *LSTM, GRU*
- **Inductive bias:** Recency
  - more recent symbols have bigger impact on hidden state
- **Advantages**
  - everything is connected
  - easy to train and robust in practice
- **Disadvantages**
  - Slow ➔ computation time linear in length of text
  - not good for (very) long range dependencies

- *Good for:* sentences, small paragraphs



$$\mathbf{y}_t = f(\mathbf{x}_t, \mathbf{y}_{t-1})$$

Tree-variants:

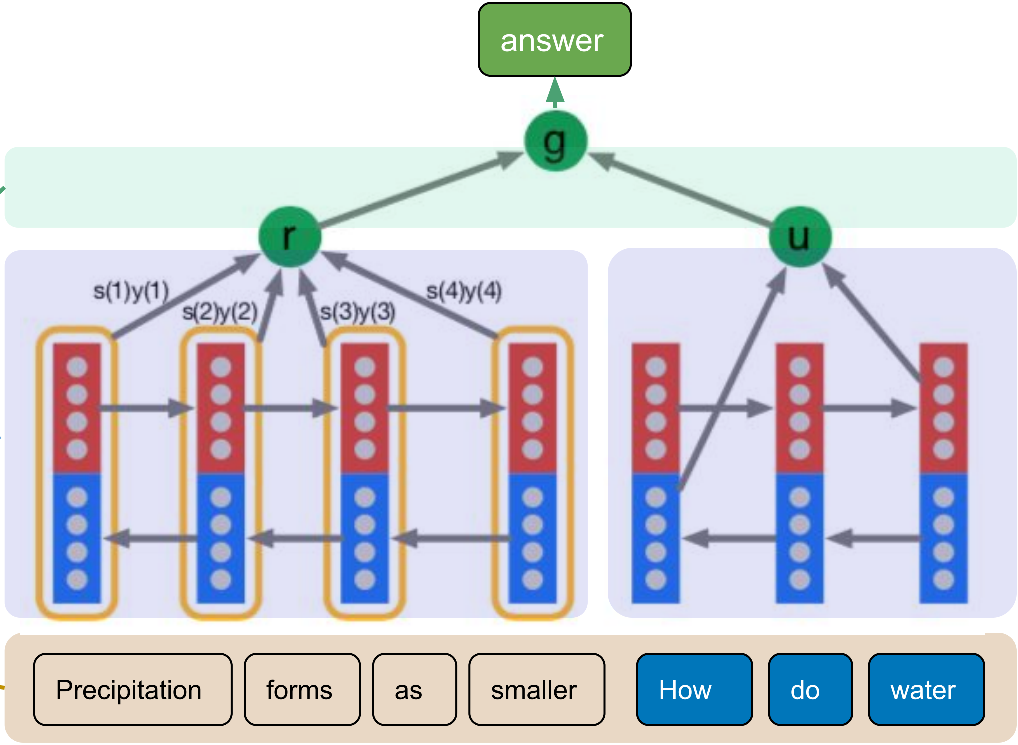
- TreeLSTM (Tai et al. 2015)
- RNN Grammars (Dyer et al. 2016)
- Bias towards syntactic hierarchy

The Attentive Reader Model: Overview

Sequence Interaction: Matching text with question

Composition: incorporating context around words

Input: Representing symbols as vectors



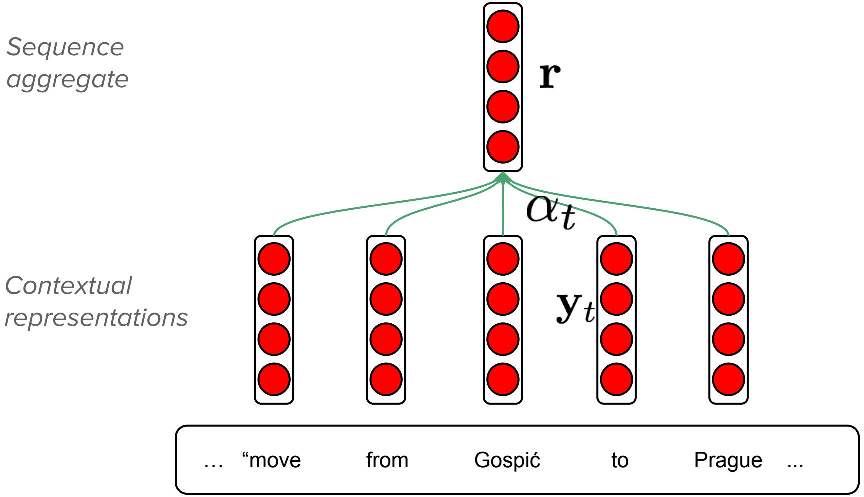
Modified visualization from Hermann et. al. 2015

Modelling Sequence Interactions

- **Why?** QA requires matching between question and text.
    - condition text representation on question (and vice versa)
- **“Naive approach”**: concatenation
    - append question after text, use RNN with longer sequence
- **Problem with naive approach:**
    - Long range dependencies: Many recurrent steps between answer and question ➔ dilution of signal

Modelling Sequence Interactions: Attention

- **Attention:**
    - relevance-weighted pooling of vectors across sequence
- attention mask computed can be conditional on question and text
- determines relevance of tokens for answering the question



f(q)

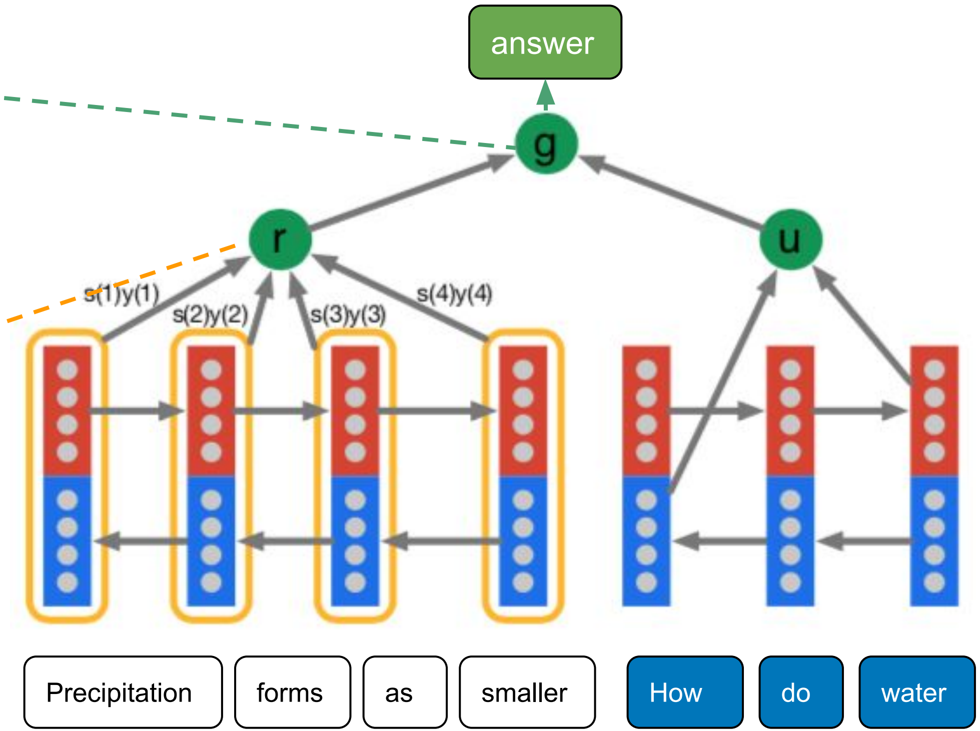
$$\mathbf{r} = \sum_{t=1}^{T} \alpha_t \mathbf{y}_t$$

$$\sum_{t=1}^{T} \alpha_t = 1; \quad \alpha_t \in [0, 1]$$

Modelling Sequence Interactions

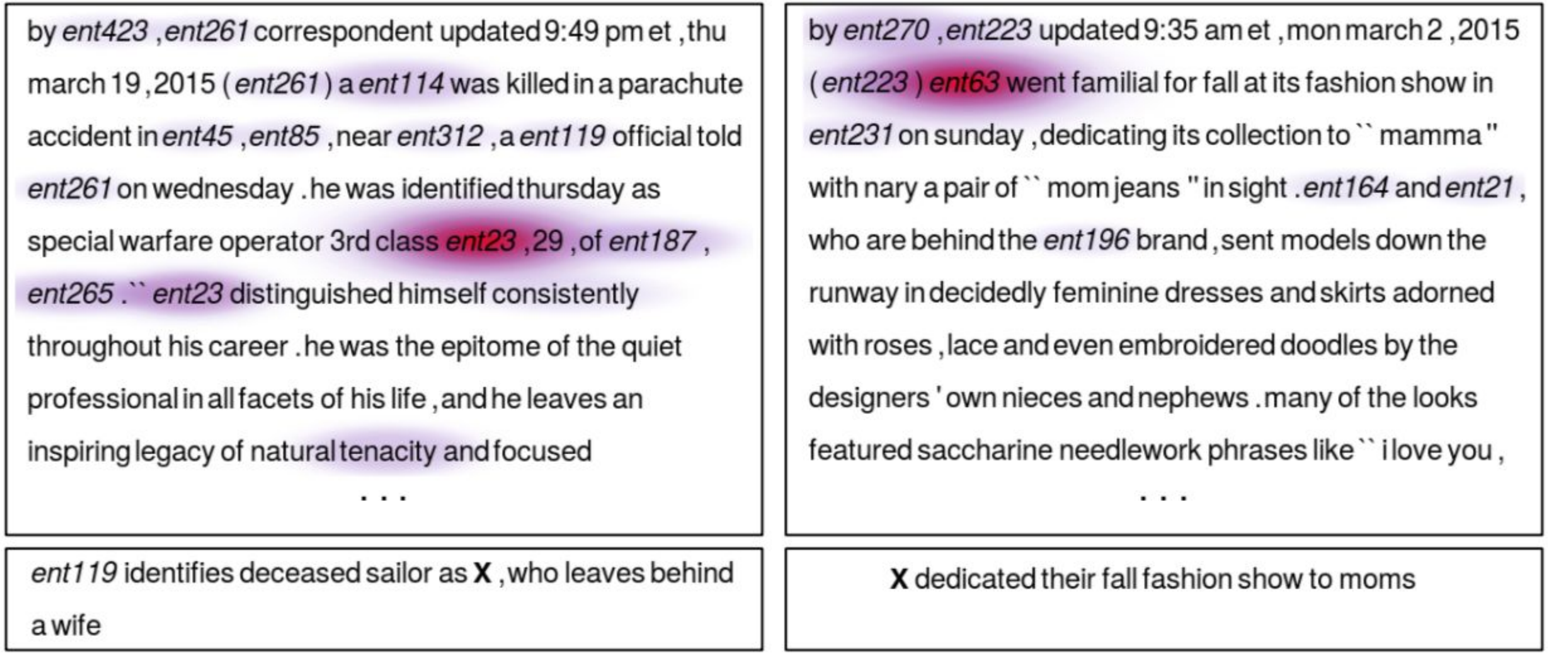
Combination of question and text representation

attention-weighted sum of contextualised word representations



Modified visualization from Hermann et. al. 2015

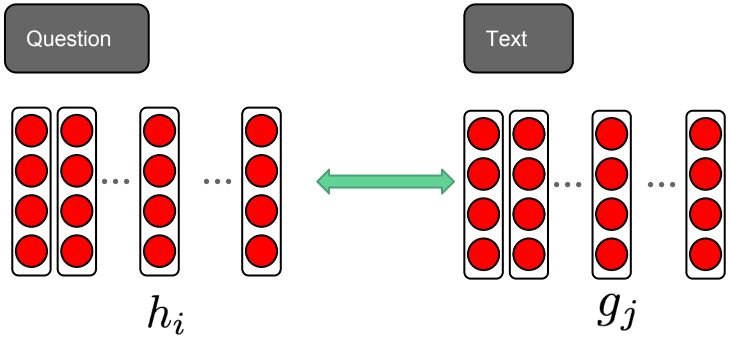
Example: Learned Attention Patterns



Intuition: Relevancy Masks

Visualization from Hermann et. al. 2015

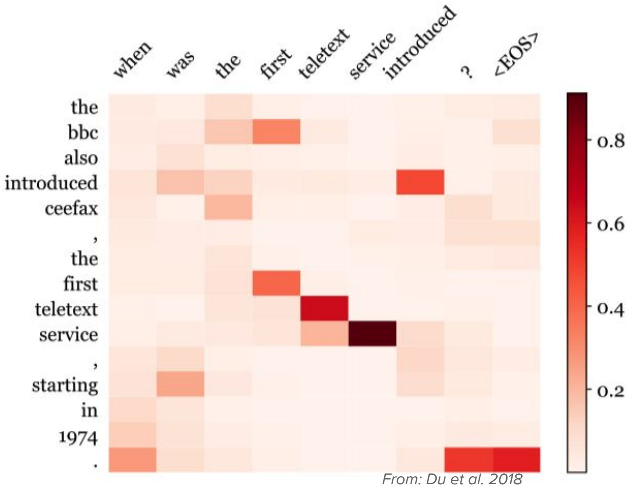
Modeling Sequence Interaction



“Naive” approach:

- **Goal in QA**: match question with text
- conditioning sequence representations **on one another**
  ➜ e.g., compute token-token attention masks from latent states
- Interpretation: per-word relevancy mask, (soft-)alignment

Modeling Sequence Interaction - Attention



From: Du et al. 2018

Word-to-word attention masks

e.g. $a_{ij} \propto \text{Bilinear}(h_i, g_j)$

- **Goal in QA**: match question with text
- conditioning sequence representations **on one another**
  ➜ e.g., compute token-token attention masks from latent states
- Interpretation: per-word relevancy mask, (soft-)alignment

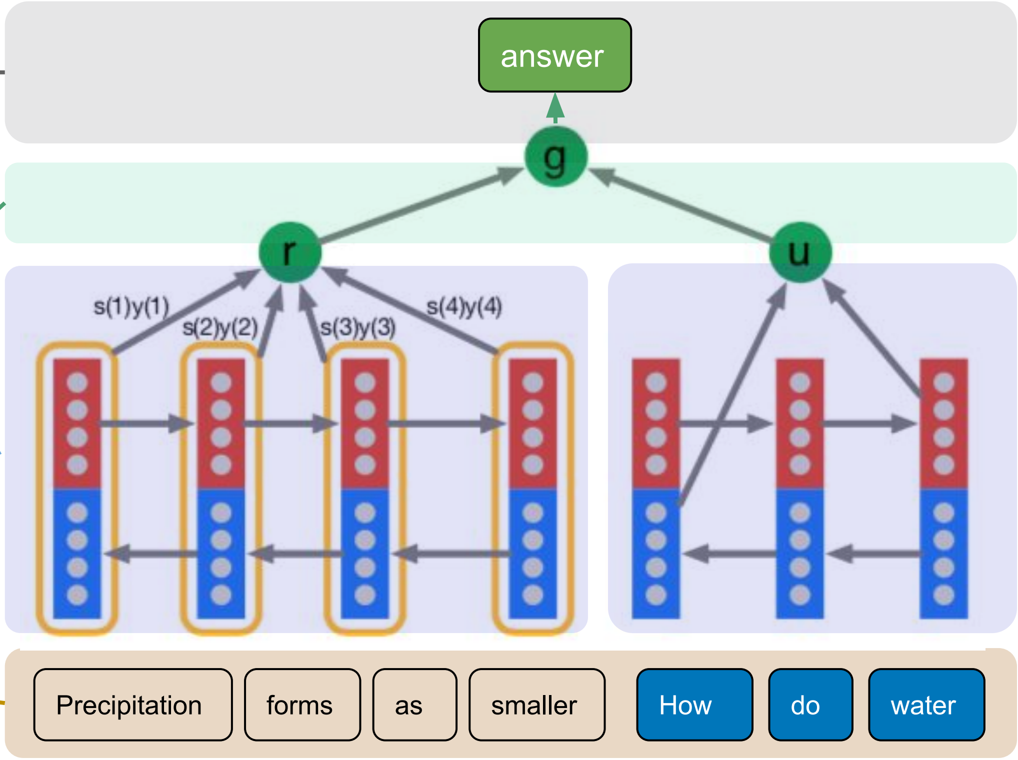
The Attentive Reader Model: Overview

Answer Selection:
answer prediction

Sequence Interaction:
Matching text with question

Composition: incorporating context around words

Input: Representing symbols as vectors



Modified visualization from Hermann et. al. 2015

Answer Prediction

- Linear projection
- **Probability distribution over different answer options**
  - spans in text -- distribution over positions for beginning and end
  - multiple choice: candidates
- **Training:** cross-entropy loss

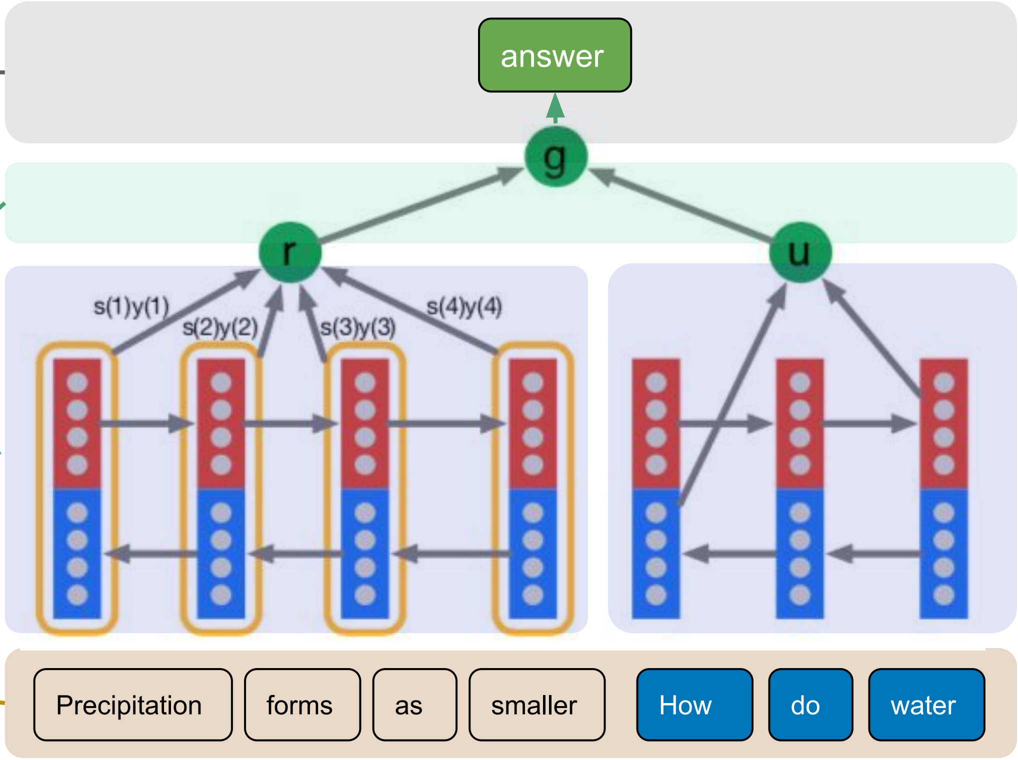
The Attentive Reader Model: Overview

Answer Selection: answer prediction

Sequence Interaction: Matching text with question

Composition: incorporating context around words

Input: Representing symbols as vectors



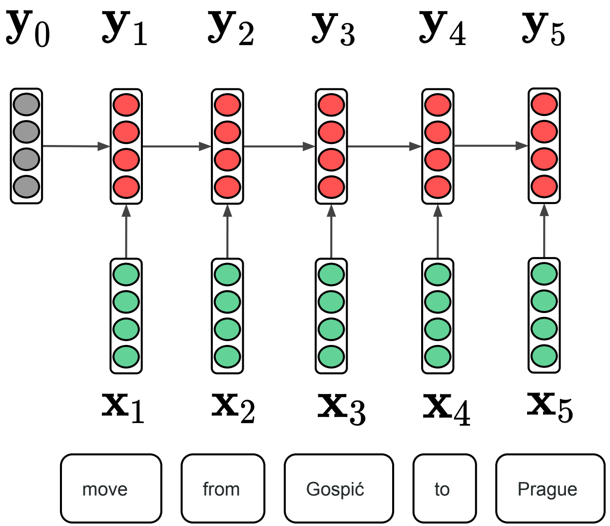
Modified visualization from Hermann et. al. 2015

Other Types of Composition Functions

Recurrent Neural Network Layers

- **Idea**: text as sequence
- Prominent types: *LSTM, GRU*
- **Inductive bias:** Recency
  - more recent symbols have bigger impact on hidden state
- **Advantages**
  - everything is connected
  - easy to train and robust in practice
- **Disadvantages**
  - Slow ➔ computation time linear in length of text
  - not good for (very) long range dependencies

- *Good for:* sentences, small paragraphs



$$\mathbf{y}_t = f(\mathbf{x}_t, \mathbf{y}_{t-1})$$

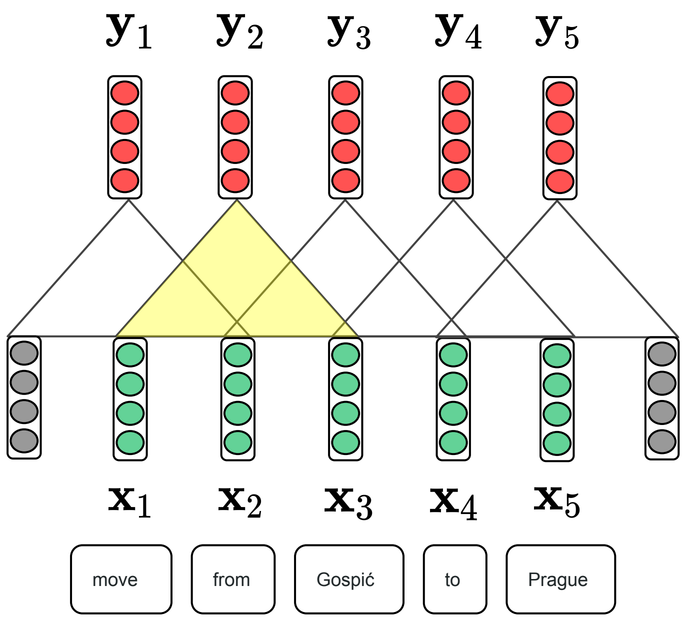
Tree-variants:

- TreeLSTM (Tai et al. 2015)
- RNN Grammars (Dyer et al. 2016)
- Bias towards syntactic hierarchy

Convolutional Layer

- **Idea**: text as collection of N-Grams
- **Inductive bias:** Locality
  - Only symbols within context window have impact on the current hidden state
- **Advantages**
  - Parallelizable and fast
- **Disadvantages**
  - Limited context window
  - remedy: stacking many layers + dilation

- *Good for:* Character-based word representations, phrases, multi-word representations

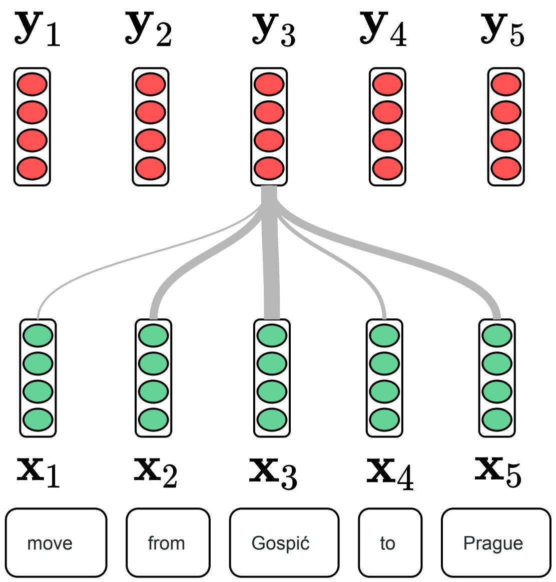


$$\mathbf{y}_t = f(\mathbf{x}_{t-k}, \ldots, \mathbf{x}_t, \ldots \mathbf{x}_{t+k})$$

See e.g.: Kim et al. 2016

Self-Attention Layer

- **Idea**: latent graph on text
- **Inductive bias:**
  - relationships between word pairs
- compute *K* separate weighted token representation(s) of the context for each token *t*
- **Advantages**
  - can capture long-range dependencies
  - Parallelizable and fast
- **Disadvantages**
  - careful setup of hyper-parameters
  - potentially memory intensive computation of attention weights for large contexts, *O(T * T * K)*
- *Good for:* phrases, sentences, paragraphs



$$\mathbf{y}_t = f(\mathbf{x}_1, \dots, \mathbf{x}_T)$$

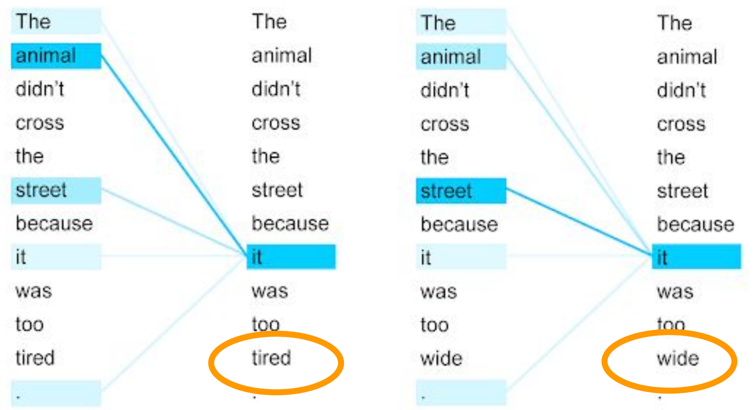
$$\tilde{\mathbf{x}}_t^k = \sum_{j=1}^{T} \alpha_{j,t}^k \mathbf{x}_j \qquad k = 1, \dots, K$$

$$f(\mathbf{x}_1, \dots, \mathbf{x}_T) = \text{nonlinear}(\tilde{\mathbf{x}}_t^1, \dots, \tilde{\mathbf{x}}_t^K)$$

α_t^k : k^{th} self-attention weights for token t

Self-Attention Layer

- **graph with weighted edges** of *K* types
- Can capture:
  - coreference chains
  - syntactic dependency structure in text
  - see for instance: Vaswani et al. 2017; Yang & Zhao et al. 2018



Transformer Self-Attention Coreference Visualization

https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html

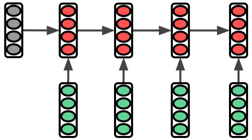
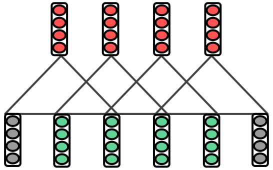
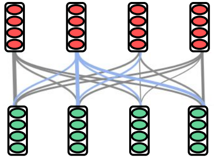
Self-Attention Layer

used in many SoTA MRC models, e.g.

- Language Modelling, Natural Language Inference: Cheng et al. 2016 (*intra-attention*)
- QA: Wang et al. 2017 (*self-matching*), Yu et al. 2018 (*self-attention)*

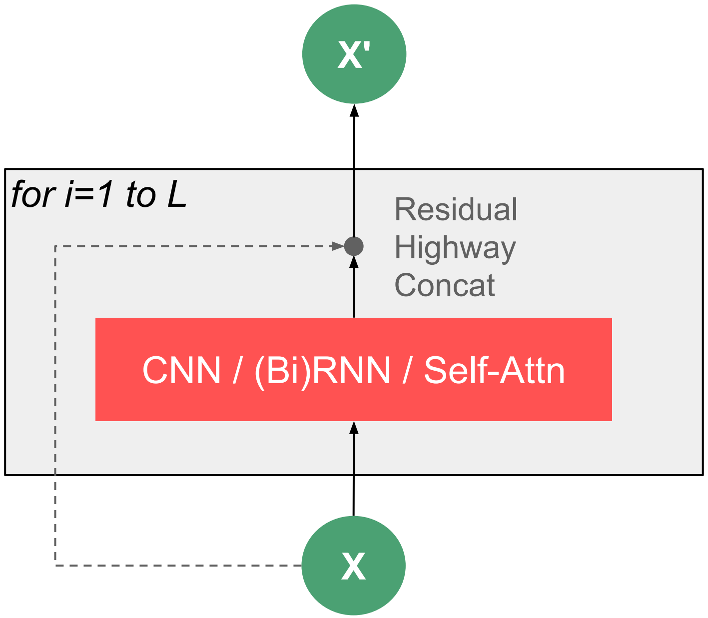
Compositional Sequence Encoders - Overview

- Language is compositional!
  - Characters ➔ Words ➔ Phrases ➔ Clauses ➔ Sentences ➔ Paragraphs ➔ Documents

Architecture	**RNN** (LSTM, GRU)	**CNN**	**Self-Attention**
Illustration			
Function $\mathbf{y}_t =$	$f(\mathbf{x}_t, \mathbf{y}_{t-1})$	$f(\mathbf{x}_{t-k}, \ldots, \mathbf{x}_{t+k})$	$f(\mathbf{x}_1, \ldots, \mathbf{x}_T)$
Advantages	- unlimited context - recency bias	- parallelizable ➔ fast - local n-gram patterns	- parallelizable ➔ fast - long-range dep
Disadvantages	- slow - strong recency bias - long-range dep	- limited context - strong locality bias - long-range dep	- harder to train - careful setup of hyper-parameters

Deep Compositional Sequence Encoders

- pure RNN based models usually not deep (typically L < 3)
  - Depth in RNNs comes naturally by processing sequentially
- CNN based models are quite deep
  - E.g. QANet: 42 CNN + 21 SelfAttn
  - use residual/highway layers or concatenation to avoid vanishing gradient
- Self-Attn. is usually applied after layers of CNN or RNN
  - exception: Transformer (Vaswani et al. 2017)

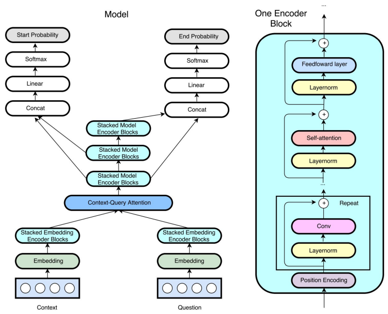


End-to-end Machine Reading for Question Answering

QANet, Yu et. al. 2018

State-of-the-Art Architecture

[...] 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”.



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

within a cloud

[Text] [Meaning] [Information Need]

QANet - A State-of-the-Art Architecture

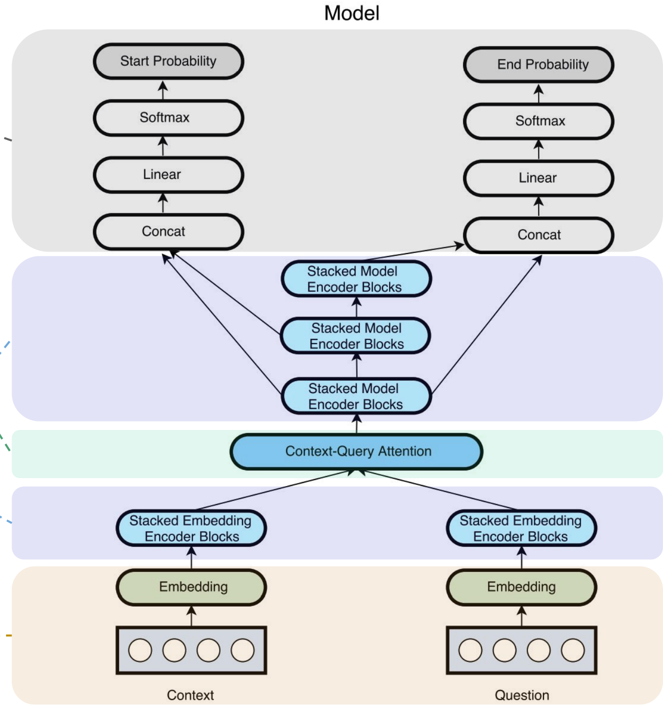
QANet, Yu et. al. 2018

Span Scoring:
answer prediction

Sequence Interaction:
Matching text with question

Composition: incorporating context around words

Input: Representing symbols as vectors



QANet - A State-of-the-Art Architecture

QANet, Yu et. al. 2018

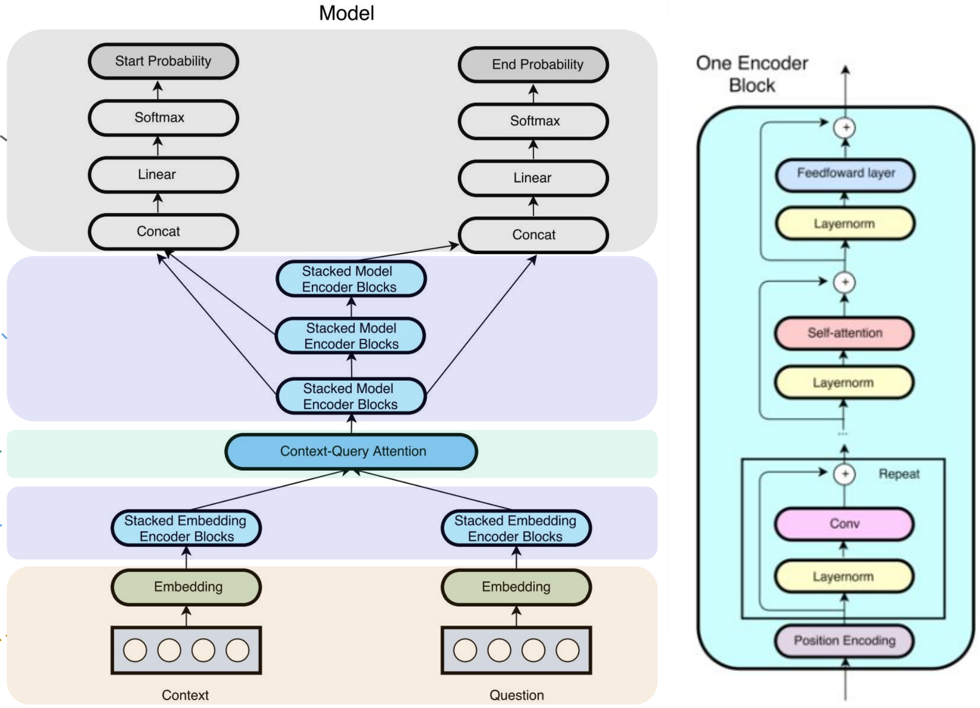
Span Scoring: linear projection, score for start and end position

Composition 2:
(2 * Conv + 1 * Self Attn) * 7 Blocks
= 21 Layers * 3 = **63** Layers

Sequence Interaction:
Bidirectional Attention

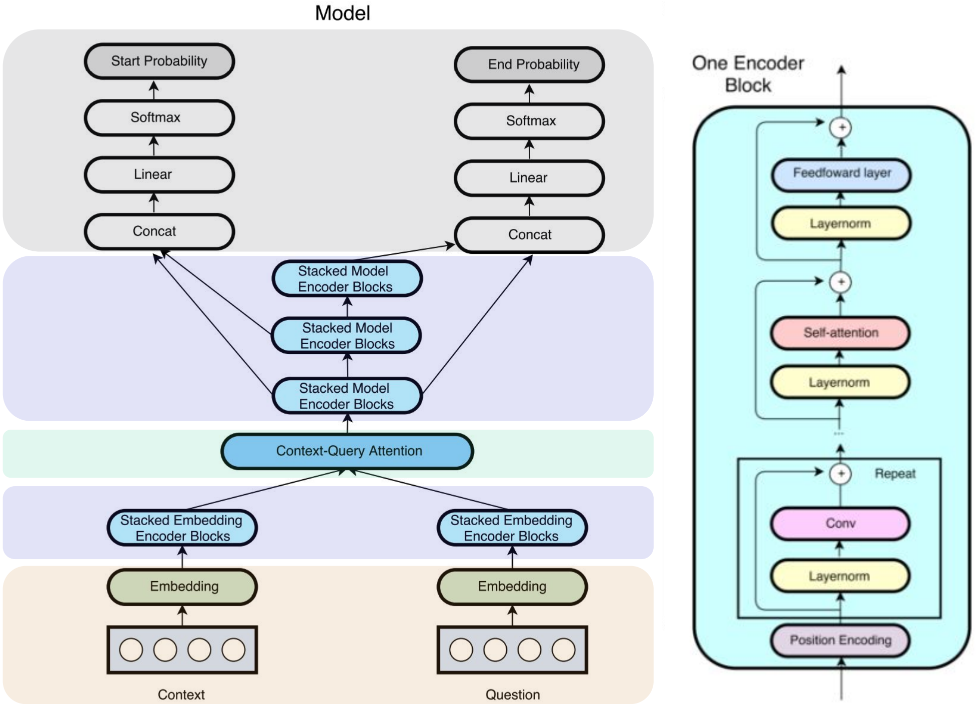
Composition 1:
(4 * Conv + 1 * Self Attn) = **5** Layers

Input: Representing symbols as vectors



QANet - A State-of-the-Art Architecture

- extremely deep
  - **68** compositional, residual layers
- but no RNNs
  - parallelizable and fast
- Currently best model on SQuAD
  - Self-attention
  - Data augmentation
  - Parallelizable ➜ faster training / tuning



References Compositional Sequence Encoders

- Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. NAACL.
- McCann, B., Bradbury, J., Xiong, C., & Socher, R. (2017). Learned in translation: Contextualized word vectors. NIPS.
- Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-Training. arXiv.
- Howard, J. & Ruder, S. (2018). Universal Language Model Fine-tuning for Text Classification. ACL.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. (2017). Attention is all you need. NIPS.
- Cheng, J., Dong, L., & Lapata, M. (2016). Long short-term memory-networks for machine reading. EMNLP.
- Wang, W., Yang, N., Wei, F., Chang, B., & Zhou, M. (2017). Gated self-matching networks for reading comprehension and question answering. ACL.
- Yu, A. W., Dohan, D., Luong, M. T., Zhao, R., Chen, K., Norouzi, M., & Le, Q. V. (2018). QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension. ICLR.
- Yang, Z., Zhao, J., Dhingra, B., He, K., Cohen, W. W., Salakhutdinov, R., & LeCun, Y. (2018). GLoMo: Unsupervisedly Learned Relational Graphs as Transferable Representations. arXiv.
- Tai, K. S., Socher, R., & Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. ACL.
- Dyer, C., Kuncoro, A., Ballesteros, M., & Smith, N. A. (2016). Recurrent Neural Network Grammars. NAACL.

References Interaction

- Cho, K., Gulcehre, B. V. M. C., Bahdanau, D., Schwenk, F. B. H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. EMNLP.
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. NIPS.
- Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. ICLR.
- Sukhbaatar, S., Weston, J., & Fergus, R. (2015). End-to-end memory networks. NIPS.
- Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., ... & Socher, R. (2016). Ask me anything: Dynamic memory networks for natural language processing. ICML.
- Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., ... & Badia, A. P. (2016). Hybrid computing using a neural network with dynamic external memory. Nature
- Grefenstette, E., Hermann, K. M., Suleyman, M., & Blunsom, P. (2015). NIPS.
- Henaff, M., Weston, J., Szlam, A., Bordes, A., & LeCun, Y. (2017). Tracking the world state with recurrent entity networks. ICLR.
- Rocktäschel, T., Grefenstette, E., Hermann, K. M., Kočiský, T., & Blunsom, P. (2016). Reasoning about entailment with neural attention. ICLR.
- Yu, A. W., Dohan, D., Luong, M. T., Zhao, R., Chen, K., Norouzi, M., & Le, Q. V. (2018). QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension. ICLR.

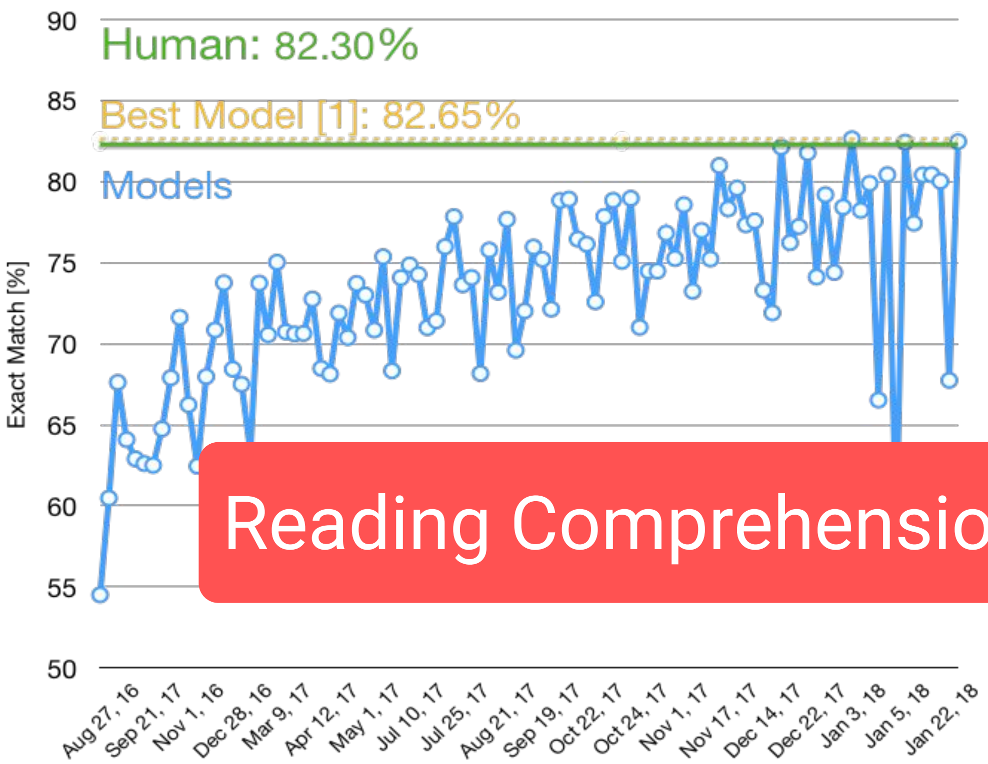
Conclusion

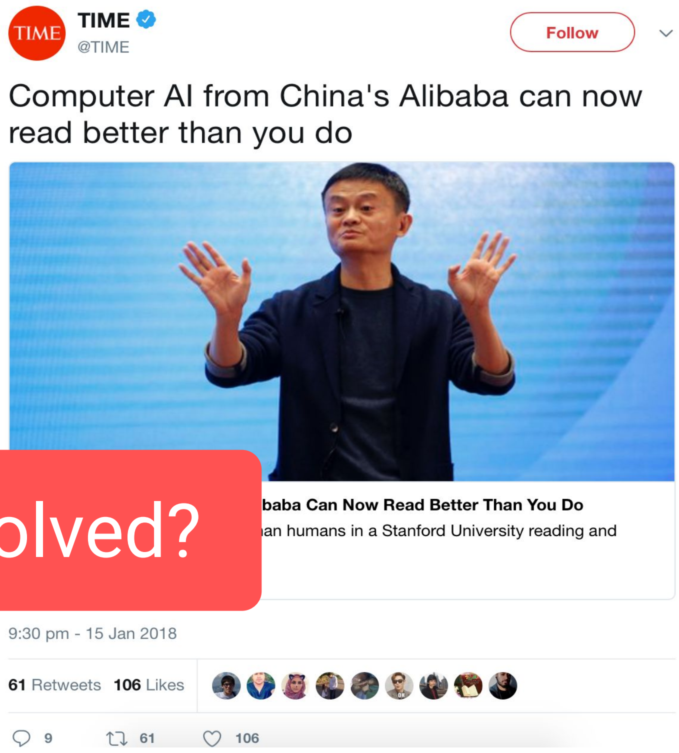
We gathered all ingredients to build state-of-the-art supervised MRQA systems!

- We know about:
  - Representing words with and without context
  - Modeling compositionality
  - Modeling sequence interaction (question-paragraph)
  - Answer questions by pointing to the start and end of the answer-span
- architectures work well in practice
  … as long as we stay in-domain and questions are simple

Trends & Open Problems

Progression of SQuAD Model Performance





Reading Comprehension Solved?

QA System Demo

Where RC models work well today

- question is answerable
- relevant paragraph / text is given
- relevant paragraph not too long
- inferring answer is not too complex
- Pattern matching / soft text alignment between question and text
- same domain during training and test time

Upon closer look...

What is the name of the quarterback who was 38 in Super Bowl XXXIII?

The past record was held by quarterback John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38.

Upon closer look...

What is the name of the quarterback who was 38 in Super Bowl XXXIII?

John Elway

The past record was held by quarterback **John Elway**, who led the Broncos to victory in Super Bowl XXXIII at age 38.

Upon closer look...

What is the name of the quarterback who was 38 in Super Bowl XXXIII?

The past record was held by quarterback John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38. Quarterback Jeff Dean had a jersey number 37 in Champ Bowl XXXIV.

Upon closer look...

What is the name of the quarterback who was 38 in Super Bowl XXXIII?

Jeff Dean

The past record was held by quarterback John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38. Quarterback **Jeff Dean** had a jersey number 37 in Champ Bowl XXXIV.

Upon closer look...

What is the name of the quarterback who was 38 in Super Bowl XXXIII?

Jeff Dean

The past record was held by quarterback John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38. Quarterback **Jeff Dean** had a jersey number 37 in Champ Bowl XXXIV.

- Reading Comprehension models can easily be fooled by adding adversarial sentences (Jia et al. 2017)

Is all this model complexity necessary?

- Simpler model (BiLSTM + word-in-question feature) still competitive on SQuAD (Weissenborn et al., 2017)
- Simple and complex models break

Should we rather:

- build model architectures more carefully?
- think more carefully about our training data?

Trends & Open Problems

Directions for Improving Robustness

Solvability

Can the question actually be answered? (Rajpurkar et al. 2018)

What was the name of the 1937 treaty?

[UNANSWERABLE]

… Other legislation followed, including the Migratory Bird Conservation Act of 1929, a 1937 treaty prohibiting the hunting of right and gray whales, and the Bald Eagle Protection Act of 1940.

From: Rajpurkar et al. 2018

Solvability

Can the question actually be answered? (Rajpurkar et al. 2018)

What was the name of the 1937 treaty?

[UNANSWERABLE]

… Other legislation followed, including the Migratory Bird Conservation Act of 1929, a 1937 treaty prohibiting the hunting of right and gray whales, and the Bald Eagle Protection Act of 1940.

System	SQuAD 1.1 test EM	SQuAD 1.1 test F1	SQuAD 2.0 dev EM	SQuAD 2.0 dev F1	SQuAD 2.0 test EM	SQuAD 2.0 test F1
BNA	68.0	77.3	59.8	62.6	59.2	62.1
DocQA	72.1	81.0	61.9	64.8	59.3	62.3
DocQA + ELMo	**78.6**	**85.8**	**65.1**	**67.6**	**63.4**	**66.3**
Human	82.3	91.2	86.3	89.0	86.9	89.5
Human–Machine Gap	3.7	5.4	**21.2**	**21.4**	**23.5**	**23.2**

From: Rajpurkar et al. 2018

Adversarial Examples for Training / Regularization

- Make models adhere to higher-level rules
- What are these rules, how can we formulate / integrate them?
  - Appending Sentences + KB rules (Jia et al. 2017)
  - Erasing words (Li et al. 2017)
  - Character flips (Ebrahimi et al. 2018)
  - Paraphrases (Iyyer et al. 2018)
  - Semantic equivalence (Ribeiro et al. 2018)
  - KB rules (Minervini et al. 2018)

Data augmentation

Adversarial regularisation

Model Diagnostics: Right for the Wrong Reason?

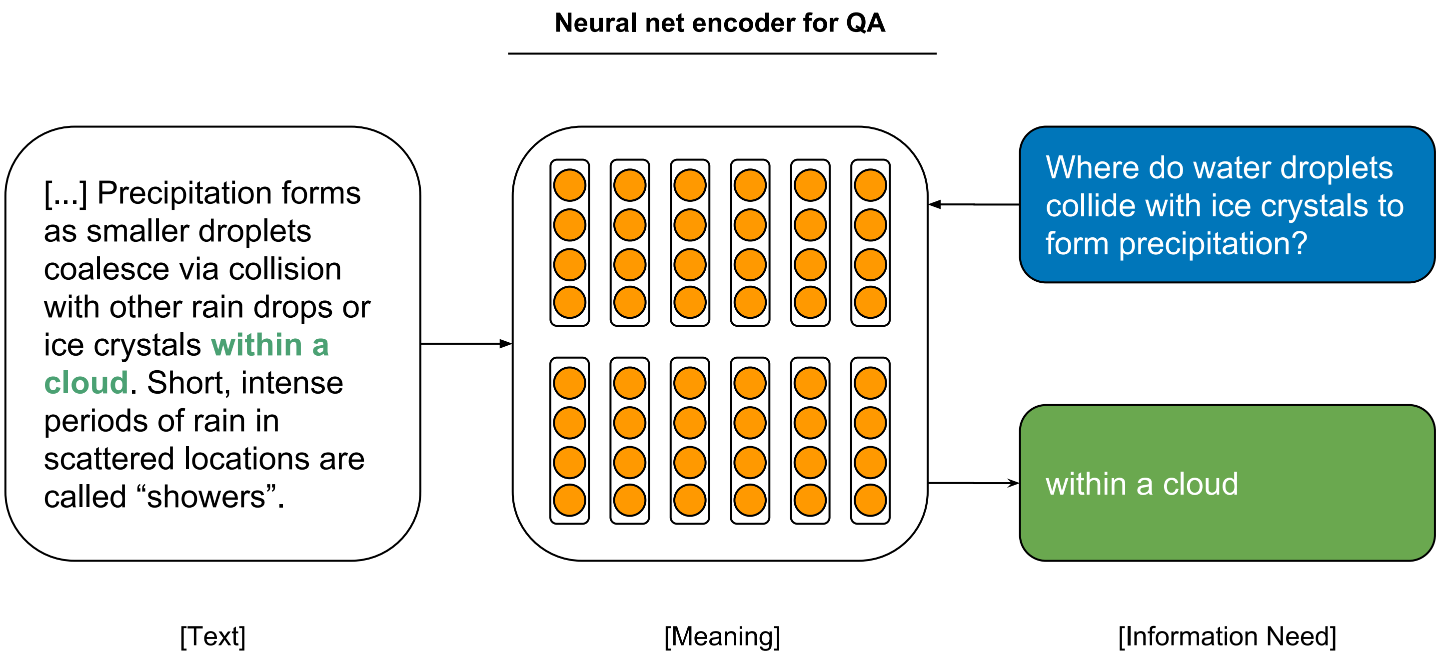
- What do models rely on to form predictions?
  - Analysing sensitivity to input: Ribeiro et al. (2016), Alvarez-Melis and Jaakkola (2017)
- Example: Anchors (Ribeiro et al. 2018)
  - Finding a minimal set of sufficient conditions to make a prediction

Anchor

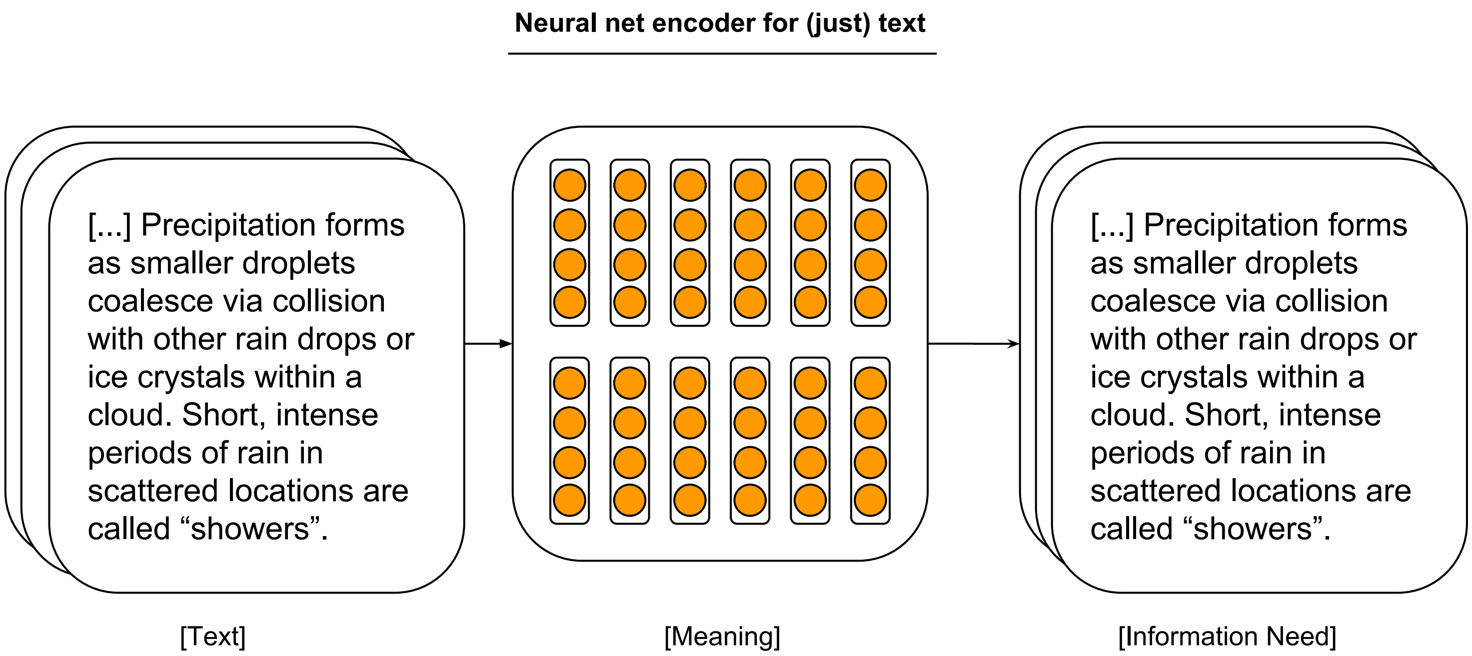
What is the mustache made of?	banana
What is the ground made of ?	banana
What is the bed made of ?	banana
What is this mustache ?	banana
What is the man made of?	banana
What is the picture of ?	banana

How **many** bananas are in the picture?	2
How **many** are in the picture?	2
many animals the picture ?	2
How **many** people are in the picture ?	2
How **many** zebras are in the picture ?	2
How **many** planes are on the picture ?	2

Pretraining Representations

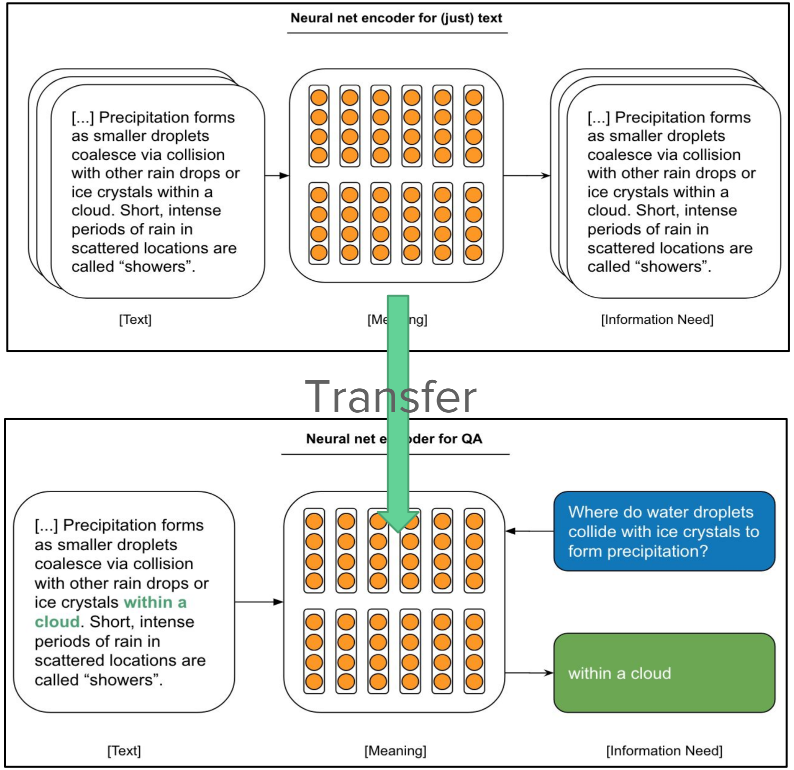


Pretraining Representations



Lifting over Pretrained Representations

Pretrained Language Model



Document QA

Pretrained Sequence Encoders

… improve NLU tasks significantly!

- ELMo, *Peters et al. 2018. NAACL (Best Paper)*
    - pre-trained bi-directional LSTM language model
    - SQuAD (+4%), SRL (+3%), SNLI (+1.5%)
- Transformer LM, *Radford et al. 2018. arXiv.*
    - pre-trained language model based on pure self-attention (Vaswani et al., 2017)
- ULMFit, *Howard & Ruder 2018. ACL.*
    - pre-trained language model, fine-tuning on classification tasks
- CoVE, *McCann et al. 2017. NIPS.*
    - pre-trained LSTM encoder from Machine Translation
- Conneau et al. 2017
    - Pre-trained representations from Natural Language Inference

Other tasks?

How is this different from pretrained word embeddings?

Pretrained <u>Word</u> Embeddings (word2vec)

- Predicting co-occurring of words
- Independent of other context

Pretrained Contextualized Embeddings (e.g. ELMo)

- Predicting whole text (using LSTM, or Self-Att.)
- Full dependence on other context

Summary: Directions for Improving Model Robustness

- Task Refinement: being more precise in what to learn
- Diagnostics: shedding insight into model failure modes
- Adversarial training / regularization
- Better prior models for contextualised representations

Trends & Open Problems

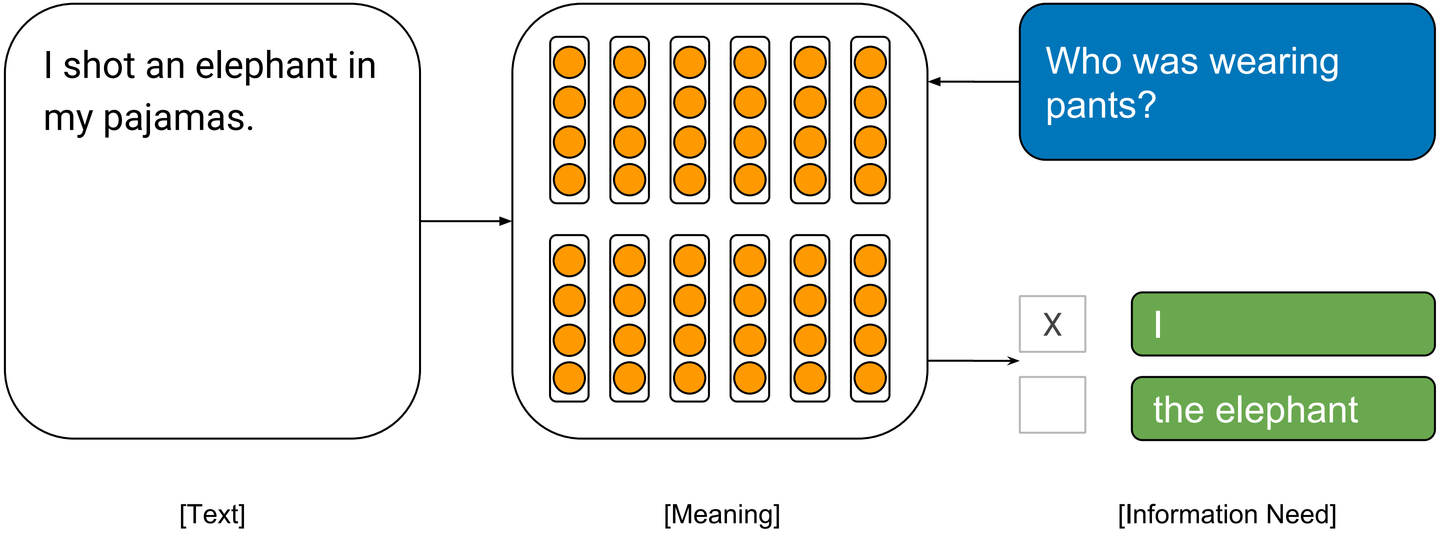
Other Challenges

Open Challenges I: Limited Supervision

- strong results with large annotated training sets
- How about smaller datasets?
    - Ideally: shift from 100K to 1K training points
    - less costly, large-scale annotation
- Approaches:
    - domain adaptation, e.g. Wiese et al. (2017)
    - Synthetic data generation, e.g. Dhingra et al. (2018)
    - transfer learning, e.g. Mihaylov et al. (2017)
    - (un?-)supervised pretraining, e.g. ELMo, Peters et al. (2018)

Challenge II: Integrating Background Knowledge

Missing context / background knowledge



Challenge II: Integrating Background Knowledge

- Sources of common sense knowledge
  - Encyclopedic descriptions (Hill et al. 2016, Bahdanau et al. 2018)
  - Knowledge Bases (Yang and Mitchell 2017, Weissenborn et al. 2017, Mihaylov and Frank, 2018)
- Example: Weissenborn et al. (2017):
  - condition context representations also on additional facts
  - Intuition: new background facts provide additional features
    ➡ refined vector representations

Challenge III: Integration of MR with Vision

- Example: Visual QA
- End-to-end trainable encoders for questions, text

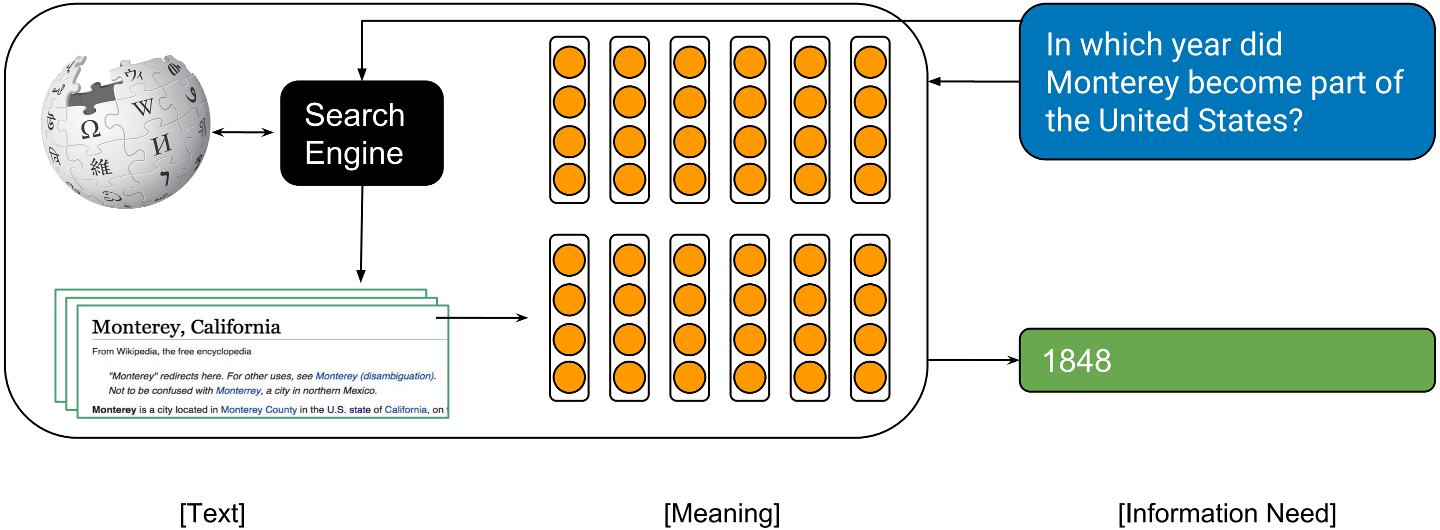
Question Complexity

Task	Question
Object Recognition	What is in the image?
Segmentation	Where are the things in the image?
Scene / Attribute Classification	What color is the dress?
Visual QA	Arbitrary Question

From: Goyal et al. (2017)

Challenge IV: End-to-End Machine Reading at Scale

Open-domain Question Answering, e.g. Chen et al. (2017)



Challenge V: Reconciling Conflicting Information

So how much does the UK pay to the EU per week?

"Once we have settled our accounts, we will take back control of roughly **£350m** per week." *Boris Johnson*

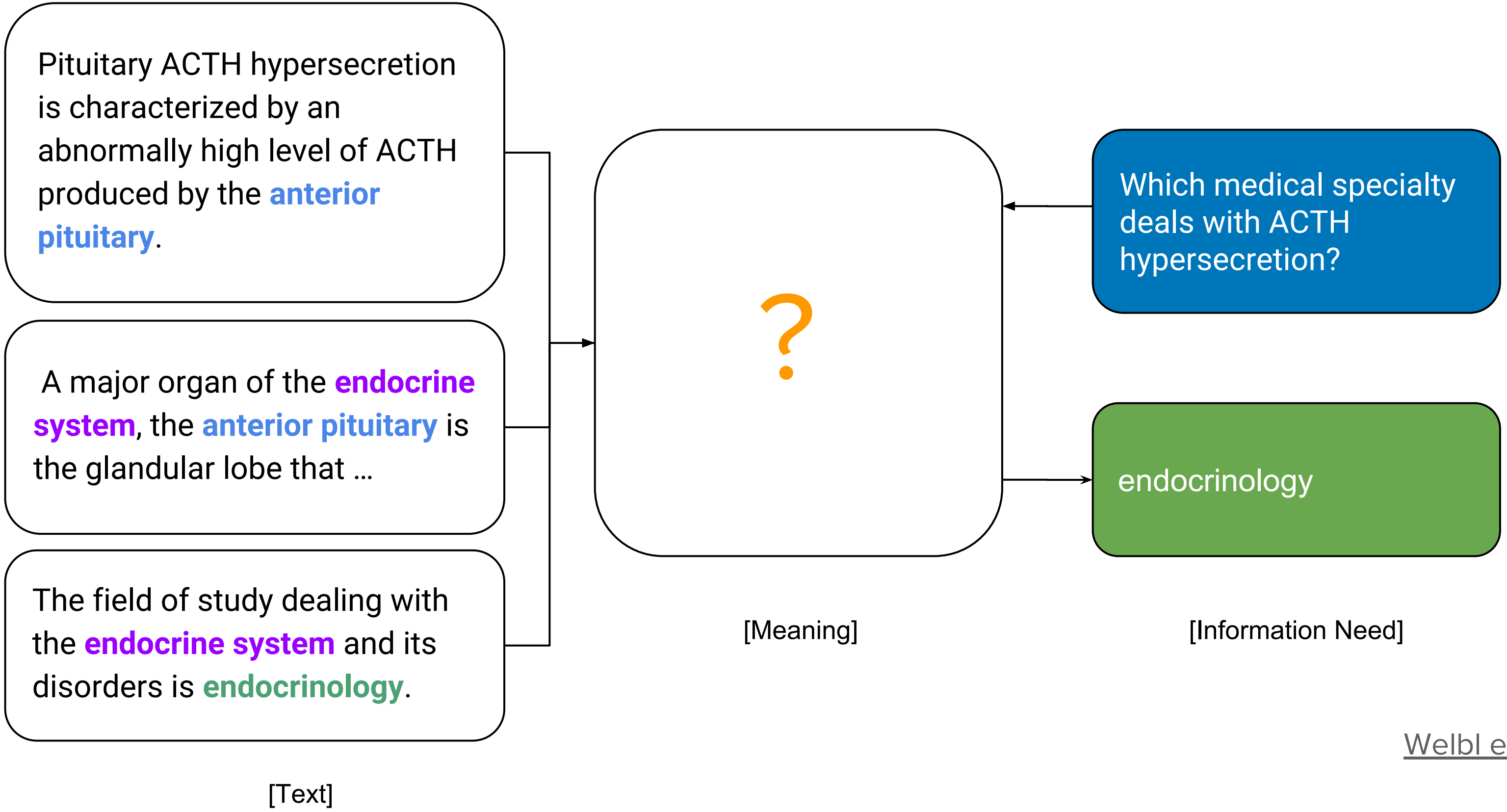
"We are not giving £20bn a year or £350m a week to Brussels - Britain pays **£276m** a week to the EU budget because of the rebate." *BBC Reality Check*

"...When those are taken into account the figure is **£250m**." *Independent*

?

Trust into source, timeline, ...

Challenge VI: Reasoning with Text



Welbl et al. (2018)

Challenge VI: Reasoning with Text

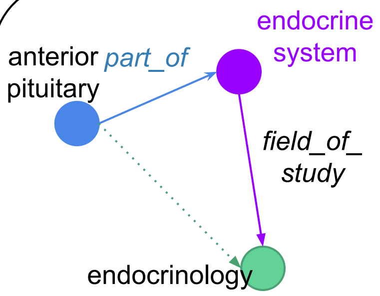
Pituitary ACTH hypersecretion is characterized by an abnormally high level of ACTH produced by the **anterior pituitary**.

A major organ of the **endocrine system**, the **anterior pituitary** is the glandular lobe that …

The field of study dealing with the **endocrine system** and its disorders is **endocrinology**.

[Text]

Reasoning with Structured Knowledge



part_of(A,B) ∧
lfield_of_study(B,C)
→ *medical_specialty*(A,C)

[Meaning]

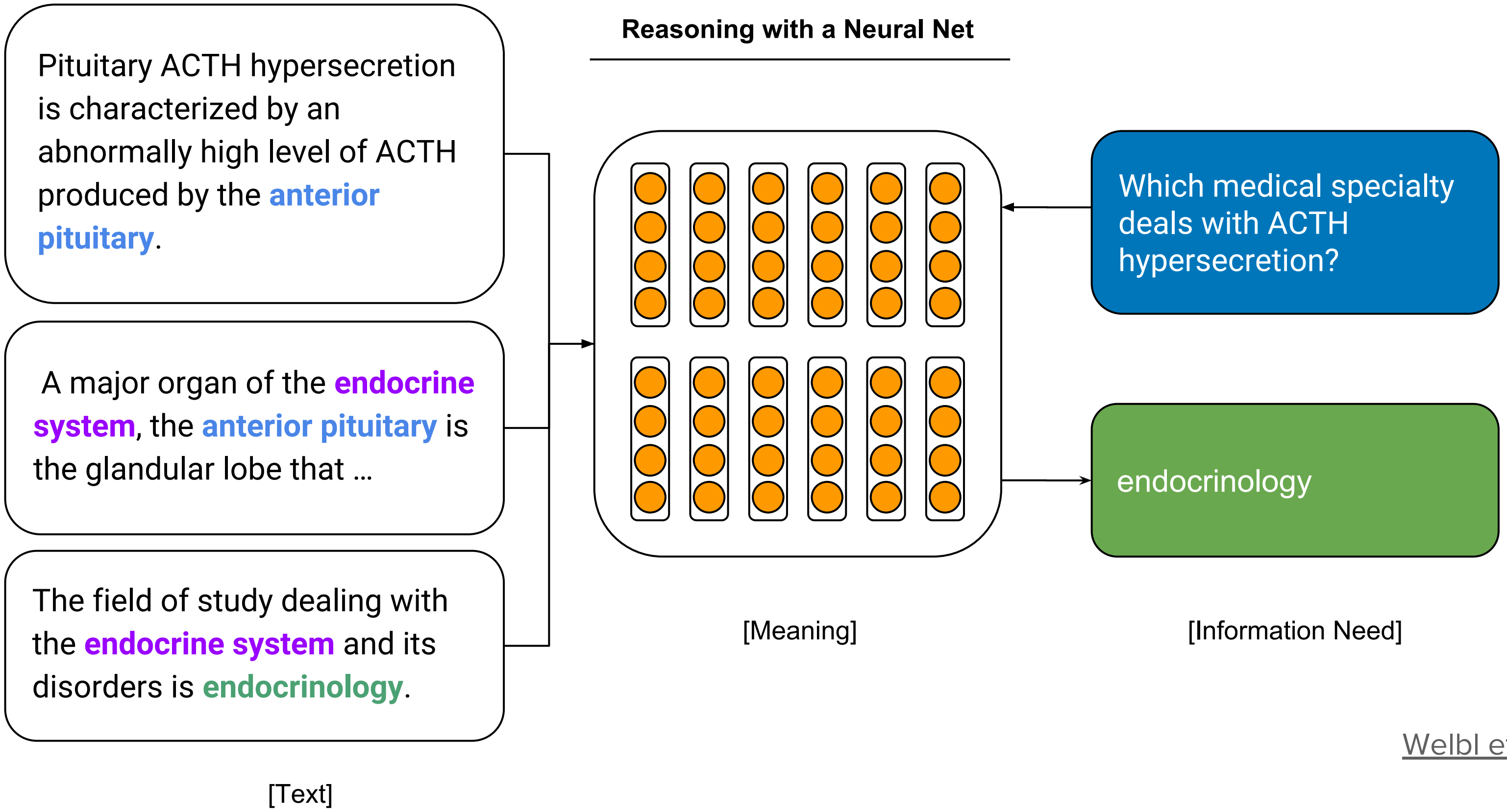
Which medical specialty deals with ACTH hypersecretion?

endocrinology

[Information Need]

Welbl et al. (2018)

Challenge VI: Reasoning with Text



Welbl et al. (2018)

Challenge VI: Reasoning with Text

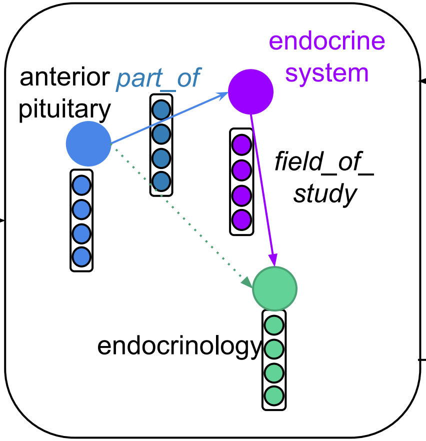
Pituitary ACTH hypersecretion is characterized by an abnormally high level of ACTH produced by the **anterior pituitary**.

A major organ of the **endocrine system**, the **anterior pituitary** is the glandular lobe that …

The field of study dealing with the **endocrine system** and its disorders is **endocrinology**.

[Text]

Neural Reasoning with Structured Knowledge



[Meaning]

Which medical specialty deals with ACTH hypersecretion?

endocrinology

[Information Need]

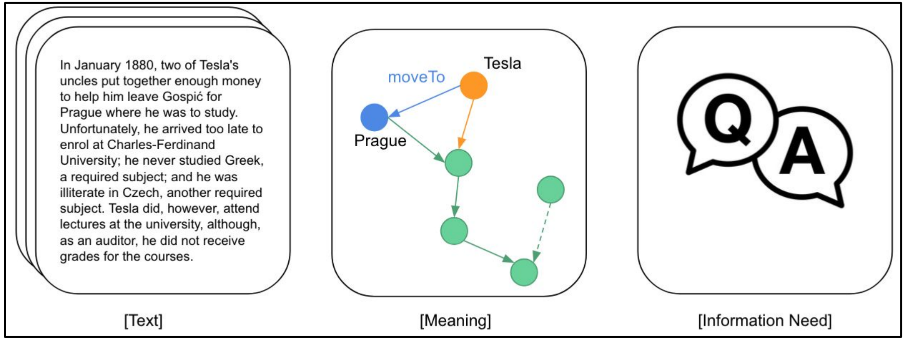
Welbl et al. (2018)

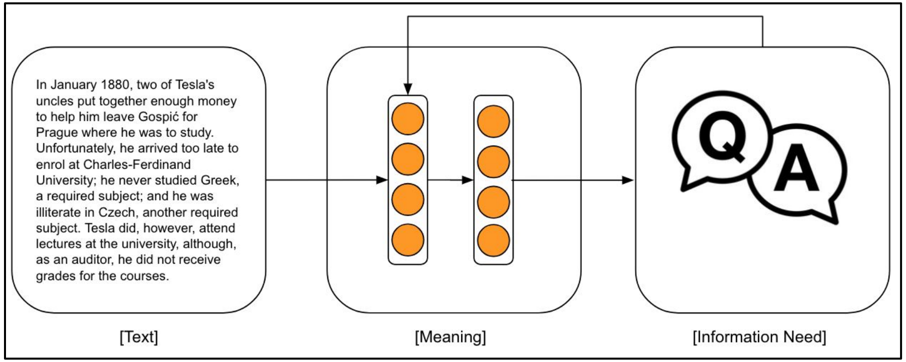
Conclusion

A Paradigm Shift

- Symbolic Meaning Representations
- ➡ Latent Vector Representations

- Feature Engineering & Domain Expertise
- ➡ Architecture Engineering & ML/DL Expertise

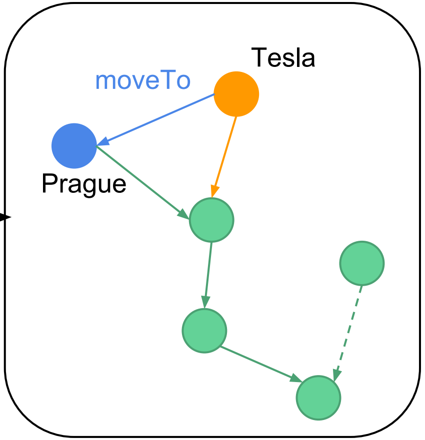




Automatic Knowledge Base Construction

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study. Unfortunately, he arrived too late to enrol at Charles-Ferdinand University; he never studied Greek, a required subject; and he was illiterate in Czech, another required subject. Tesla did, however, attend lectures at the university, although, as an auditor, he did not receive grades for the courses.

[Text]



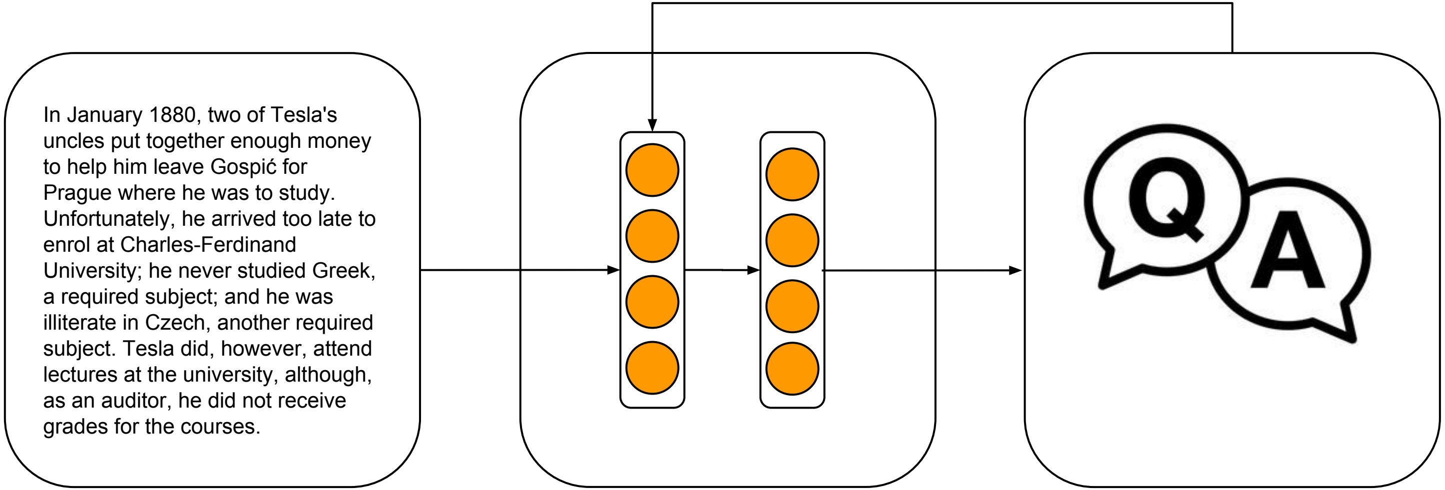
[Meaning]

[Information Need]

Structured Representations

- Advantages
  - Fast access
  - Scalable
  - Interpretable
  - Supports reasoning
  - Universality of representations: independent of question
- Disadvantages
  - Less robust to variation in language
  - Cascading errors
  - Schema engineering
  - Annotation requires experts

End-to-End Machine Reading

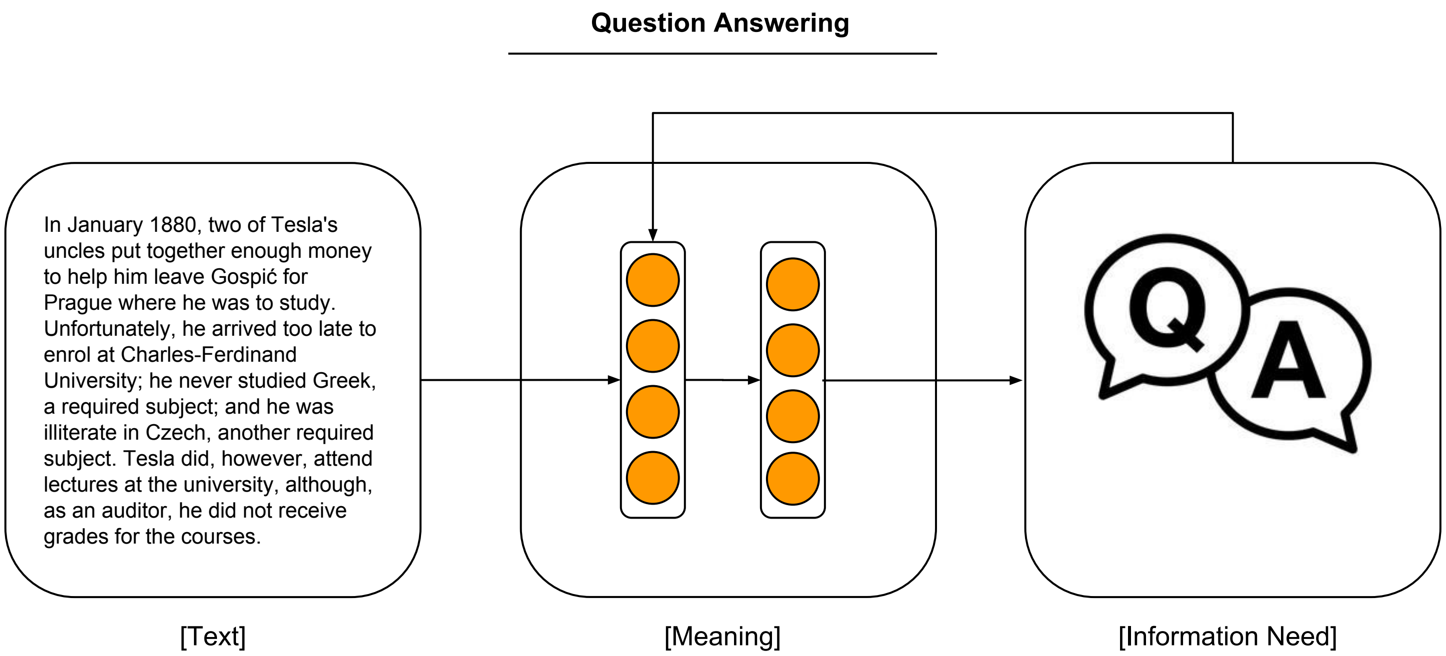


[Text] [Meaning] [Information Need]

Distributed Representations

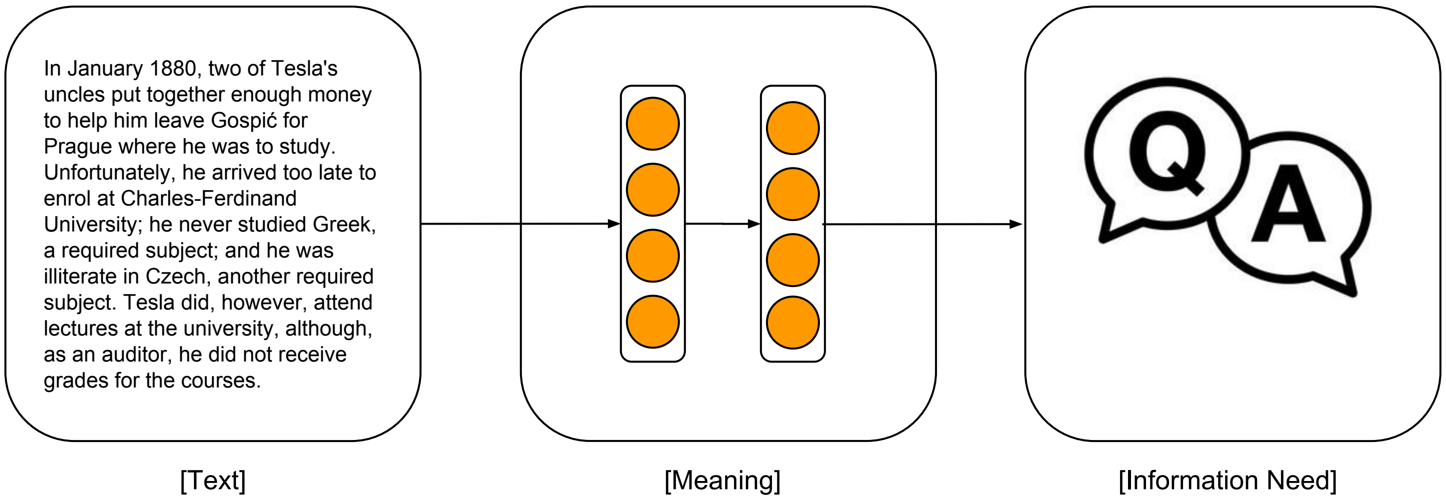
- Advantages
  - More robust to variation in language
  - No cascading errors
  - No domain expertise required
  - Multiple modalities (e.g., VQA) much easier
  - Easy annotation for end-to-end task (e.g., QA)
- Disadvantages
  - Scalability
  - Data efficiency
  - No interpretability
  - No support for reasoning
  - Representations not universal, but question-specific

End-to-End Machine Reading



End-to-End Machine Reading

universality?



Distributed Representations

- Advantages
  - More robust to variation in language
  - No cascading errors
  - No domain expertise required
  - Multiple modalities (e.g., VQA) much easier
  - Easy annotation for end-to-end task (e.g., QA)
- Disadvantages
  - Scalability
  - Data efficiency
  - No interpretability
  - No support for reasoning
  - Representations not universal, but question-specific [?]

Great research opportunities

References

- EDWARD A. FEIGENBAUM, Knowledge Engineering, The Applied Side of Artificial Intelligence, Annals of the New York Academy of Sciences, 1984
- Poon, Hoifung and Quirk, Chris and Toutanova, Kristina and Yih, Wen-tau, NLP for Precision Medicine, Proceedings of ACL 2017, Tutorial Abstracts, https://aclanthology.coli.uni-saarland.de/papers/P17-5001/p17-5001
- Vlachos, Andreas and Riedel, Sebastian, Fact Checking: Task definition and dataset construction , Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science, 2014
- M Banko, MJ Cafarella, S Soderland, M Broadhead, O Etzioni, Open information extraction from the web., IJCAI, 2007
- Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E. R., & Mitchell, T. M. (2010, July). Toward an architecture for never-ending language learning. In AAAI (Vol. 5, p. 3).
- Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. In Advances in Neural Information Processing Systems (pp. 1693-1701).
- Z. H. Huang, W. Xu, and K. Yu, Bidirectional LSTM-CRF Models for Sequence Tagging, arXiv:1508.01991, 2015.
- Lin, Y., Shen, S., Liu, Z., Luan, H., & Sun, M. (2016). Neural relation extraction with selective attention over instances. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 2124-2133).
- Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009, August). Distant supervision for relation extraction without labeled data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2 (pp. 1003-1011). Association for Computational Linguistics.
- Riedel, S., Yao, L., & McCallum, A. (2010, September). Modeling relations and their mentions without labeled text. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 148-163). Springer, Berlin, Heidelberg.
- Ratner, A. J., De Sa, C. M., Wu, S., Selsam, D., & Ré, C. (2016). Data programming: Creating large training sets, quickly. In Advances in neural information processing systems(pp. 3567-3575).
- Durrett, G., & Klein, D. (2013). Easy victories and uphill battles in coreference resolution. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 1971-1982).
- Levesque, H. J., Davis, E., & Morgenstern, L. (2011, March). The Winograd schema challenge. In Aaai spring symposium: Logical formalizations of commonsense reasoning (Vol. 46, p. 47).
- Le, P., & Titov, I. (2018). Improving Entity Linking by Modeling Latent Relations between Mentions. arXiv preprint arXiv:1804.10637.
- He, L., Lewis, M., & Zettlemoyer, L. (2015). Question-answer driven semantic role labeling: Using natural language to annotate natural language. In Proceedings of the 2015 conference on empirical methods in natural language processing (pp. 643-653).

References

- A Neural Probabilistic Language Model (Bengio et al. 2003, JMLR)
- A unified architecture for natural language processing. (Collobert & Weston 2008, ICML)
- Word representations: a simple and general method for semi-supervised learning. (Turian et al. 2010, ACL)
- Efficient Estimation of Word Representations in Vector Space. (Mikolov et al. 2013a, ICLR)
- Distributed Representations of Words and Phrases and their Compositionality. (Mikolov et al. 2013b, NIPS)
- GloVE: Global Vectors for Word Representation. (Pennington et al., 2014, EMNLP)
- Neural word embedding as implicit matrix factorization. (Levy & Goldberg 2014, NIPS)
- Character-Aware Neural Language Models. (Kim et al. 2016, AAAI)

- Blogs: http://ruder.io/word-embeddings-1/ , http://colah.github.io/posts/2015-01-Visualizing-Representations/

References

- QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension. (Yu et al. 2018, ICLR)
- Teaching machines to read and comprehend (Herrmann et al. 2015, NIPS)
- Attention is all you need. (Vaswani et al. 2017, NIPS)
- Long short-term memory-networks for machine reading. (Cheng et al. 2016, EMNLP)
- Gated self-matching networks for reading comprehension and question answering. (Wang et al. 2017, ACL)
- Improved semantic representations from tree-structured long short-term memory networks. (Tai et al. 2015, ACL)
- Recurrent Neural Network Grammars. (Dyer et al. 2016, NAACL)

References

- Adversarial Examples for Evaluating Reading Comprehension Systems (Jia et al. 2017, EMNLP)
- Know What You Don't Know: Unanswerable Questions for SQuAD (Rajpurkar et al. 2018, ACL)
- Visual question answering: Datasets, algorithms, and future challenges (Kafle et al. 2017, Computer Vision and Image Understanding)
- Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering (Goyal et al. 2017, CVPR)
- Reading Wikipedia to Answer Open-Domain Questions (Chen et al. 2017, ACL)
- Event2Mind: Commonsense Inference on Events, Intents, and Reactions (Rashkin et al. 2018, arXiv)
- Semantically Equivalent Adversarial Rules for Debugging NLP Models (Ribeiro 2018, ACL)
- Understanding Neural Networks through Representation Erasure (Li et al. 2016, arXiv)
- HotFLip: White-Box Adversarial Examples for NLP (Ebrahimi et al. 2017, arXiv)
- Anchors: High-Precision Model-Agnostic Explanations (Ribeiro et al. 2018, AAAI)
- Deep contextualized word representations (Peters et al. 2018, NAACL)
- Learned in Translation: Contextualized Word Vectors (McCann et al. 2017, NIPS)
- Supervised Learning of Universal Sentence Representations from Natural Language Inference Data (Conneau et al. 2017, EMNLP)
- Efficient Estimation of Word Representations in Vector Space (Mikolov et al. 2013, NIPS)
- Simple and Effective Semi-Supervised Question Answering (Dhingra et al. NAACL 2018)
- Neural Domain Adaptation for Biomedical Question Answering (Wiese et al. 2017, CoNLL)
- Improving Language Understanding by Generative Pre-Training (Radford et al. 2018, arXiv)
- Neural Skill Transfer from Supervised Language Tasks to Reading Comprehension (Mihaylov et al. 2017, arXiv)
- Representing General Relational Knowledge in ConceptNet 5 (Speer and Havasi, LREC 2012)
- Learning to understand phrases by embedding the dictionary (Hill et al. 2016, TACL)
- Leveraging knowledge bases in lstms for improving machine reading (Yang et al. 2017, ACL)
- Knowledgeable Reader: Enhancing Cloze-Style Reading Comprehension with External Commonsense Knowledge. (Mihaylov and Frank, 2018, ACL)
- Reading Wikipedia to Answer Open-Domain Questions (Chen et al. 2017, ACL)
- Evidence aggregation for answer re-ranking in open-domain question answering (Wang et al. ICLR 2018)
- Marco Baroni and Gemma Boleda: https://www.cs.utexas.edu/~mooney/cs388/slides/dist-sem-intro-NLP-class-UT.pdf
- News article: https://www.independent.co.uk/infact/brexit-second-referendum-false-claims-eu-referendum-campaign-lies-fake-news-a8113381.html

References for Datasets

- Building a question answering test collection, *Voorhees & Tice* SIGIR 2000
- Besting the Quiz Master: Crowdsourcing Incremental Classification Games, *Boyd-Graber et al.* EMNLP 2012
- Semantic Parsing on Freebase from Question-Answer Pairs, *Berant et al.* EMNLP 2013
- Mctest: A challenge dataset for the open-domain macchine comprehension of text, *Richardson et al.* EMNLP 2013
- Teaching Machines to Read and Comprehend, *Hermann et al.* NIPS 2015
- WikiQA: A challenge dataset for open-domain question answering, *Yang et al.* EMNLP 2015
- Large-scale Simple Question Answering with Memory Networks, *Bordes et al. 2015* arXiv:1506.02075.
- The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations, *Hill et al.* ICLR 2016
- SQuAD: 100,000+ Questions for Machine Comprehension of Text, *Rajpurkar et al.* EMNLP 2016
- [SQuAD 2.0] Know What You Don't Know: Unanswerable Questions for SQuAD, *Rajpurkar and Jia et al.* ACL 2018
- Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks, *Weston et al.* ICLR 2016
- Constraint-Based Question Answering with Knowledge Graph, *Bao et al*. COLING 2016
- MovieQA: Understanding Stories in Movies through Question-Answering, *Tapawasi et al.* CVPR 2016
- Who did What: A Large-Scale Person-Centered Cloze Dataset, *Onishi et al.* EMNLP 2016
- MS MARCO: A Human Generated MAchine Reading COmprehension Dataset, *Nguyen et al.* NIPS 2016
- The LAMBADA dataset: Word prediction requiring a broad discourse context, *Paperno et al.* ACL 2016
- WIKIREADING: A Novel Large-scale Language Understanding Task over Wikipedia, *Hewlett et al.* ACL 2016
- TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension, *Joshi et al.* ACL 2017
- Crowdsourcing Multiple Choice Science Questions, *Welbl et al.* WNUT 2017
- RACE: Large-scale ReAding Comprehension Dataset From Examinations, *Lai et al.* EMNLP 2017
- NewsQA: a Machine Comprehension Dataset, *Trischler et al.* RepL4NLP 2017
- Science Exam Datasets by the Allen Institute for Artificial Intelligence: https://allenai.org/data/data-all.html
- SearchQA: A New Q&A Dataset Augmented with Context from a Search Engine, *Dunn et al*. https://arxiv.org/pdf/1704.05179.pdf
- Quasar: Datasets for Question Answering by Search and Reading. *Dhingra et al.* 2017 https://arxiv.org/abs/1707.03904
- Constructing Datasets for Multi-Hop Reading Comprehension across Documents, *Welbl et al.* TACL 2018
- The NarrativeQA Reading Comprehension Challenge, *Kocisky et al.* TACL 2018

Differentiable Program Interpreters

Gains

End-to-end differentiable models are great:

- They can learn arbitrarily abstract representations
- They can process noisy, and ambiguous data
- State-of-the-art for many Machine Reading tasks

Gains and Limitations

End-to-end differentiable models are great:

- They can learn arbitrarily abstract representations
- They can process noisy, and ambiguous data
- State-of-the-art for many Machine Reading tasks

However:

- They cannot really **extrapolate** outside the training data manifold
- They require large amounts of data
- Hard to interpret and analyse models, and explain predictions.

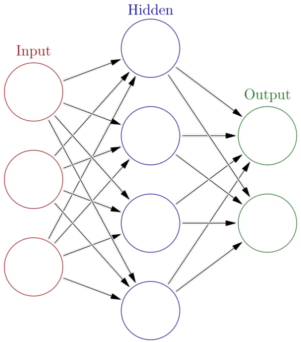
Differentiable Program Interpreters

A possible solution is using models that can learn **algorithms** - decoupling **data** (what) and **computation** (how) - from **multiple training signals** with a differentiable architecture that can be trained end-to-end:

- Learn to operate Memory - Neural Turing Machines
- Learn from Program Traces - Neural Programmer-Interpreters
- Learn from Sketches - Differentiable Forth
- Combining Logic and Learning - Neural Theorem Provers

Differentiable Memory Access

IDEA - turn Neural Networks into **differentiable computers**, by giving them **read-write access** to an **external memory**

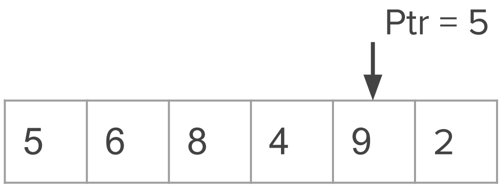


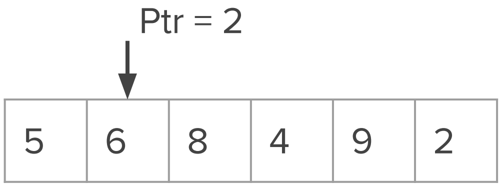
+

- Neural Turing Machines
- Memory Networks
- Stack-Augmented Recurrent Nets
- Neural Random-Access Machines
- Neural GPUs Learn Algorithms
- Neural Programmer-Interpreters
- Hierarchical Attentive Memory
- Dynamic External Memory
- ...

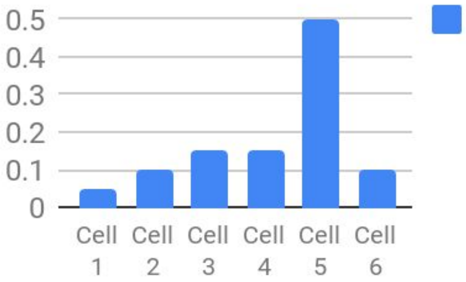
Differentiable Memory Access

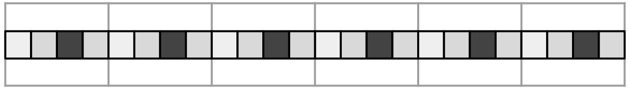
Discrete Representation





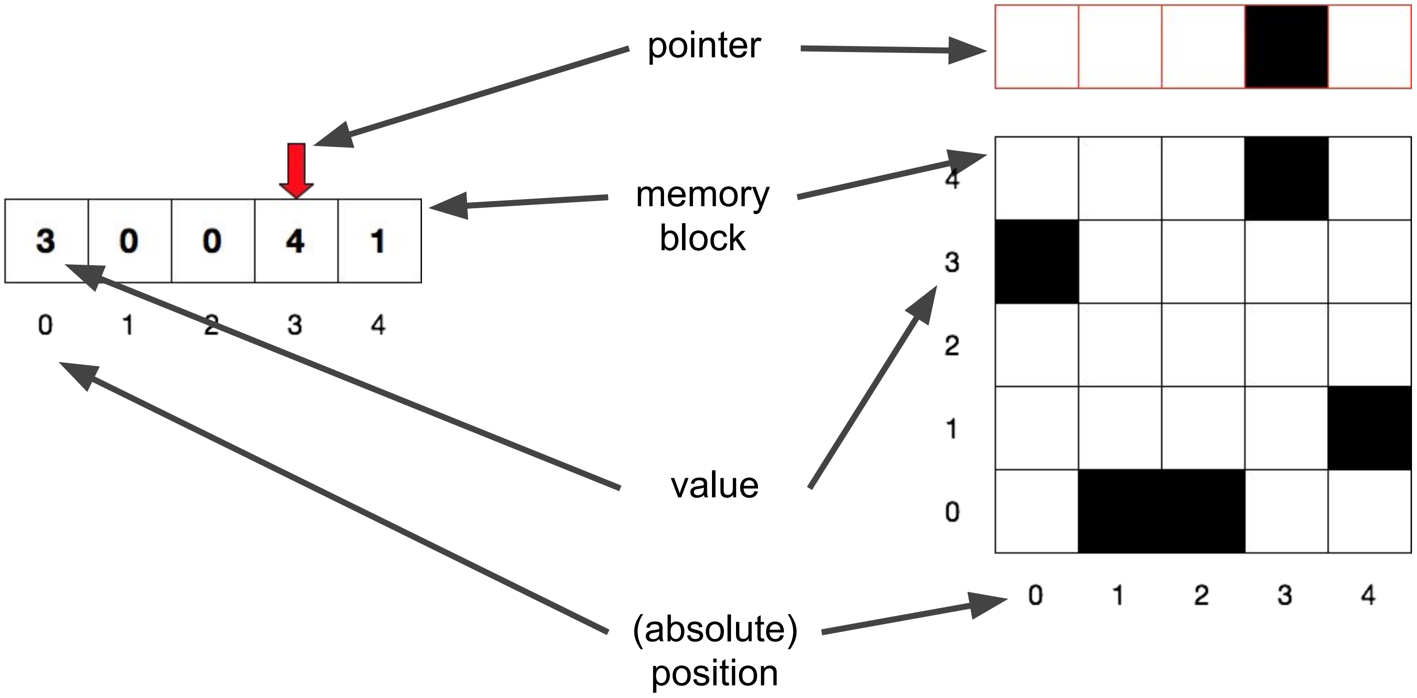
Differentiable Representation





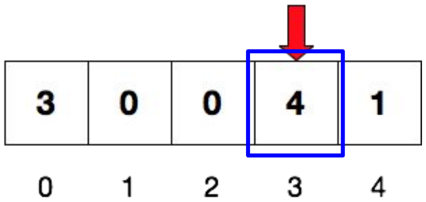
Differentiable Memory Access

Discrete Representation **Sample Continuous (One-Hot) Representation**



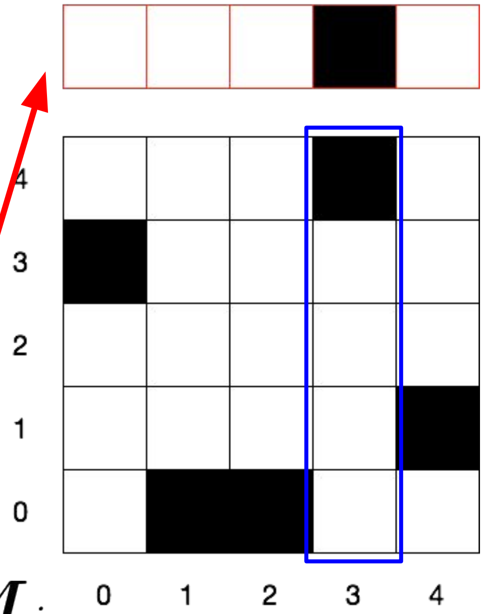
Differentiable Memory Access - Read

Discrete Representation



$$r = M[w]$$

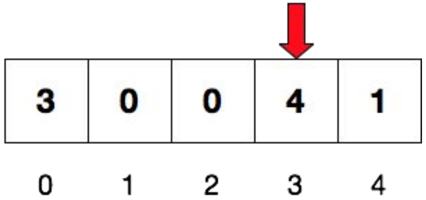
Sample Continuous (One-Hot) Representation



$$\boldsymbol{r} = \sum_i \boldsymbol{w}_i \boldsymbol{M}_i$$

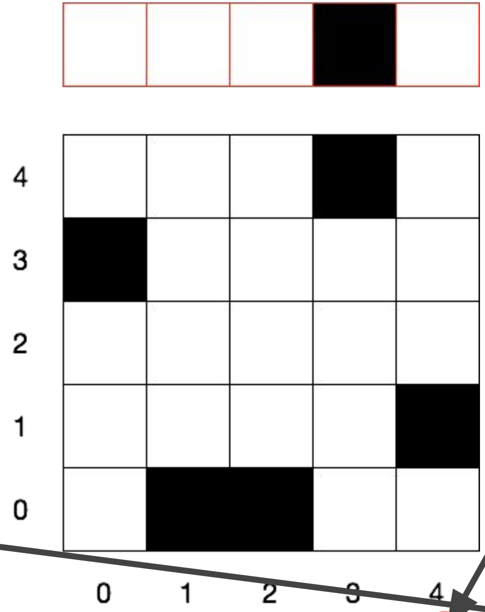
Differentiable Memory Access - Write

Discrete Representation



$$M[w] \leftarrow \alpha$$

Sample Continuous (One-Hot) Representation



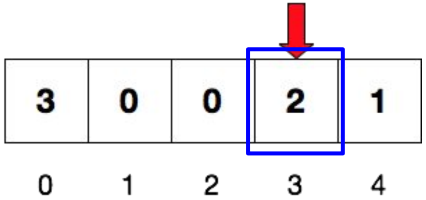
'erase' vector

new value

$$\boldsymbol{M}_i \leftarrow \boldsymbol{M}_i(1 - \boldsymbol{w}_i\boldsymbol{e}) + \boldsymbol{w}_i\boldsymbol{a}$$

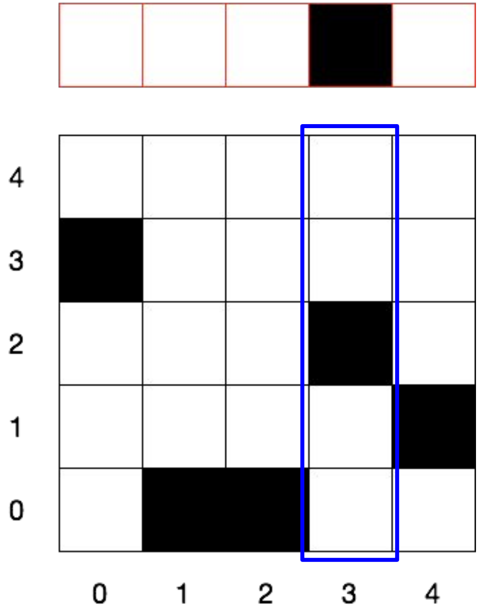
Differentiable Memory Access - Write

Discrete Representation



$$M[w] \leftarrow \alpha$$

Sample Continuous (One-Hot) Representation



$$\boldsymbol{M}_i \leftarrow \boldsymbol{M}_i(1 - \boldsymbol{w}_i \boldsymbol{e}) + \boldsymbol{w}_i \boldsymbol{a}$$

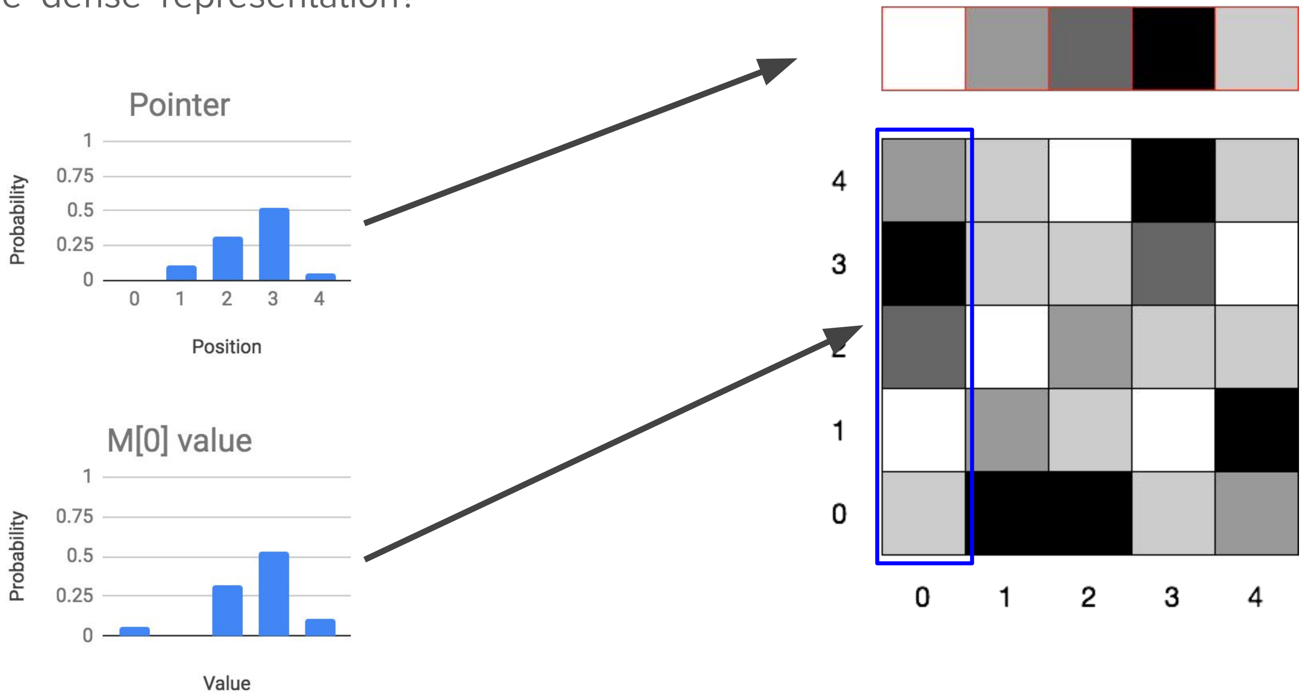
Differentiable Memory Access

One-hot representation is clear, but what is the meaning of the ‘dense’ representation?

Differentiable Representation

Distribution over positions

Distribution over values



Differentiable Memory Access - Read and Write

Reading is a weighted sum of all the values in the memory:

$$\boldsymbol{r} = \sum_i \boldsymbol{w}_i \boldsymbol{M}_i$$

Writing erases the previous value with an **erase vector e**, and then **adds a vector a** to it, all weighted by w:

$$\boldsymbol{M}_i \leftarrow \boldsymbol{M}_i(\boldsymbol{1} - \boldsymbol{w}_i \boldsymbol{e}) + \boldsymbol{w}_i \boldsymbol{a}$$

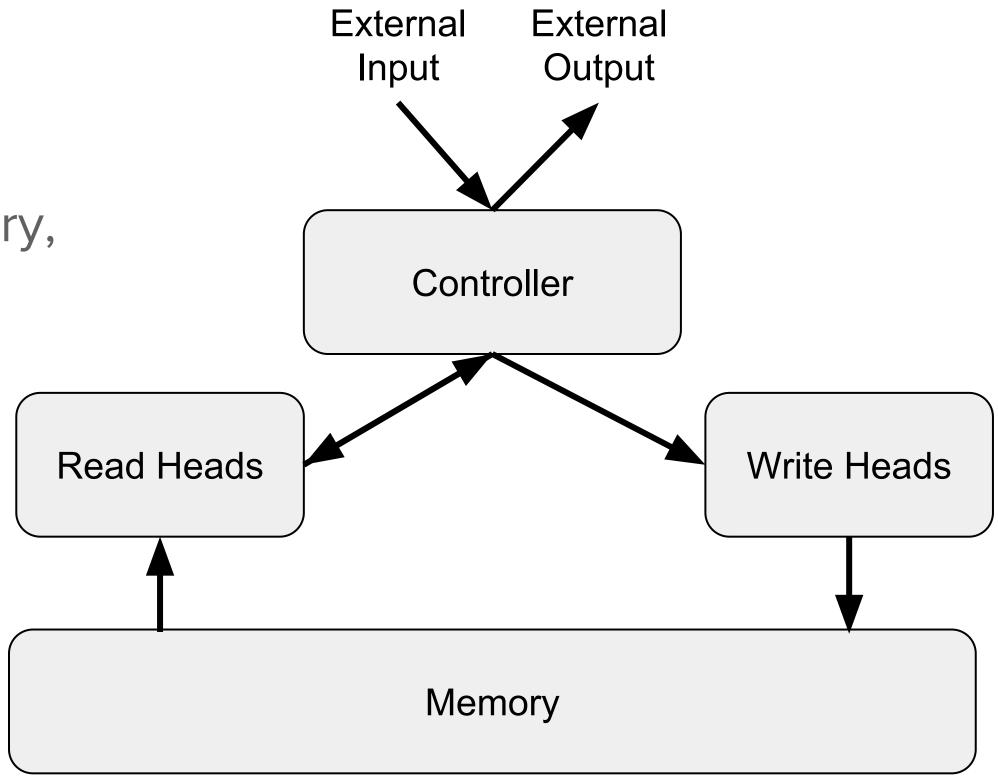
Neural Turing Machines

Controller is a neural network.

Heads select portions of memory, and **read**/**write** to them.

Memory is a real-valued matrix.

<u>**End-to-end Differentiable**</u>



Selective Attention

- **Focus** on parts of memory the network will read and write to
  - **Attention** model

- Controller outputs parametrise a distribution (**weighting**) over the rows (**memory locations**) in the memory matrix.
- **Weighting** is defined by two main attention mechanisms:
  - **Content**-based lookup
  - **Location**-based lookup

Addressing by Content

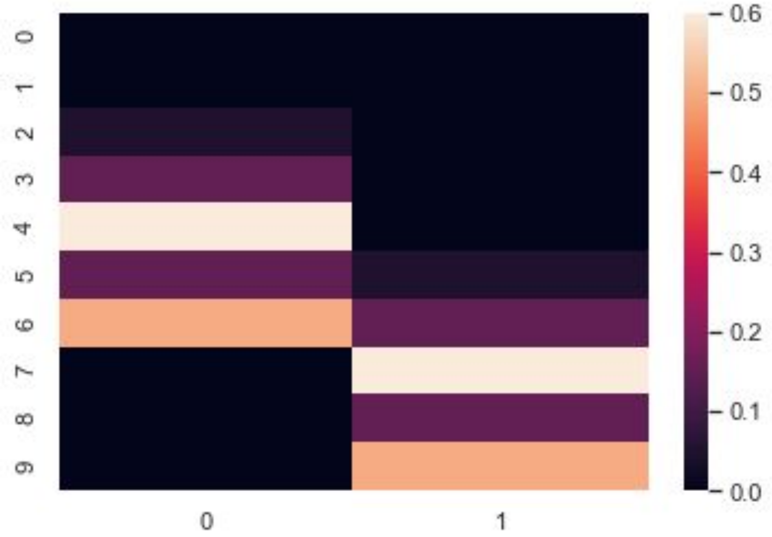
A **key vector *k*** is emitted by the controller and compared with each memory location ***M*** using a similarity measure *s*, then normalised via a **softmax** operation.

$$\boldsymbol{w}_i = \frac{\exp(\beta s(\boldsymbol{k}, \boldsymbol{M}_i))}{\sum_j \exp(\beta s(\boldsymbol{k}, \boldsymbol{M}_j))}$$

Addressing by Location

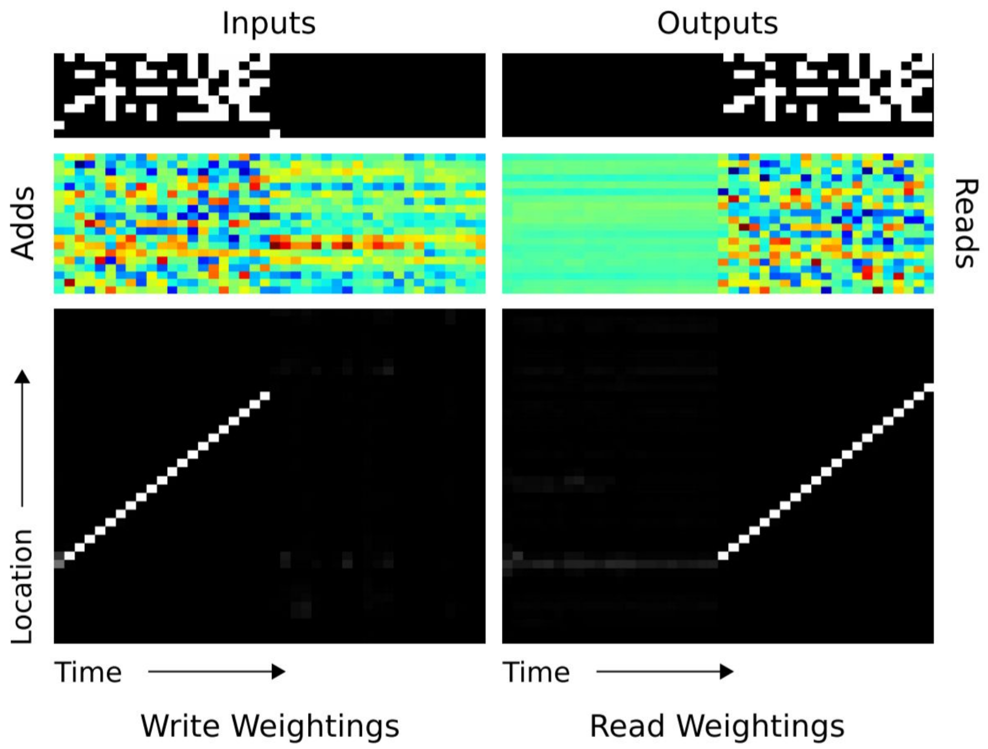
The **Controller** outputs a **shift kernel *s*** (for instance a softmax over *[-n, n]*), which is combined with a distribution over locations **w** to produce a shifted weighting:

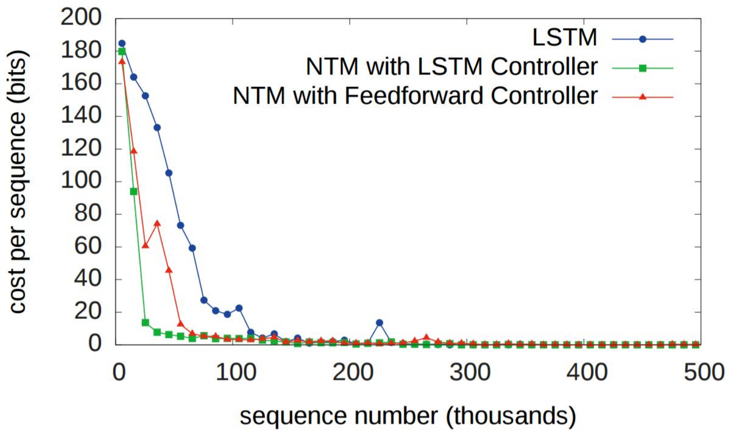
$$\hat{\boldsymbol{w}}_i = \sum_j \boldsymbol{w}_j \boldsymbol{s}(i-j)$$



The addressing mechanisms jointly interact with memory.

Neural Turing Machine - (Repeated) Copying





NTM learns its first **for loop**, using **content** to jump, **iteration** to step, and a **variable** to **count** to N.

Reading and Writing

Once weightings are defined, each **read head** returns a **read vector** ***r*** as input to the controller at the next time step.

$$\boldsymbol{r} = \sum_i \boldsymbol{w}_i \boldsymbol{M}_i$$

Each **write head** receives an **erase vector e** and an **add vector *a*** from the controller, and resets then writes to modify the memory.

$$\boldsymbol{M}_i \leftarrow \boldsymbol{M}_i(\mathbf{1} - \boldsymbol{w}_i\boldsymbol{e}) + \boldsymbol{w}_i\boldsymbol{a}$$

Neural Programmer-Interpreters

Recurrent compositional neural network that **learns to represent and execute programs**, composed by three components:

- Task-agnostic recurrent core (similar to a **controller**)
- A key-value program **memory**
- Domain-specific **encoders** for observations and args

NPIs can be trained from program traces

Neural Programmer Interpreters

NPIs have the following goals:

- **Long-Term Predictions** - generalise to longer sequences of action by exploiting a program's compositional structure
- **Continual/Never-Ending Learning** - possible to learn new programs by compositing previously-learned programs.
- **Data Efficiency** - Use multiple training signals - traces - for learning more generalizable programs.
- **Interpretability** - By observing commands generated by NPIs, we can understand what it is doing at various levels of granularity.

Neural Programmer Interpreters - Training Data

Environment Observations

Program Index

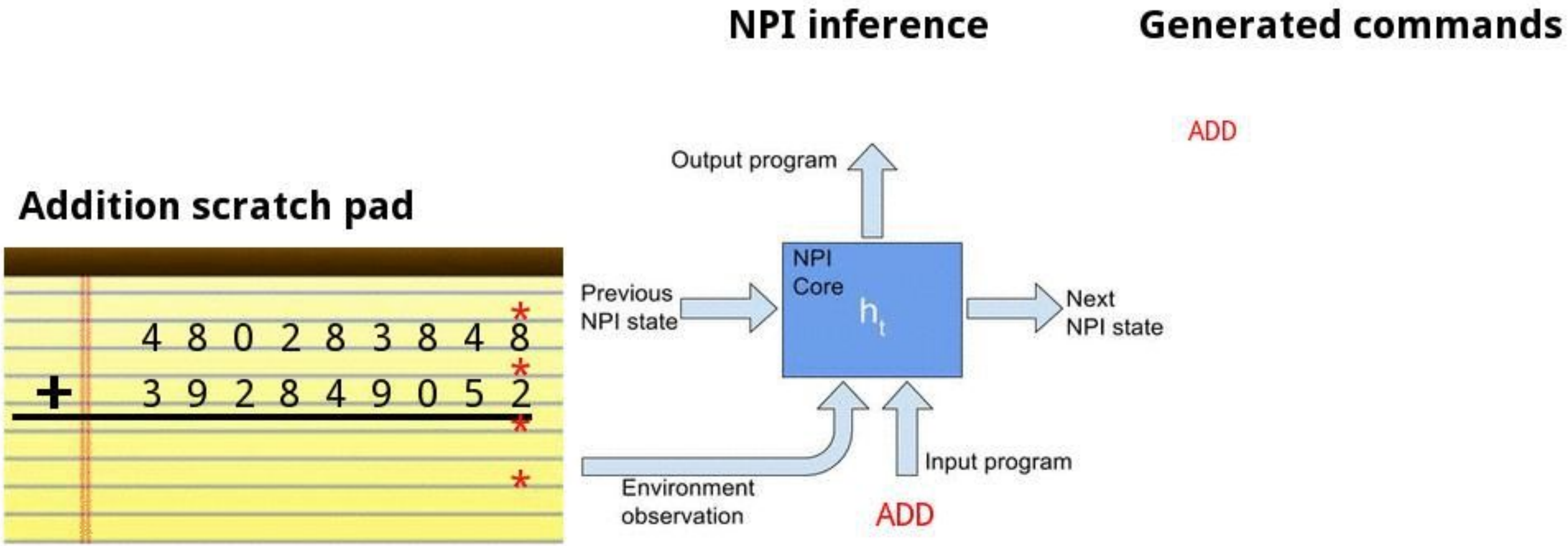
Program Arguments

$$\xi_t^{\text{input}} = \{(e_t, i_t, a_t) \mid t = 1, \ldots, T-1\}$$

$$\xi_t^{\text{output}} = \{(\boxed{i_{t+1}}, \boxed{a_{t+1}}, r_t) \mid t = 1, \ldots, T-1\}$$

Return Bit

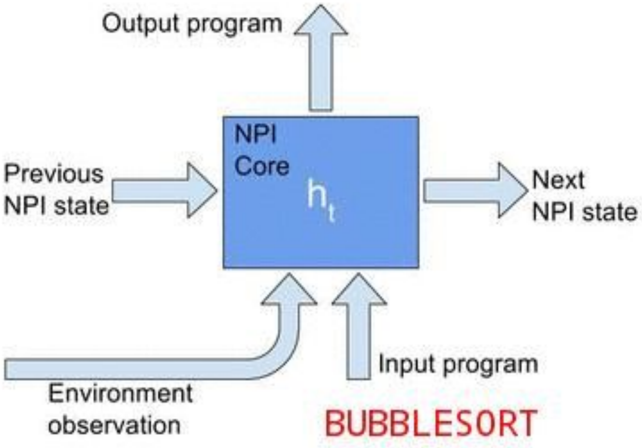
Neural Programmer Interpreters



Neural Programmer Interpreters

Input array

NPI inference



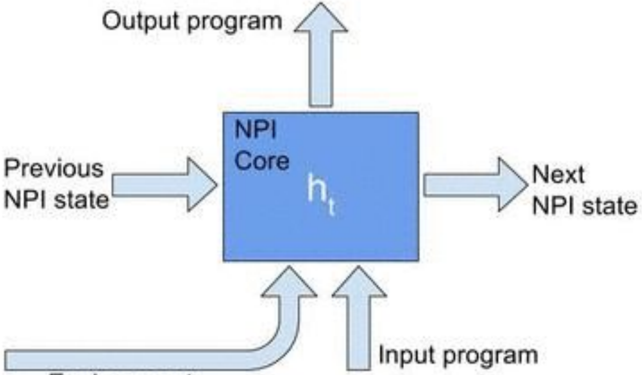
Generated commands

BUBBLESORT

Neural Programmer Interpreters

Car rendering

NPI inference



GOTO 1 2

Generated commands

GOTO 1 2

Differentiable Forth

Forth Abstract Machine

- Program Counter
- Memory Heap
- Data Stack
- Return Stack

```
: BUBBLE
    DUP IF >R
      OVER OVER < IF SWAP THEN
      R> SWAP >R 1- BUBBLE R>
    ELSE
      DROP
    THEN
;
: SORT
    1- DUP 0 DO >R R@ BUBBLE R> LOOP DROP
;
```

Simple abstract machine: one **Heap** and two **Stacks**.

Differentiable Forth

Forth Abstract Machine

- Program Counter
- Memory Heap
- Data Stack
- Return Stack

∂4 - Neural Forth Abstract Machine

- Softmax over all commands
- Matrix (RW ops like NTM)
- Matrix + ToS vector
- Matrix + ToS vector

When learning comparison in sorting, and digit addition:

- Generalise to longer sequences
- Extrapolate from a smaller number of samples
- Still difficult to learn sorting longer sequences (longer term dependencies)

Prolog - Backward Chaining

Knowledge Base

fatherOf(abe, homer)
parentOf(homer, bart)

grandFatherOf(X, Y) ⇐
    fatherOf(X, Z),
    parentOf(Z, Y)

Intuition:

- **Backward chaining** translates a **query** into **subqueries** via **rules**, e.g.
  grandFatherOf(abe, homer) becomes
  fatherOf(abe, Z), parentOf(Z, bart)

- Prolog attempts this for all rules in the Knowledge Base, in a **depth-first** fashion

Prolog - Unification

Knowledge Base

fatherOf(abe, homer)
parentOf(homer, bart)

grandFatherOf(X, Y)
⇐ fatherOf(X, Z),
parentOf(Z, Y)

Query

grandFatherOf	abe	bart
fatherOf	abe	homer
FAIL	SUCCESS	FAIL

Prolog - Unification

Knowledge Base

fatherOf(abe, homer)
parentOf(homer, bart)

grandFatherOf(X, Y)
⇐ fatherOf(X, Z),
parentOf(Z, Y)

Query

grandFatherOf	abe	bart
⬇	⬇	⬇
grandFatherOf	X	Y
⬇	⬇	⬇
SUCCESS	X/abe	X/bart

Prolog - Unification

Knowledge Base

fatherOf(abe, homer)
parentOf(homer, bart)

grandFatherOf(X, Y)
⇐ fatherOf(X, Z),
parentOf(Z, Y)

Query

grandPaOf	abe	bart
grandFatherOf	X	Y
FAIL	X/abe	X/bart

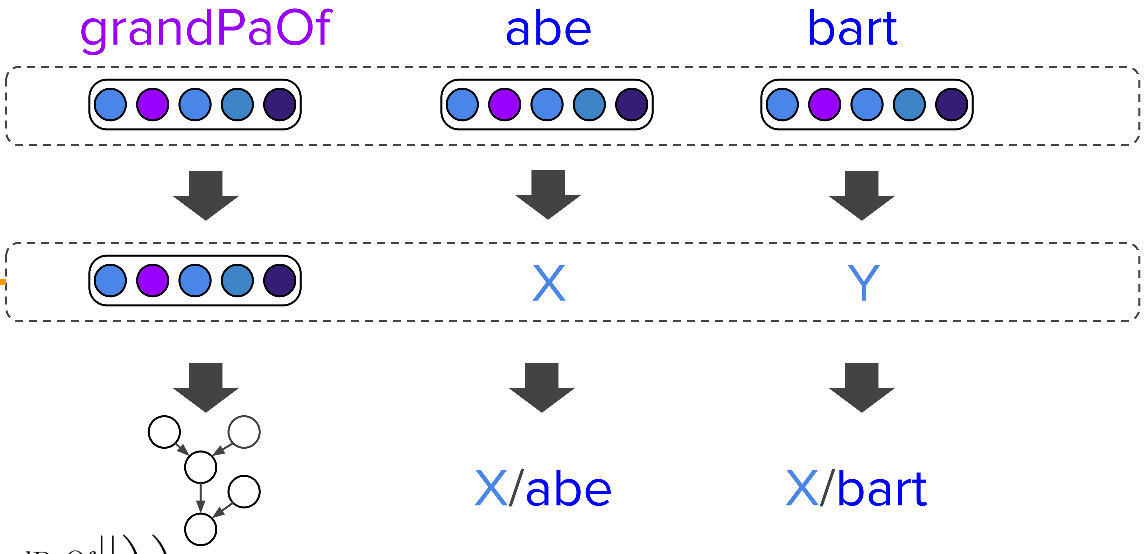
Prolog - **Neural** Unification

Knowledge Base

fatherOf(abe, homer)
parentOf(homer, bart)

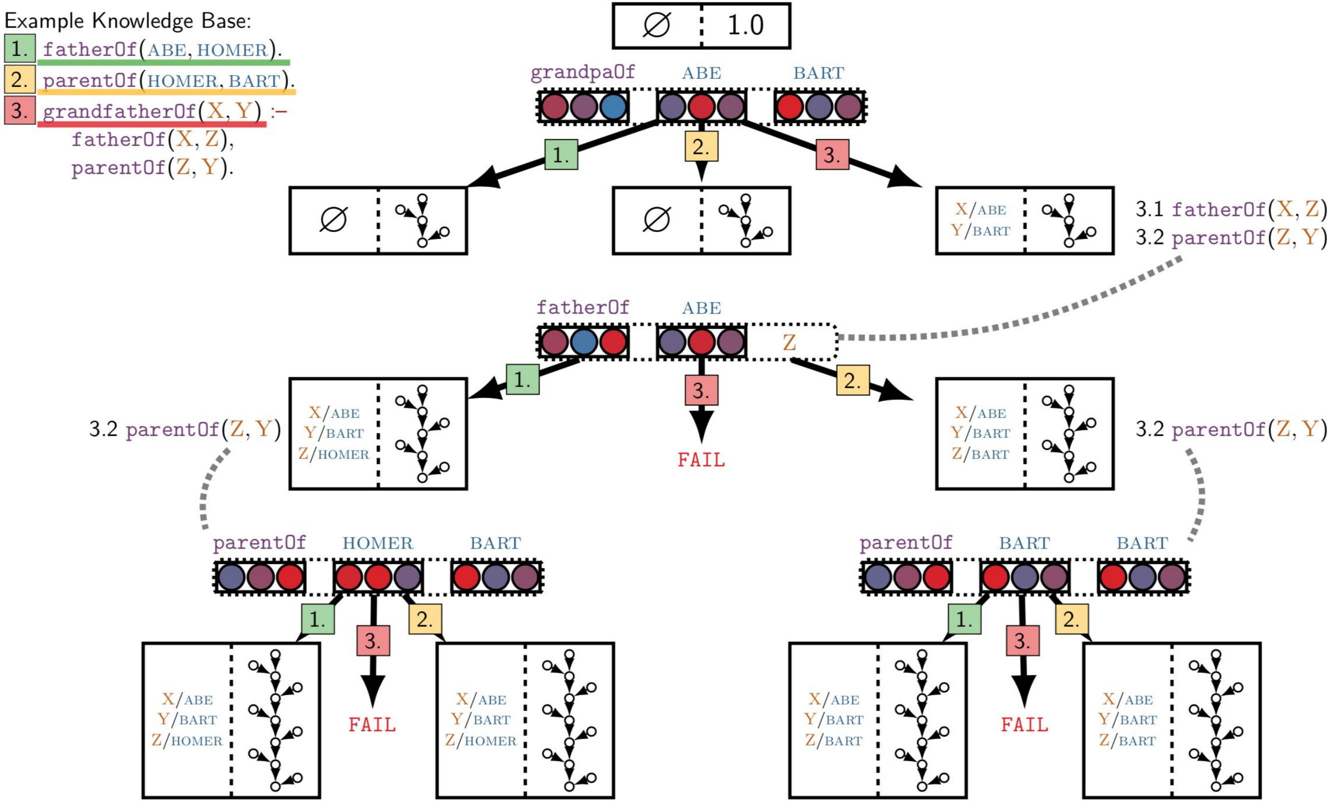
grandFatherOf(X, Y)
$\Leftarrow$ fatherOf(X, Z),
parentOf(Z, Y)

Query



$$\min\left(1.0, \exp\left(\frac{-||\theta_{\text{grandFatherOf}} - \theta_{\text{grandPaOf}}||}{2\mu^2}\right)\right)$$

End-to-end Differentiable Theorem Proving

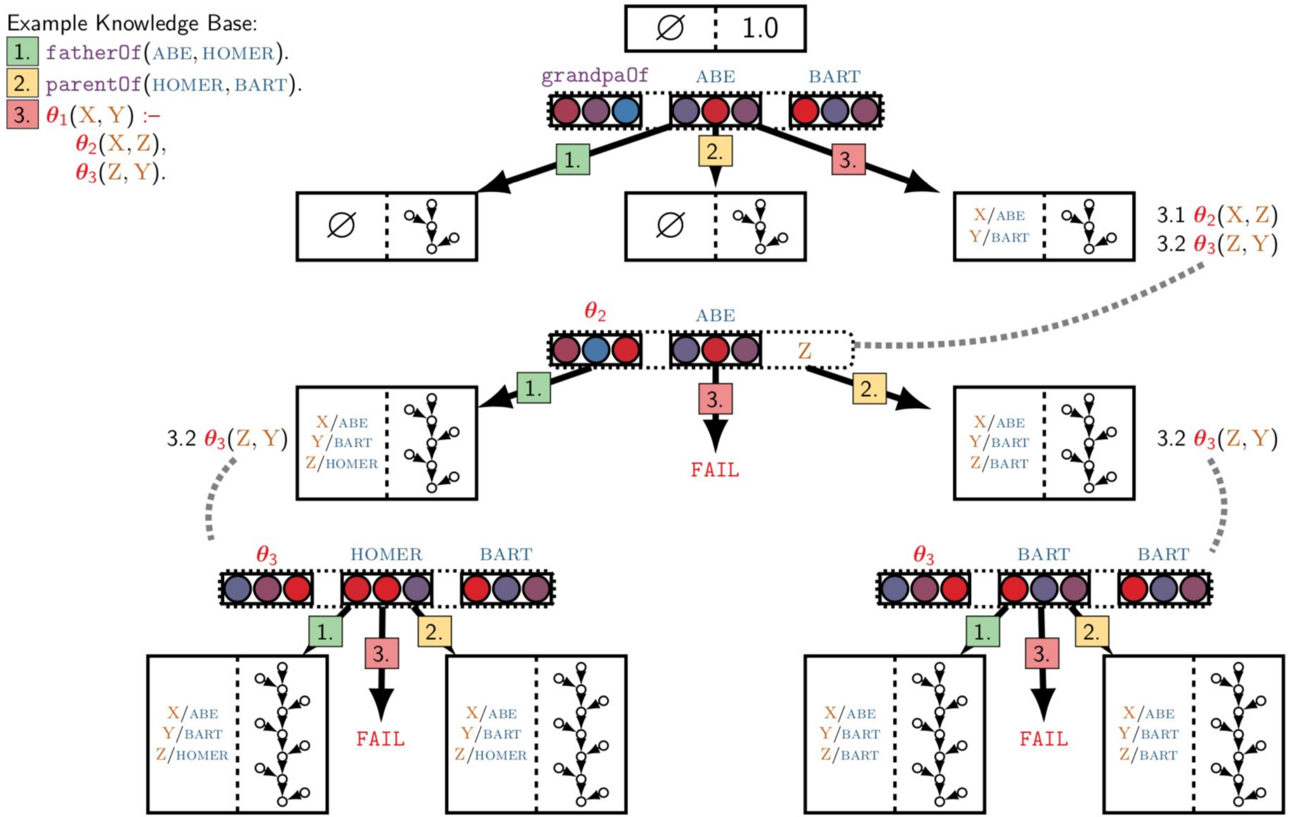


Idea - use Prolog's backward chaining to recursively construct a neural network aggregating all possible proof trees for a given goal - each proof tree returning a different **proof score**.

Final score - maximum proof score across all proof trees.

Rocktäschel et al. - NIPS 2017
End-to-end Differentiable Proving

End-to-end Differentiable Rule Induction



Idea - use Prolog's backward chaining to recursively construct a neural network aggregating all possible proof trees for a given goal - each proof tree returning a different **proof score**.

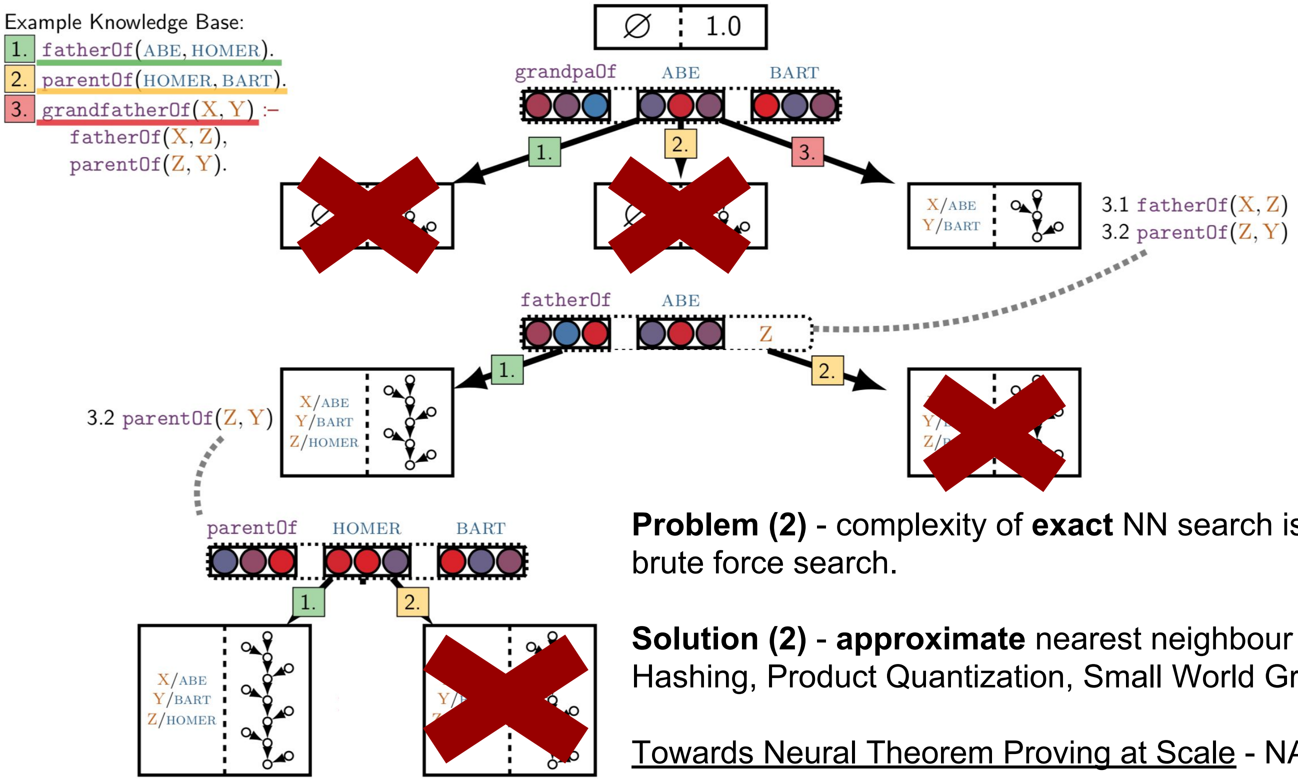
Rocktäschel et al. - NIPS 2017
End-to-end Differentiable Proving

Final score - maximum proof score across all proof trees.

However -

Problem - exponential blow-up in the number of proof trees in the **depth** and **width** of the network.

Differentiable Theorem Proving at Scale



Problem - exponential blow-up in the number of proof trees in the **depth** and **width** of the network.

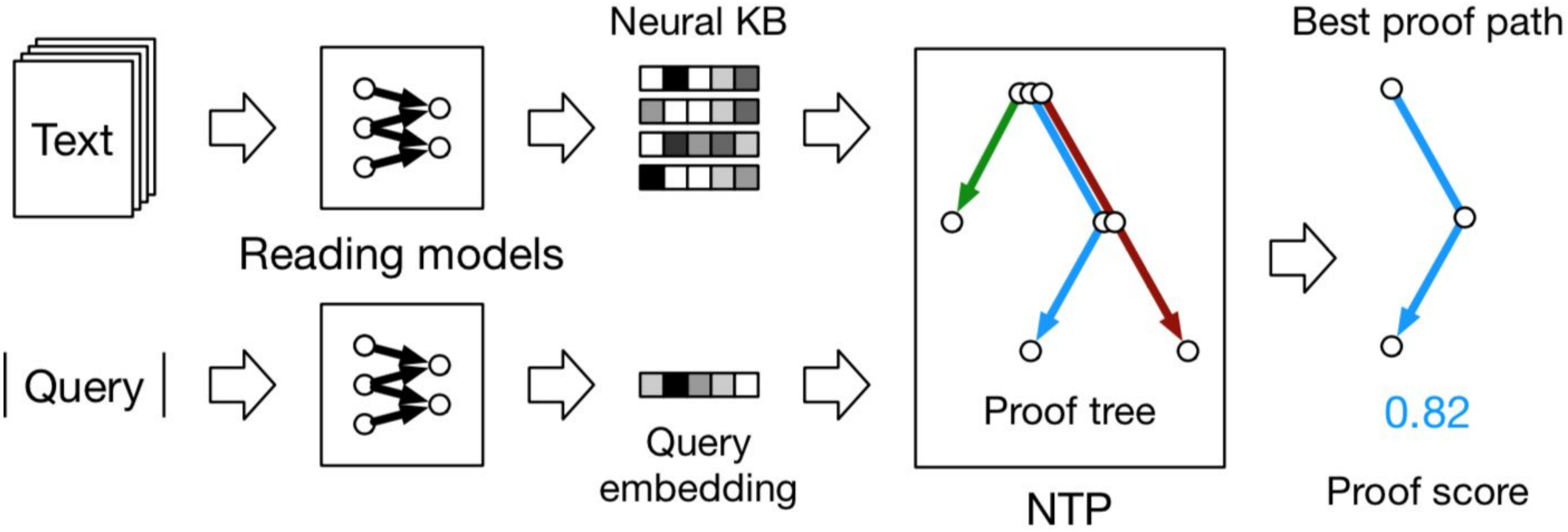
Solution - during the construction of the neural network, dynamically avoid constructing proof trees that will likely lead to low proof scores by using nearest neighbour search.

Problem (2) - complexity of **exact** NN search is approximately the same as brute force search.

Solution (2) - **approximate** nearest neighbour search (via Local-Sensitive Hashing, Product Quantization, Small World Graphs..)

Towards Neural Theorem Proving at Scale - NAMPI 2018

Challenge: Reasoning at Scale on Multiple Modalities



What if my model is not end-to-end differentiable?

If your model or loss function has non-differentiable steps in it, you can still train it:

- Reinforcement Learning
- Evolution Strategies
- Bayesian Optimisation
- Other gradient-free optimisation methods

One example of a simple technique for computing noisy gradient estimates:

$$\nabla f(\theta) \approx \frac{1}{\sigma^2} \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\sigma^2)} \left[\epsilon f(\theta + \epsilon) \right]$$

Salimans et al. 2017 -
Evolution Strategies as a Scalable Alternative to Reinforcement Learning

Conclusions

Neural networks are not perfect:

- Hard time generalising from small data samples
  - They are universal functions approximators - without a proper inductive bias, they may find the wrong solutions for a given learning (optimisation) problem.
- Hard to incorporate procedural or declarative knowledge
  - Our knowledge is symbolic (e.g. language), but neural networks are inherently subsymbolic.

Things we can do:

- Try to differentiate **computation** (how) from **data** (what)
- Use multiple supervision signals - e.g. auxiliary objectives, program traces, partial programs, declarative background knowledge..

Thank You!

Backup or Old Slides

Why do we need compositional phrase representations in QA?

What city did Tesla move to in 1880?

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study.

- **Goal**: similar representations for phrases with similar meaning, even with lexical / syntactic variation

"move from Gospić to Prague" ≈ "leave Gospić for Prague"

Synthesis: Symbolic vs. Subsymbolic Machine Reading

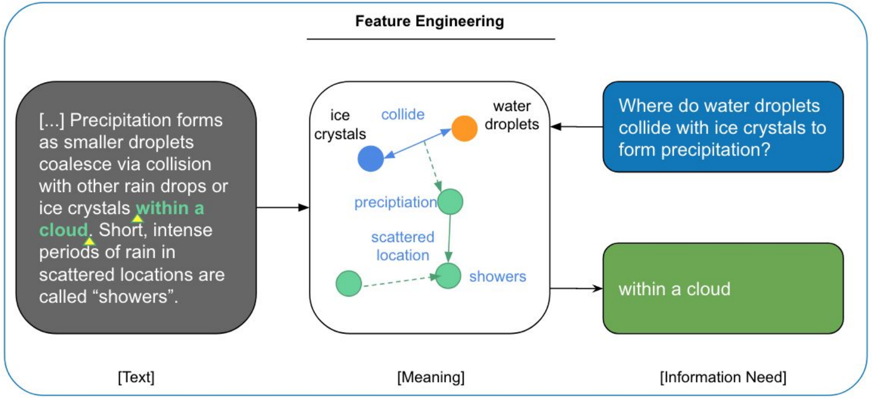
- A transferrable representation of text
  - that humans and machine can interface with.

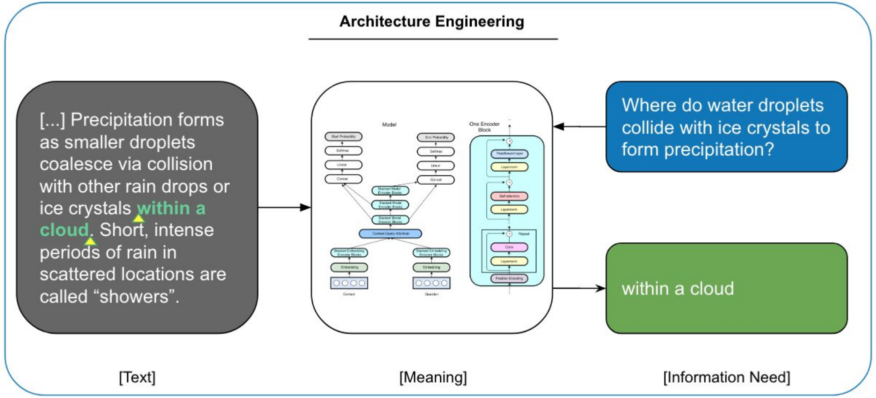
	Knowledge Base	**Neural Networks**
Knowledge Representation	structured / explicit	distributed / implicit
Means of Construction	Information Extraction	(Un)supervised Learning
Interface	Query Language	Vectors
Optimization	discrete	gradient-based

A Paradigm Shift

- Symbolic Meaning Representations
- ➡ Latent Vector Representations

- Feature Engineering & Domain Expertise
- ➡ Architecture Engineering & ML/DL Expertise





Gains and Losses of this Shift

- Gains
  - Generalization and domain transferability (mainly due to unsupervised learning)
  - No domain expertise
  - Multiple modalities (e.g., VQA) much easier
  - Easy annotation for end-to-end task (e.g., QA)

- Losses
  - Ability to do reasoning
  - Data efficiency
  - Incorporating background knowledge
  - Scalability
  - Interpretability

Great research opportunities

Synthesis: Symbolic vs. Subsymbolic Machine Reading

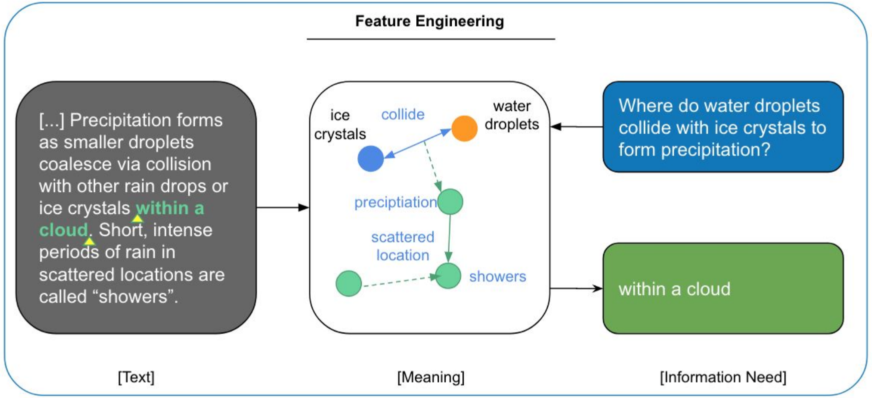
- A transferrable representation of text
  - that humans and machine can interface with.

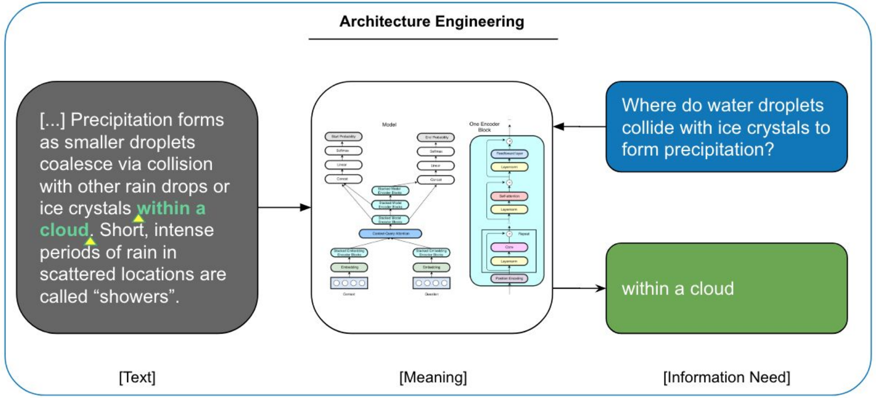
	Knowledge Base	**ELMo Vectors**
Knowledge Representation	structured / explicit	distributed / implicit
Means of Construction	Information Extraction	Applying Language Model
Interface	Query Language	Neural Net
Optimization	discrete	gradient-based

A Paradigm Shift

- Symbolic Meaning Representations
- ➡ Latent Vector Representations

- Feature Engineering & Domain Expertise
- ➡ Architecture Engineering & ML/DL Expertise





A Synthesis ?!

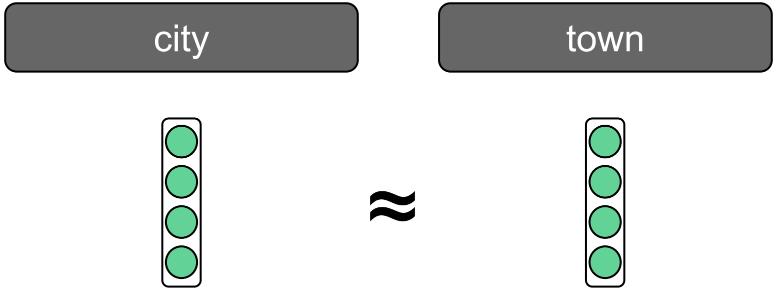
- Can we solve the challenges of end-to-end solutions that could be addressed more easily with intermediate symbolic meaning representations?

- Or can we find a way to synthesize the best of both worlds?

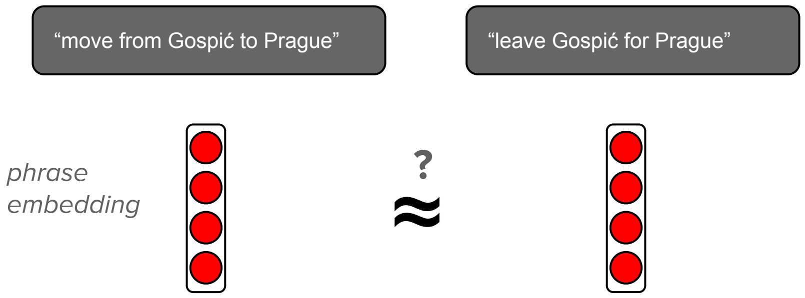
Best Practices

- Exploit pre-trained models:
  - (Minimum) word embeddings and language models
  - Modeling innovations such as (self-)attention
  -
- ...
- Nice reference: ruder.io/deep-learning-nlp-best-practices/

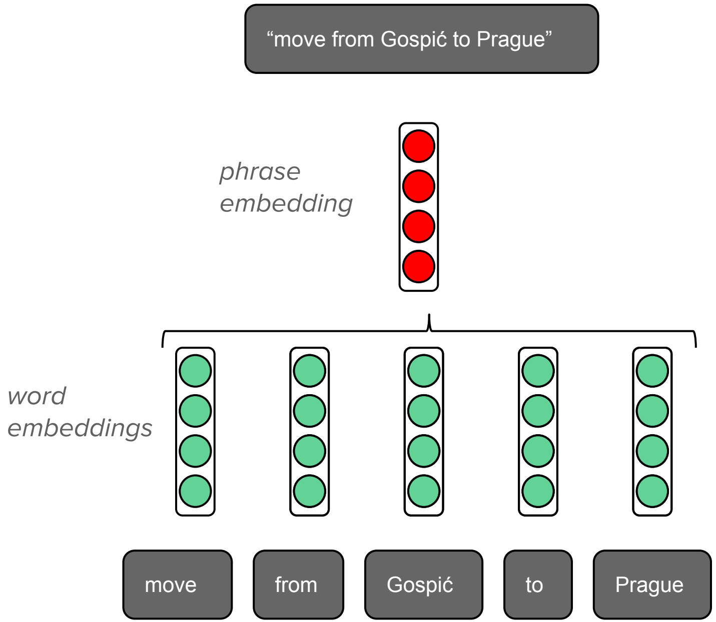
Similarity between words: word embeddings



Similarity between phrases?

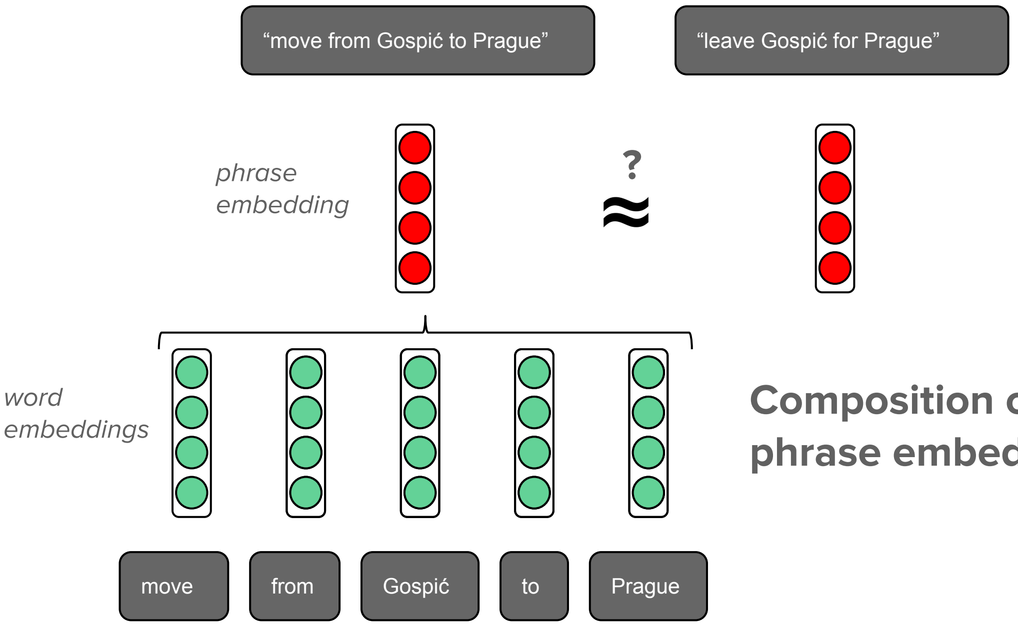


Similarity between phrases?



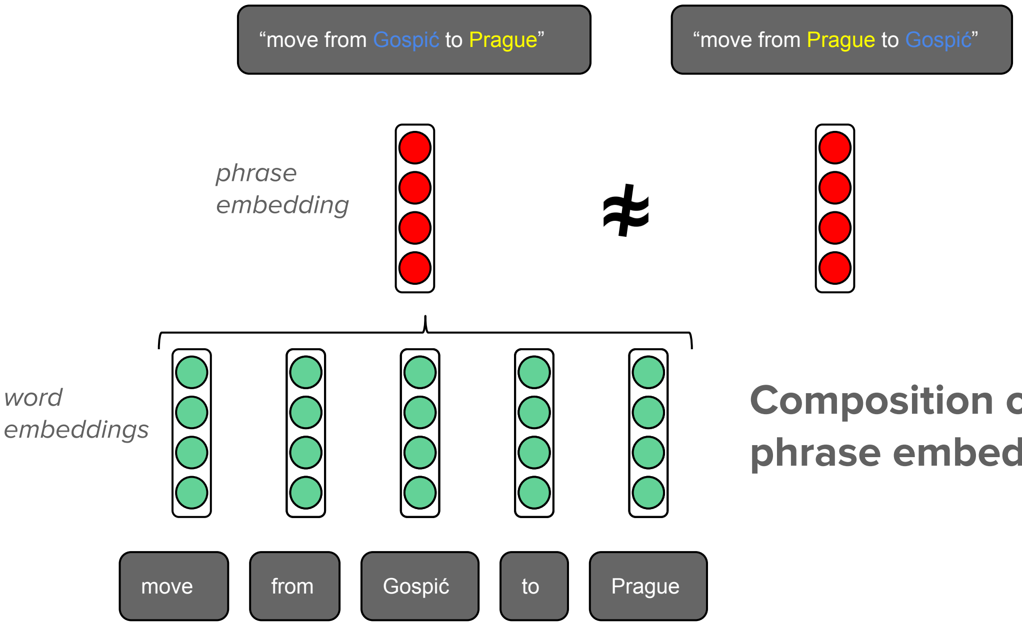
Composition of word into phrase embeddings

Similarity between phrases?



Composition of word into phrase embeddings

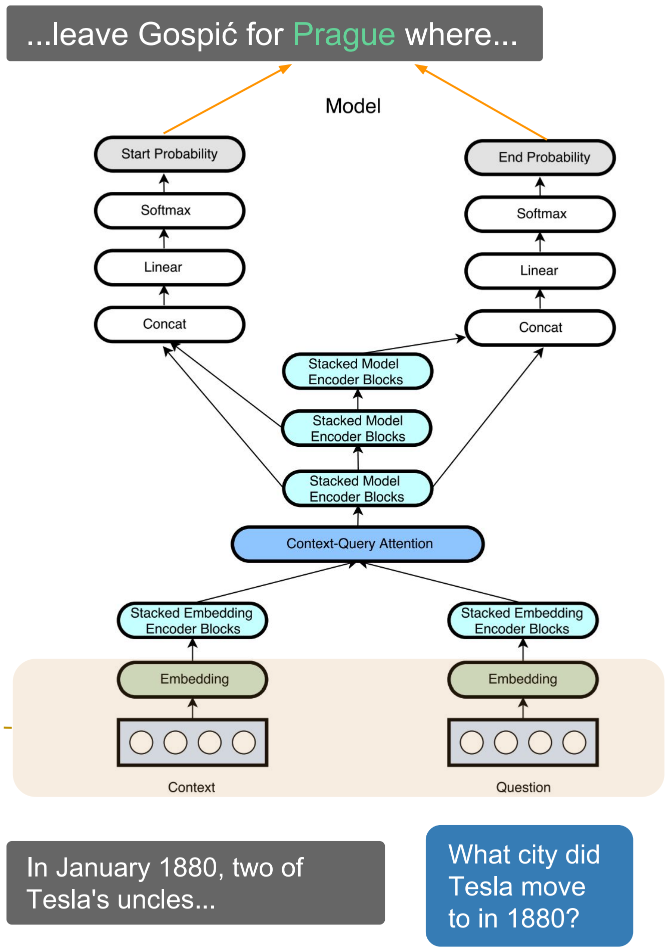
Similarity between phrases?



Word order matters!

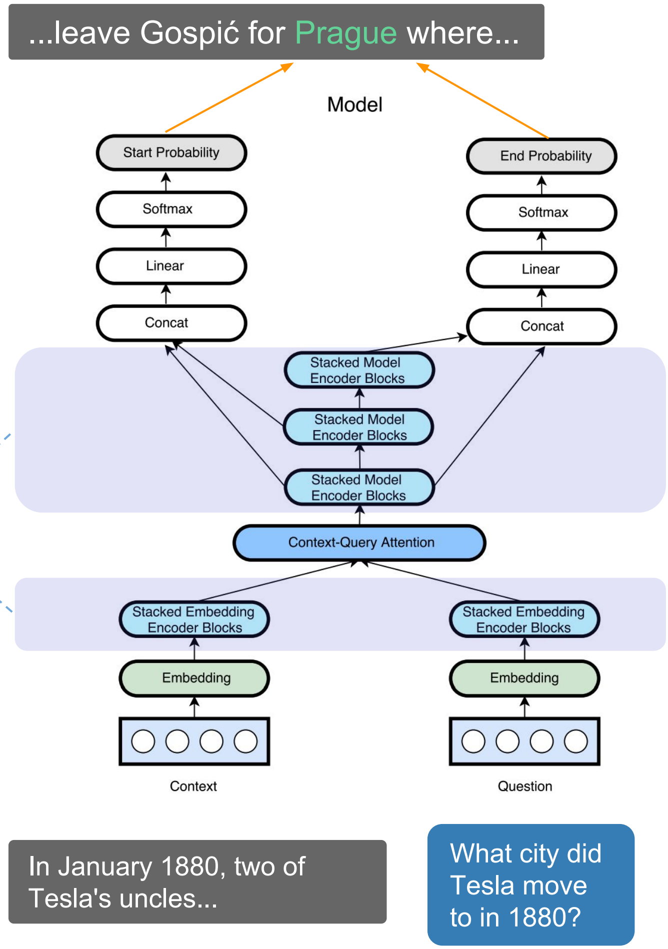
Composition of word into phrase embeddings

How to represent symbols?



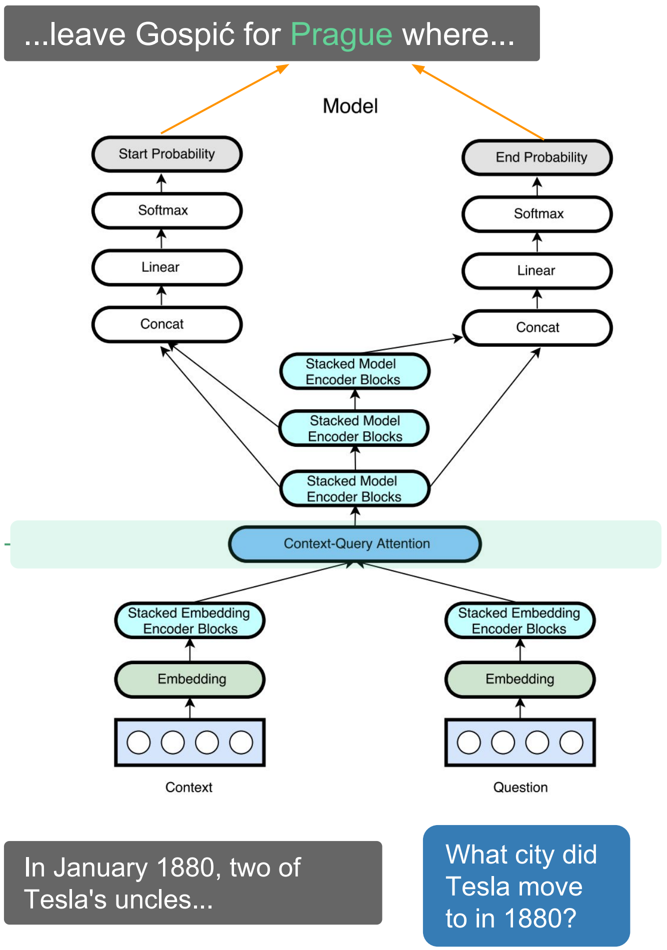
QANet, Yu et al. (2018)

How to condition word representations on one another



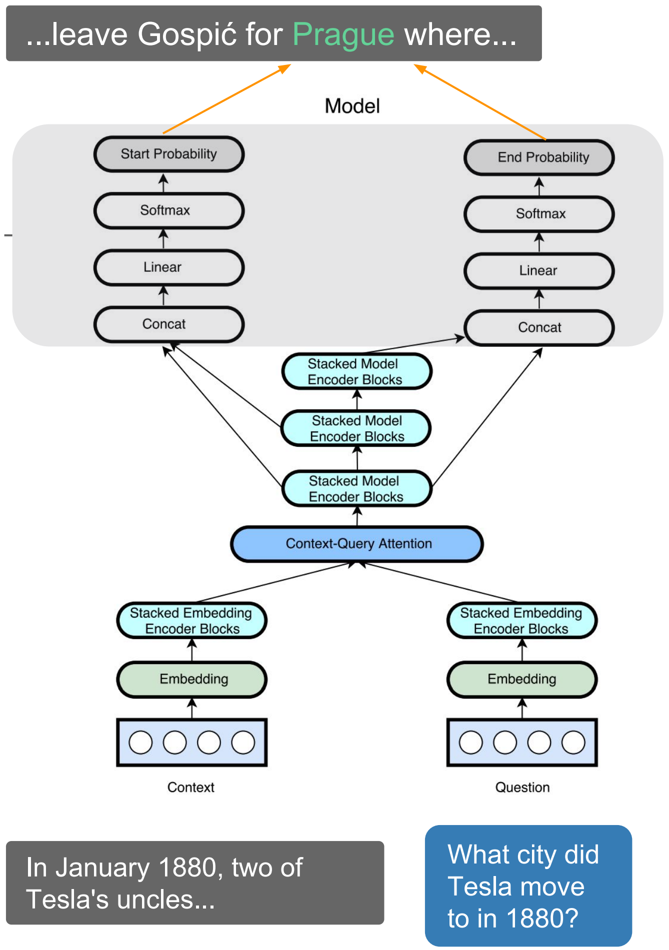
QANet, Yu et al. (2018)

sequence interaction between question and text



QANet, Yu et al. (2018)

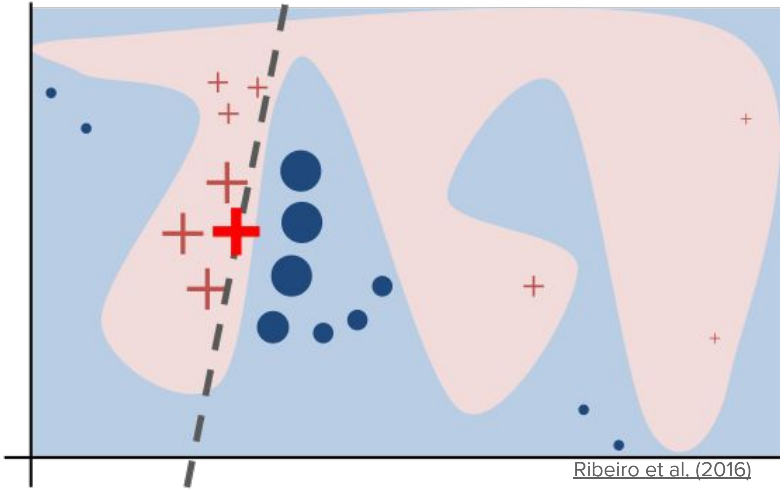
Span Scoring: linear projection, score for start and end position



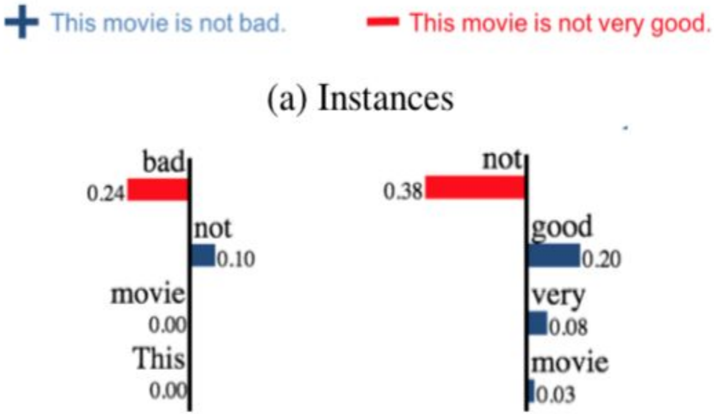
QANet, Yu et al. (2018)

Model Diagnostics: Right for the Wrong Reason?

- Example 2: LIME (Ribeiro et al. 2016)
  - Idea: Find features that predictions are sensitive to
  - Local perturbations, fit linear model on predictions



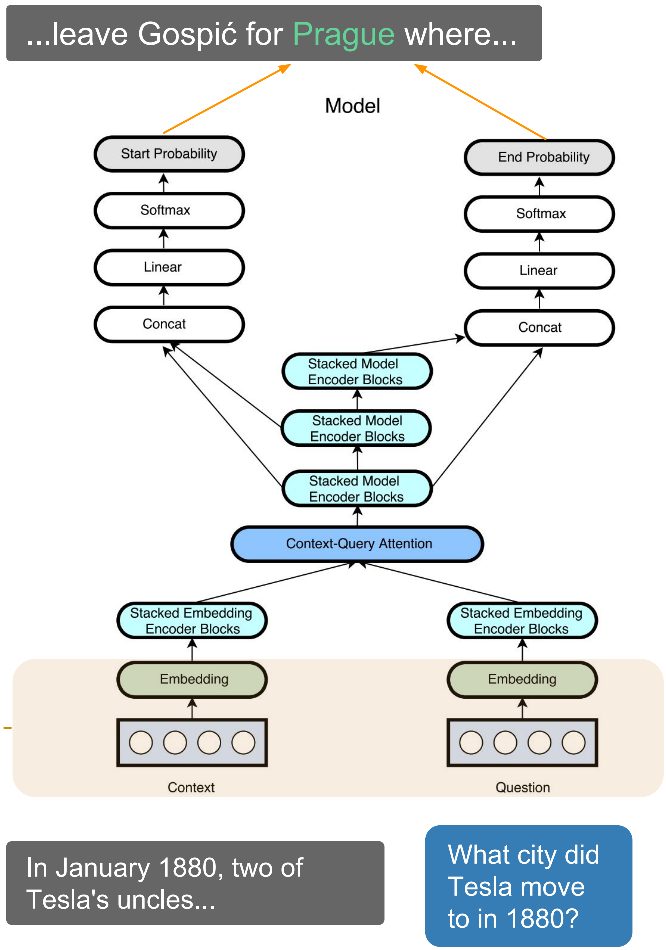
Ribeiro et al. (2016)



Ribeiro et al. (2016)

- Alvarez-Melis and Jaakkola (2017): similar, but with sequences.

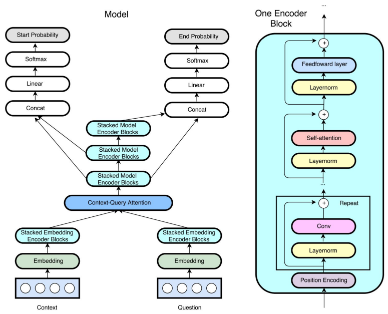
How to represent symbols?



Architecture Engineering

Architecture Engineering

[...] 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”.



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

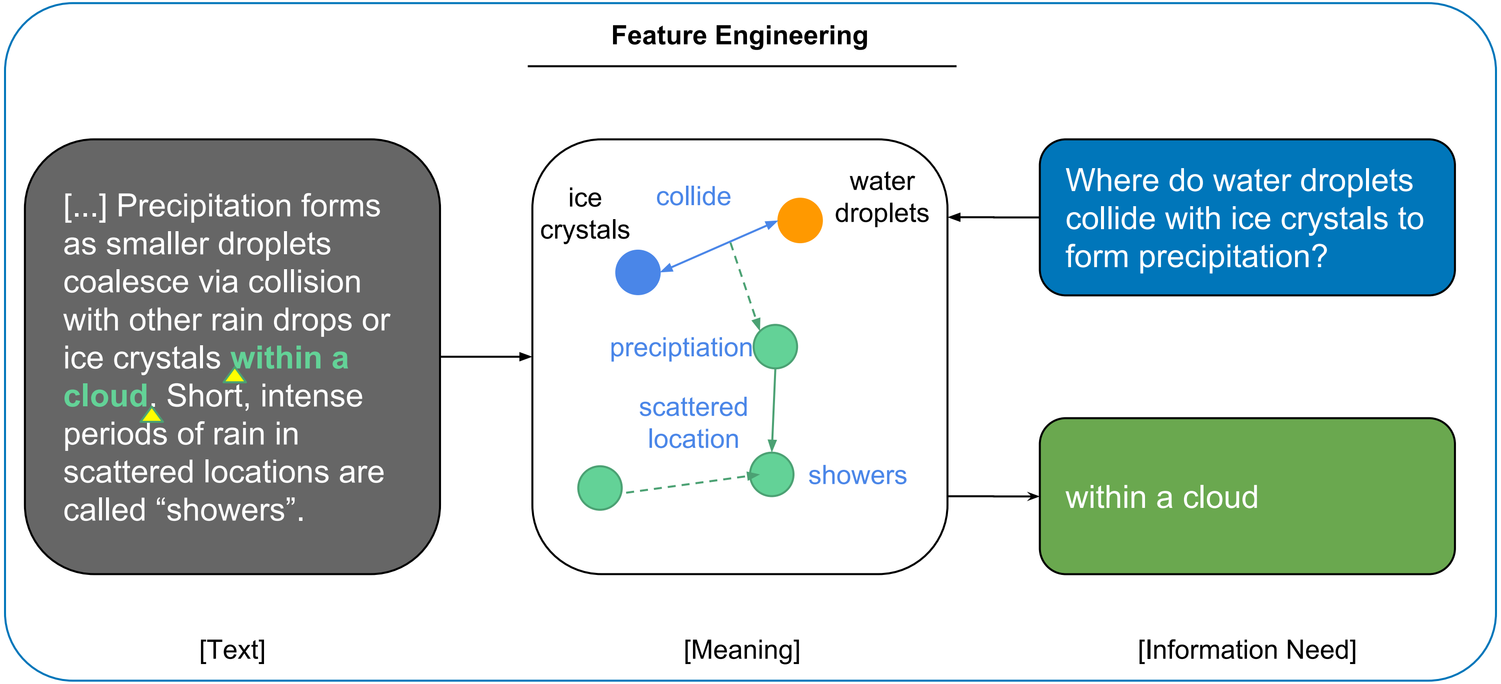
within a cloud

[Text]

[Meaning]

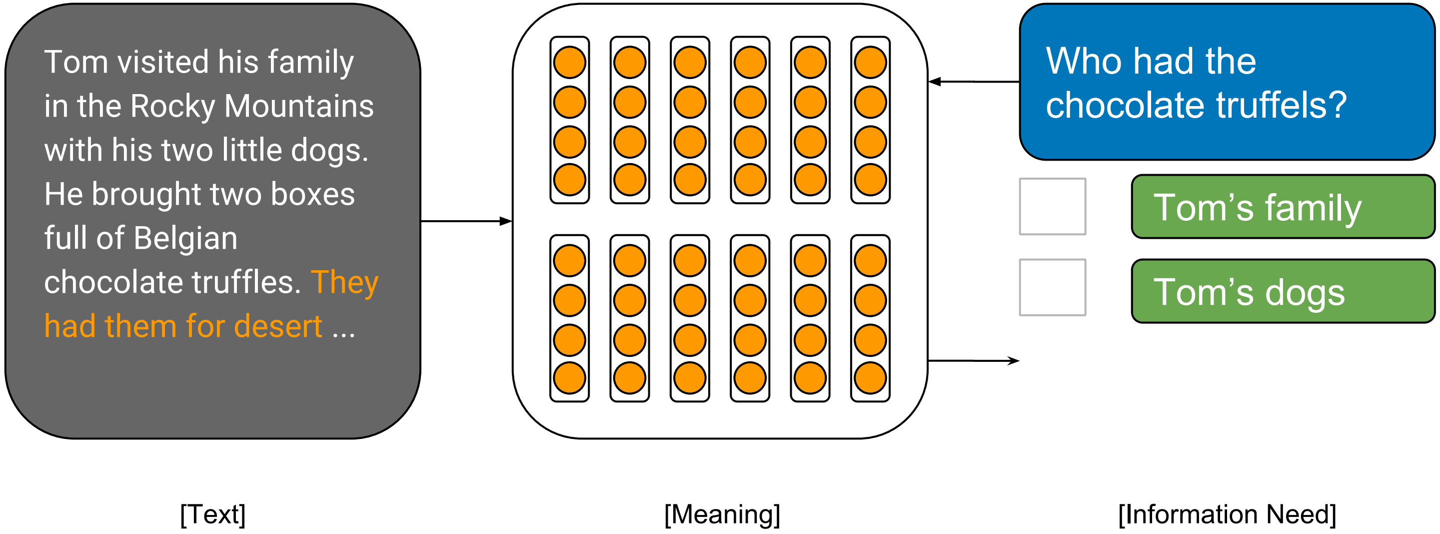
[Information Need]

Architecture Engineering



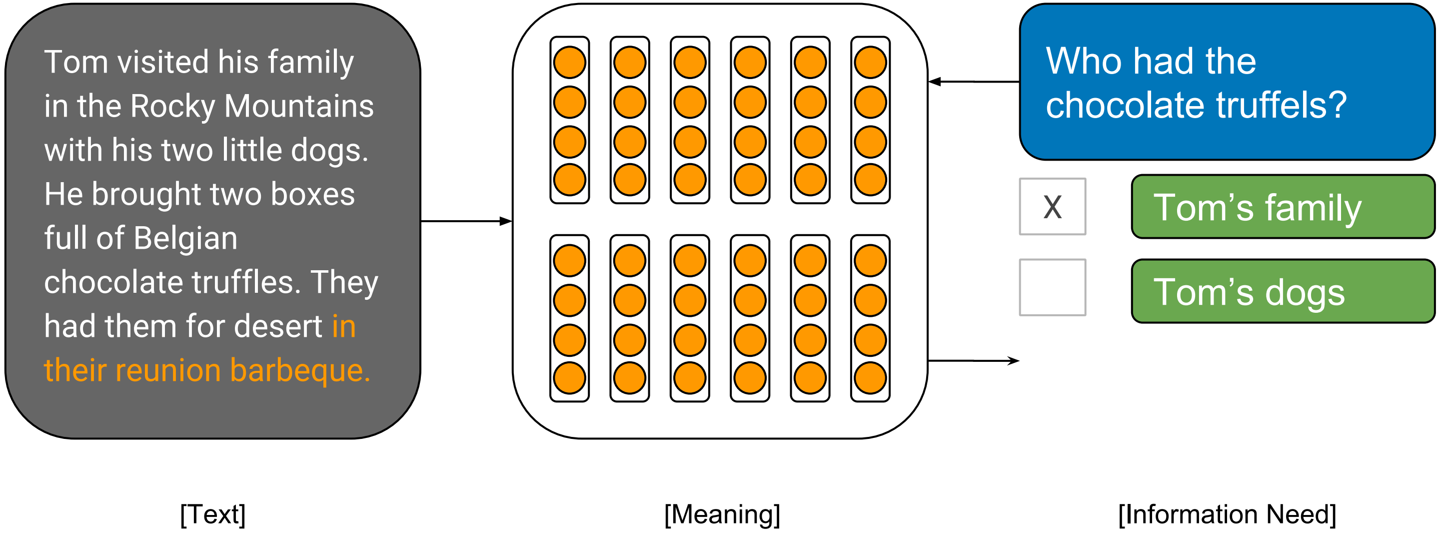
Challenge II: Ambiguity

References gradually become certain

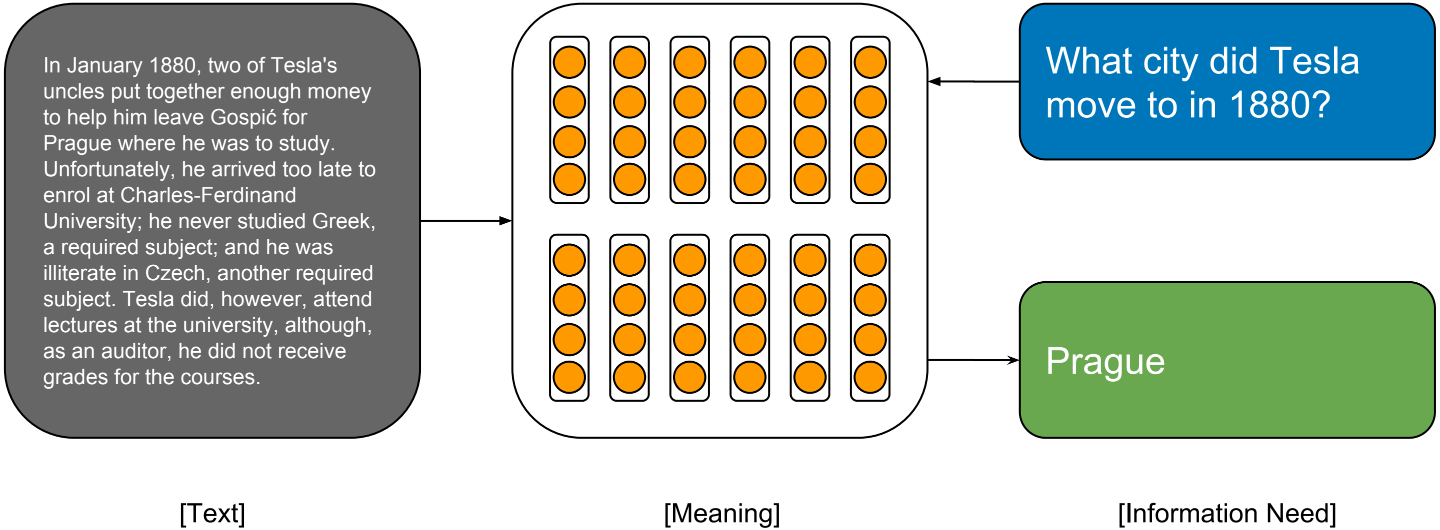


Challenge II: Ambiguity

References gradually become certain



End-to-end Machine Reading for Question Answering



Representing Words in Context

Why do we need compositional representations in QA?

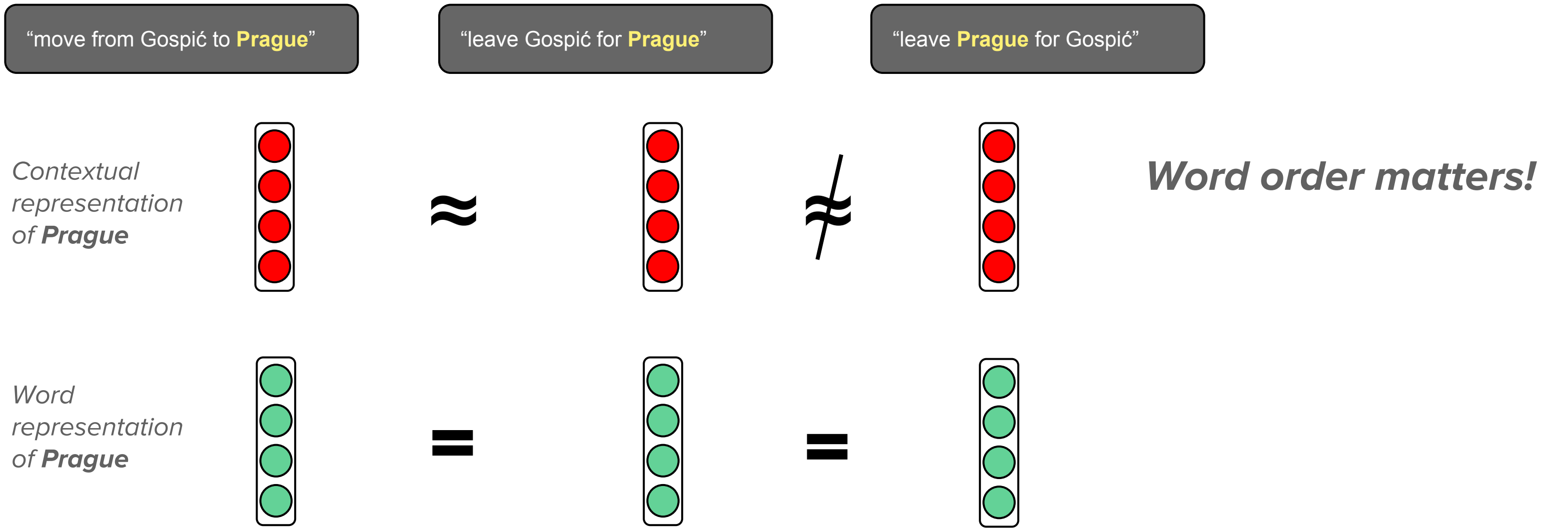
What **city** did Tesla move to in 1880?

In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for **Prague** where he was to study.

- **Goal**: similar representations for tokens in similar contexts, for instance through lexical / syntactic variation

"move from Gospić to **Prague**" ≈ "leave Gospić for **Prague**"

Similarity between contexts?



Word Similarity

"Words are defined by the company they keep."

➔ Two words are similar if they appear in the same documents.

Term-Document matrix:

	d1	**d2**	**d3**	**d4**	**...**	**d*M***
resident	2	0	0	0	...	1
street	0	1	0	1	...	0
city	4	2	0	1	...	1
...	...	...	...	...	...	...
town	1	1	0	1	...	1
mozarella	0	0	3	0	...	0
balsamico	0	0	1	0	...	0

Somewhat collinear,
but very sparse