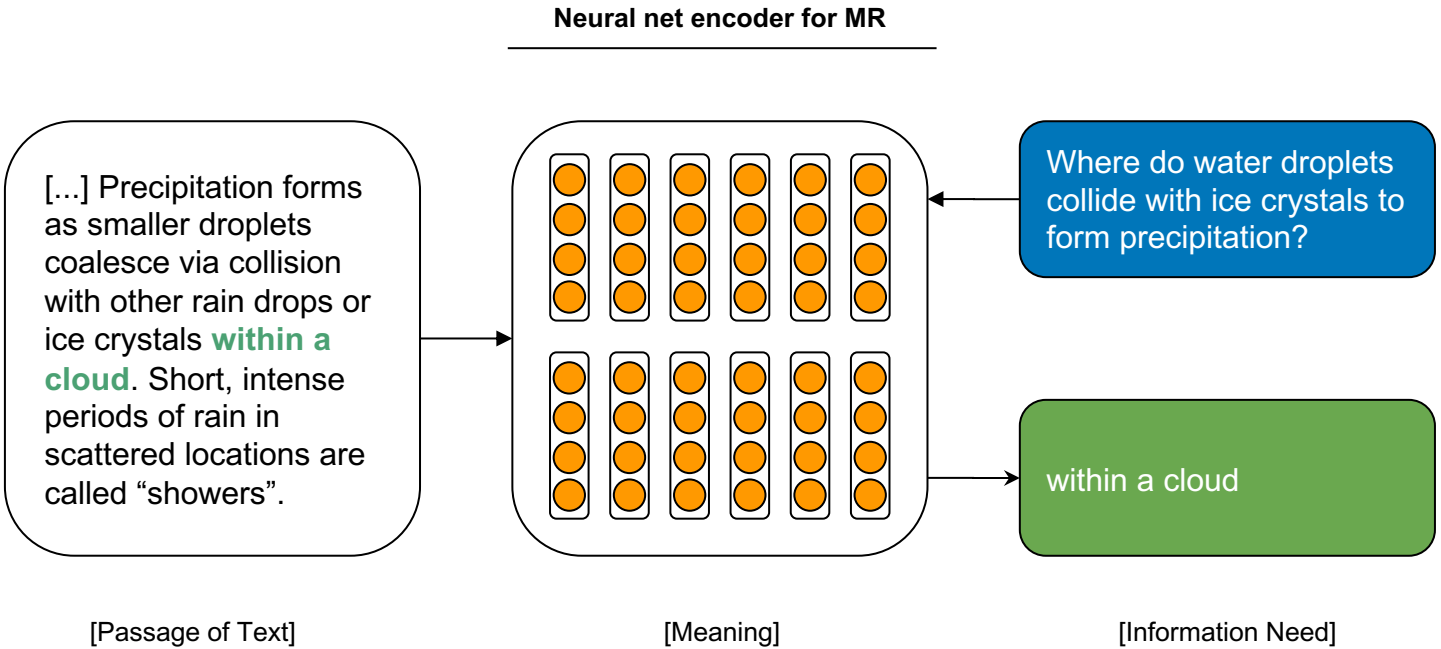
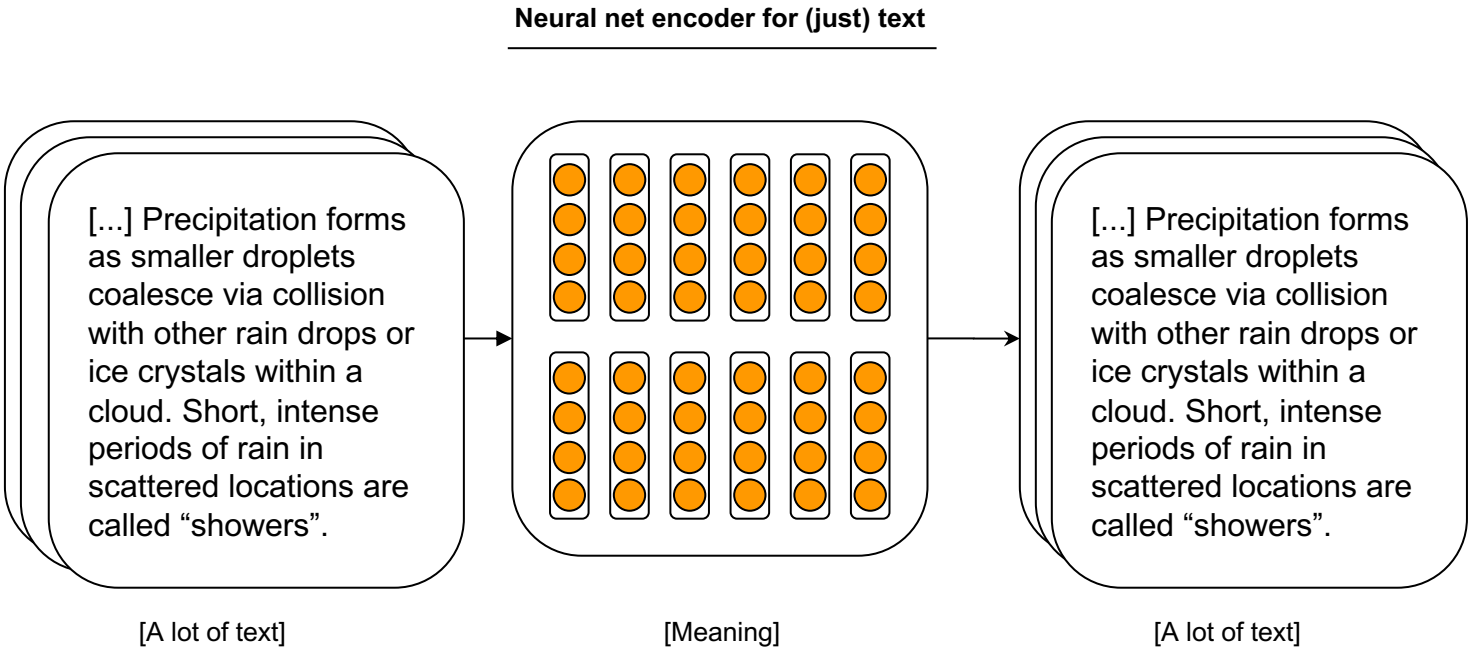
# Machine Reading / Current Trend

# Supervised training

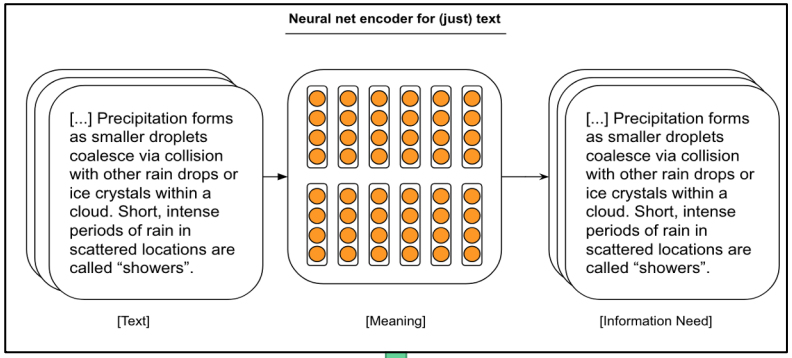


# Unsupervised pretrained representations



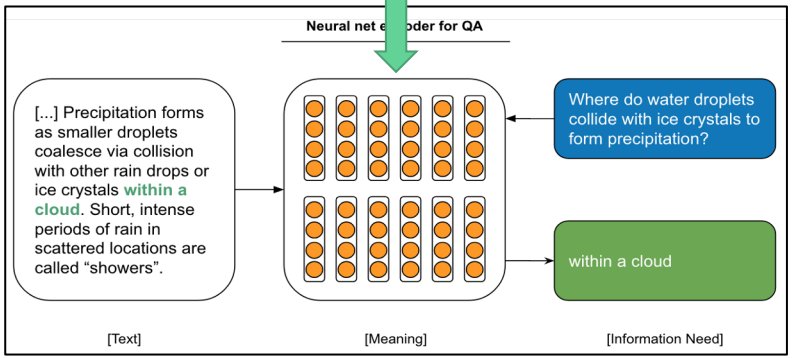
# Lifting over pretrained representations

**Pretrained Language Model**




Transfer

**Machine Reading**



# How is this different from pretrained word embeddings?

**Pretrained Word Embeddings (word2vec)**

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

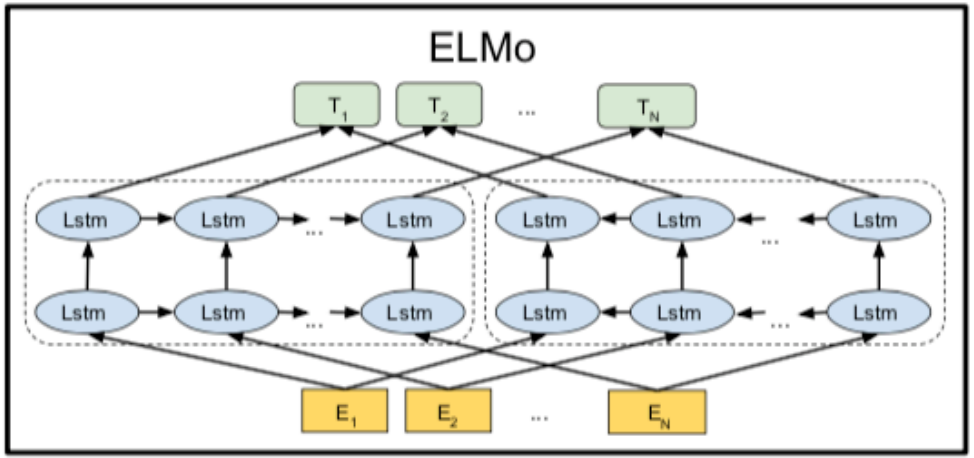
**Pretrained Contextualized Embeddings (e.g. ELMo, BERT)**

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

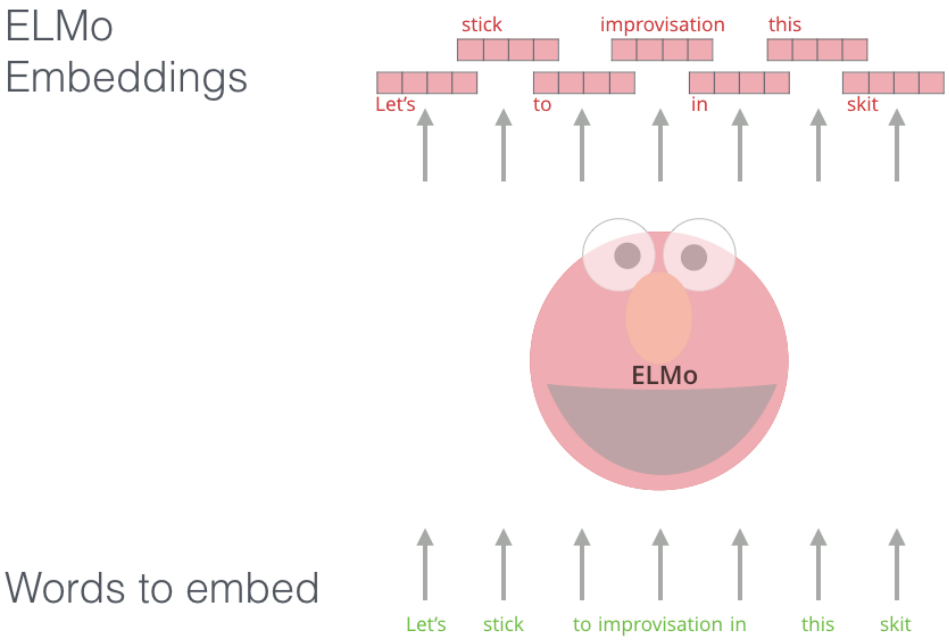
# ELMo: **E**mbeddings from **L**anguage **Mo**dels

Peters et al., NAACL'18

- Train a BiLSTM for Bidirectional language modeling on a large dataset
- Run the sentence to encode through both forward and backward LSTMs
- Combine forward and backward representations into final contextual embeddings



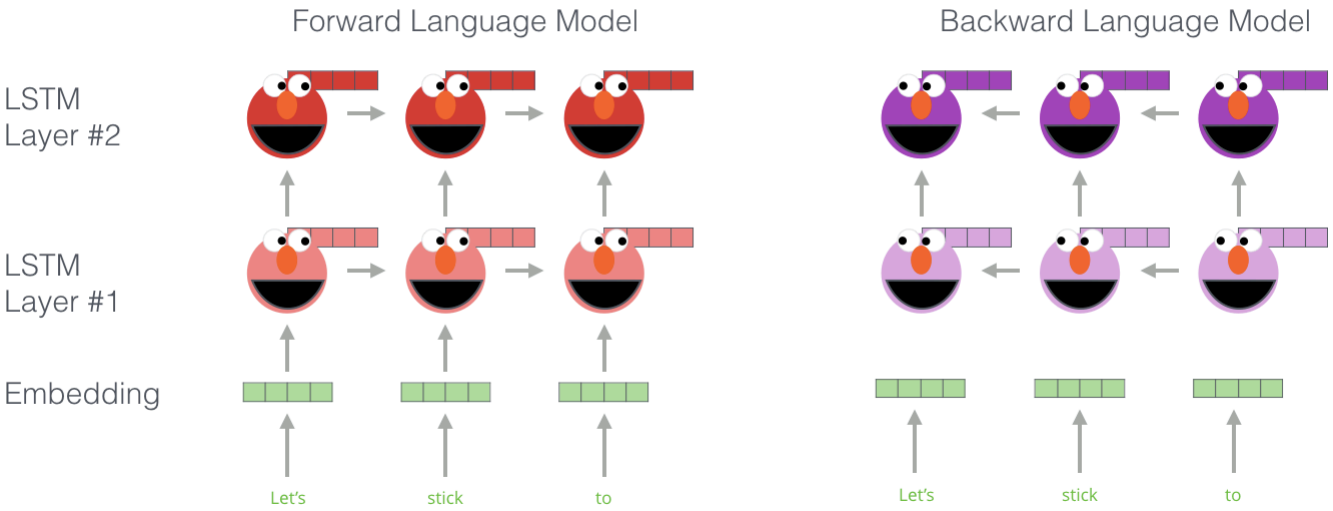
# ELMo: **E**mbeddings from **L**anguage **Mo**dels




Figures from http://jalammar.github.io/illustrated-bert/

# ELMo: **E**mbeddings from **L**anguage **Mo**dels

Embedding of “stick” in “Let’s stick to” - Step #1




Figures from http://jalammar.github.io/illustrated-bert/

# ELMo: **E**mbeddings from **L**anguage **Mo**dels

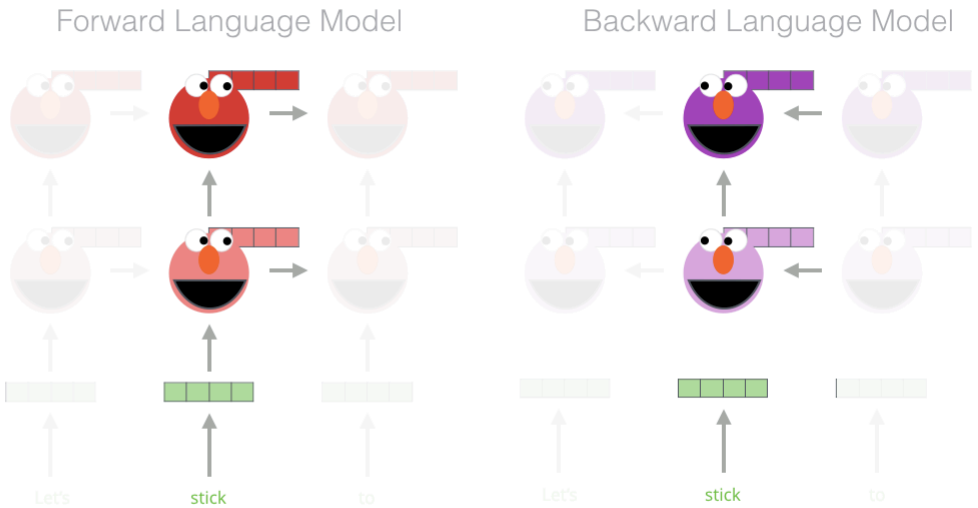
Embedding of “stick” in “Let’s stick to” - Step #2

1- Concatenate hidden layers

2- Multiply each vector by a weight based on the task


3- Sum the (now weighted) vectors

ELMo embedding of “stick” for this task in this context




Figures from http://jalammar.github.io/illustrated-bert/

# ELMo performance

| | Task | Previous SOTA | | Our baseline | ELMo + baseline | Increase (absolute/ relative) |
|---|---|---|---|---|---|---|
| **Machine Reading-** | SQuAD | Liu et al. (2017) | 84.4 | 81.1 | 85.8 | 4.7 / 24.9% |
| Textual Entailment - | SNLI | Chen et al. (2017) | 88.6 | 88.0 | 88.7 ± 0.17 | 0.7 / 5.8% |
| Semantic Labeling - | SRL | He et al. (2017) | 81.7 | 81.4 | 84.6 | 3.2 / 17.2% |
| Coreference Resolution - | Coref | Lee et al. (2017) | 67.2 | 67.2 | 70.4 | 3.2 / 9.8% |
| Entity Extraction - | NER | Peters et al. (2017) | 91.93 ± 0.19 | 90.15 | 92.22 ± 0.10 | 2.06 / 21% |
| Sentiment Analysis - | SST-5 | McCann et al. (2017) | 53.7 | 51.4 | 54.7 ± 0.5 | 3.3 / 6.8% |

# What is ELMo learning ?

- Meaning of words in context
  - POS, word sense, etc.

| | Source | Nearest Neighbors |
|---|---|---|
| GloVe | play | playing, game, games, played, players, plays, player, Play, football, multiplayer |
| biLM | Chico Ruiz made a spectacular play on Alusik 's grounder {...} | Kieffer , the only junior in the group , was commended for his ability to hit in the clutch , as well as his all-round excellent play . |
| | Olivia De Havilland signed to do a Broadway play for Garson {...} | {...} they were actors who had been handed fat roles in a successful play , and had talent enough to fill the roles competently , with nice understatement . |

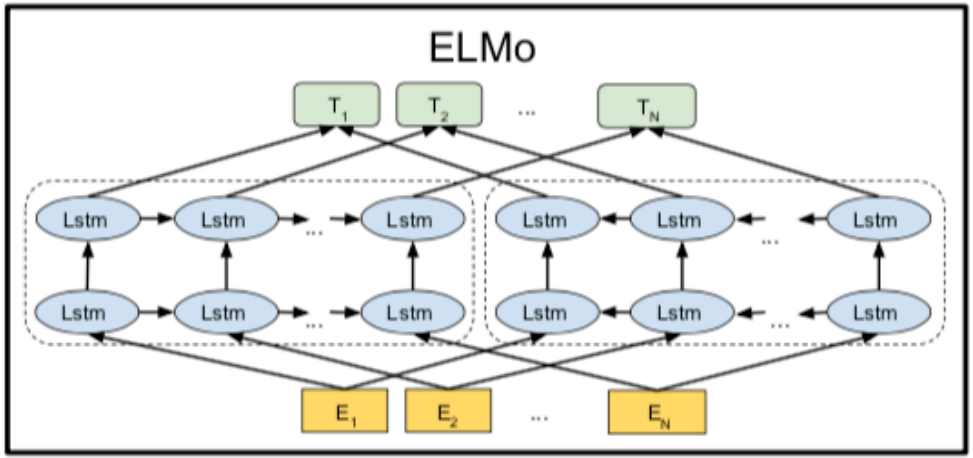
# Problems with ELMo

- Need to use different architectures for different tasks
- Retraining models is slow, transfer learning is fast
- Need to deal with long term dependencies in LSTMs!

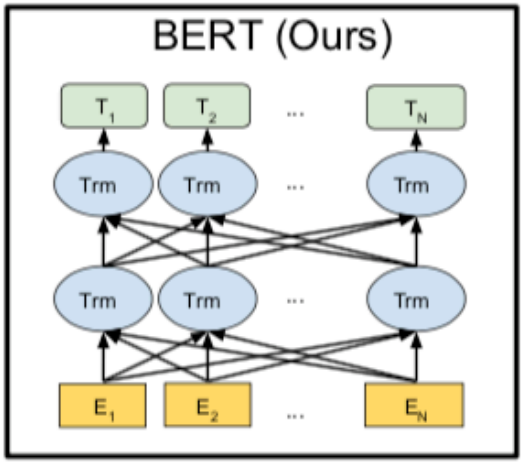
# BERT - **B**idirectional **E**ncoder **R**epresentations from **T**ransformers

Devlin et al., NAACL'19

Solutions: use Transformer + encoder layers instead of decoder layers




(OpenAI GPT)

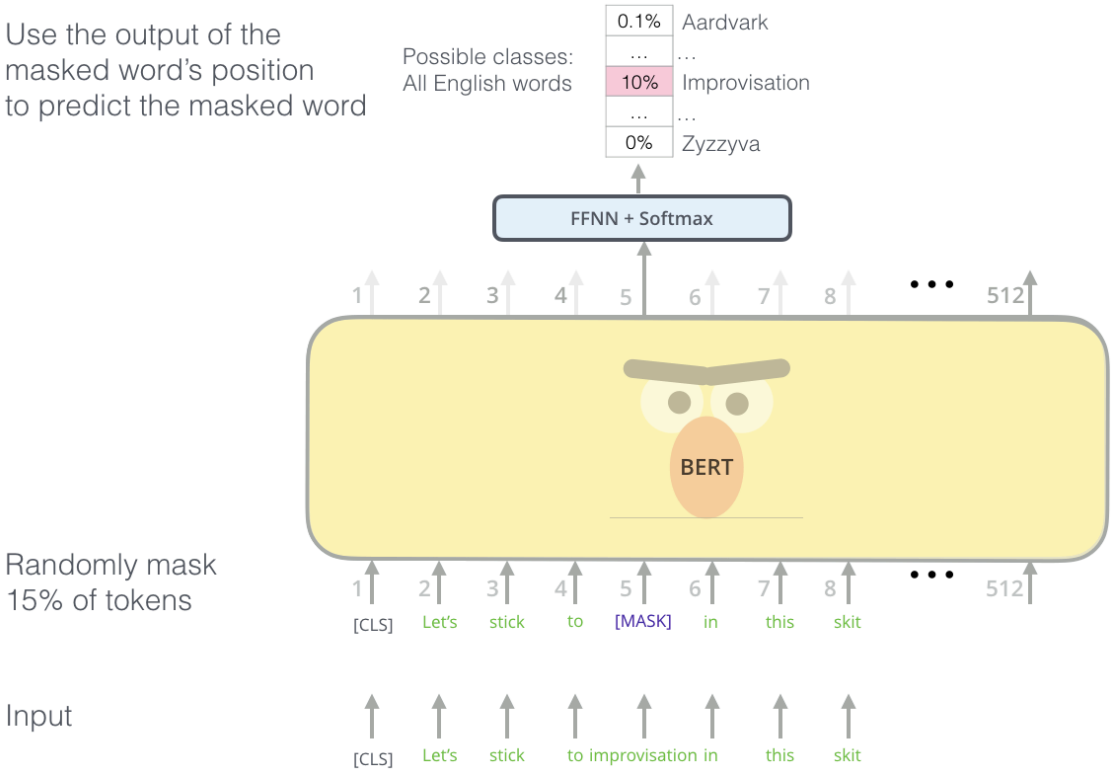



Innovation with multiple pretraining tasks

# BERT – Pretraining 1: masked language modeling

- Given a sentence with some words masked at random, can we predict them?
- Randomly select 15% of tokens to be replaced with “<MASK>”

# BERT – Pretraining 1: masked language modeling

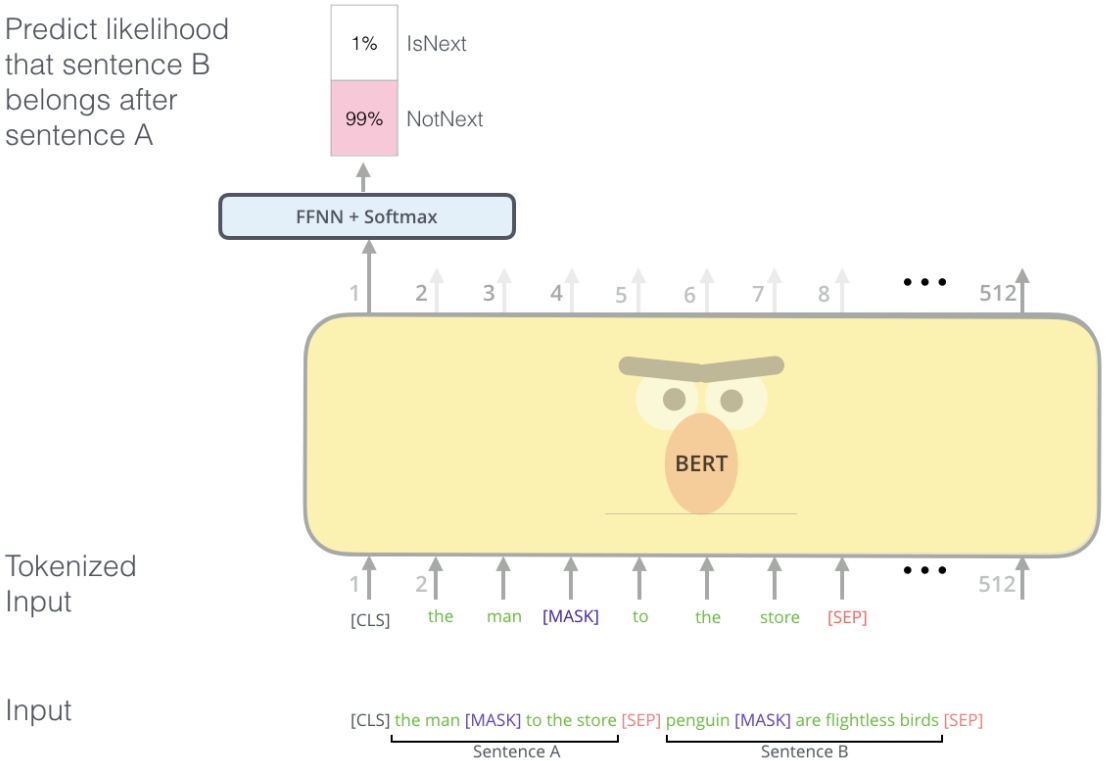



Figures from http://jalammar.github.io/illustrated-bert/

# BERT – Pretraining 2: next sentence prediction

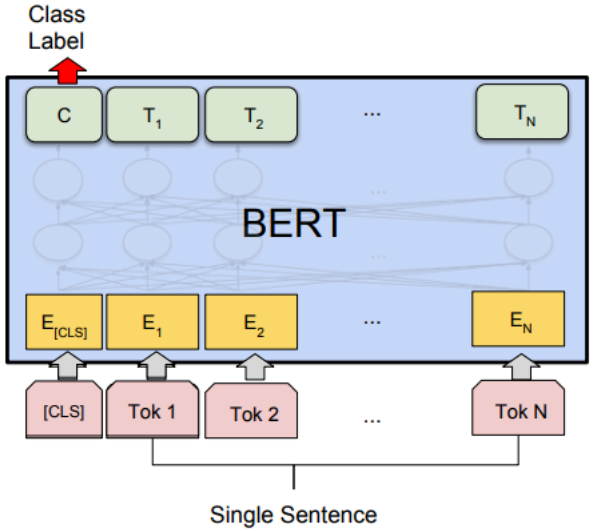
- Given two sentences, does the first follow the second?
- Teaches BERT about relationship between two sentences
- 50% of the time the actual next sentence, 50% random

# BERT – Pretraining 2: next sentence prediction



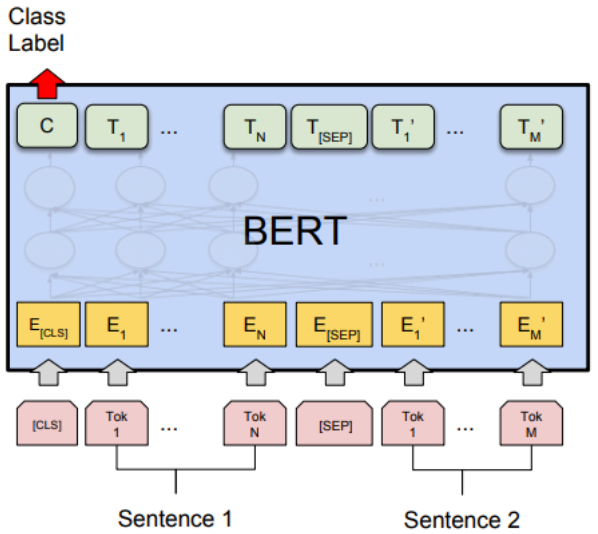

Figures from http://jalammar.github.io/illustrated-bert/

# BERT – Fine-tuning for Classification




Single sentence classification

Sentiment analysis, spam detection, etc.

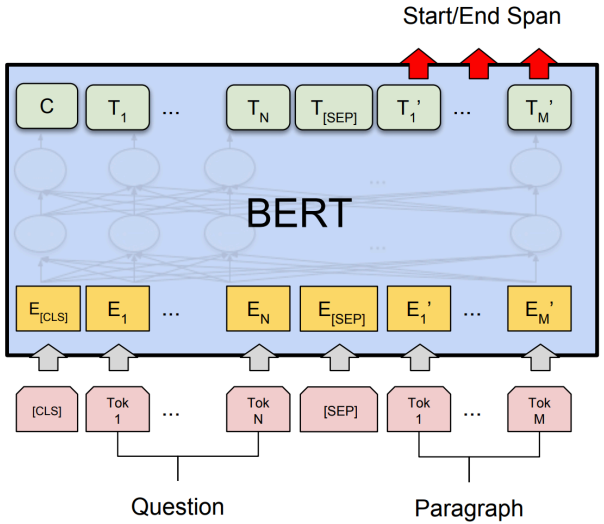



Pair of sentences classification

Entailment, paraphrase detection, etc.

Figures from Devlin et al. 18'

# BERT – Fine-tuning for Machine Reading




(c) Question Answering Tasks: SQuAD v1.1

| System | Dev EM | Dev F1 | Test EM | Test F1 |
|---|---|---|---|---|
| Leaderboard (Oct 8th, 2018) | | | | |
| Human | - | - | 82.3 | 91.2 |
| #1 Ensemble - nlnet | - | - | 86.0 | 91.7 |
| #2 Ensemble - QANet | - | - | 84.5 | 90.5 |
| #1 Single - nlnet | - | - | 83.5 | 90.1 |
| #2 Single - QANet | - | - | 82.5 | 89.3 |
| Published | | | | |
| BiDAF+ELMo (Single) | - | 85.8 | - | - |
| R.M. Reader (Single) | 78.9 | 86.3 | 79.5 | 86.6 |
| R.M. Reader (Ensemble) | 81.2 | 87.9 | 82.3 | 88.5 |
| Ours | | | | |
| $\text{BERT}_{\text{BASE}}$ (Single) | 80.8 | 88.5 | - | - |
| $\text{BERT}_{\text{LARGE}}$ (Single) | 84.1 | 90.9 | - | - |
| $\text{BERT}_{\text{LARGE}}$ (Ensemble) | 85.8 | 91.8 | - | - |
| $\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA) | **84.2** | **91.1** | **85.1** | **91.8** |
| $\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA) | **86.2** | **92.2** | **87.4** | **93.2** |

Figures from Devlin et al. 18'

# RoBERTa / ALE

## Derivatives from BE

& RoBERTa

| Rank | Model | EM | F1 |
|---|---|---|---|
| | Human Performance<br>*Stanford University*<br>(Rajpurkar & Jia et al. '18) | 86.831 | 89.452 |
| 1<br>Mar 12, 2020 | ALBERT + DAAF + Verifier (ensemble)<br>*PINGAN Omni-Sinitic* | **90.386** | **92.777** |
| 2<br>Jan 10, 2020 | Retro-Reader on ALBERT (ensemble)<br>*Shanghai Jiao Tong University*<br>http://arxiv.org/abs/2001.09694 | 90.115 | 92.580 |
| 3<br>Nov 06, 2019 | ALBERT + DAAF + Verifier (ensemble)<br>*PINGAN Omni-Sinitic* | 90.002 | 92.425 |
| 4<br>Sep 18, 2019 | ALBERT (ensemble model)<br>*Google Research & TTIC*<br>https://arxiv.org/abs/1909.11942 | 89.731 | 92.215 |
| 4<br>Feb 25, 2020 | Albert_Verifier_AA_Net (ensemble)<br>*QIANXIN* | 89.743 | 92.180 |
| 5<br>Jan 23, 2020 | albert+transform+verify (ensemble)<br>*qianxin* | 89.528 | 92.059 |
| 6<br>Mar 06, 2020 | ALBert-LSTM (ensemble)<br>*oppo.tensorlab* | 89.269 | 91.777 |
| 7<br>Dec 08, 2019 | ALBERT+Entailment DA (ensemble)<br>*CloudWalk* | 88.761 | 91.745 |

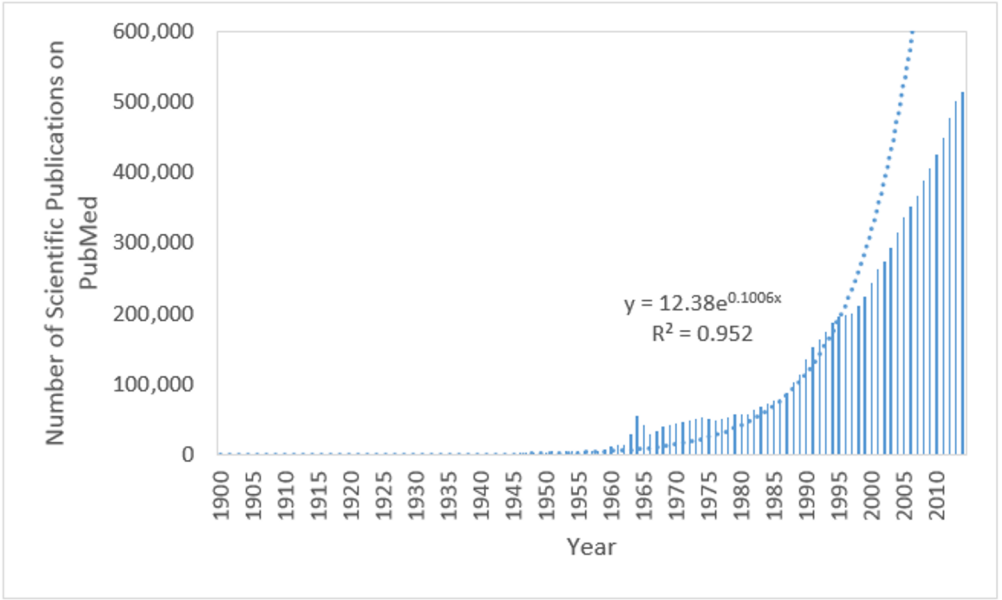
| Models | /High) |
|---|---|
| *Single model (from lead* | |
| BERT-large | 1) |
| XLNet | 2) |
| RoBERTa | 3) |
| UPM | |
| XLNet + SG-Net Verifi | |
| ALBERT (1M) | 1) |
| ALBERT (1.5M) | **5)** |
| *Ensembles (from leader* | |
| BERT-large | |
| XLNet + SG-Net Verifi | |
| UPM | |
| XLNet + DAAF + Veri | |
| DCMN+ | 3) |
| ALBERT | **6)** |

+5%

# Open Domain Question Answering

Is Machine Reading actually useful?

# Motivation 1: Information Overload



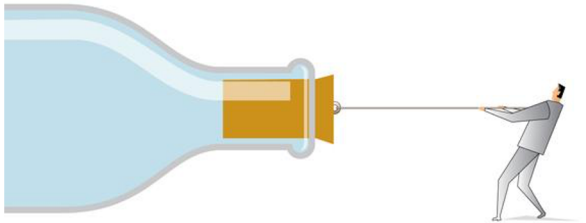


[Information Need]

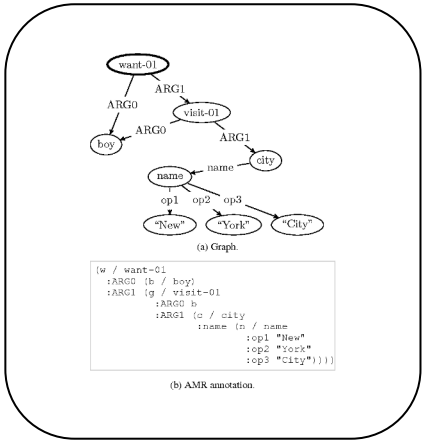
**uses for**

# Motivation 2: The Knowledge Acquisition Bottleneck

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



[Knowledge]




[Meaning]

**uses for**


[Information Need]

# Open domain Question Answering

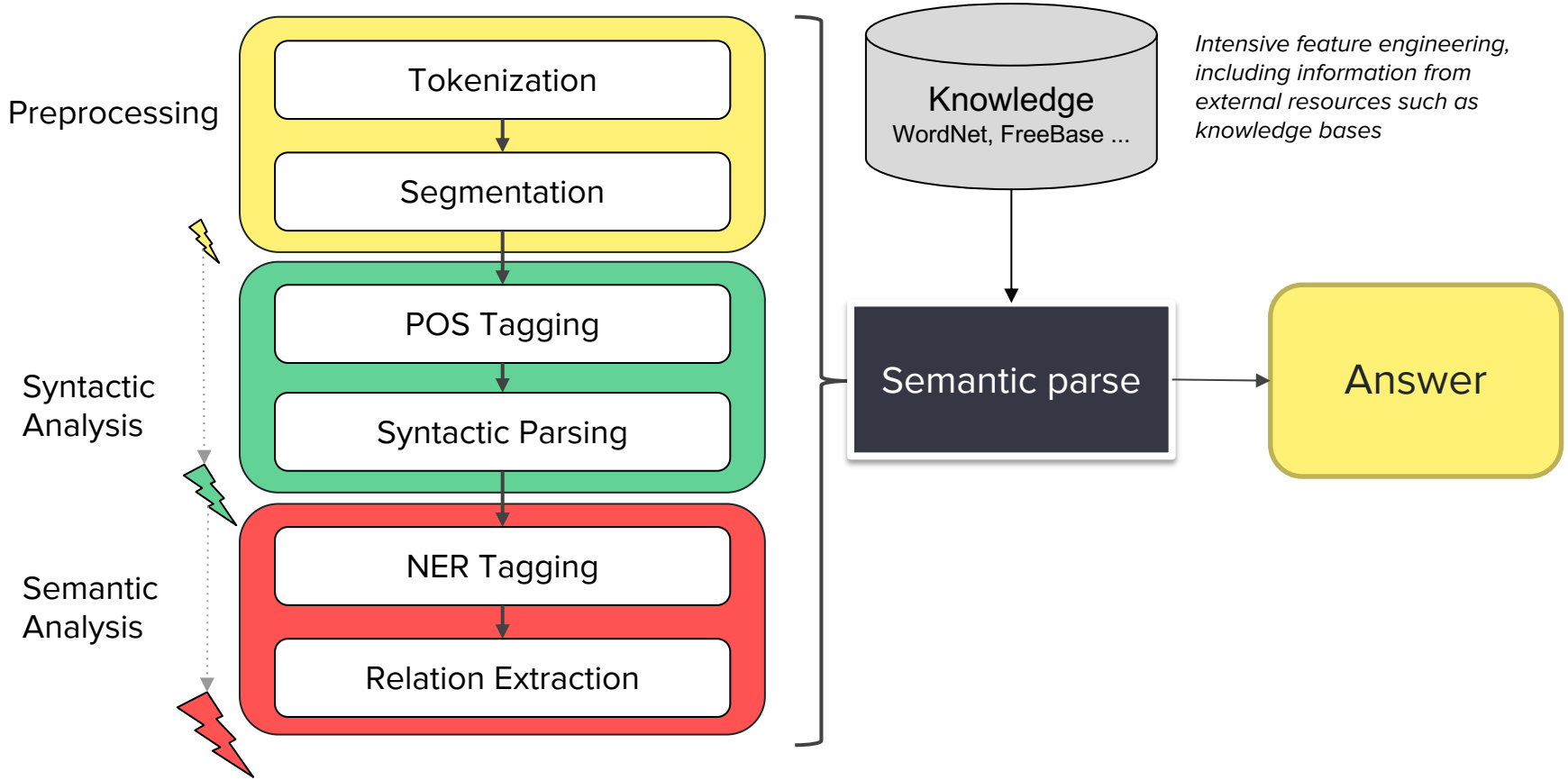
- Open domain QA: answer any question using very large knowledge sources
- Goes beyond Machines Reading that expects a paragraph to be given

- Open domain = question on any topic not a restricted subset

- In the following

1. Traditional approaches using Knowledge Bases
2. New approaches based on end-to-end Machine Reading

# “Traditional” NLP for open domain QA



# Semantic Parsing

Can We use ML to automate this

Ewan forgot the mozarella in his car

$\exists x0\ named(x0, ewan, person) \wedge$
$\exists x1\ mozzarella(x1) \wedge$
$\exists x2\ car(x2) \wedge of(x2,x0) \wedge in(x1, x2) \wedge$
$\exists e\ event(e) \wedge forget(e) \wedge agent(e, x0) \wedge$
$patient(e, x1)$


[Text] [Meaning] [Information Need]

Semantic parses are logical forms in PROLOG, SQL, SPARQL, etc.

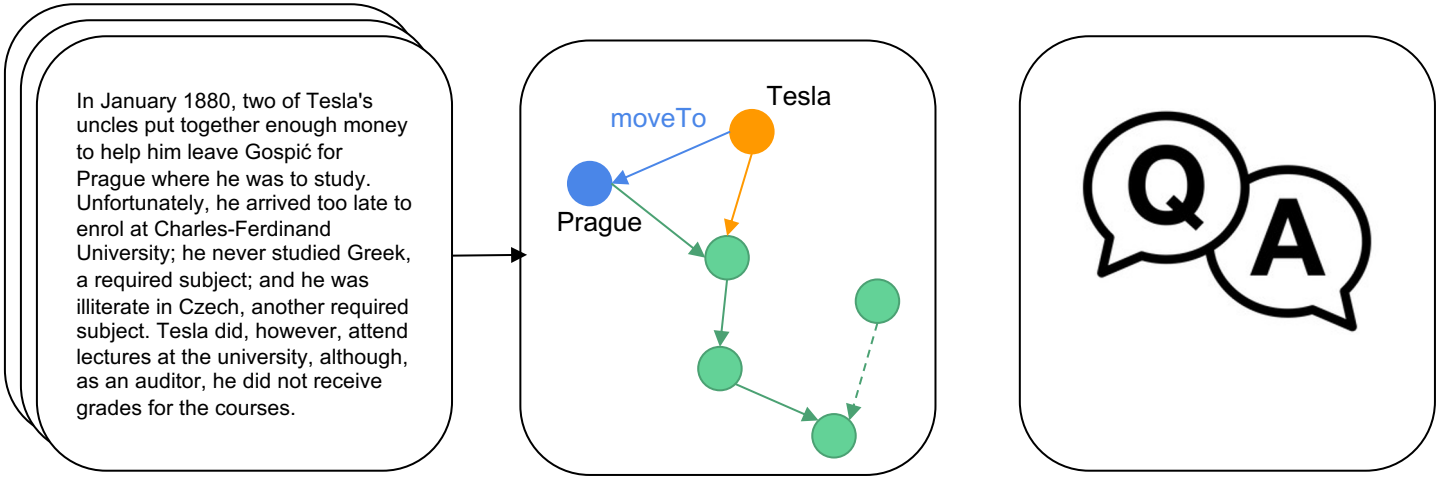
# Knowledge Bases

- KB: structured repository of knowledge (usually relational DB)
- Goal: encode knowledge so that it can be queried by semantic parses **efficiently**
- Scale can be huge: billions of facts, millions of entities

- KB can be generic or specific
- Examples: Cyc, WikiData, DBPedia, Google KG, GeneOntology, IMDB, etc.

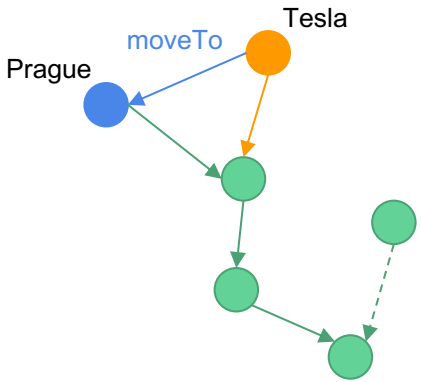
- Key challenge is their construction!
    - Manually: Crowdsourcing, paid experts
    - Automatically: Information extraction or Automatic KB Construction

# Automatic Knowledge Base Construction



# Knowledge Graph Construction

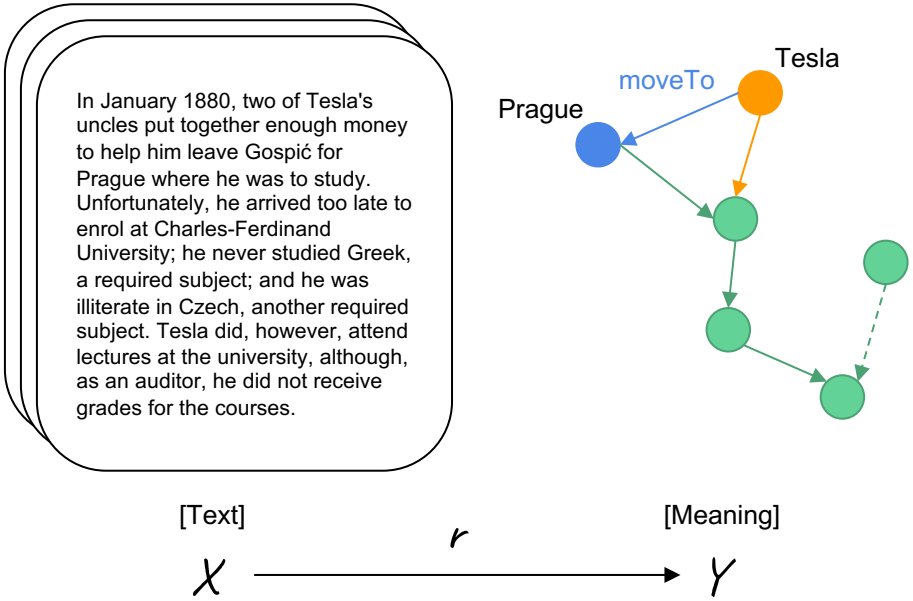
In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study. Unfortunately, he arrived too late to enrol at Charles-Ferdinand University; he never studied Greek, a required subject; and he was illiterate in Czech, another required subject.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



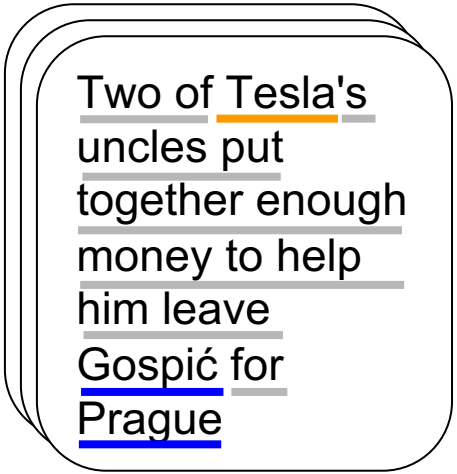

What city did Tesla move to in 1880?

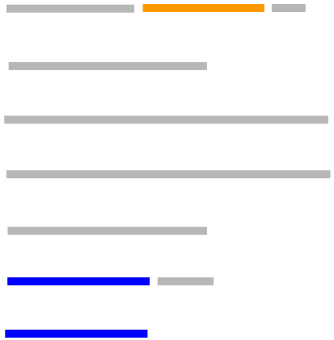
Prague

# Knowledge Graph Construction



# Entity Extraction

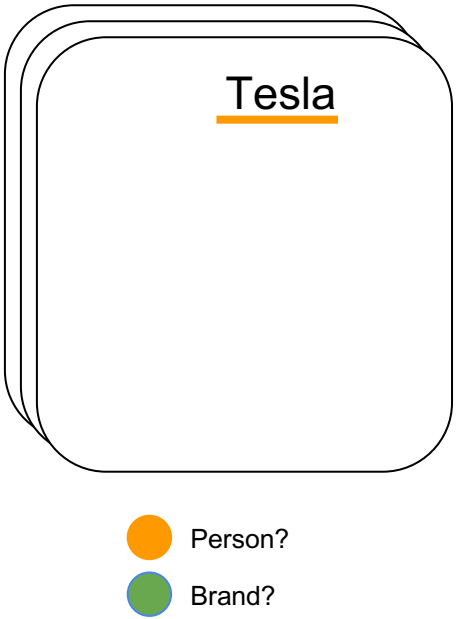


- Linear Chain CRF
- Bi-directional RNNs
- Hybrid RNN & CRFs

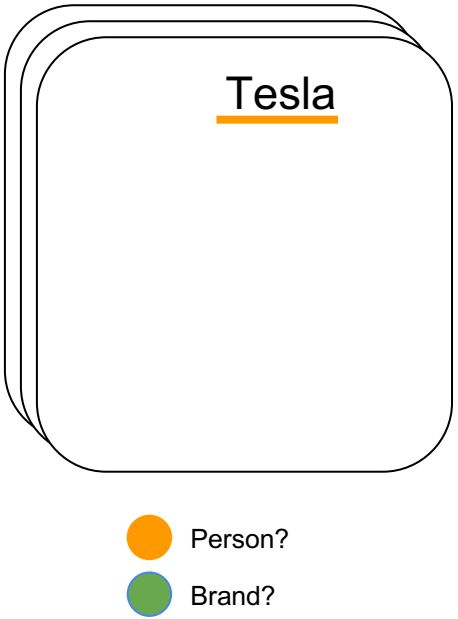
Person
Location

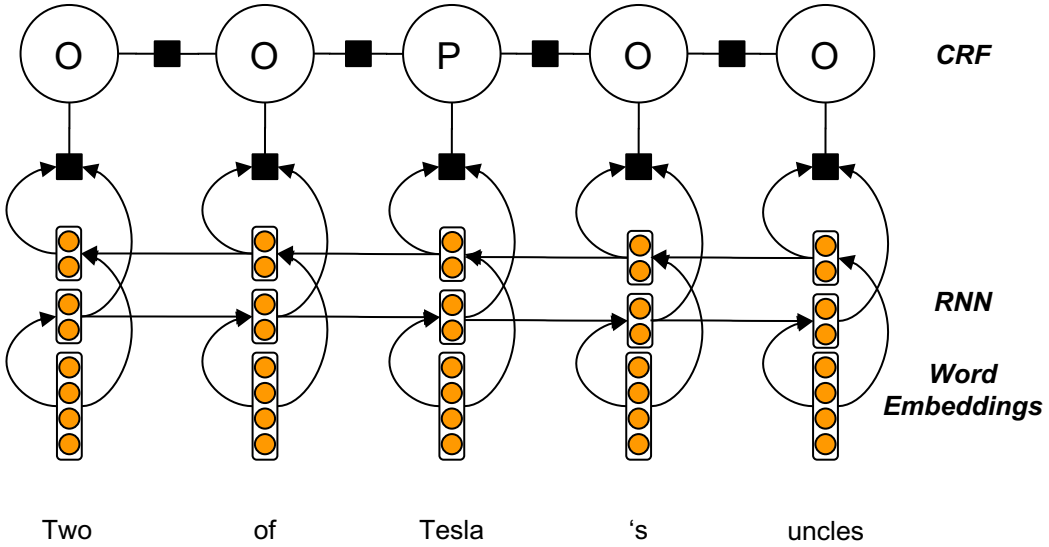
# Challenge: Ambiguity



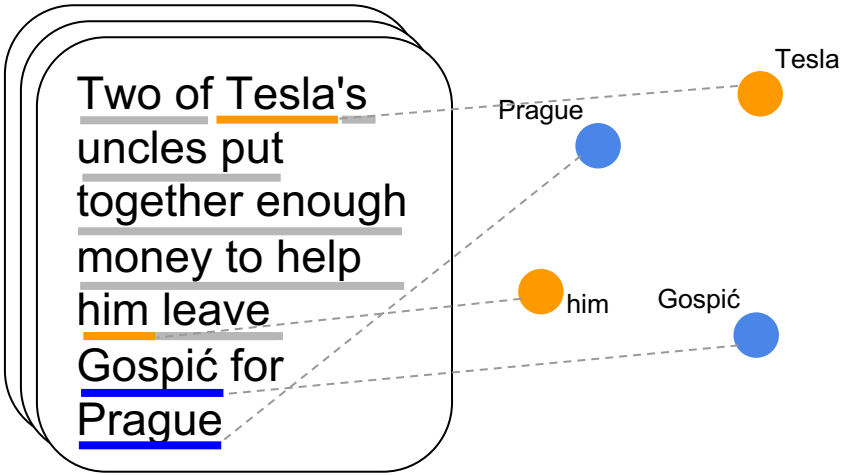
# Conditional Random Fields with RNN Potentials

Huang et al., 2015

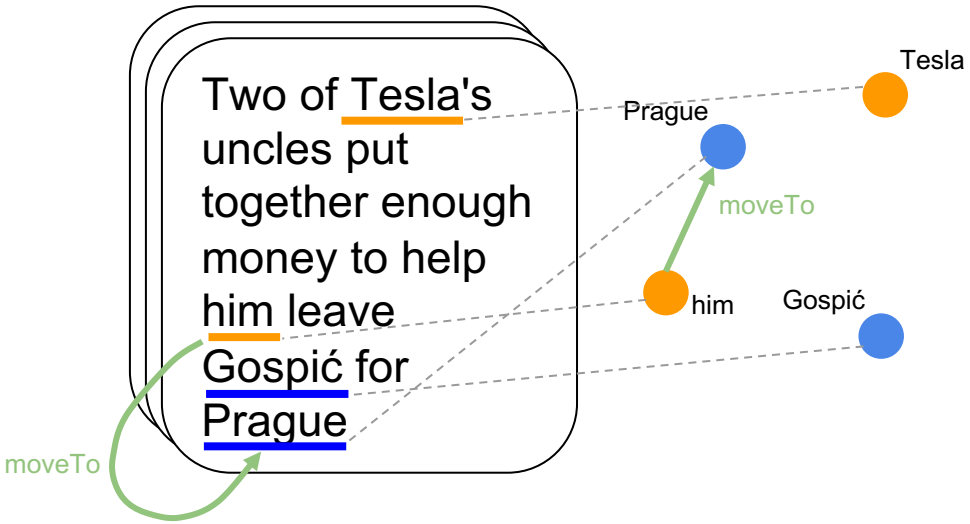


# Instantiate Nodes

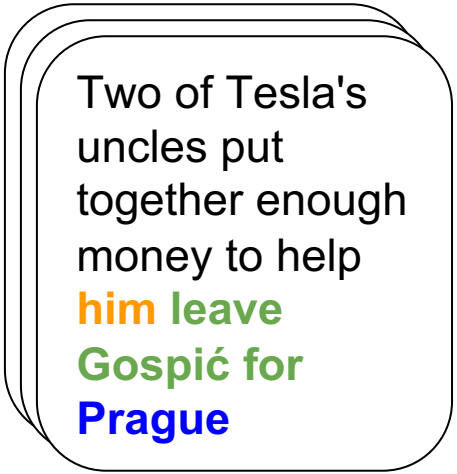


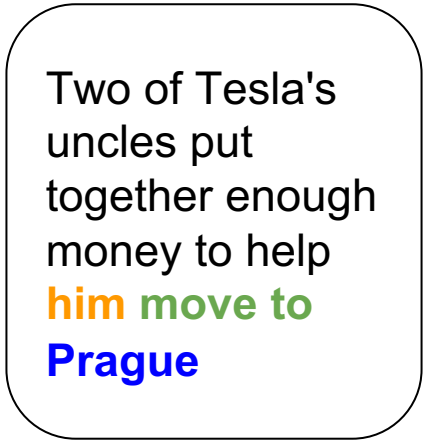
# Relation Extraction

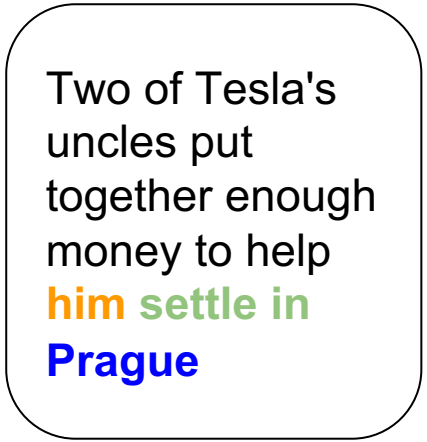



- Neural Classification
- Distant Supervision

# Challenge: Variation

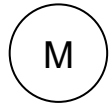


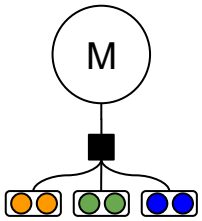



# Relation Classification

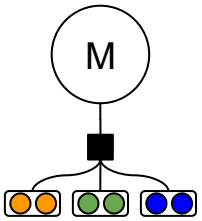
[Current SOTA neural RE model]



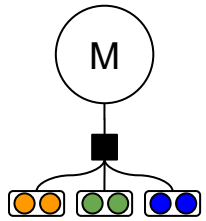

(Tesla, moveTo, Prague)


him leave Gospić for Prague



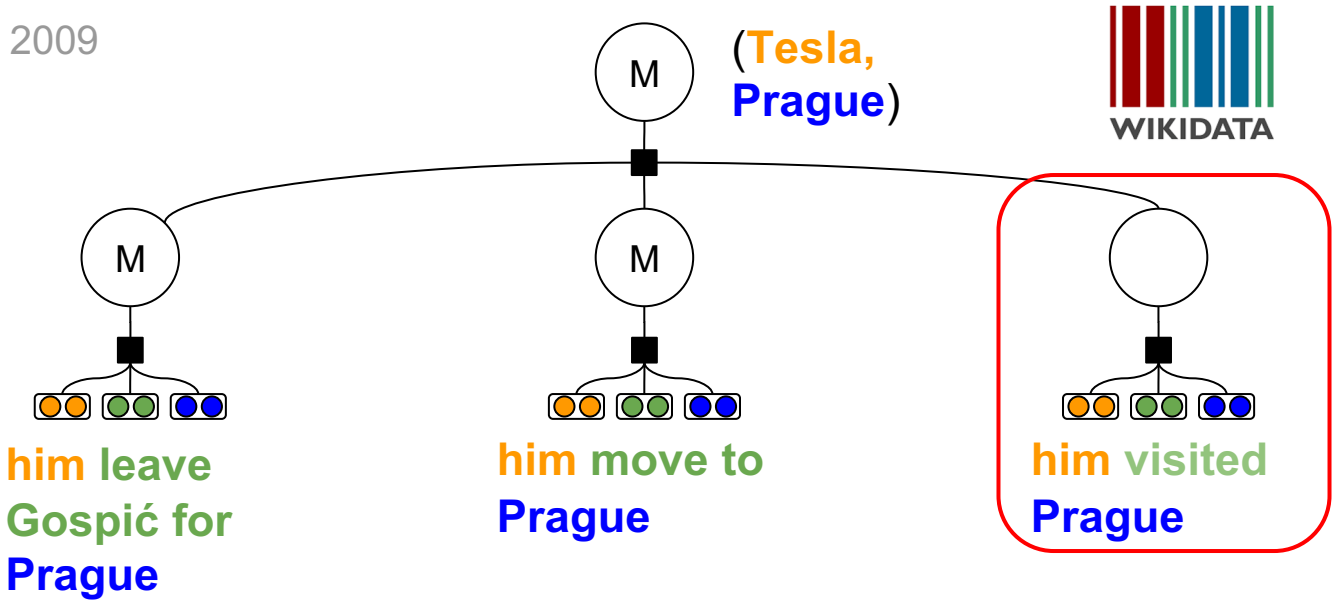

him move to Prague



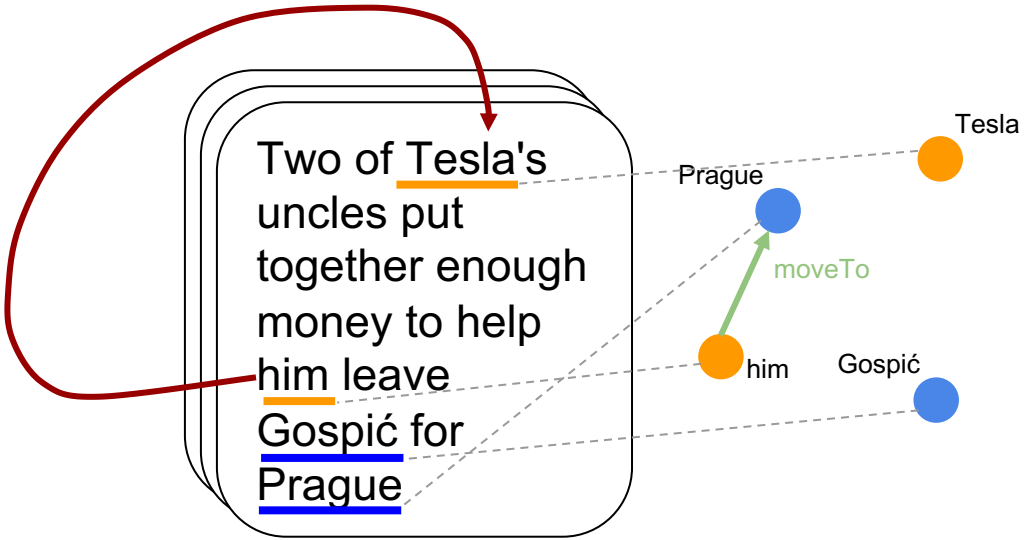

him settle in Prague

# Distant Supervision & Multiple Instance Learning

Mintz et al., 2009

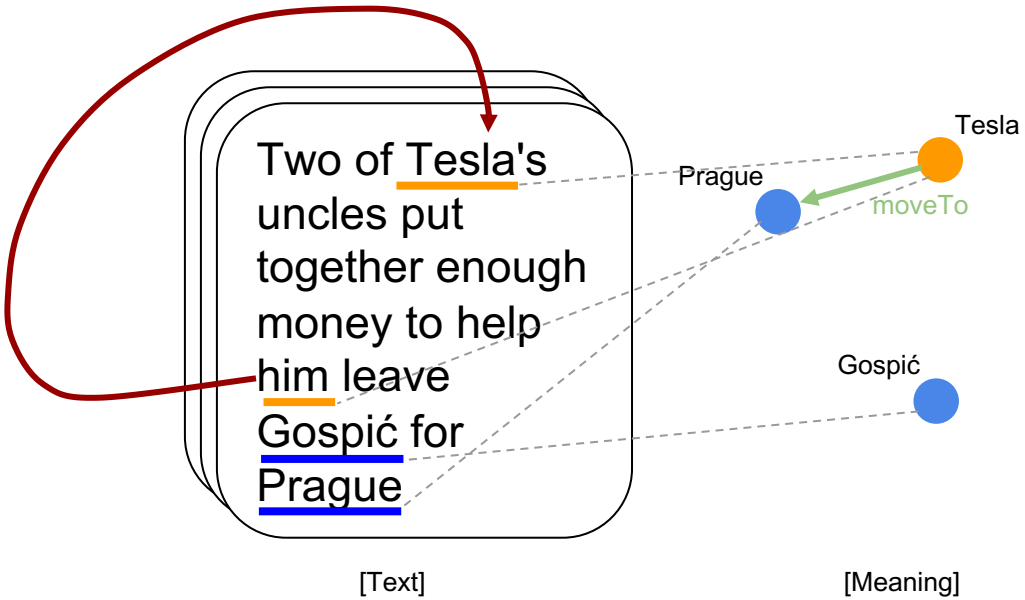


# Coreference Resolution

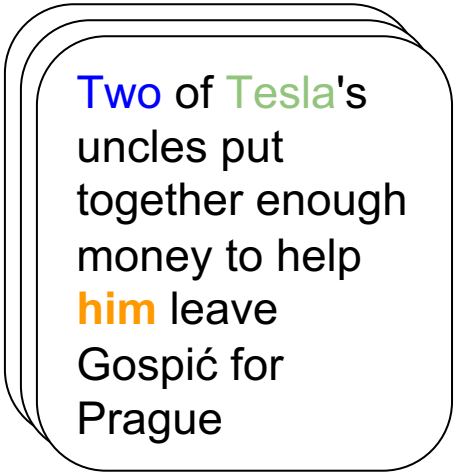



- Neural Classification
- Latent Variables

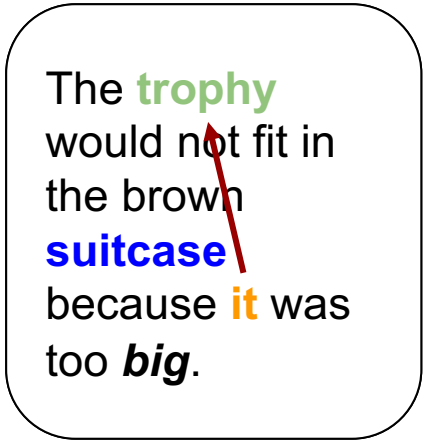
# Collapsing Nodes

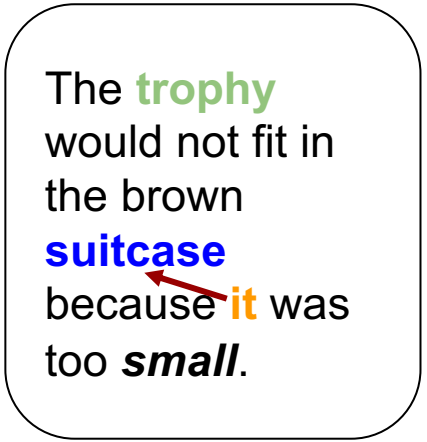


# Challenge: Common Sense



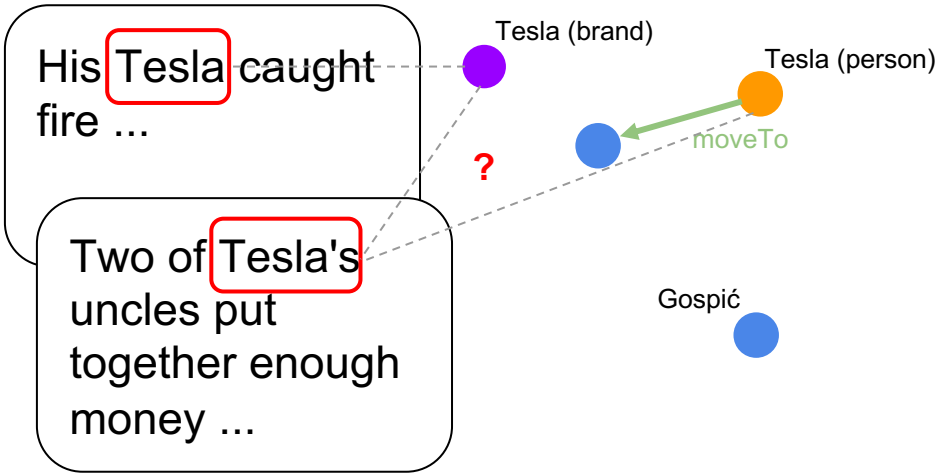

Surface



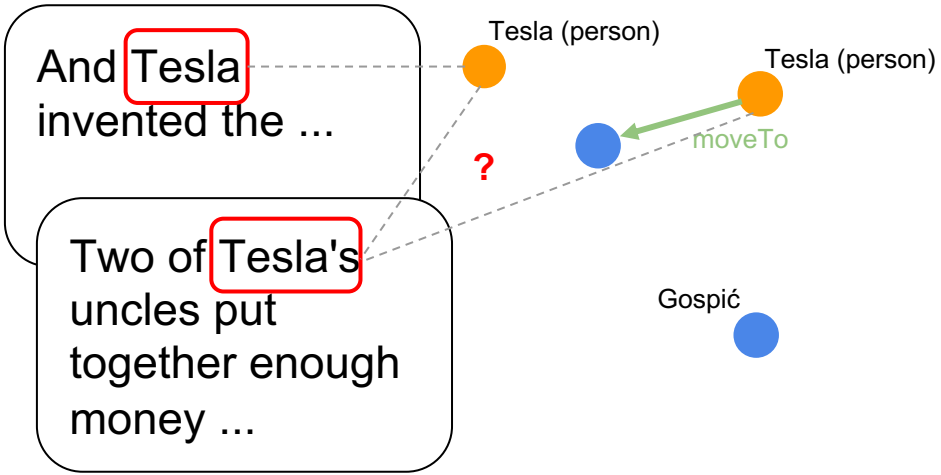

Common Sense

# Entity Linking



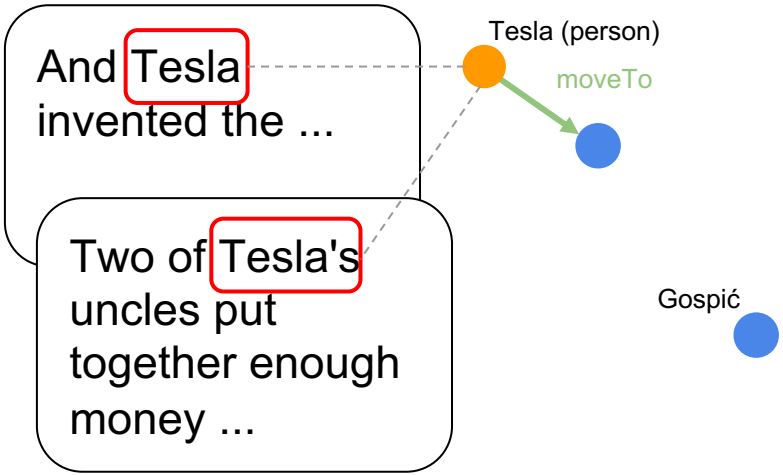

- Reranking ...
- Embeddings ...

# Entity Linking

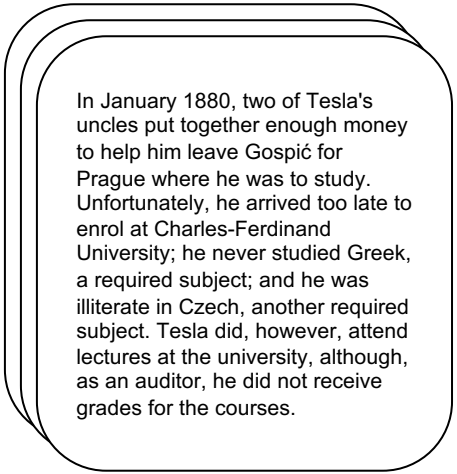


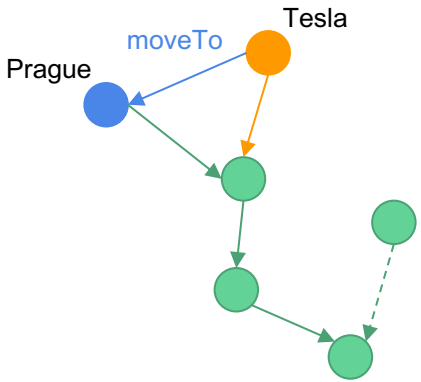

- Reranking ...
- Embeddings ...

# Collapsing


And Tesla invented the ...
Two of Tesla's uncles put together enough money ...
Tesla (person)
moveTo
Gospić

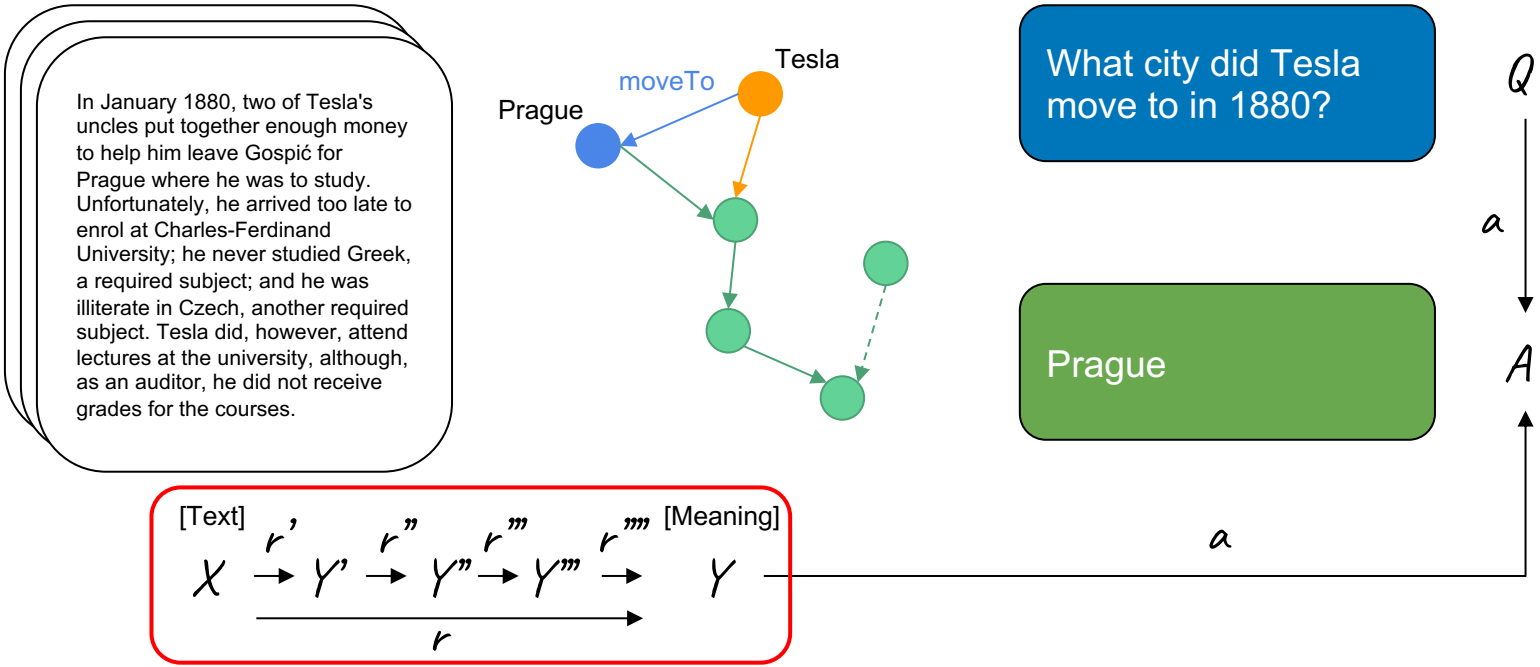
# Strengths




- Supports Reasoning
- Fast access
- Generalisation
- Interpretable
- Existing KBs can serve as supervision signal!

# Weakness: Cascading errors

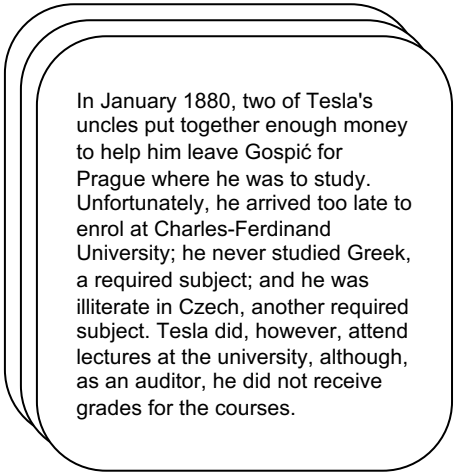


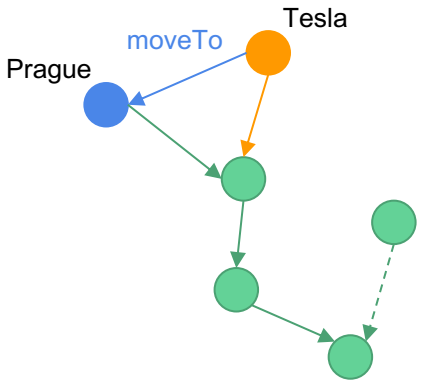
# Weakness: Cascading errors

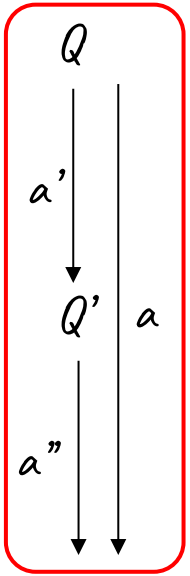
What city did Tesla move to in 1880?

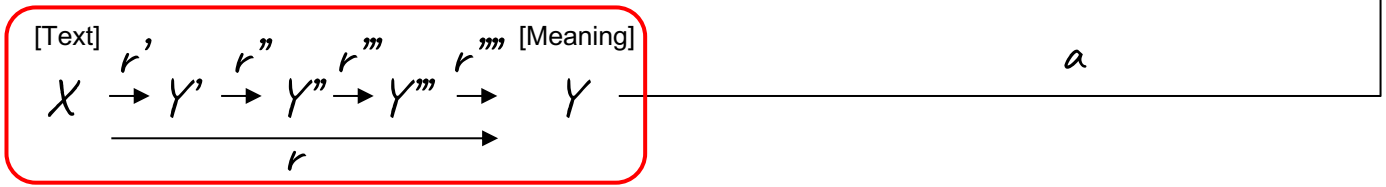
moveTo(Tesla,X)?

Prague



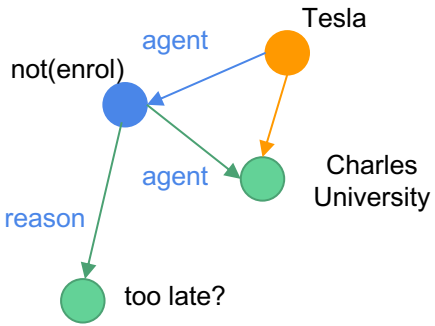



# Weakness: Engineering Schemas and Formalisms

Unfortunately, he arrived too late to enrol at Charles University



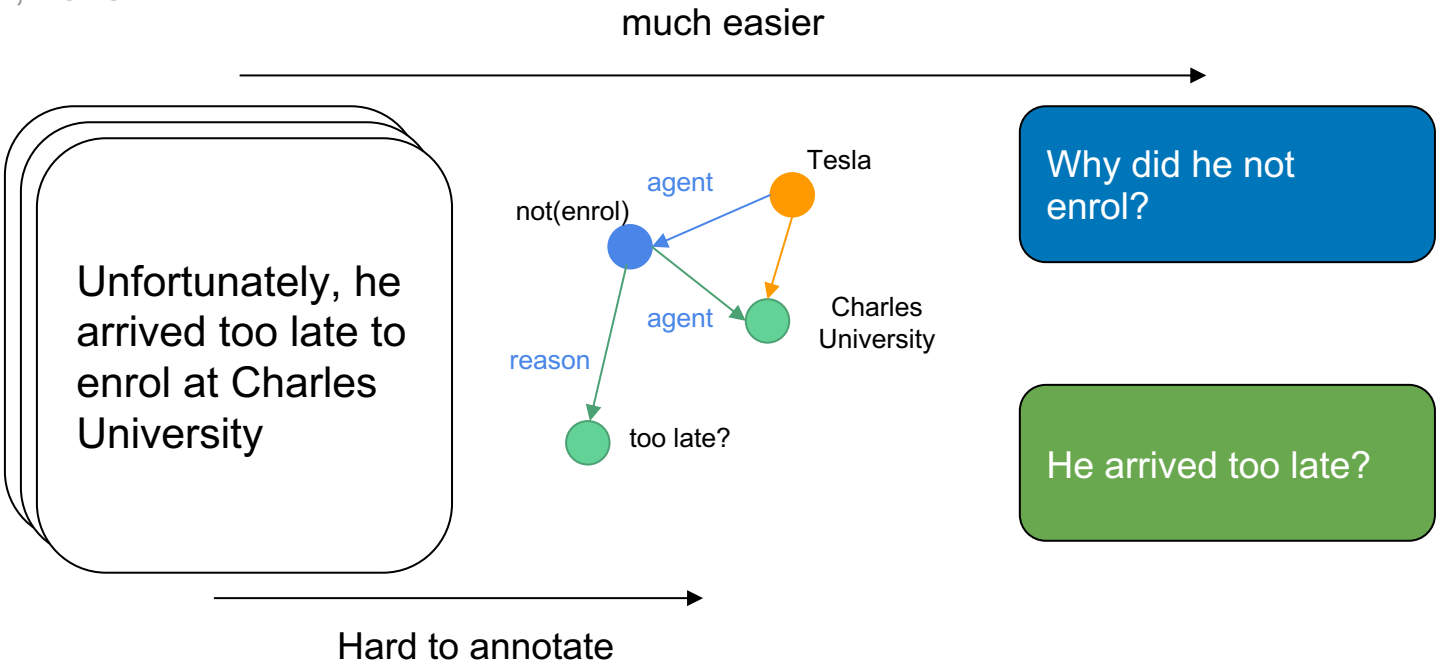

Why did he not enrol?

He arrived too late?

getting this right is hard

# Weakness: Annotation

He et al., 2015

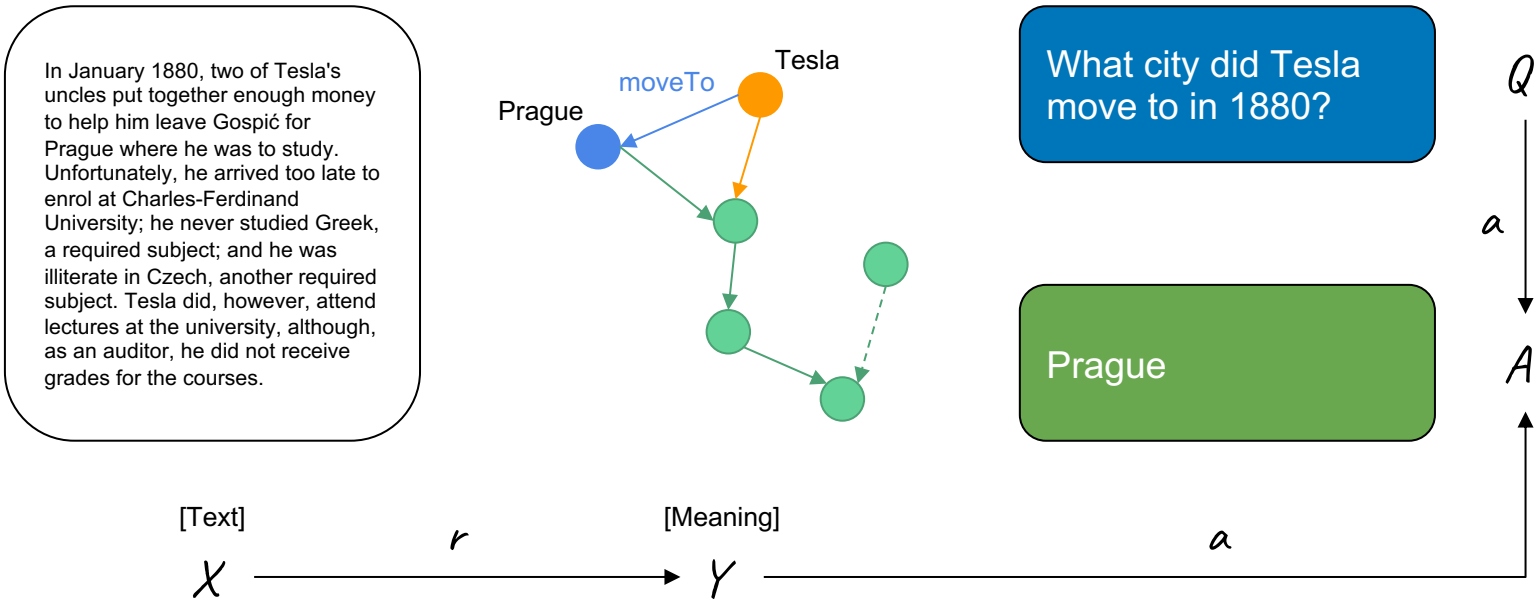


# Structured Representations

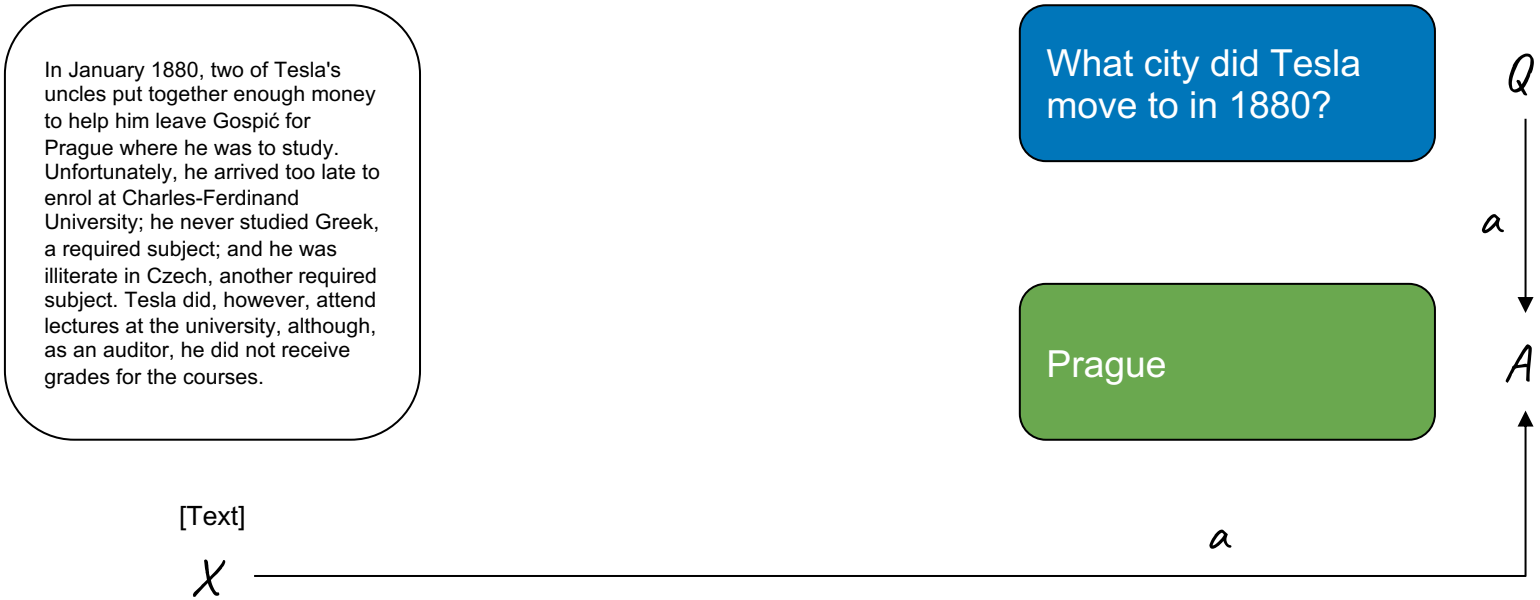
- Advantages
  - Fast access
  - Scalable
  - Interpretable
  - Supports reasoning
  - Universality of representations: independent of question

- Disadvantages
  - Less robust to variation in language
  - Cascading errors
  - Schema engineering
  - Annotation requires experts

# The end-to-end way



# Omitting Symbolic Meaning Representations !!



# Machine Reading AT SCALE

A **machine** processes **a (very) large collection of texts** to satisfy an **information need**

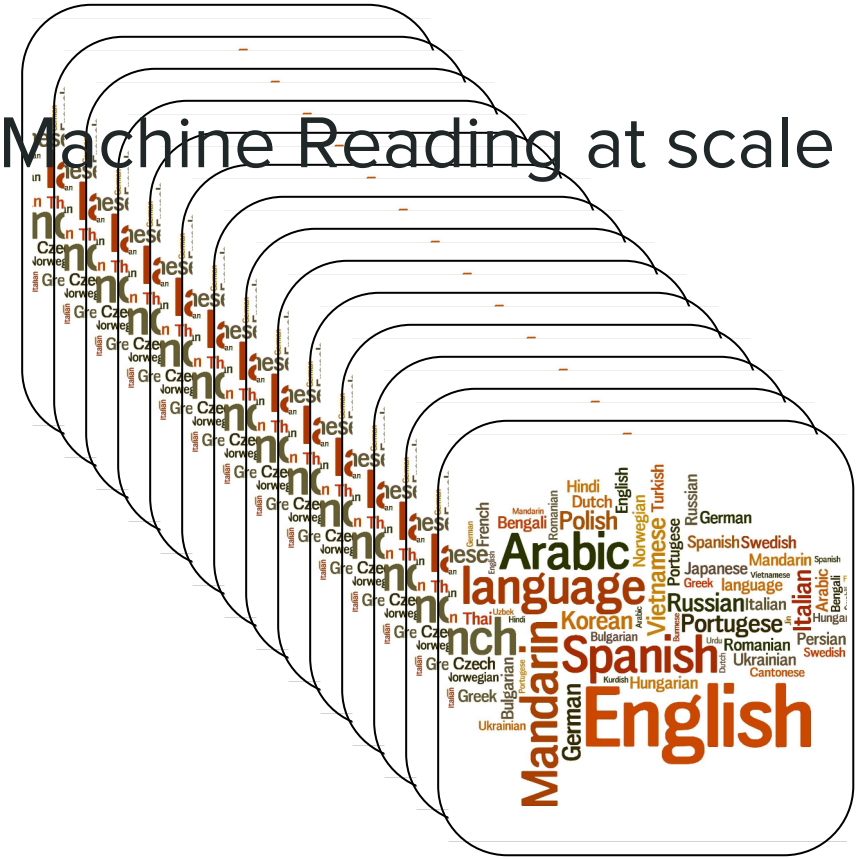
# Machine Reading


[Text]

**uses for**


[Information Need]

# Machine Reading at scale



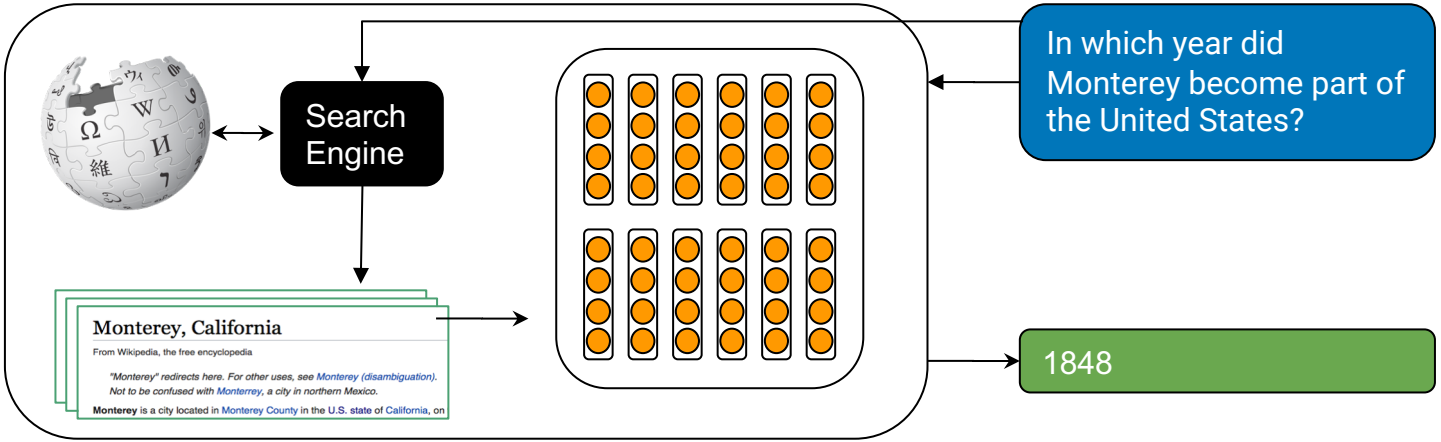

[Large collection of Text]

**uses for**


[Information Need]

# Typical Machine Reading at Scale System

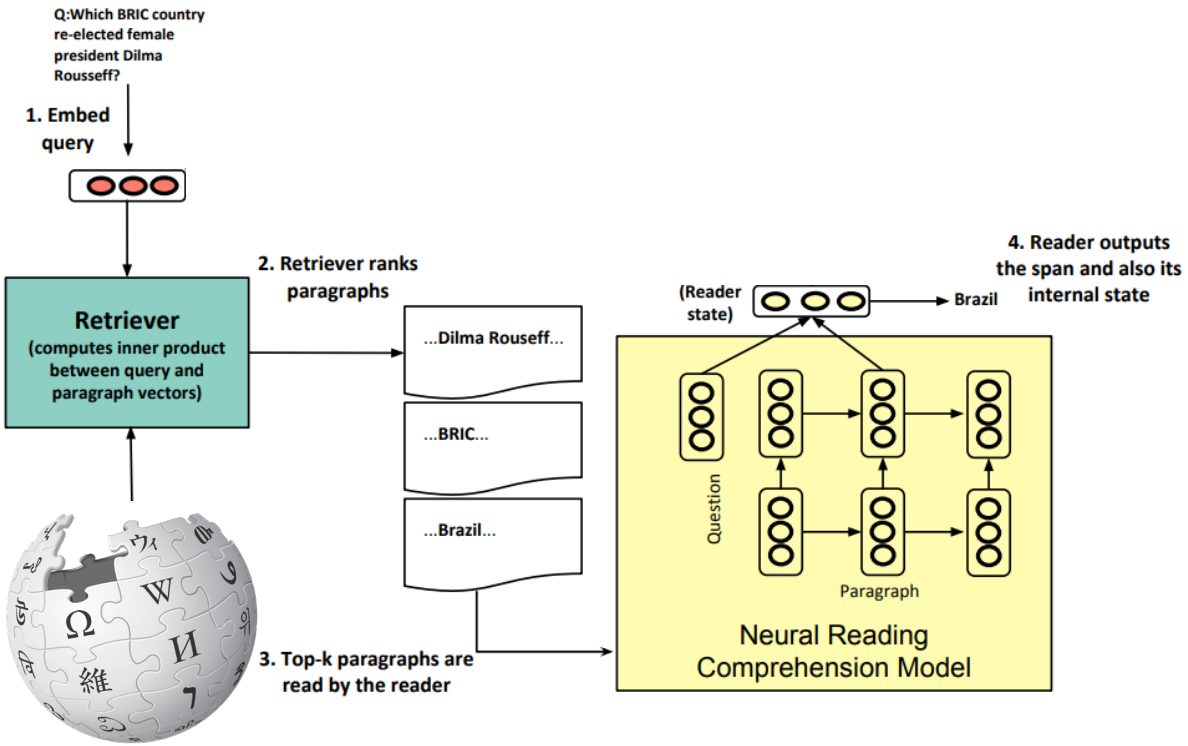
Dr.QA Chen et al., 2017




No way to recover if the search engine is wrong!

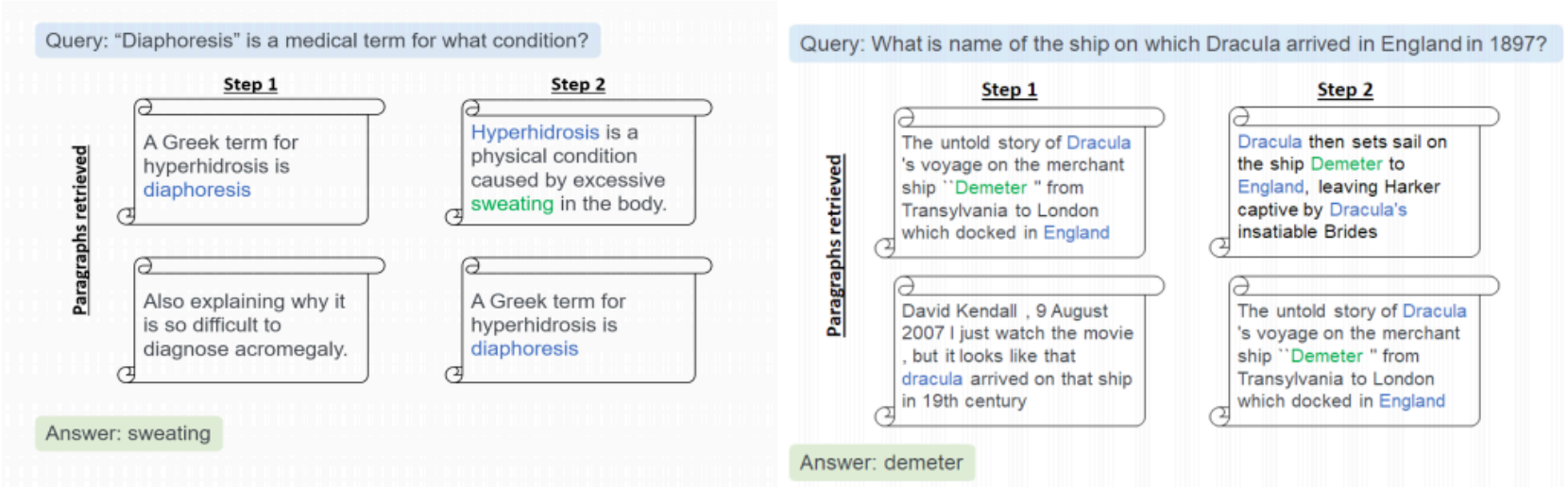
# Follow-up 1: Multi-Step Retriever-Reader

Das et al., 2019



# Follow-up 1: Multi-Step Retriever-Reader

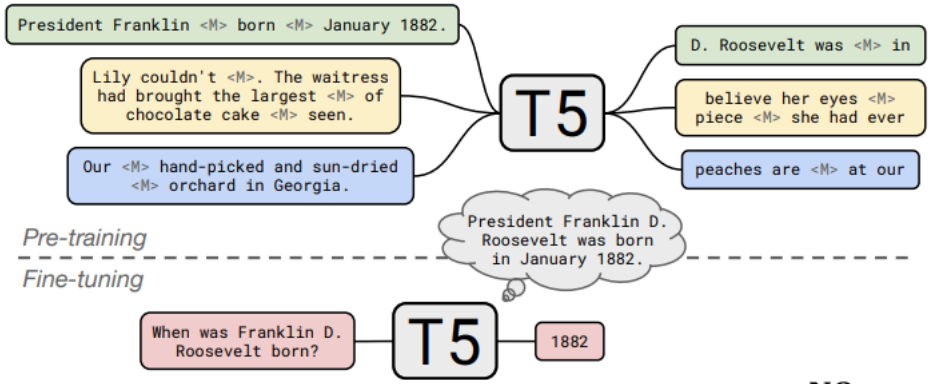
Das et al., 2019




Between 40 and 60% of correct responses (for rather simple questions)

# Follow-up 2: T5 – No retrieval at all!

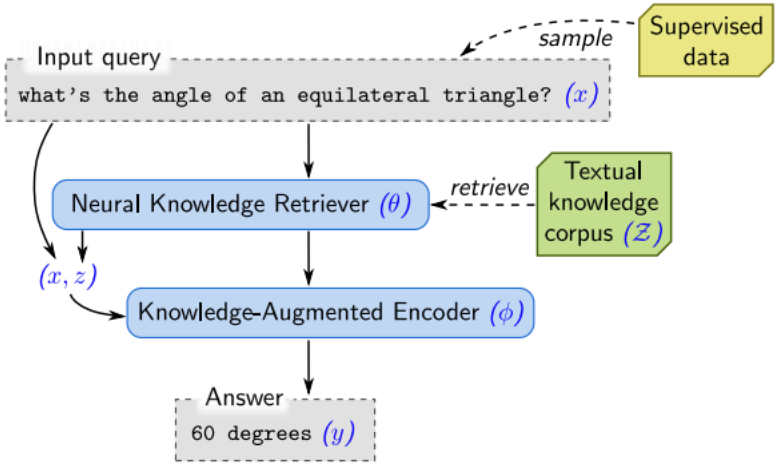
Roberts et al., 2020



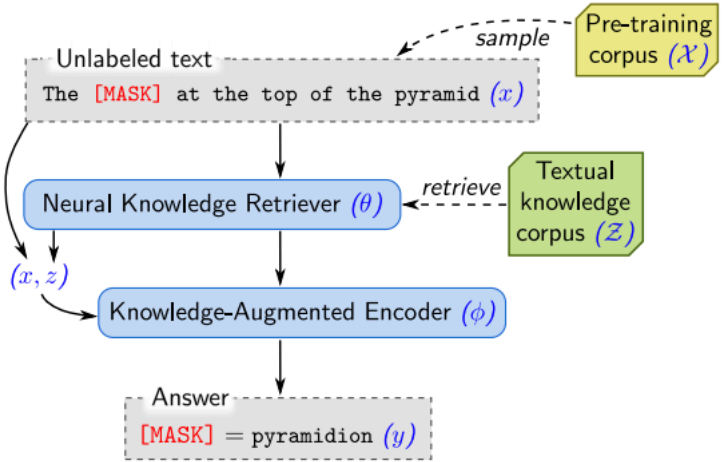

| Name | Architectures | Pre-training | NQ (79k/4k) | WQ (3k/2k) | CT (1k /1k) | # params |
|---|---|---|---|---|---|---|
| BERT-Baseline (Lee et al., 2019) | Sparse Retr.+Transformer | BERT | 26.5 | 17.7 | 21.3 | 110m |
| DrQA (Chen et al., 2017) | Sparse Retr.+DocReader | N/A | - | 20.7 | 25.7 | 34m |
| HardEM (Min et al., 2019a) | Sparse Retr.+Transformer | BERT | 28.1 | - | - | 110m |
| GraphRetriever (Min et al., 2019b) | GraphRetriever+Transformer | BERT | 31.8 | 31.6 | - | 110m |
| PathRetriever (Asai et al., 2019) | PathRetriever+Transformer | MLM | 32.6 | - | - | 110m |
| ORQA (Lee et al., 2019) | Dense Retr.+Transformer | ICT+BERT | 33.3 | 36.4 | 30.1 | 330m |
| T5 (base) (Roberts et al., 2020) | Transformer Seq2Seq | T5 (Multitask) | 27.0 | 29.1 | - | 223m |
| T5 (large) (Roberts et al., 2020) | Transformer Seq2Seq | T5 (Multitask) | 29.8 | 32.2 | - | 738m |
| T5 (11b) (Roberts et al., 2020) | Transformer Seq2Seq | T5 (Multitask) | 34.5 | 37.4 | - | 11318m |

# Follow-up 3: REALM: Pretraining with retrieval

Guu et al., 2020




Open domain QA system




Retrieval-based pretraining!

# Follow-up 3: REALM: Pretraining with retrieval

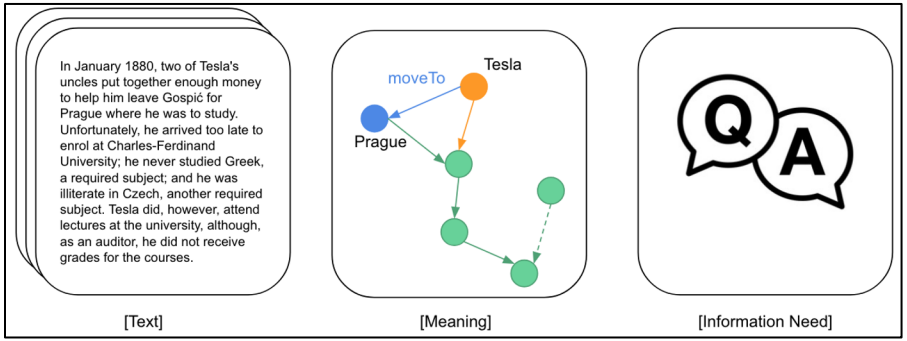
Guu et al., 2020

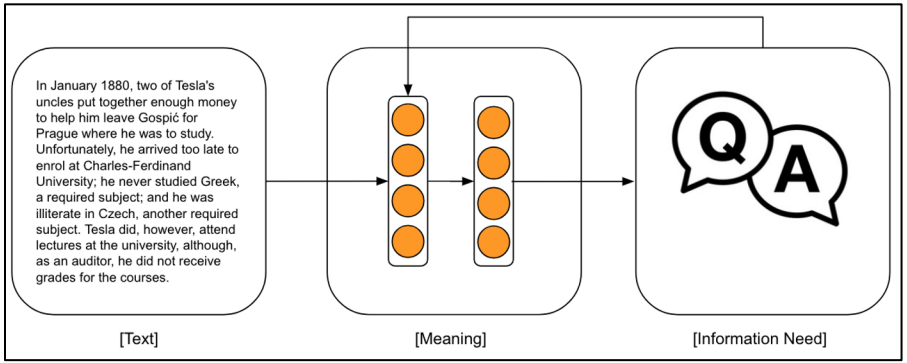
| Name | Architectures | Pre-training | NQ (79k/4k) | WQ (3k/2k) | CT (1k /1k) | # params |
|---|---|---|---|---|---|---|
| BERT-Baseline (Lee et al., 2019) | Sparse Retr.+Transformer | BERT | 26.5 | 17.7 | 21.3 | 110m |
| DrQA (Chen et al., 2017) | Sparse Retr.+DocReader | N/A | - | 20.7 | 25.7 | 34m |
| HardEM (Min et al., 2019a) | Sparse Retr.+Transformer | BERT | 28.1 | - | - | 110m |
| GraphRetriever (Min et al., 2019b) | GraphRetriever+Transformer | BERT | 31.8 | 31.6 | - | 110m |
| PathRetriever (Asai et al., 2019) | PathRetriever+Transformer | MLM | 32.6 | - | - | 110m |
| ORQA (Lee et al., 2019) | Dense Retr.+Transformer | ICT+BERT | 33.3 | 36.4 | 30.1 | 330m |
| T5 (base) (Roberts et al., 2020) | Transformer Seq2Seq | T5 (Multitask) | 27.0 | 29.1 | - | 223m |
| T5 (large) (Roberts et al., 2020) | Transformer Seq2Seq | T5 (Multitask) | 29.8 | 32.2 | - | 738m |
| T5 (11b) (Roberts et al., 2020) | Transformer Seq2Seq | T5 (Multitask) | 34.5 | 37.4 | - | 11318m |
| REALM ($\mathcal{X}$ = Wikipedia, $\mathcal{Z}$ = Wikipedia) | Dense Retr.+Transformer | REALM | 39.2 | 40.2 | **46.8** | 330m |
| ($\mathcal{X}$ = CC-News, $\mathcal{Z}$ = Wikipedia) | Dense Retr.+Transformer | REALM | **40.4** | **40.7** | 42.9 | 330m |

# A Paradigm Shift

- Symbolic Meaning Representations
- ➡ Latent Vector Representations

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



# Pros and cons

| End-to-end models | Symbolic systems |
| --- | --- |
| *Neural Networks* | *KBs, Inductive Logic Programming, etc.* |
| • Scale to very large datasets<br>• Can be used by non domain experts<br>• Robust to noise and ambiguity in data<br>• Game changers in multiple applications | • Small scale conditions<br>• Require heavy expert knowledge<br>• Very brittle with noisy, ambiguous data<br>• Limited applicative success |
| • Very data hungry (unsupervised data helps)<br>• Can't learn easily new tasks from old ones<br>• Not interpretable<br>• Relatively simple reasoning | Great research opportunities!<br>ostractions |

# Current Challenge: 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, ... Fact Checking

# Conclusion

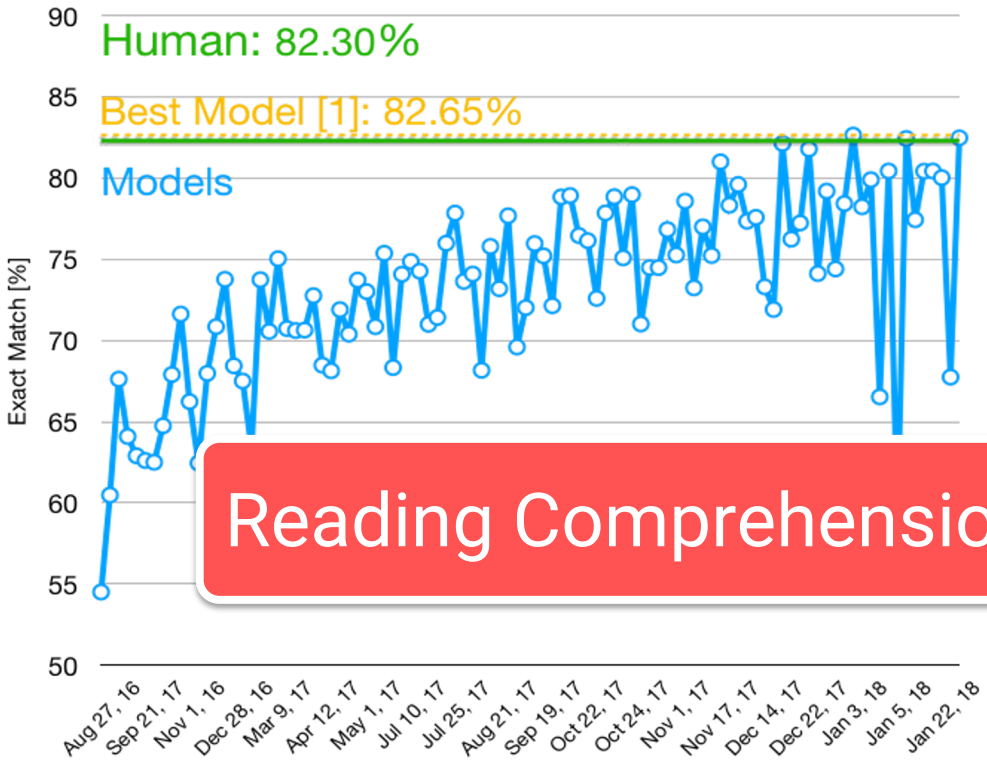
- We've seen 2 approaches for building system to answer any question
- Most deployed systems still rely on traditional pipelines for the most part (+ some DL here and there)

- Why? Scale, reliability, interpretability

- Open questions:
  - All shortcomings of Machine Reading → Open domain QA. Need to solve them
  - Will pretrained contextual embeddings change everything forever?
  - Can we combine both symbolic and end-to-end approaches?

# Machine Reading & QA / Open Problems

# Progression of SQuAD Model Performance




**Reading Comprehension Solved?**

# Challenge 1: Robustness

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.

# Challenge 1: Robustness

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.

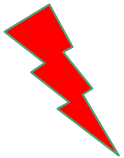
# Challenge 1: Robustness

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.

# Challenge 1: Robustness

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

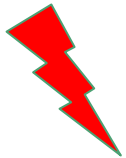


Jeff Dean

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.

# Challenge 1: Robustness

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., ACL'17)

# 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

# Challenge 2: 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

# Challenge 2: 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.

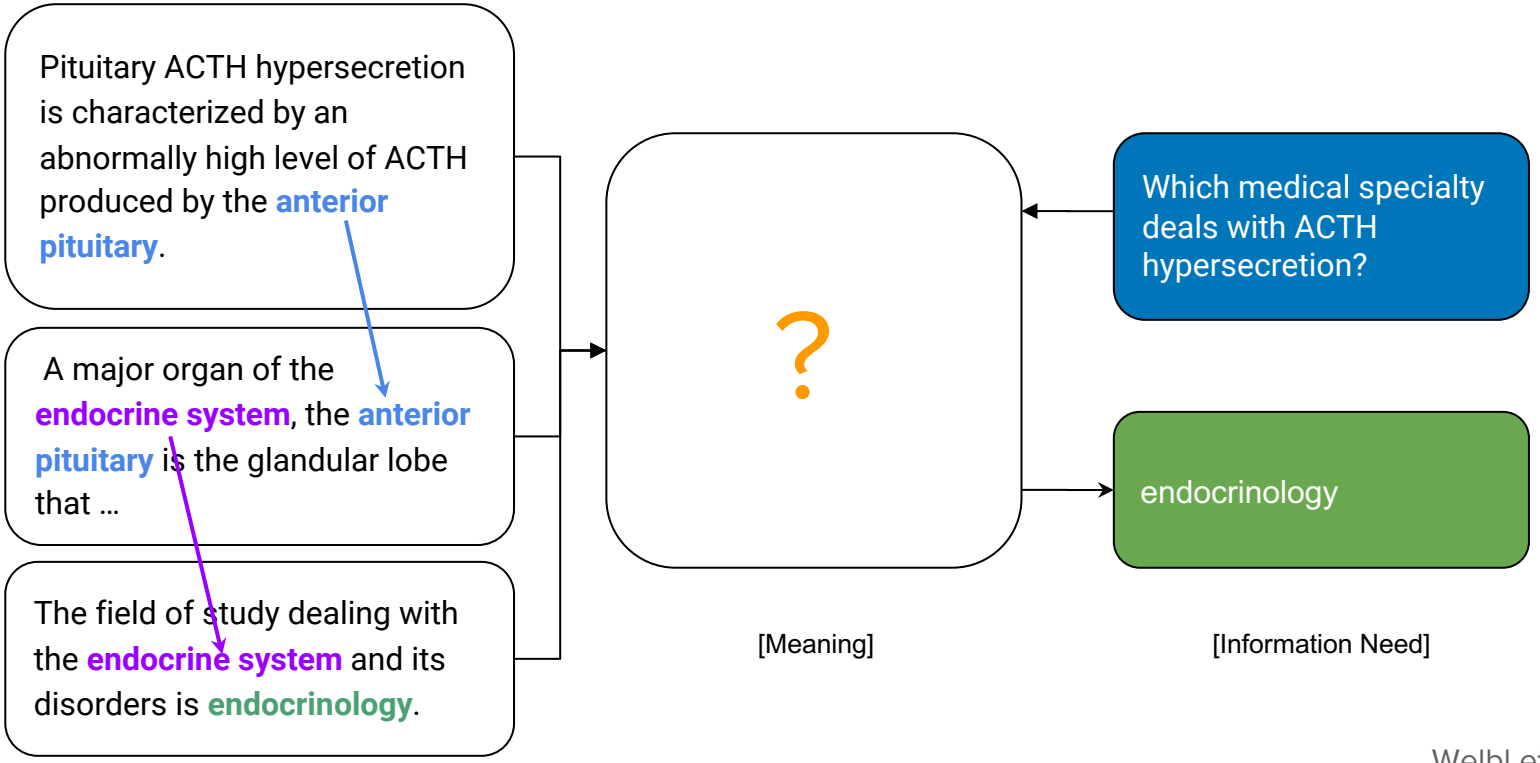
| System | SQuAD 1.1 test | | SQuAD 2.0 dev | | SQuAD 2.0 test | |
|---|---|---|---|---|---|---|
| | EM | F1 | EM | F1 | EM | F1 |
| BNA | 68.0 | 77.3 | 59.8 | 62.6 | 59.2 | 62.1 |
| DocQA | 72.1 | 81.0 | 61.9 | 64.8 | 59.3 | 62.3 |
| DocQA + ELMo | **78.6** | **85.8** | **65.1** | **67.6** | **63.4** | **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

# Challenge 3: Limited Supervision

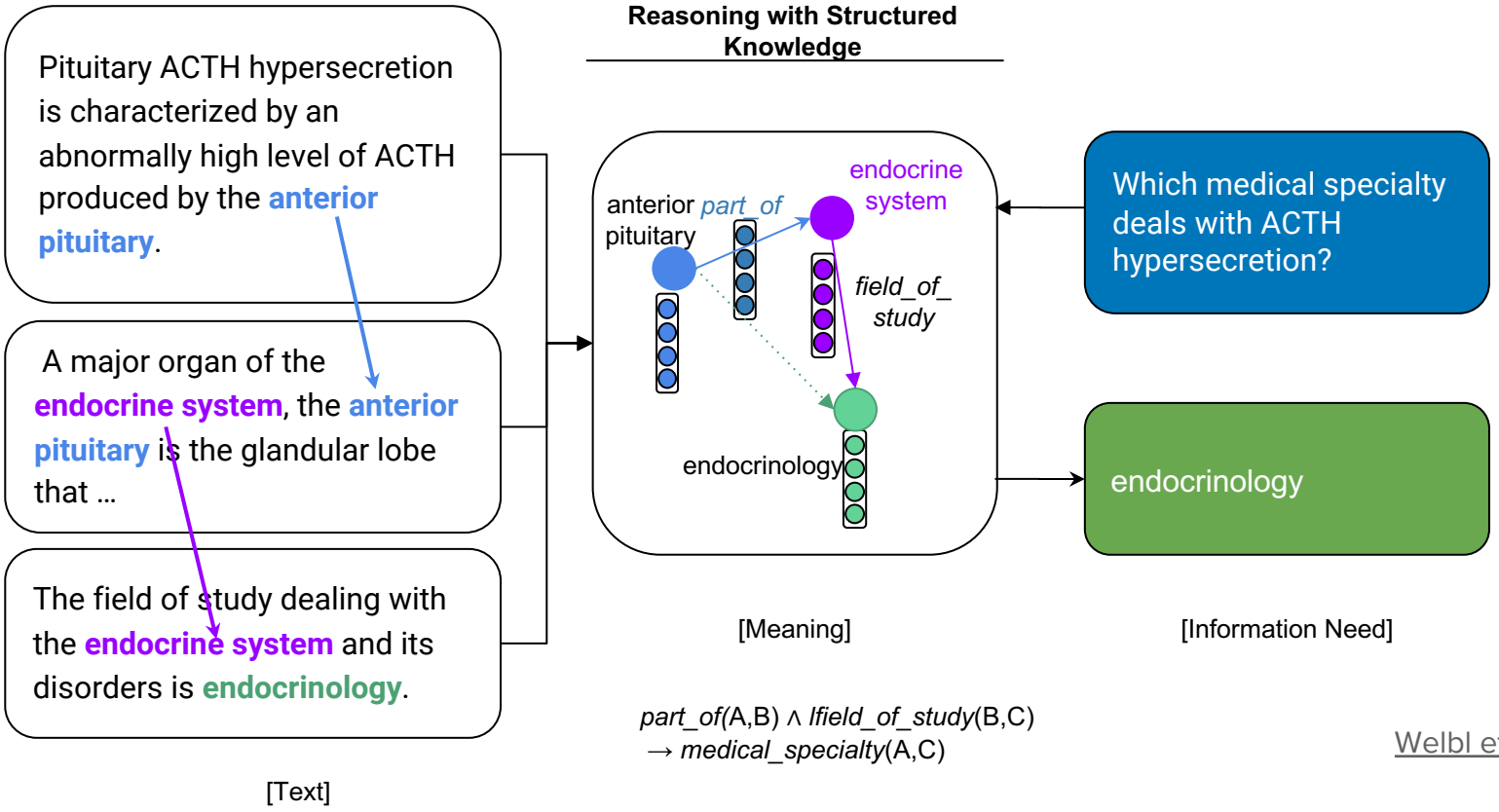
- Strong results with large annotated training sets
- How about smaller datasets?
  - Ideally: shift from 100K to 1K training points
  - less costly, large-scale annotation
- Approaches:
  - domain adaptation, e.g. Wiese et al. (2017)
  - Synthetic data generation, e.g. Dhingra et al. (2018)
  - transfer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

# Challenge 4: Reasoning with Text



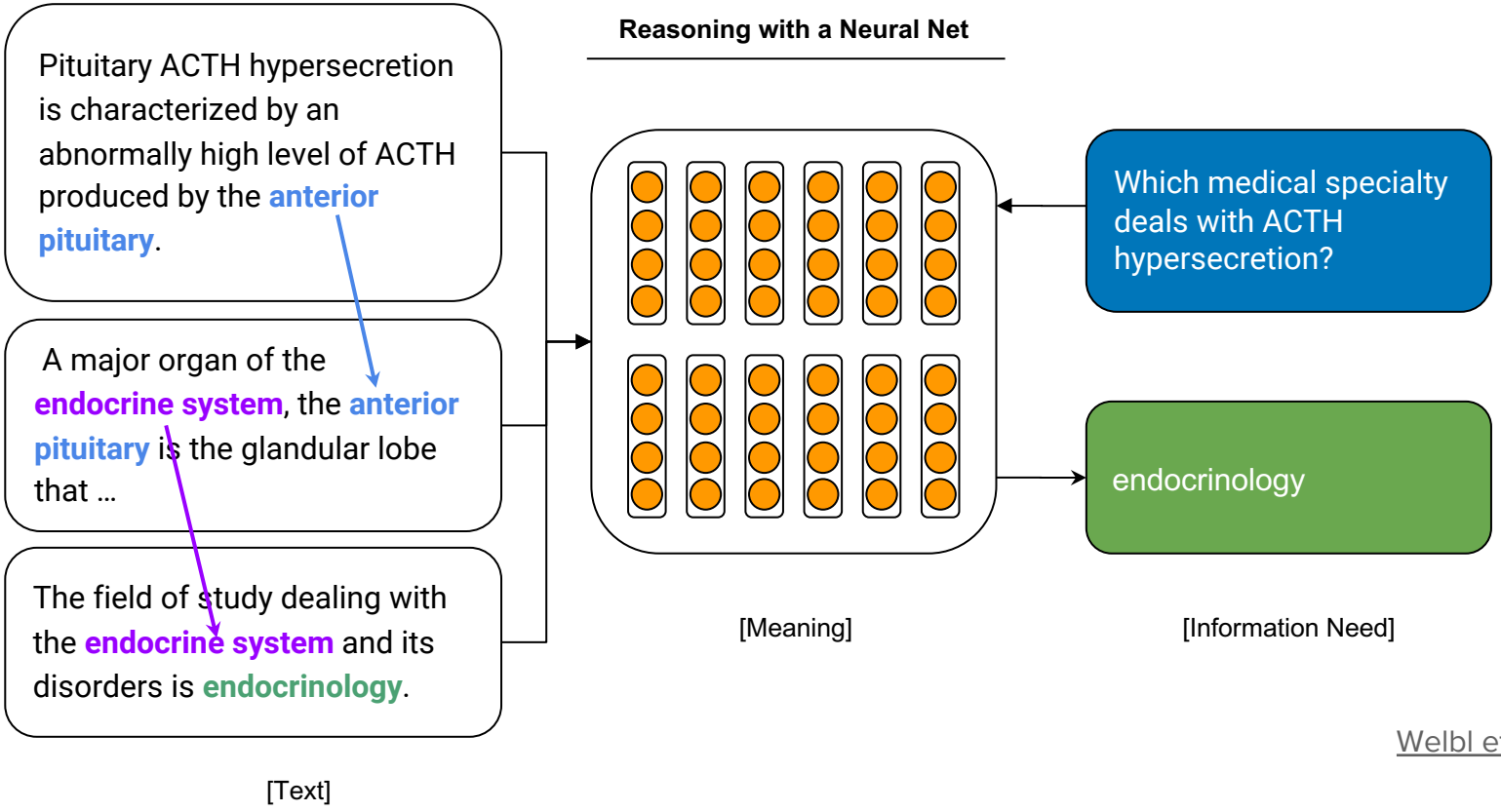

Welbl et al. (2018)

# Challenge 4: Reasoning with Text



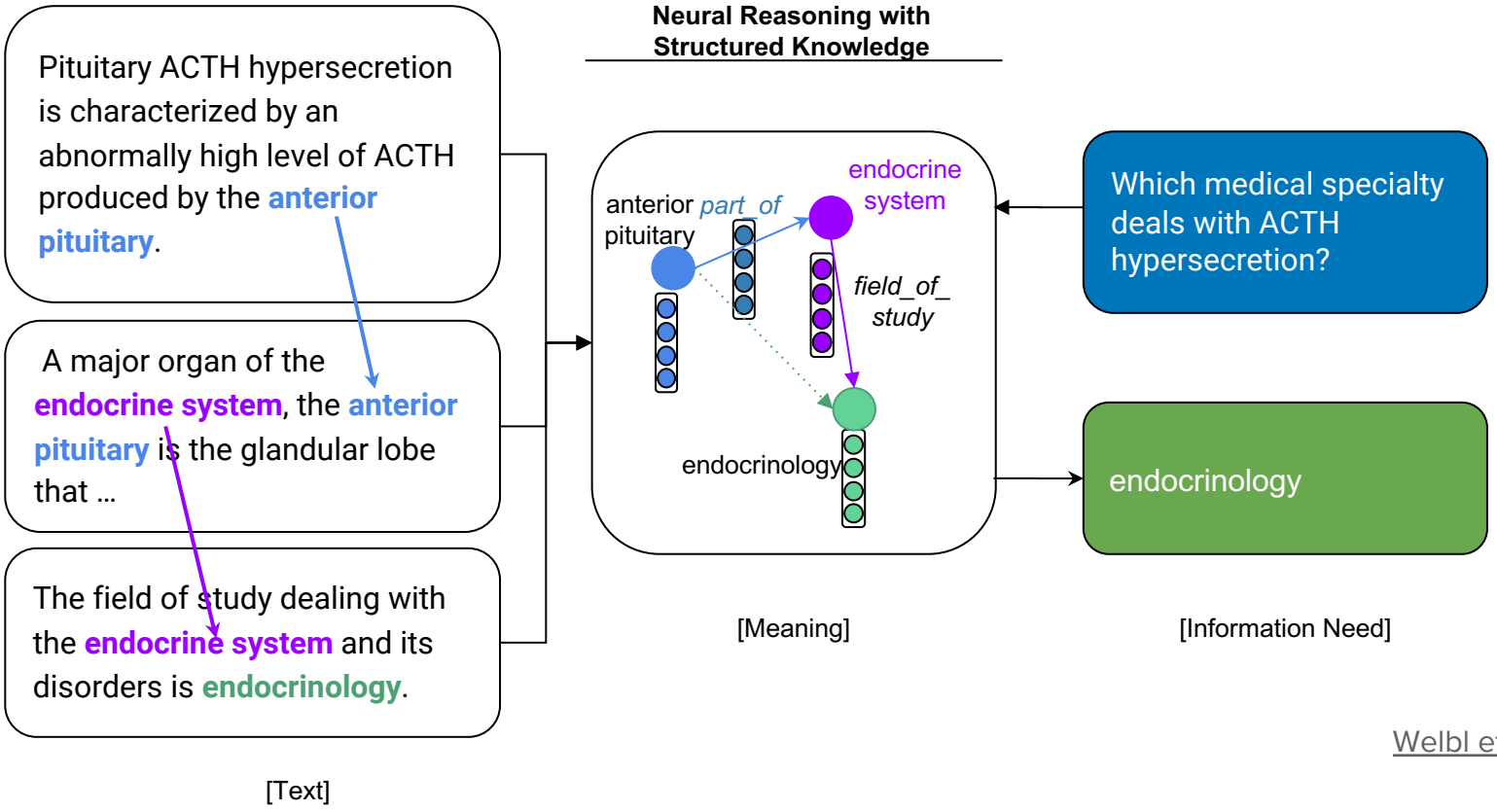

Welbl et al. (2018)

# Challenge 4: Reasoning with Text




Welbl et al. (2018)

# Challenge 4: Reasoning with Text




Welbl et al. (2018)

# Summary: Where models work well today

- Question is answerable
- 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
- Relevant paragraph / text is given

# Is all this model complexity necessary?

**Should we rather:**

- Build model architectures more carefully?
- Think more carefully about our training data?

**Take home:**

- **Don’t over-engineer** before establishing a decent baseline
- **Look at your datasets**! Are they challenging enough for the research you want to conduct?