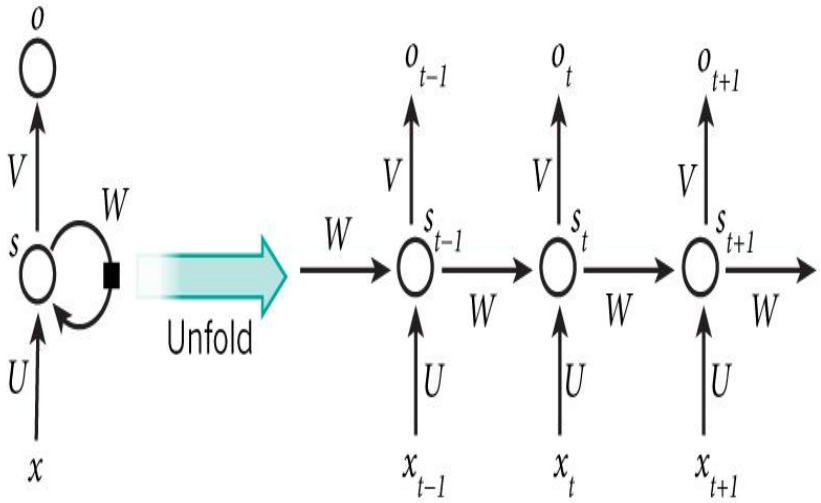
Recurrent Neural Nets and Language Models

Anup Deshmukh
IMT2014013

Advised by: Prof J. Dinesh

Sections

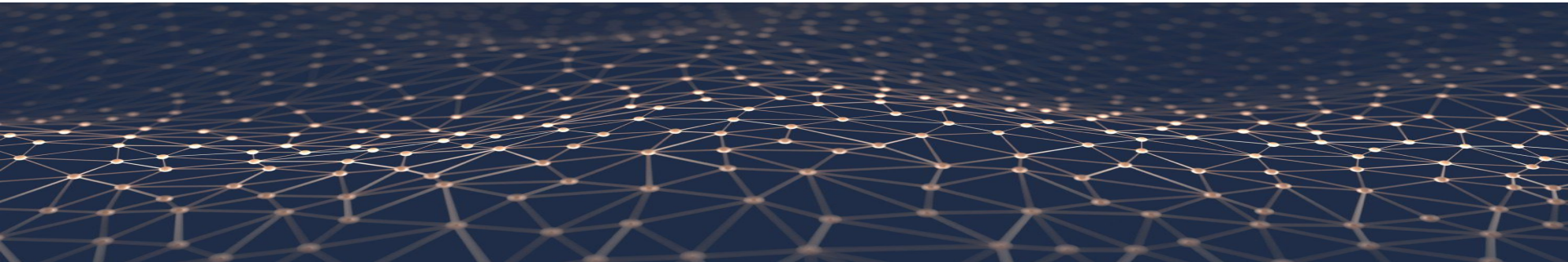
- Introduction to RNN's
  - Language Modelling
  - Vanishing Gradient in RNN's
- Understanding the problem of Image Description Generation
- Multimodal RNN architecture



1.1 Universal Approximation Theorem

"The universal approximation theorem states that a feed-forward network with a single hidden layer containing a finite number of neurons can approximate continuous functions on subsets of R^n."

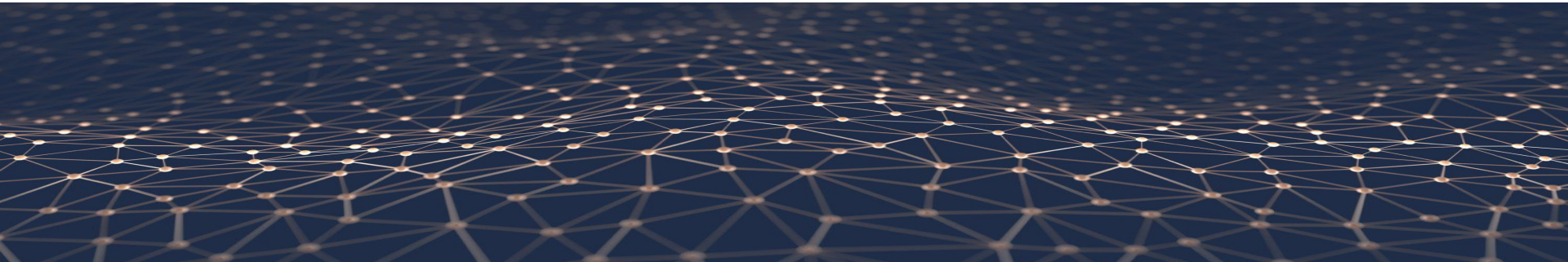
http://neuralnetworksanddeeplearning.com/chap4.html



1.2 RNN's take it a step further

"Some RNN's with proper weights and architecture qualify as Turing Complete." Turing Complete means in principle (although often not in practice) that this machine could be used to solve any computation problem.

https://stats.stackexchange.com/questions/220907/meaning-and-proof-of-rnn-can-approximate-any-algorithm/221142#221142



1.3 Importance in Language Modelling

1. RNN's have variable architecture. They allow us to have input data and/or output data of variable size

2. RNN's are more "biologically realistic" because of the recurrent connectivity found in the visual cortex of the brain

3. RNN's can deal with sequential or "temporal" data.

The person X is a criminal, and should be sent to ***Jail***

1.4 Traditional Language Model: N-gram Model

- The assumption that the probability of a word depends only on the previous word Markov is called a **Markov assumption**.
- Bigram probability of a word w_n given a previous word w_{n-1}, we'll compute the count of the bigram $C(w_n\ w_{n-1})$ and normalize by the sum of all the bigrams that share the same first word w_{n-1}.

$$P(w_n|w_1^{n-1}) \approx P(w_n|w_{n-N+1}^{n-1})$$

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w)}$$

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

1.5 Vanishing Gradient in RNN's

$$
\begin{aligned}
h_t &= \sigma\left(W^{(hh)} h_{t-1} + W^{(hx)} x_{[t]}\right) \\
\hat{y}_t &= \text{softmax}\left(W^{(S)} h_t\right)
\end{aligned}
$$

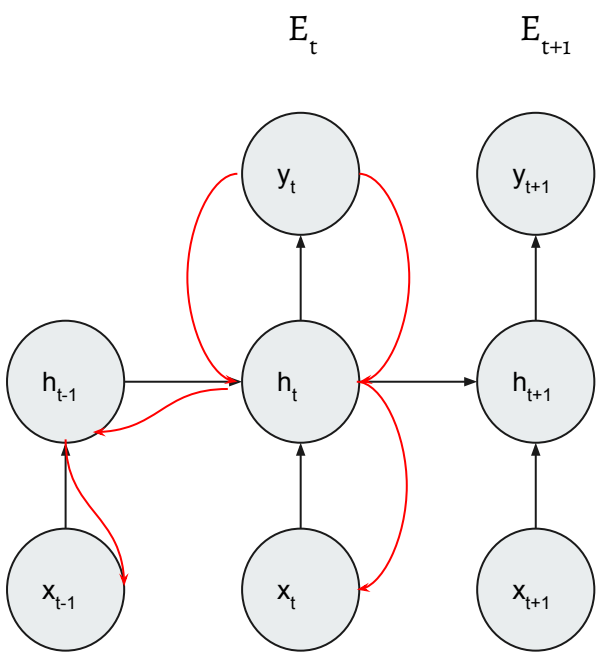
$$
J = -\frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{|V|} y_{t,j} \log \hat{y}_{t,j}
$$

1.6 Vanishing Gradient in RNN's

$$\frac{\partial E}{\partial W} = \sum_{t=1}^{T} \frac{\partial E_t}{\partial W}$$

$$\frac{\partial E_t}{\partial W} = \sum_{k=1}^{t} \frac{\partial E_t}{\partial y_t} \frac{\partial y_t}{\partial h_t} \frac{\partial h_t}{\partial h_k} \frac{\partial h_k}{\partial W}$$

$$\frac{\partial h_t}{\partial h_k} = \prod_{j=k+1}^{t} \frac{\partial h_j}{\partial h_{j-1}}$$



2.1 Image Description Generation

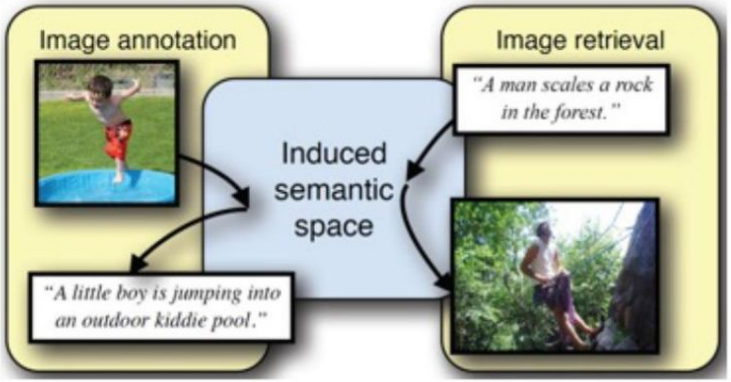
"girl in pink dress is jumping in air."

"black and white dog jumps over bar."

"a young boy is holding a baseball bat."

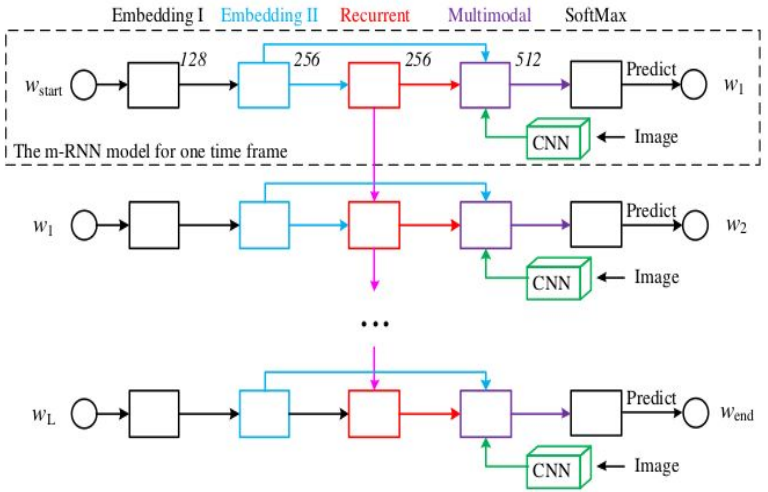
2.2 Retrieval based models

- Represent the given query image by specific visual features.
- Retrieve a candidate set of images from the training set based on a similarity measure in the feature space used.
- Re-rank the descriptions of the candidate images by further making use of visual and/or textual information contained in the retrieval set.



2.3 Multimodal RNN for Deep Captioning

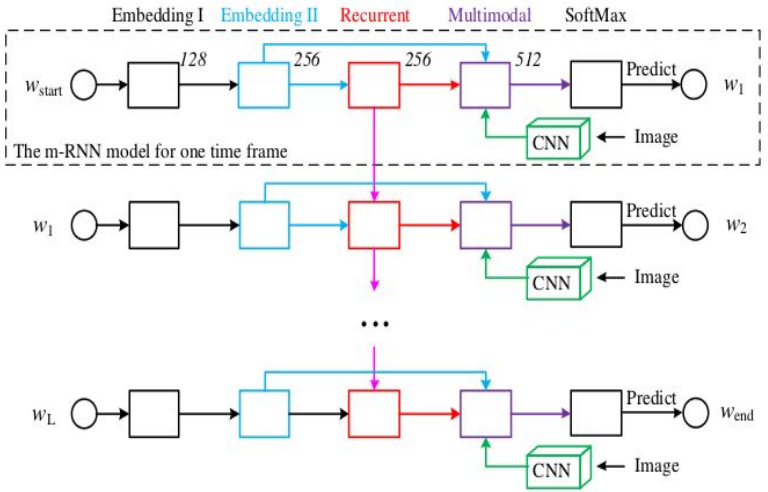
- Language model
  - Two word embedding layers and recurrent layer
- Vision model
  - CNN layer
- Multimodal model
  - Multimodal layer



2.4 Multimodal RNN for Deep Captioning

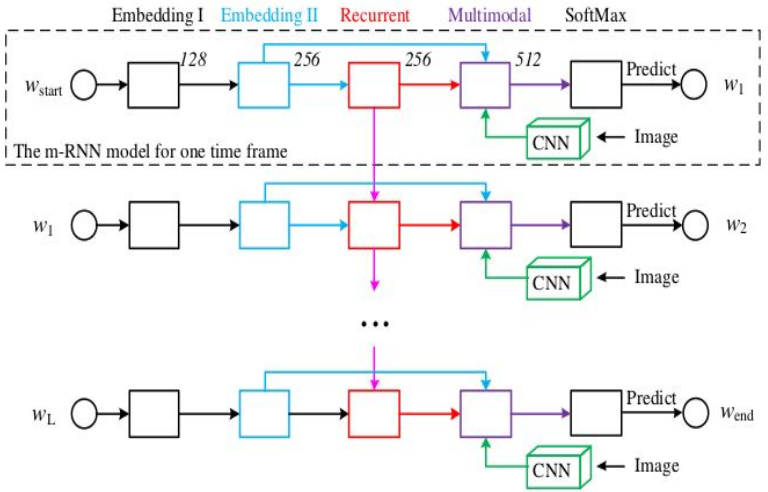
- **The two word embedding layers and recurrent layer**
  - Encodes both the syntactic and semantic meaning of the words.
  - Randomly initialize our word embedding layers and learn word embedding vectors from the training data.

$$r(t) = f_1(U_r.r(t-1) + w(t))$$



2.5 Multimodal RNN for Deep Captioning

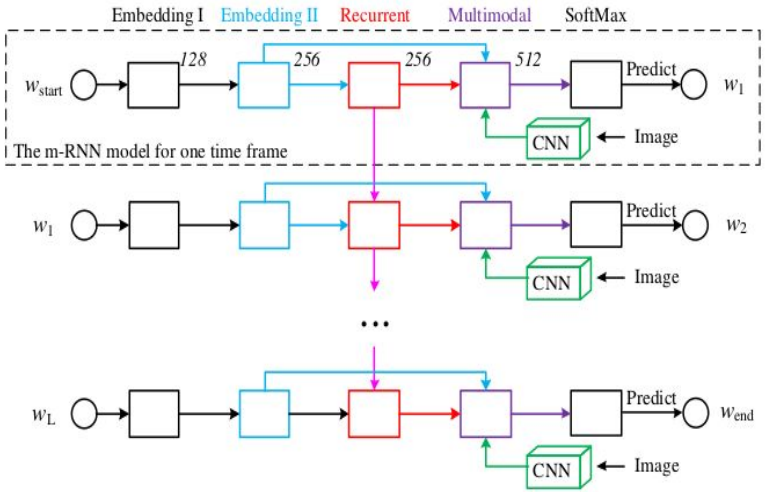
- **The CNN layer**
  - For the image representation, the activation of the 7 th layer of AlexNet or 15 th layer of VggNet are used.



2.6 Multimodal RNN for Deep Captioning

- **The multimodal layer**
  - The activation of the three layers are then mapped to the same multimodal feature space and added obtain the activation of the multimodal layer

$$m(t) = f_2(V_w.w(t) + V_r.r(t) + V_I.I)$$



2.7 Other Retrieval based models

State of art Technique	Model used
Socher et al. (2014)	DT-RNN (Dependency tree RNN)
Donahue et al. (2015)	Stack of 4 LSTM neural networks
Mao et al. (2015)	RNN
Karpathy and Fei-Fei (2015)	CNN and bi-directional RNN

3.1 Future Directions

1. Description of objects that are not depicted
2. Comprehensive but concise
3. Background Knowledge

Thank you.