# DEEP LEARNING MODELS FOR QUESTION ANSWERING

Sujit Pal & Abhishek Sharma

Elsevier Search Guild Question Answering Workshop

October 5-6, 2016

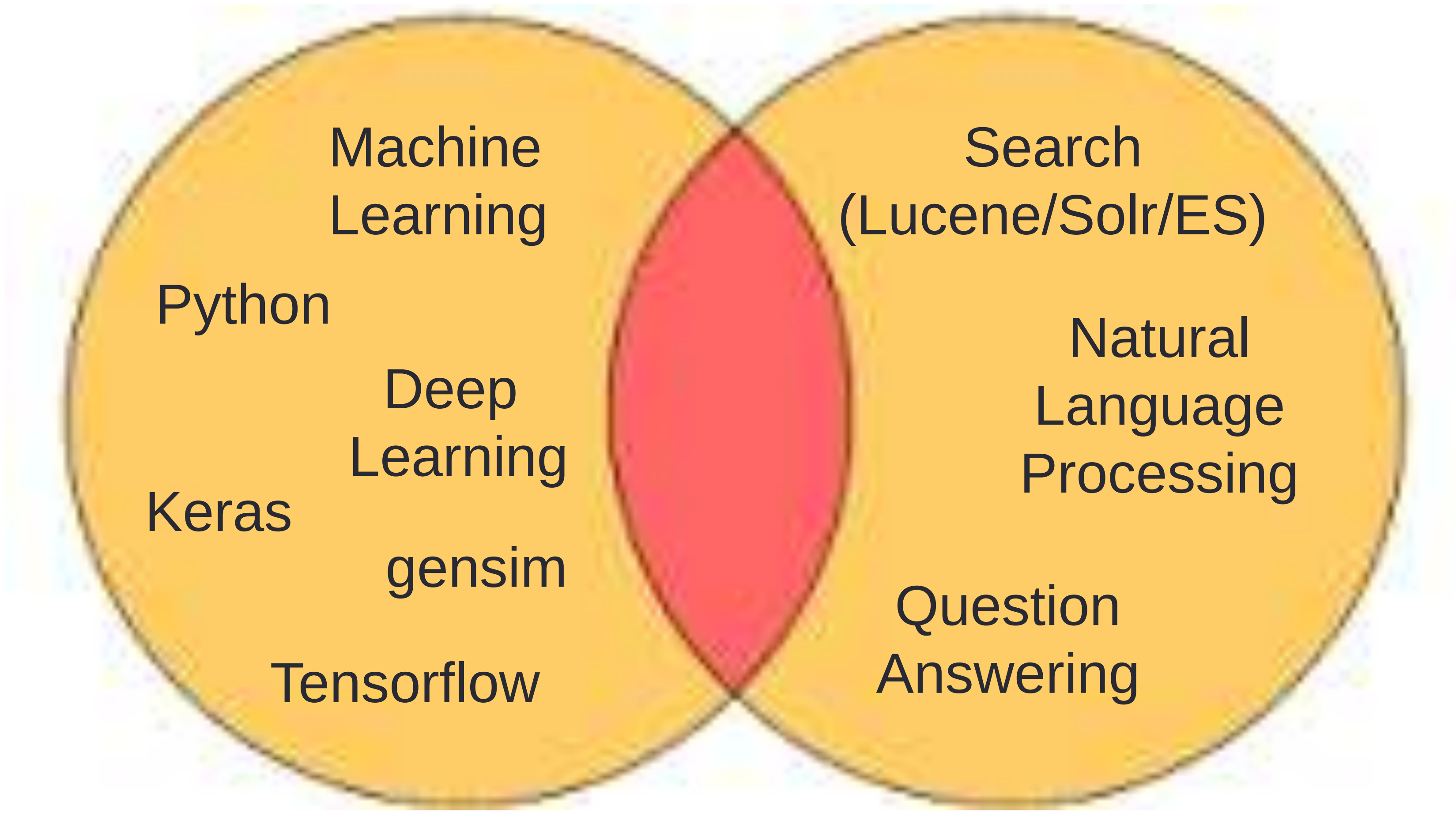
# About Us

| **Sujit Pal** | **Abhishek Sharma** |
|---|---|
| Technology Research Director<br>Elsevier Labs | Organizer, DLE Meetup<br>and<br>Software Engineer, Salesforce |

# How we started

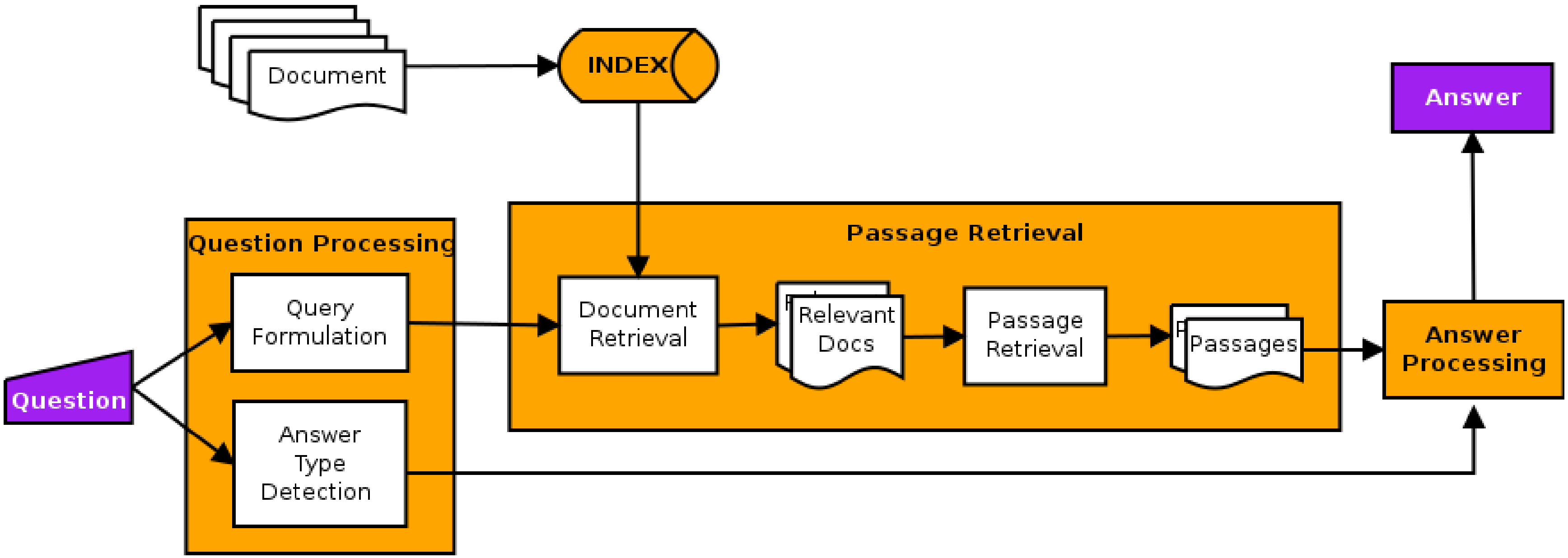
- Watching Udacity “Deep Learning” videos taught by Vincent Vanhoucke.
- Watching “Deep Learning for Natural Language Processing” (CS 224d) videos taught by Richard Socher.
- Thinking that it might be interesting to do “something” around Question Answering and Deep Learning.

# What we knew

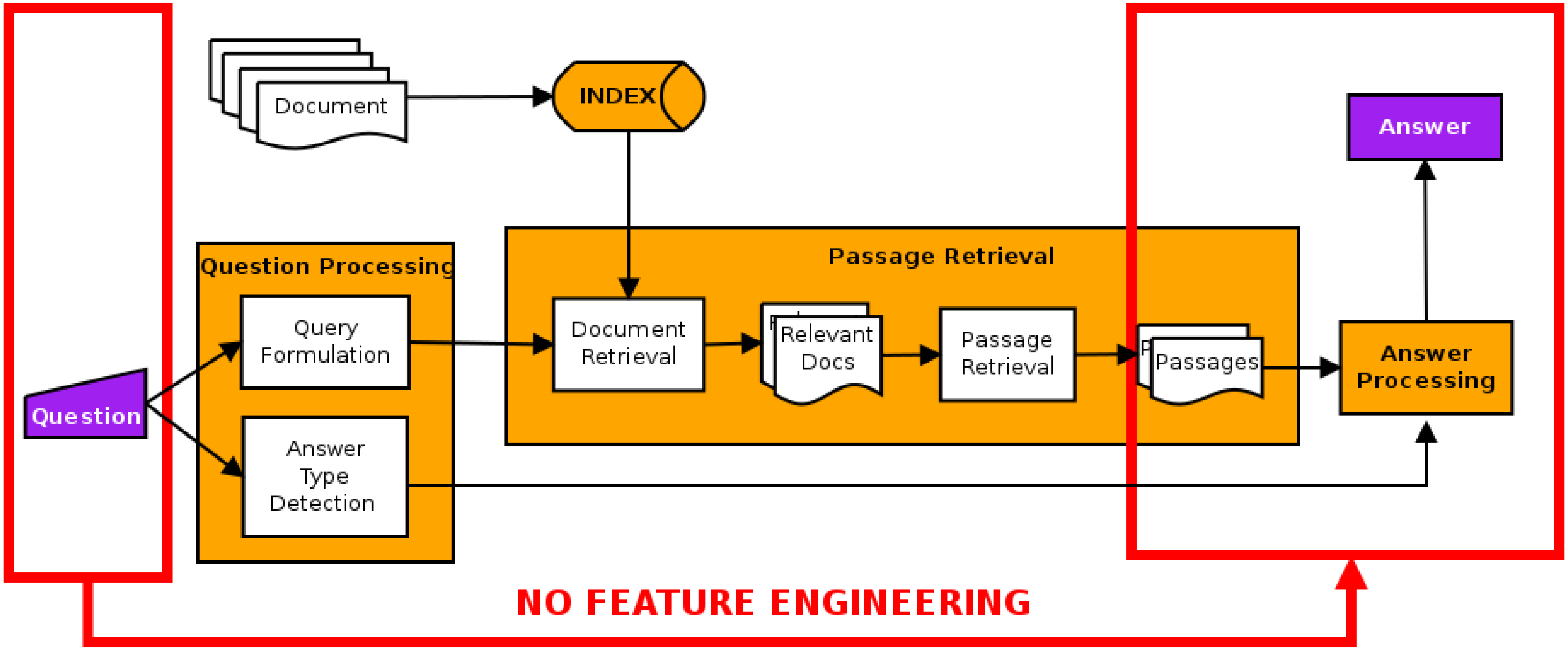

Machine Learning
Python
Deep Learning
Keras
gensim
Tensorflow
Search (Lucene/Solr/ES)
Natural Language Processing
Question Answering

# Identify Scope

# Question Answering Pipeline



# Question Answering Pipeline




IR Based Factoid QA // Coursera NLP // Jurafsky and Manning

# Research

# This had just ended…

## The Allen AI Science Challenge

Wed 7 Oct 2015 – Sat 13 Feb 2016 (7 months ago)

Competition Details » Get the Data » Make a submission

### Is your model smarter than an 8th grader?


and the #4 ranked entry used Deep Learning for their solution.

Computer Science > Artificial Intelligence

# Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks

Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M. Rush, Bart van Merriënboer, Armand Joulin, Tomas Mikolov

*(Submitted on 19 Feb 2015 (v1), last revised 31 Dec 2015 (this version, v10))*

One long-term goal of machine learning research is to produce methods that are applicable to reasoning and natural language, in particular building an intelligent dialogue agent. To measure progress towards that goal, we argue for the usefulness of a set of proxy tasks that evaluate reading comprehension via question answering. Our tasks measure understanding in several ways: whether a system is able to answer questions via chaining facts, simple induction, deduction and many more. The tasks are designed to be prerequisites for any system that aims to be capable of conversing with a human. We believe many existing learning systems can currently not solve them, and hence our aim is to classify these tasks into skill sets, so that researchers can identify (and then rectify) the failings of their systems. We also extend and improve the recently introduced Memory Networks model, and show it is able to solve some, but not all, of the tasks.

arXiv.org > cs > arXiv:1503.08895

Search or A

Computer Science > Neural and Evolutionary Computing

# End-To-End Memory Networks

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, Rob Fergus

*(Submitted on 31 Mar 2015 (v1), last revised 24 Nov 2015 (this version, v5))*

We introduce a neural network with a recurrent attention model over a possibly large external memory. The architecture is a form of Memory Network (Weston et al., 2015) but unlike the model in that work, it is trained end-to-end, and hence requires significantly less supervision during training, making it more generally applicable in realistic settings. It can also be seen as an extension of RNNsearch to the case where multiple computational steps (hops) are performed per output symbol. The flexibility of the model allows us to apply it to tasks as diverse as (synthetic) question answering and to language modeling. For the former our approach is competitive with Memory Networks, but with less supervision. For the latter, on the Penn TreeBank and Text8 datasets our approach demonstrates comparable performance to RNNs and LSTMs. In both cases we show that the key concept of multiple computational hops yields improved results.

arXiv.org > cs > arXiv:1511.04108

Search or A

Computer Science > Computation and Language

# LSTM-based Deep Learning Models for Non-factoid Answer Selection

Ming Tan, Cicero dos Santos, Bing Xiang, Bowen Zhou

*(Submitted on 12 Nov 2015 (v1), last revised 28 Mar 2016 (this version, v4))*

In this paper, we apply a general deep learning (DL) framework for the answer selection task, which does not depend on manually defined features or linguistic tools. The basic framework is to build the embeddings of questions and answers based on bidirectional long short-term memory (biLSTM) models, and measure their closeness by cosine similarity. We further extend this basic model in two directions. One direction is to define a more composite representation for questions and answers by combining convolutional neural network with the basic framework. The other direction is to utilize a simple but efficient attention mechanism in order to generate the answer representation according to the question context. Several variations of models are provided. The models are examined by two datasets, including TREC-QA and InsuranceQA. Experimental results demonstrate that the proposed models substantially outperform several strong baselines.

arXiv.org > cs > arXiv:1506.07285



Computer Science > Computation and Language

# Ask Me Anything: Dynamic Memory Networks for Natural Language Processing

Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, Richard Socher

*(Submitted on 24 Jun 2015 (v1), last revised 5 Mar 2016 (this version, v5))*

Most tasks in natural language processing can be cast into question answering (QA) problems over language input. We introduce the dynamic memory network (DMN), a neural network architecture which processes input sequences and questions, forms episodic memories, and generates relevant answers. Questions trigger an iterative attention process which allows the model to condition its attention on the inputs and the result of previous iterations. These results are then reasoned over in a hierarchical recurrent sequence model to generate answers. The DMN can be trained end-to-end and obtains state-of-the-art results on several types of tasks and datasets: question answering (Facebook's bAbI dataset), text classification for sentiment analysis (Stanford Sentiment Treebank) and sequence modeling for part-of-speech tagging (WSJ-PTB). The training for these different tasks relies exclusively on trained word vector representations and input-question-answer triplets.

arXiv.org > cs > arXiv:1603.01417

Search or A

Computer Science > Neural and Evolutionary Computing

# Dynamic Memory Networks for Visual and Textual Question Answering

Caiming Xiong, Stephen Merity, Richard Socher

*(Submitted on 4 Mar 2016)*

Neural network architectures with memory and attention mechanisms exhibit certain reasoning capabilities required for question answering. One such architecture, the dynamic memory network (DMN), obtained high accuracy on a variety of language tasks. However, it was not shown whether the architecture achieves strong results for question answering when supporting facts are not marked during training or whether it could be applied to other modalities such as images. Based on an analysis of the DMN, we propose several improvements to its memory and input modules. Together with these changes we introduce a novel input module for images in order to be able to answer visual questions. Our new DMN+ model improves the state of the art on both the Visual Question Answering dataset and the \babi-10k text question-answering dataset without supporting fact supervision.

Computer Science > Computation and Language

# Attention-Based Convolutional Neural Network for Machine Comprehension

Wenpeng Yin, Sebastian Ebert, Hinrich Schütze

*(Submitted on 13 Feb 2016)*

Understanding open-domain text is one of the primary challenges in natural language processing (NLP). Machine comprehension benchmarks evaluate the system's ability to understand text based on the text content only. In this work, we investigate machine comprehension on MCTest, a question answering (QA) benchmark. Prior work is mainly based on feature engineering approaches. We come up with a neural network framework, named hierarchical attention-based convolutional neural network (HABCNN), to address this task without any manually designed features. Specifically, we explore HABCNN for this task by two routes, one is through traditional joint modeling of passage, question and answer, one is through textual entailment. HABCNN employs an attention mechanism to detect key phrases, key sentences and key snippets that are relevant to answering the question. Experiments show that HABCNN outperforms prior deep learning approaches by a big margin.

# Building Blocks

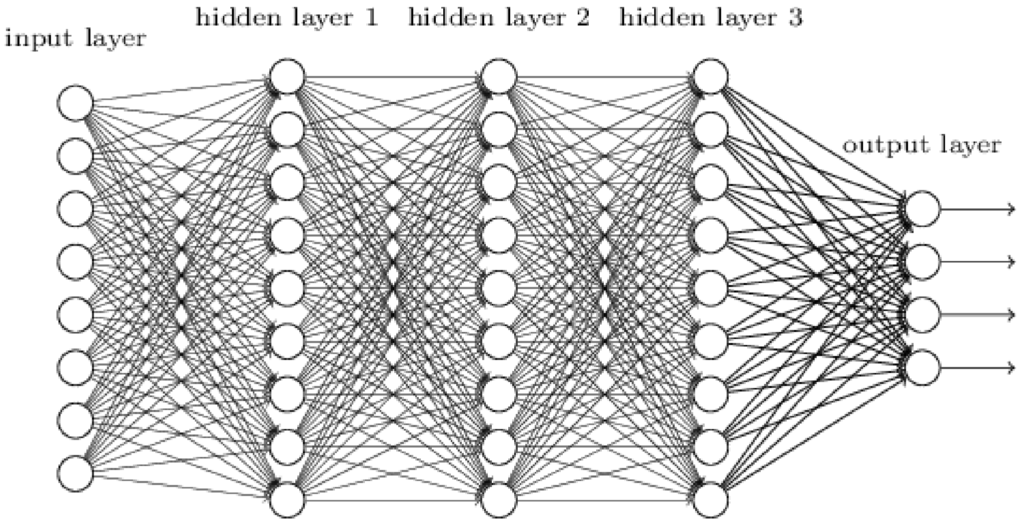
# Building blocks

Most DL Networks (including Question Answering models) composed out of these basic building blocks.

- Fully Connected Network
- Word Embedding
- Convolutional Neural Network
- Recurrent Neural Network

# Fully Connected Network

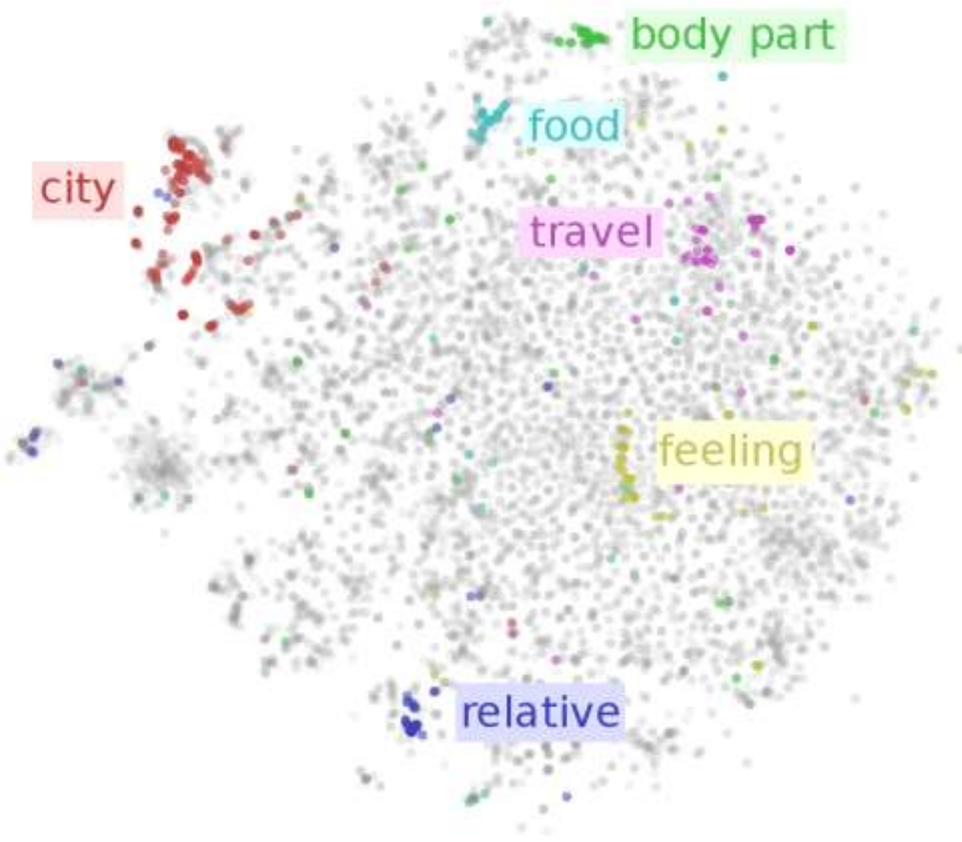
- Workhorse architecture of Deep Learning.
- Number of layers and number of units per layer increased for more complex models.
- Used for all kinds of problem spaces.




Credit: neuralnetworksanddeeplearning.com

# Word Embedding

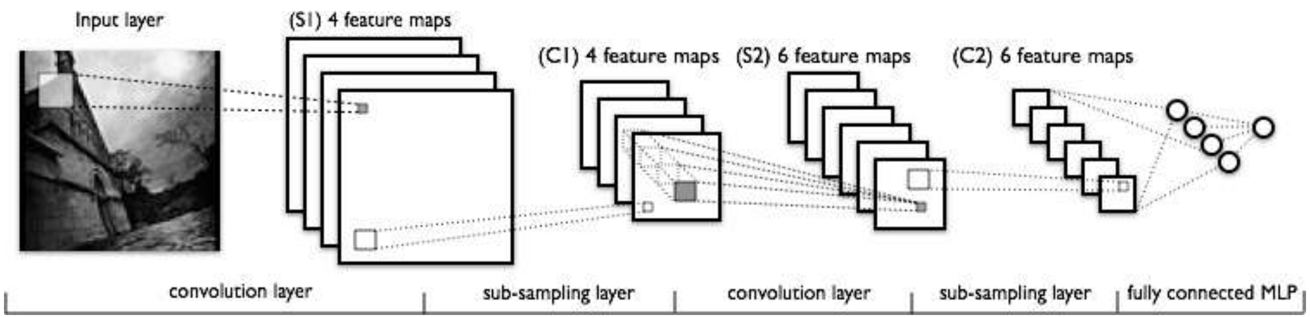
- Projects sparse 1-hot vector representation onto denser lower dimensional space.
- Unsupervised technique.
- Embedding space exhibits Distributional Semantics.
- Has almost completely replaced traditional distributional features in NLP (Deep Learning and non Deep Learning).
- Word2Vec (CBOW and Skip-gram), GloVe.




Credit:Sebastian Ruder // sebastianruder.com

# Convolutional Neural Network

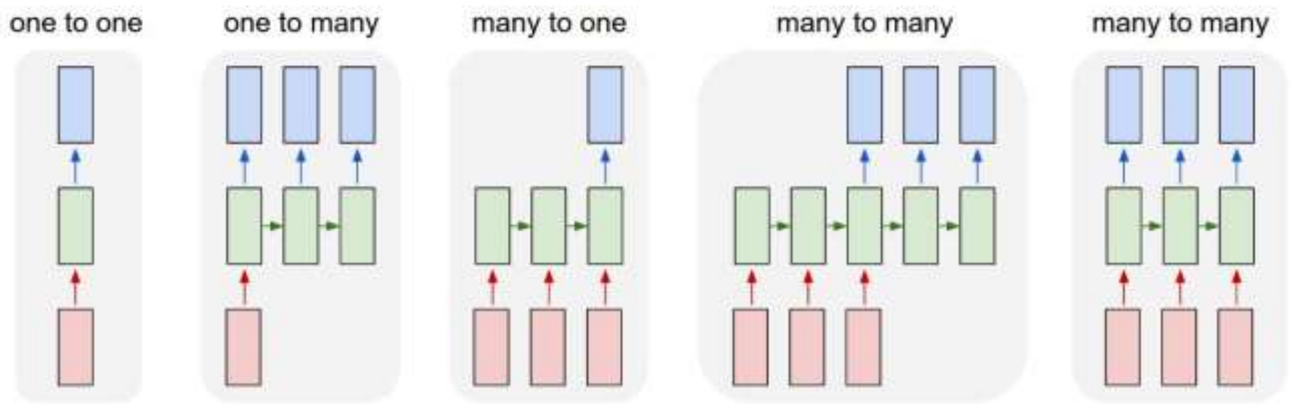
- Alternate Convolution and Pooling Operations to extract relevant features
- Mainly used in Image recognition, exploits geometry of image.
- 1D variant (Convolution and Pooling) used for text.
- Exploits word neighborhoods and extracts “meaning” of sentences or paragraphs.




Credit: deeplearning.net

# Recurrent Neural Network

- Works with sequence input (such as text and audio).
- Exploits temporal nature of the data.
- Many variations possible (shown below).
- Basic RNN suffers from vanishing gradient problem – addressed by Long Short Term Memory (LSTM) RNNs.
- Gated Recurrent Unit (GRU) another variant with simpler structure and better performance than LSTM.




Credit: Andrej Karpathy // karpathy.github.io

# Start with bAbI

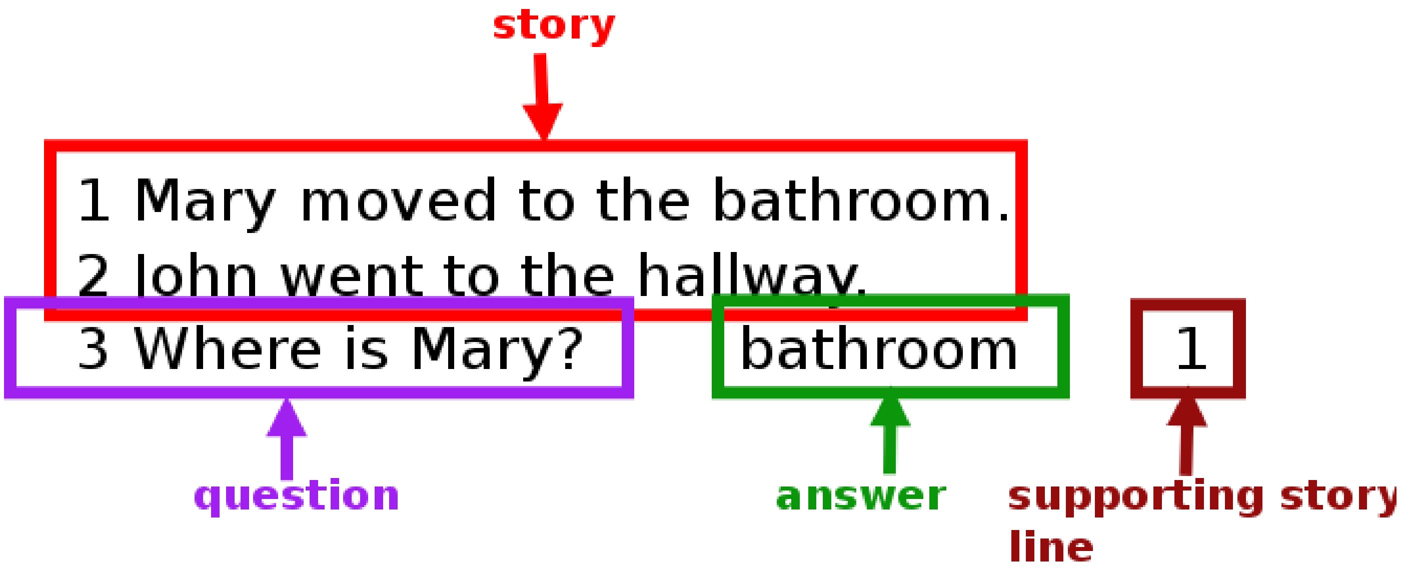
# bAbI Dataset

- Synthetic Dataset (1000, 10k, … records)
- Composed of actors, places, things, actions, etc.
- Released by Facebook Research

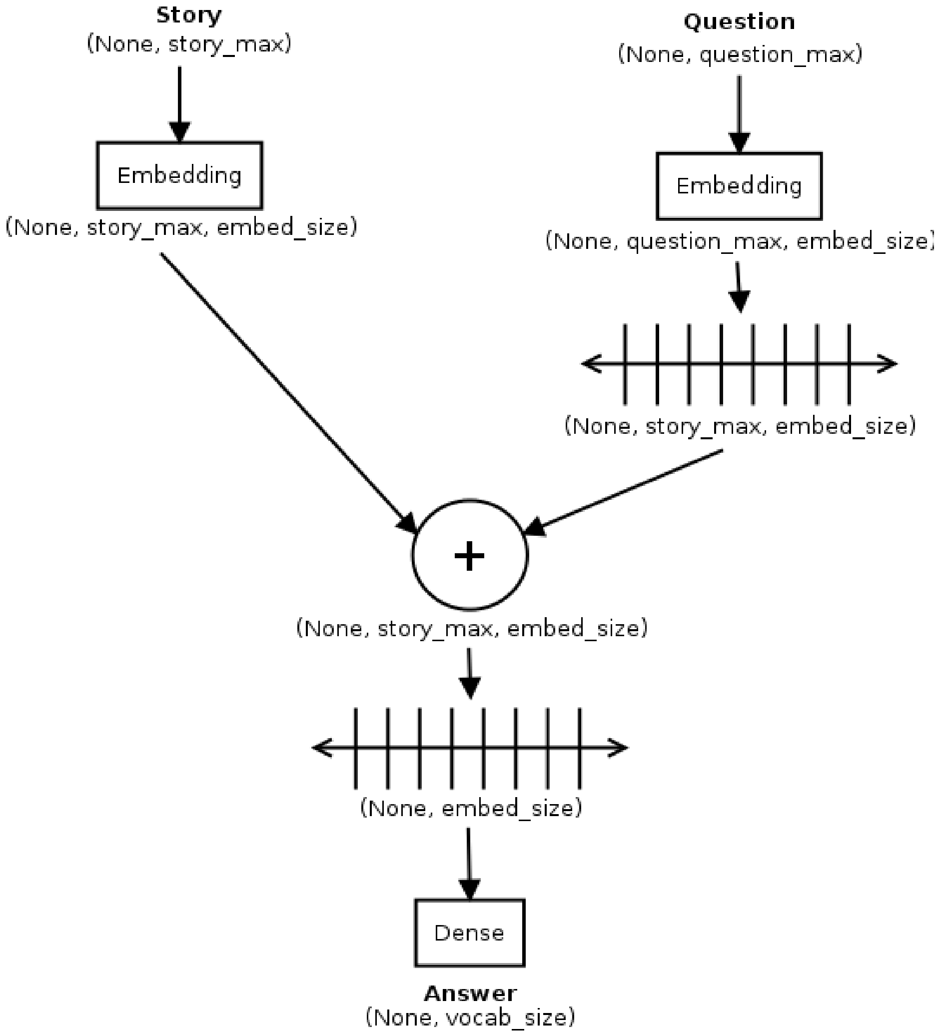
- **Single supporting fact**
- Two supporting facts
- Three supporting facts
- Two argument relations
- Three argument relations
- Yes/No questions
- Counting
- Lists/sets
- Simple Negation
- Indefinite Knowledge

- Basic Coreference
- Conjunction
- Compound Coreference
- Time Reasoning
- Basic Deduction
- Basic Induction
- Positional Reasoning
- Size Reasoning
- Path Finding
- Agent's Motivations

# bAbI Format (task 1)

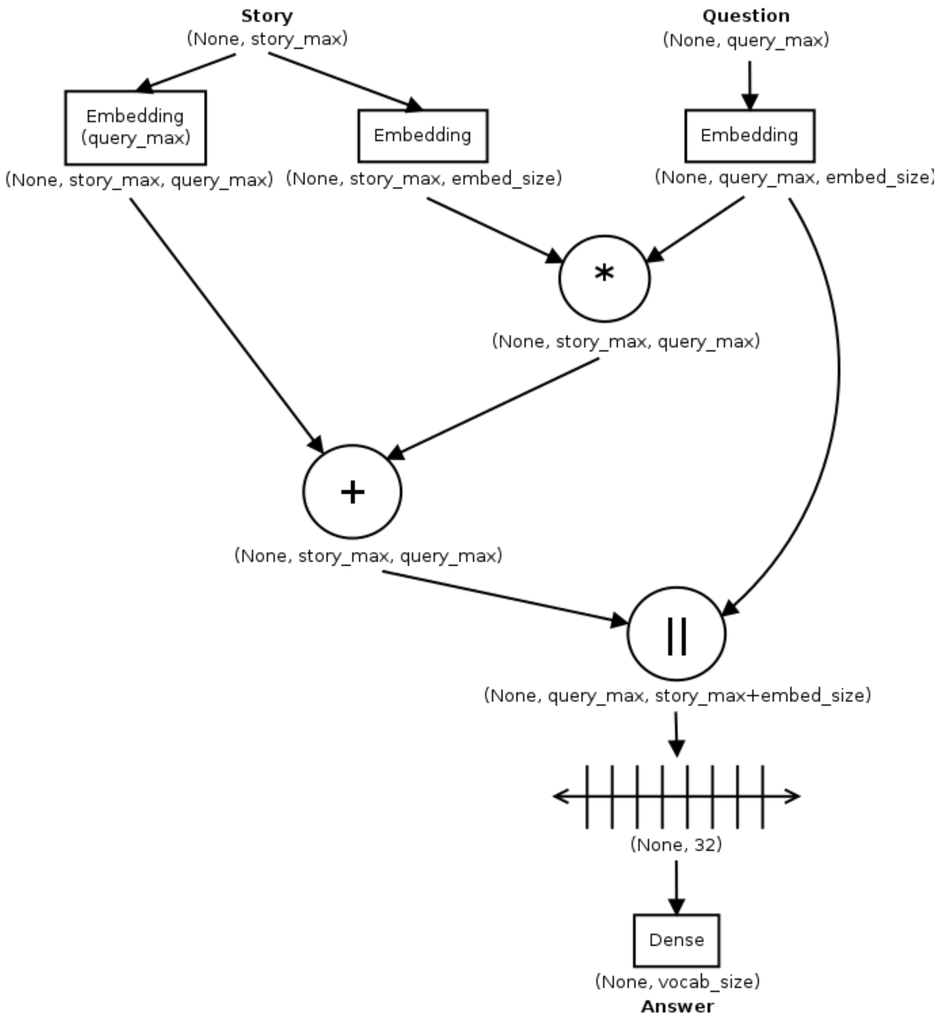


# bAbI LSTM




- Implementation based on the paper: **Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks**.
- Test accuracy reported in paper: 50%.
- Test accuracy achieved by implementation: 56%.
- Code is here.

# bAbI MemNN

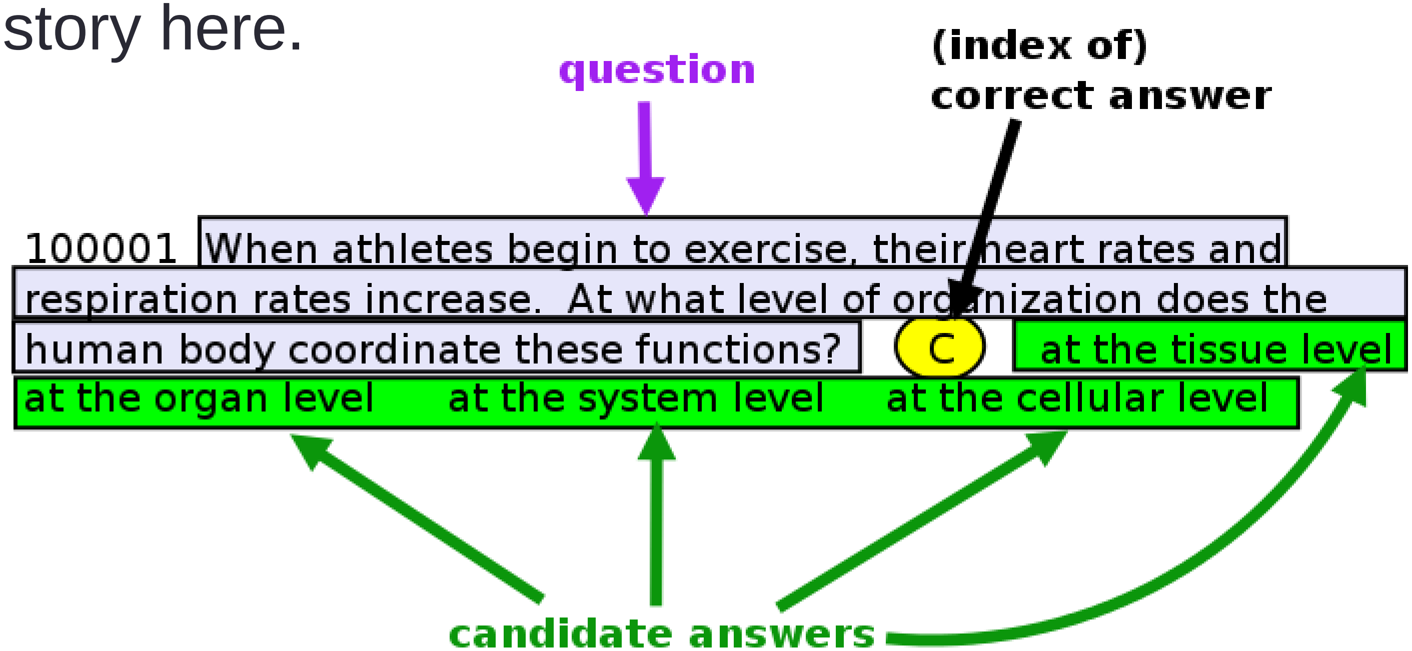



- Implementation based on the paper: **End-to-end Memory Networks**.
- Test accuracy reported in paper: 99%.
- Test accuracy achieved by implementation: 42%.
- Code is [here].

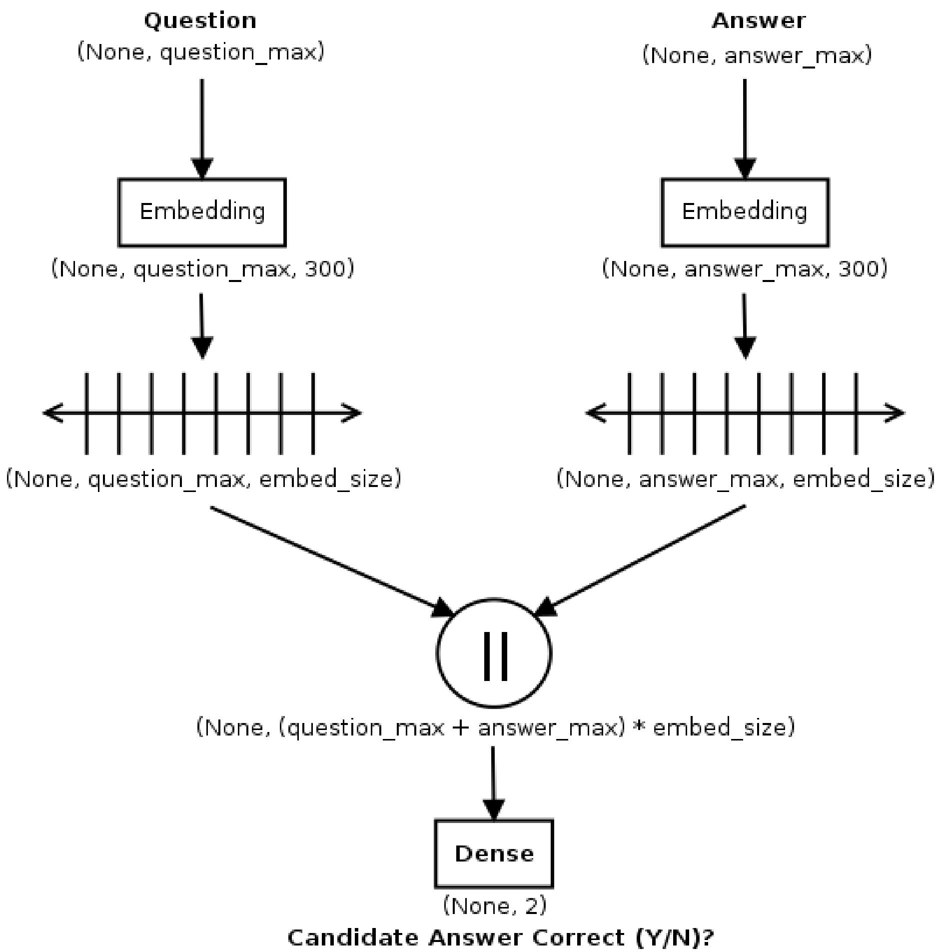
# Back to Kaggle

# Data Format

- 2000 multiple choice 8th Grade Science questions with 4 candidate answers and correct answer label.
- 2000 questions without correct answer label.
- Each question = 1 positive + 3 negative examples.
- No story here.



# QA-LSTM

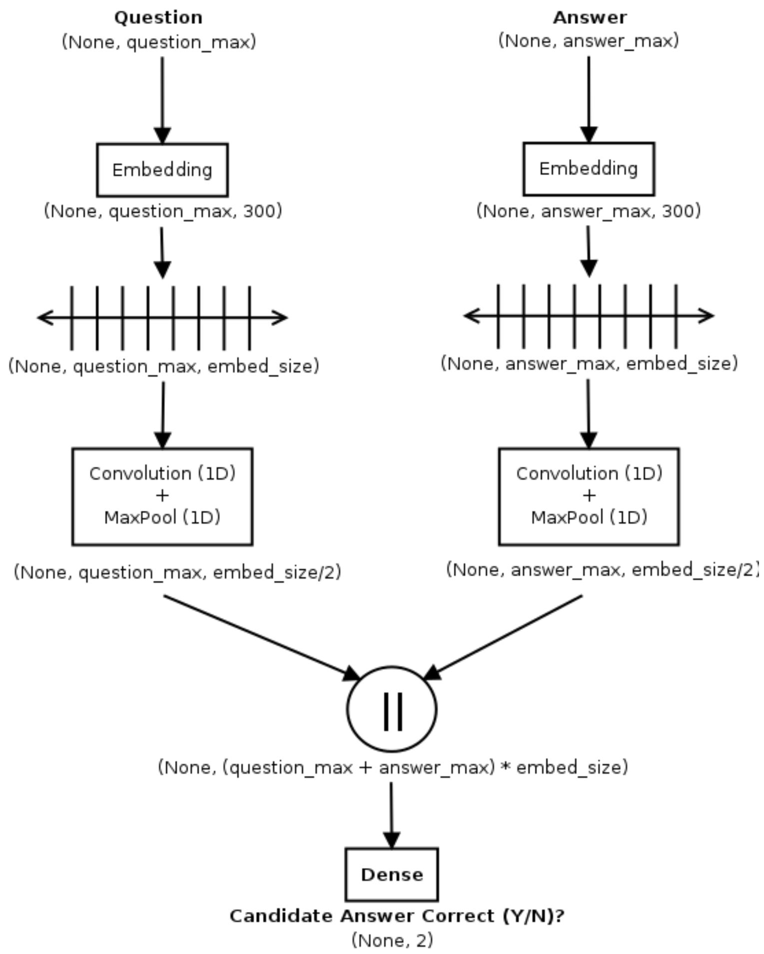



- Implementation based on the paper: **LSTM-based Deep Learning Models for Non-factoid Answer Selection**.
- Test accuracy reported in paper: 64.3% (InsuranceQA dataset).
- Test accuracy achieved by implementation: 56.93% (unidirectional) and 57% (bidirectional).
- Code: unidirectional, bidirectional

# Our Embedding Approach

- 3 approaches to Embedding
  - Generate from Data
  - Use External model for lookup
  - Initialize with External model, then fine tune.
- Not enough question data to generate good embedding.
- Used pre-trained Google News word2vec model (trained with 3 billion words)
- Model has 3 million word vectors of dimension (300,).
- Uses gensim to read model.

# QA-LSTM-CNN

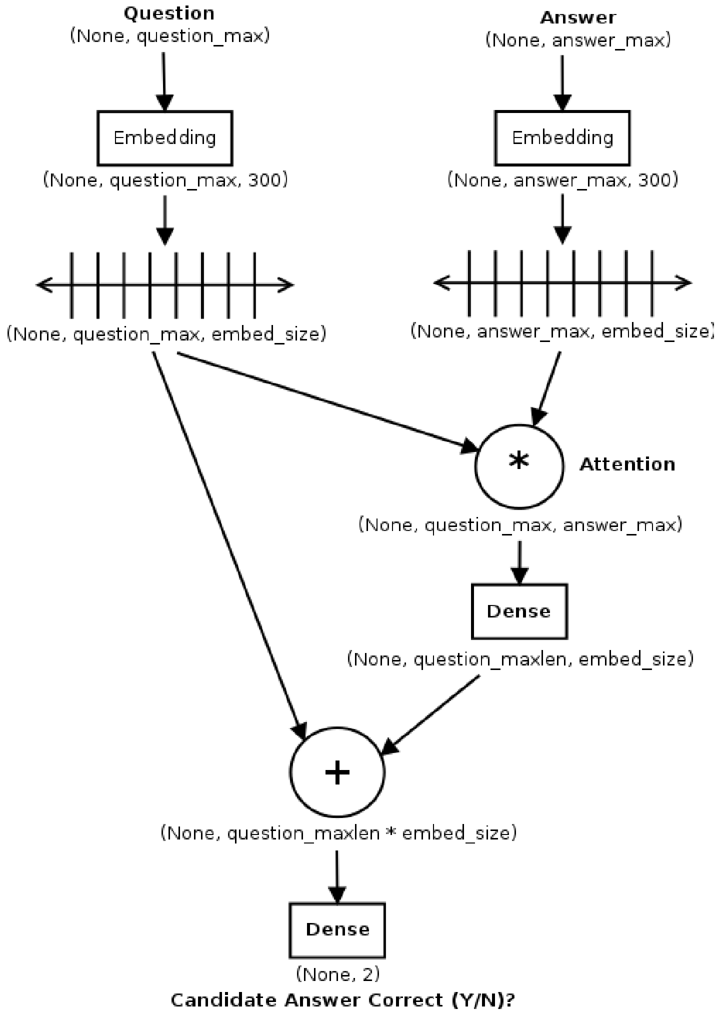



- Additional CNN Layer for more effective summarization.
- Test accuracy reported in paper: 62.2% (InsuranceQA dataset).
- Test accuracy achieved by implementation: 56.3% (unidirectional), did not try bidirectional.
- Code is [here](here).

# Incorporating Attention

- Vanishing Gradient problem addressed by LSTMs, but still shows up in long range Q+A contexts.
- Solved using [Attention Models]
  - Based on visual models of human attention.
  - Allow the network to focus on certain words in question with “high resolution” and the rest at “low resolution”.
  - Similar to advice given for comprehension tests about reading the questions, then scanning passage for question keywords.
  - Implemented here as a dot product of question and answer, or question and story vectors.

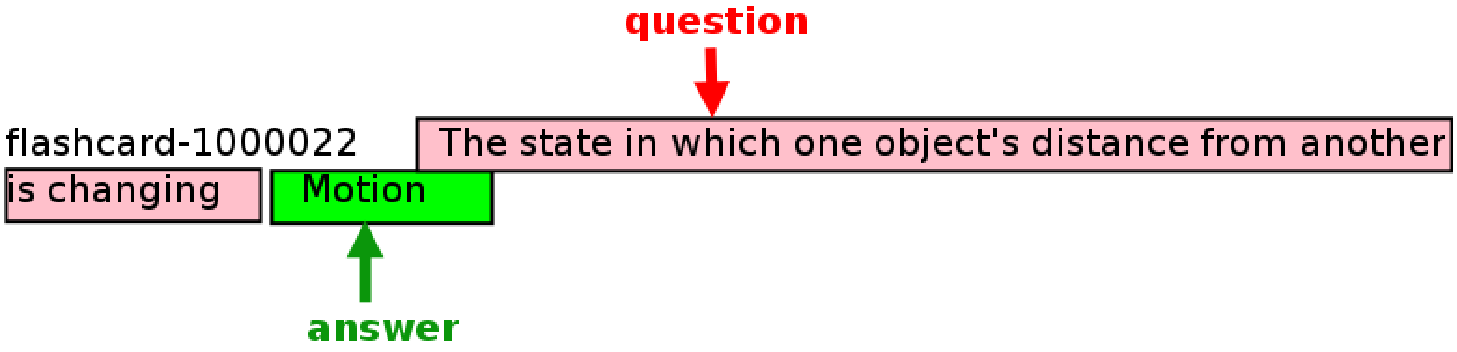
# QA-LSTM + Attention




- Attention vector from question and answer combined with question.
- Test accuracy reported in paper: 68.4% (InsuranceQA dataset).
- Test accuracy achieved by implementation: 62.93% (unidirectional), 60.43% (bidirectional)
- Code: unidirectional, bidirectional.

# Incorporating External Knowledge

- Contestants were allowed/advised to use external sources such as [ConceptNet], [CK-12 books], [Quizlets], [Flashcards from StudyStack], etc.
- Significant crawling/scraping and parsing effort involved.
- 4th place winner (tambietm) provides parsed download of StudyStack Flashcards on his Google drive.
- Flashcard "story" = question || answer



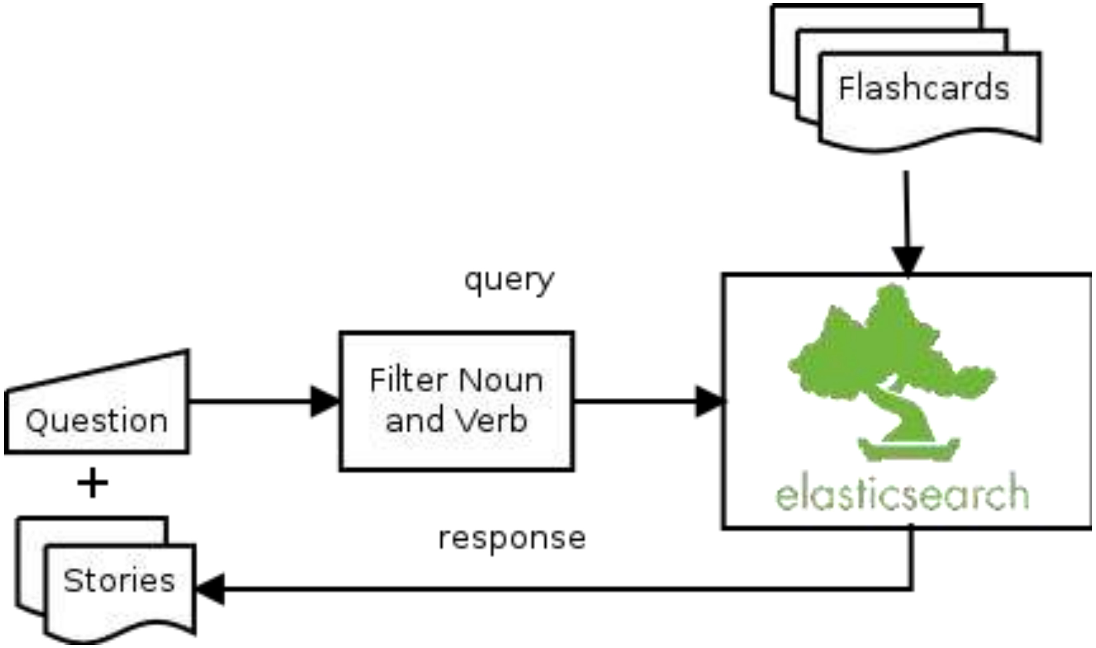
# Using Story Embedding

- Build Word2Vec model using words from Flashcards.
- Approximately 500k flashcards, 8,000 unique words.
- Provides smaller, more focused embedding space.
- Good performance boost over default Word2Vec embedding.

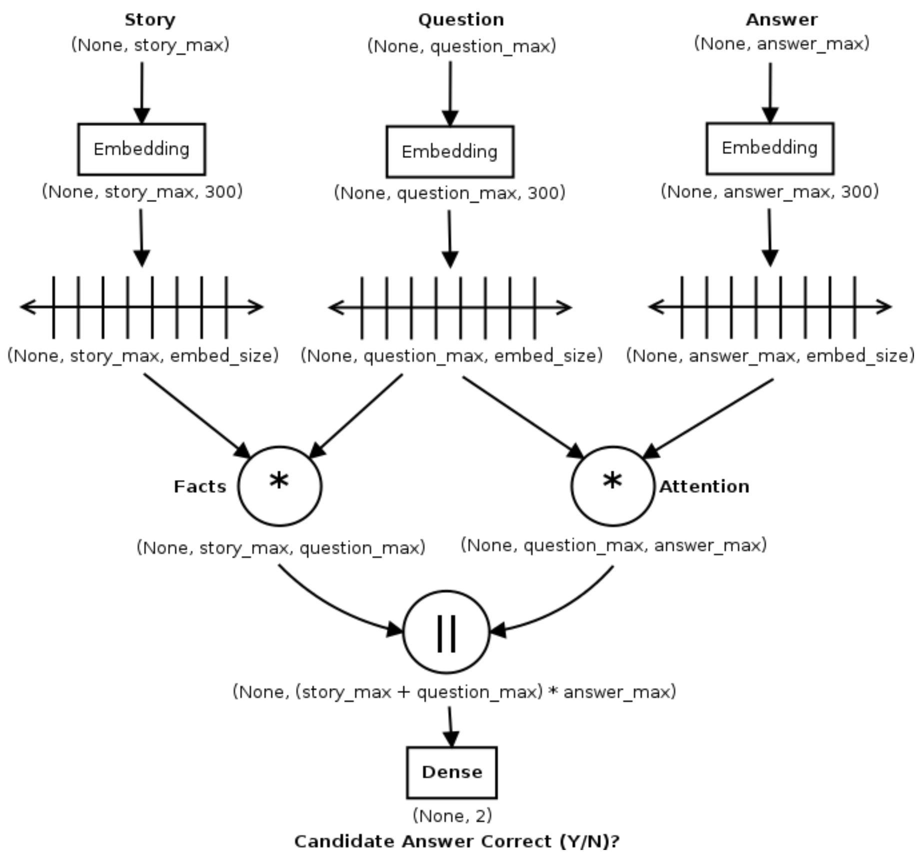
| Model | Default Embedding | Story Embedding |
|---|---|---|
| QA-LSTM with Attention | 62.93 | 76.27 |
| QA-LSTM Bidirectional with Attention | 60.43 | 76.27 |

# Relating Story to Question

- Replicate bAbI setup: (story, question, answer).
- Only a subset of flashcards relate to given question.
- Using traditional IR methods to generate flashcard stories for each question.



# QA-LSTM + Story




- Story and Question combined to create Fact vector.
- Question and Answer combined to create Attention vector
- Fact and Attention vectors concatenated.
- Test accuracy achieved by implementation: 70.47% (unidirectional), 61.77% (bidirectional).
- Code: unidirectional, bidirectional.

# Results

| Model Specifications | Test Accuracy (%) |
|---|---|
| QA-LSTM (Baseline) | 56.93 |
| QA-LSTM Bidirectional | 57.0 |
| QA-LSTM + CNN | 55.7 |
| QA-LSTM with Attention | 62.93 |
| QA-LSTM Bidirectional with Attention | 60.43 |
| QA-LSTM with Attention + Custom Embedding | 76.27 * |
| QA-LSTM Bidirectional w/Attention + Custom Embedding | 76.27 * |
| QA-LSTM + Attention + Story Facts | 70.47 |
| QA-LSTM Bidirectional + Attention + Story Facts | 61.77 |

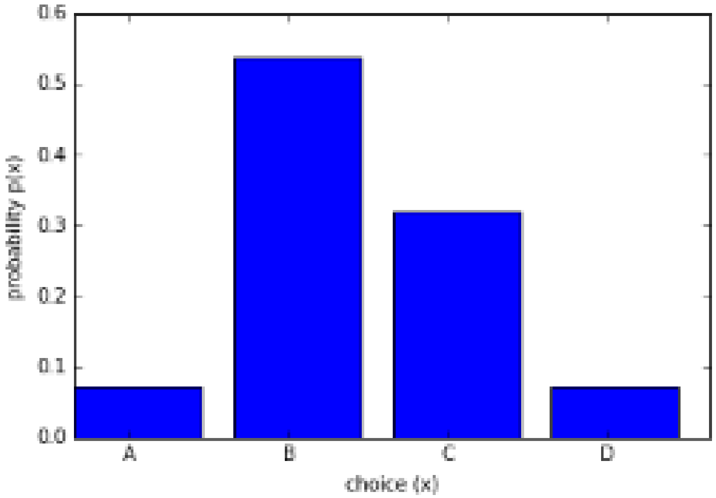
# Model Deployment

- Our models predict answer correct vs. incorrect.
- Task is to choose the correct answer from candidate answers.
- Re-instantiate trained model with Softmax layer removed.
- Run batch of (story, question, answer) for each candidate answer.
- Select best scoring answer as correct answer.

# Deploying Model - Example

Which is a distinction between an epidemic and a pandemic

[A] the symptoms of the disease
[B] the geographical area affected
[C] the species of organisms infected
[D] the season in which the disease spreads



# Future Work

- Would like to implement Dynamic Memory Network (DMN) and Hierarchical Attention Based Convolutional Neural Network (HABCNN) models against this data.
- Would like to submit to Kaggle to see how I did once Keras Issue 3927 is resolved.
- Would like to try out these models against Stanford Question Answering Dataset (SQuAD) based on Wikipedia articles.
- Would like to investigate Question Generation from text in order to generate training sets for Elsevier corpora.

# Administrivia

- Code for this talk:https://github.com/sujitpal/dl-models-for-qa
- Contact me for questions and suggestions: sujit.pal@elsevier.com

# Closing Thoughts

- Deep Learning is rapidly becoming a general purpose solution for nearly all learning problems.
- Information Retrieval approaches are still more successful on Question Answering than Deep Learning, but there are many efforts by Deep Learning researchers to change that.

Thank you.