

# Neural Machine Translation

Sargur N. Srihari

srihari@cedar.buffalo.edu

This is part of lecture slides on [Deep Learning](http://www.cedar.buffalo.edu/~srihari/CSE676):  
<http://www.cedar.buffalo.edu/~srihari/CSE676>

# Topics in NLP

1. N-gram Models
2. Neural Language Models
3. High-Dimensional Outputs
4. Combining Neural Language Models with n-grams
5. Neural Machine Translation
6. Historical Perspective

# Topics in Neural Machine Translation

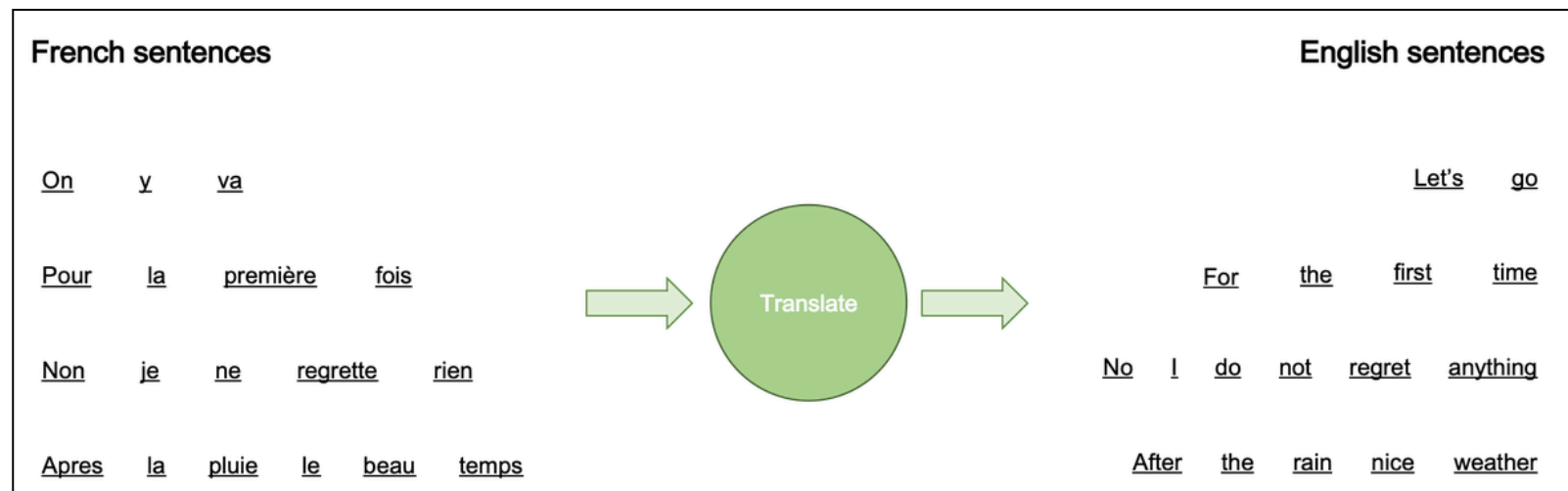
- Overview of Machine Translation (MT)
- An MLP approach to MT
- An RNN approach to MT
- Using an Attention Mechanism and Aligning Pieces of Data

# Example of Translation Task

- Source Language: English:
  - Would you like coffee or tea?
- Target Language:
  1. French: voulez-vous du café ou du thé
  2. German: Möchtest du Kaffee oder Tee
  3. Kannada: ನೀವು ಕಾಫಿ ಅಥವಾ ಚಹಾ ಬಯಸುವಿರಾ?
    - Neevu coffee athava chaha bayasuvira?
  4. Hindi: आप कॉफी या चाय पीना पसंद करेंगे
    - aap kophee ya chaay peena pasand karenge
  5. Tamil: நீங்கள் காபி அல்லது தேநீர் விரும்புகிறீர்களா?
    - Nīṅkaḷ kāpi allatu tēnīr virumpukirīrkaḷā?
  6. Japanese: コーヒーやお茶が好きですか？ Kōhī ka ocha ga īdesu ka
  7. Chinese: 你要咖啡还是茶 Nǐ yào kāfēi háishì chá

# What is Machine Translation (MT)?

- Read a sentence in a natural language and emit equivalent sentence in another language
- Computer program to convert source text to target text



# Importance of Machine Translation

Neural Machine Translation is eliminating demarcation between human and machine translation

Improved human productivity

Making machines more accurate going forward

## Machine Translation Engines

- [Amazon Translate](#)
- [CrossLang](#)
- [DeepL](#)
- [Google Translate](#)
- [Microsoft Translator](#)
- [Unbabel](#)
- [Watson Language Translator](#)

# Proposal and Evaluation Approach

- Two components

## 1. Proposal component suggests translations

- Many translations will not be grammatical
  - Many languages put adjectives after nouns, so when translated to English yield phrases such as “apple red”
- Proposal mechanism suggests translation variants
  - Ideally including “red apple”

## 2. Language model evaluates translations

- Assigns higher score to “red apple” than to “apple red”

# History of Machine Translation (MT)

- Early systems used variants of  $n$ -gram models
  - $n$ -gram models
    - Back-off  $n$ -gram models
    - Maximum entropy language models
      - an affine-softmax layer predicts the next word given the presence of frequent  $n$ -grams in the context
    - Report probability of a natural language sentence
- First neural networks upgraded the language models



# Extending to Conditional Models

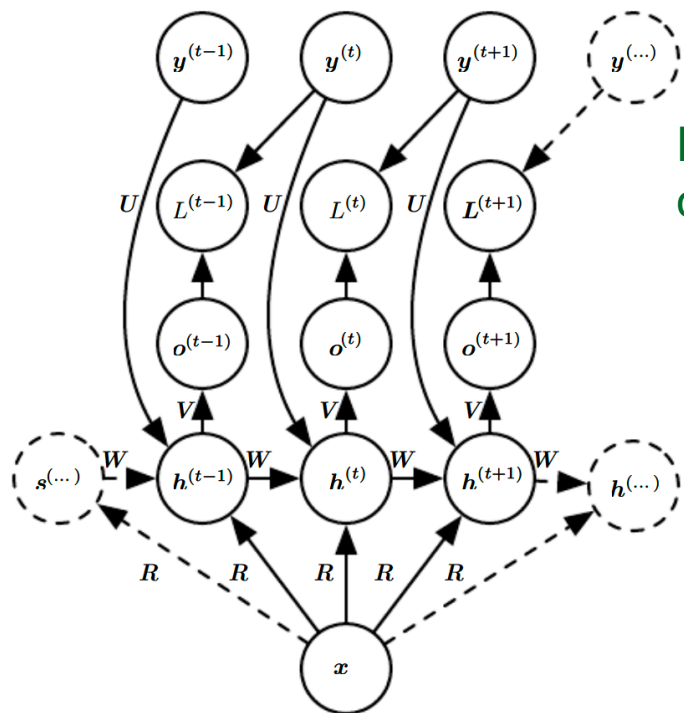
- Traditional language models simply report the probability of a natural language sentence
- Because MT produces an output sentence given an input sentence, extend the model to be conditional
- Straightforward to extend a model that defines a marginal distribution over some variable to define a conditional distribution over that variable given a context  $C$ , where  $C$  might be a single variable or a list of variables

# A Successful Conditional Model

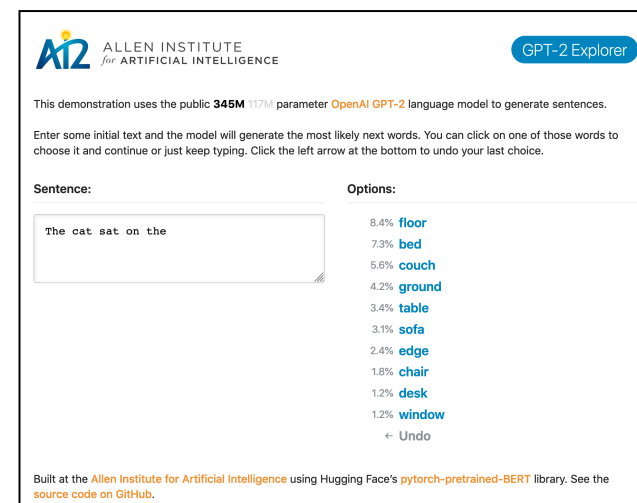
- An MLP MT model
  - Produces a conditional distribution given context  $C$ 
    - Where  $C$  is a single variable or a list of variables
  - An MLP scores a phrase  $t_1, \dots, t_k$  in the target language given a phrase  $s_1, \dots, s_n$  in the source language by estimating  $P(t_1, \dots, t_k | s_1, \dots, s_n)$
  - Beat state-of-the-art in statistical MT benchmarks
- Disadvantage of MLP model
  - Requires inputs to be processed be of fixed length

# An RNN model is an improvement

- RNN provides ability to accommodate variable length inputs and variable length outputs
- RNN represents a conditional distribution over a sequence given some input

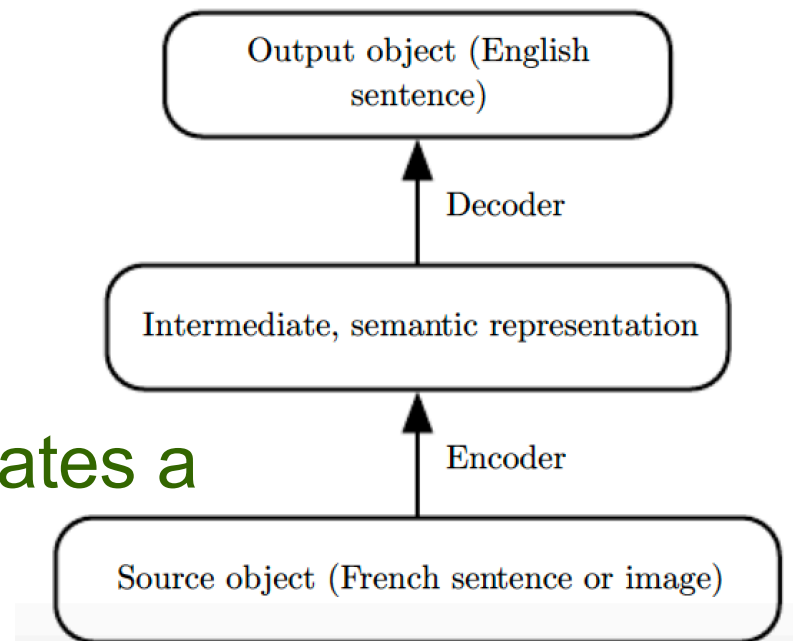


RNN maps a fixed-length vector  $x$  into a distribution over sequences  $Y$



# RNN Model

- One model reads input sequence and emits a data structure that summarizes the input
  - We call this summary “context”  $C$ 
    - $C$  may be a list of vectors, or a vector, or a tensor
  - This model may be an RNN
- A second model is an RNN
  - It reads context  $C$  and generates a sentence in target language
- This is an encoder-decoder framework



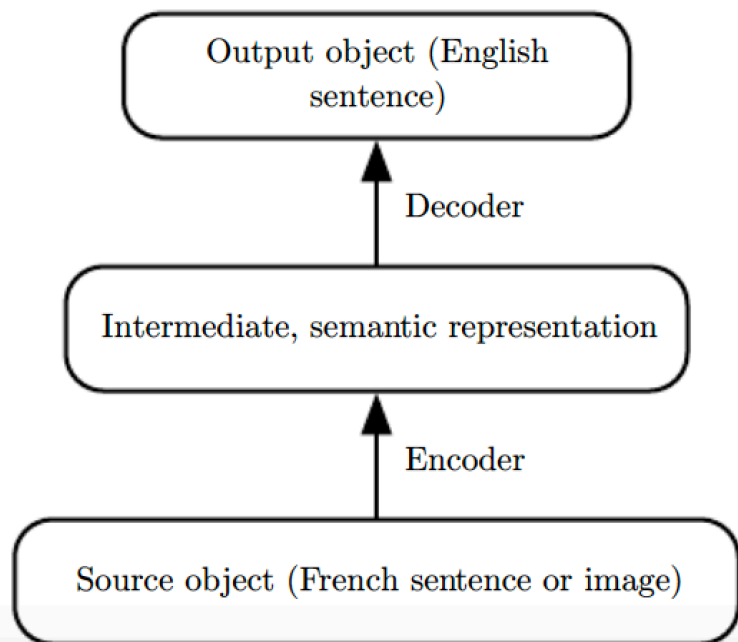
# The encoder-decoder architecture

Map back and forth between a surface representation (sequence of words) and a semantic representation

- Called an inter-lingua

Uses output of encoder of data from one modality (maps French to hidden representation capturing meaning)

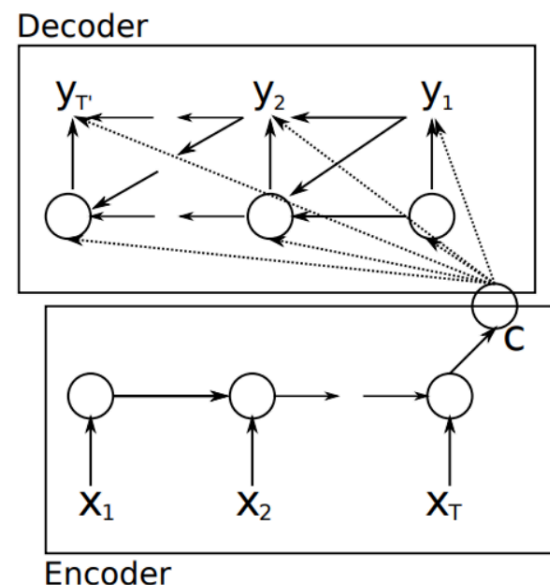
Provides as input to a decoder for another modality (maps from hidden to English)



This idea has been applied successfully not just to machine translation but also to caption generation from images

# RNN Encoder-Decoder

- To generate output sentence conditioned on source sentence, model represents entire source sentence
  1. Early models only able to represent individual words or phrases
  2. Neural models learn a representation in which
    - Sentences with same meaning have similar representations regardless of whether they were written in the source or target language



# Using an attention mechanism and aligning pieces of data

- Using a fixed-size representation to capture all the semantic details of a very long sentence of 60 words is very difficult
- Although it can be achieved by an RNN trained well-enough and long enough, a more efficient approach exists
  - Attention model!

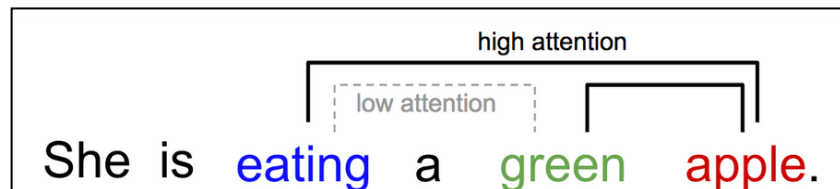
# Attention mechanism

- It is to read the whole sentence or paragraph (to get gist or context) then produce translated words one at a time each time focusing on a different part of the input sentence
- The attention mechanism is used to focus on specific parts of the input sequence at each time step



# What is Attention?

- It is how we correlate words in one sentence



<https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html>

- When we see “eating”, we expect a food word soon
  - “green” describes food, but more with “eating” directly
  - the word “chair” correlates with “green” but not with “eat”
- Attention in deep learning is a vector of importance weights
  - in order to predict or infer a word in a sentence, we estimate using the attention vector how strongly it is correlated with (or “*attends to*”) other elements

# AM in Sentiment Analysis

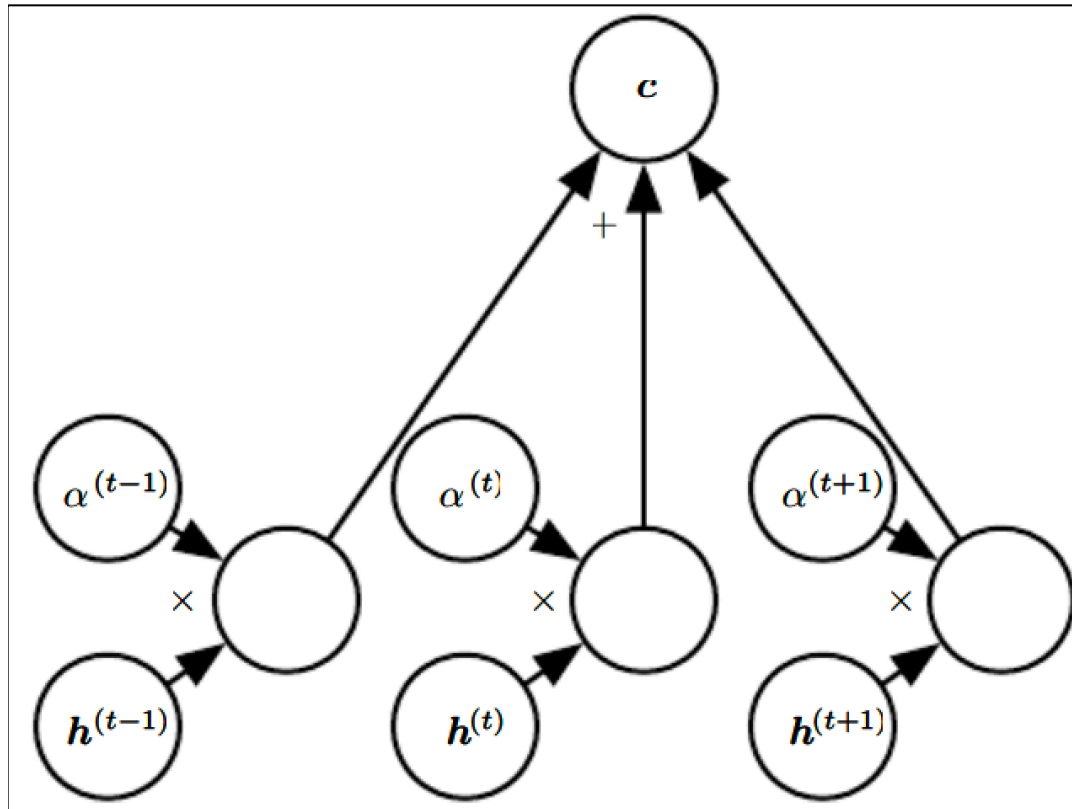
## An example review

1. pork belly= delicious.
2. scallops?
3. I don't even like scallops, and these were a-m-a-z-i-n-g
4. fun and tasty cocktails
5. next time I in Phoenix, I will go back here .
  - Highly recommend.

AM learns that out of five sentences, first and third sentences are more relevant

Furthermore, the words **delicious** and **amazing** within those sentences are more meaningful to determine the sentiment of the review

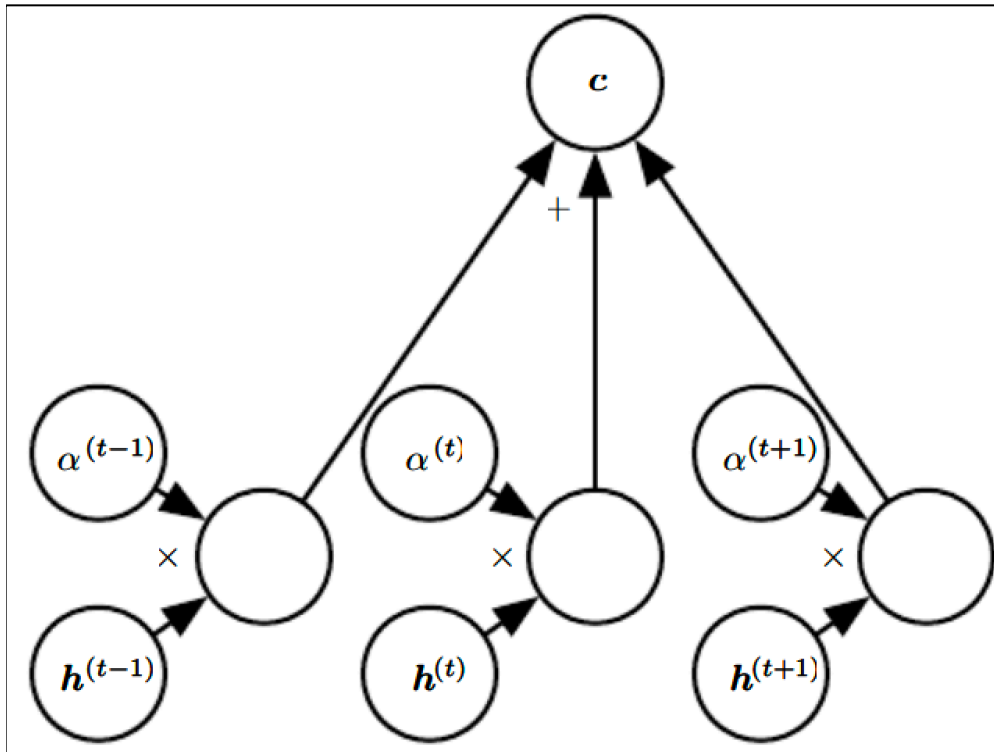
# Attention mechanism



$c$  is a context vector  
It is a weighted average of  
feature vectors  $h^{(t)}$  and  
weights  $\alpha^{(t)}$

The feature vectors  $h$  are  
hidden units of a neural network,  
but they may also be raw input  
to the model

# Weights of attention model

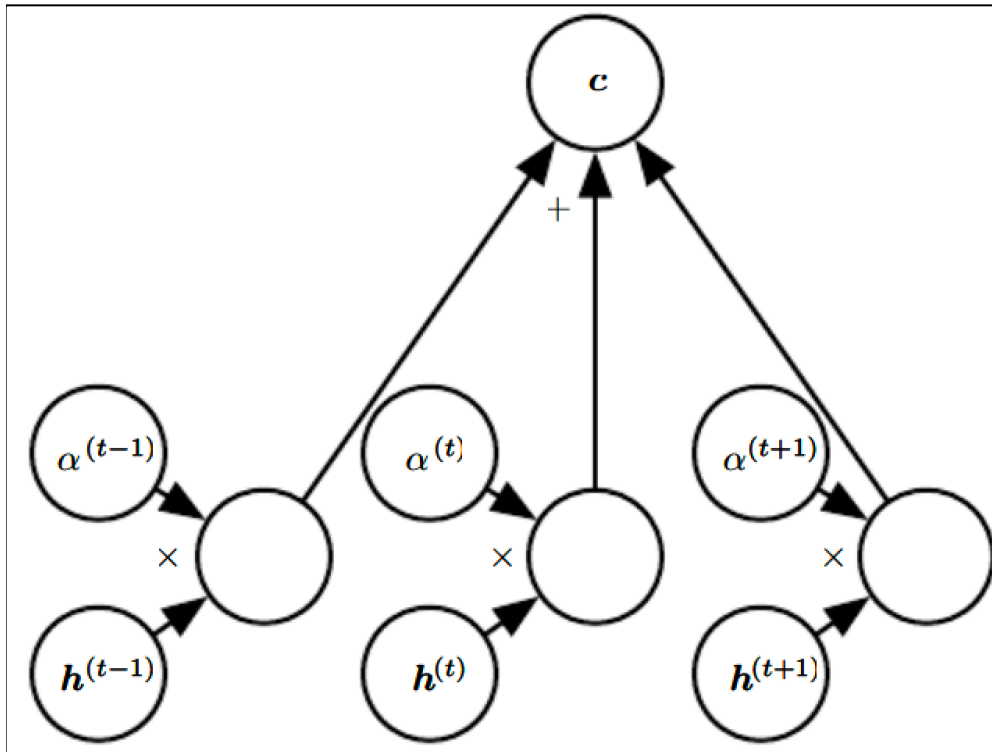


Weights  $\alpha^{(t)}$  are produced by the model itself

They are usually values in the interval  $[0,1]$  and are intended to concentrate around one  $h^{(t)}$  so that the weighted average approximates reading that one specific time precisely

Weights  $\alpha^{(t)}$  are produced by applying a softmax function to the relevant scores emitted by another portion of the model

# Attention model can be trained



The attention mechanism is more expensive computationally than directly indexing the desired  $h^{(t)}$

But direct indexing cannot be trained with gradient descent.

The attention mechanism based on weighted averages is a smooth, differentiable approximation that can be trained with existing approximation algorithms

# Three Components of Attention

- An attention-based system has 3 components:
  1. A process that *reads* raw data (such as source words in a source sentence) and converts them into distributed representations with one feature vector associated with each word position
  2. A list of feature vectors storing the output of the reader. This can be thought of as *memory* containing a sequence of facts, which can be retrieved, not necessarily in order
  3. A process that *exploits* the content of the memory to sequentially perform a task at each time step having the ability to put attention on one memory element
- The third component generates the translated sentence

# Relating word embeddings

- When words in one language are aligned with corresponding words in a translated sentence, we can relate corresponding word embeddings
- Earlier work:
  - Learn translation matrix relating word embeddings in a language with embeddings in another
    - Yielding lower alignment error rates than traditional methods based on frequency counts in phrase tables
- Extensions:
  - Cross-lingual word vectors
    - Allows training on larger datasets

# Importance of Attention Models

- Attention Model (AM) was first introduced for Machine Translation [Bahdanau et al., 2014]
- Now, widely used in neural networks for
  - NLP
  - Statistical Learning
  - Speech
  - Computer Vision



# Reasons for AM Advancement

1. AM models are state-of-the-art for tasks of
  - Machine Translation, Question Answering, Sentiment Analysis, Part-of-Speech tagging, Constituency Parsing and Dialogue Systems
2. Advantages beyond improving performance
  - Improving interpretability of neural networks, which are otherwise black-box models
3. Overcome challenges with RNNs
  - Performance with increase in length of input