Natural Language Understanding

Lecture 17: Distributed Representations for Documents

Mirella Lapata

School of Informatics
University of Edinburgh
mlap@inf.ed.ac.uk

March 25, 2016

1. Introduction
   - Sentences
   - Documents
2. Li and Hovy's Model
3. Other Models

Reading: Li and Hovy (EMNLP, 2014)

Representations for Sentences

1. Vector addition/multiplication
2. Convolutional Neural Networks (CNNs)
3. Sequential language models (RNNs, LSTMs)
4. Recursive neural networks

Representations for Sentences

1. Vector addition/multiplication
   - bag of words models
   - no tuning of representations
2. Convolutional Neural Networks (CNNs)
3. Sequential language models (RNNs, LSTMs)
4. Recursive neural networks

Representations for Sentences

1. Vector addition/multiplication
   - bag of words models
   - no tuning of representations
2. Convolutional Neural Networks (CNNs)
   - feature learning over word subsequences
   - not strictly compositional
3. Sequential language models (RNNs, LSTMs)
4. Recursive neural networks

Representations for Sentences

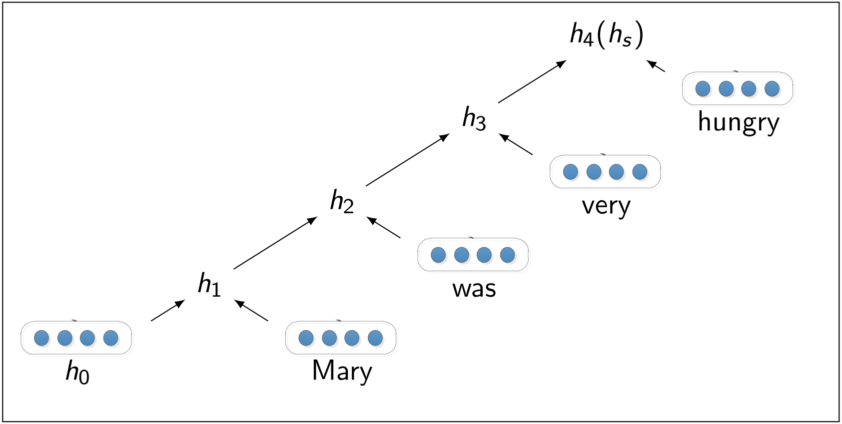
1. Vector addition/multiplication
   - bag of words models
   - no tuning of representations
2. Convolutional Neural Networks (CNNs)
   - feature learning over word subsequences
   - not strictly compositional
3. Sequential language models (RNNs, LSTMs)
   - structure $\approx$ linear order
   - compositional representations for length n sequences
4. Recursive neural networks

Representations for Sentences

1. Vector addition/multiplication
   - bag of words models
   - no tuning of representations
2. Convolutional Neural Networks (CNNs)
   - feature learning over word subsequences
   - not strictly compositional
3. Sequential language models (RNNs, LSTMs)
   - structure $\approx$ linear order
   - compositional representations for length n sequences
4. Recursive neural networks
   - structure $\approx$ binary trees
   - learn representations for linguistic units

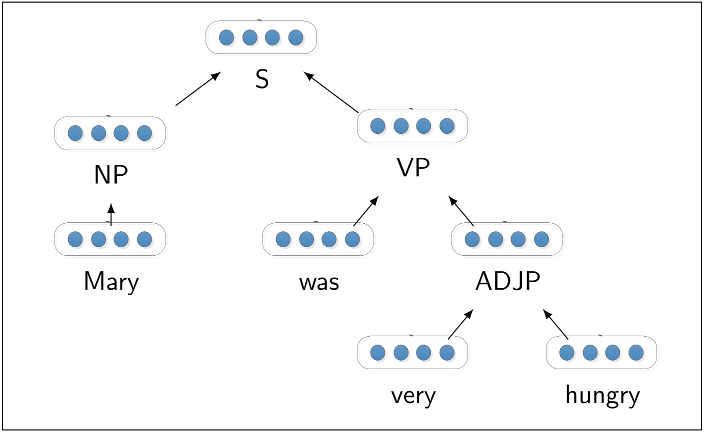
Representations for Documents

- We have to first decide how to represent sentences.
- Then, we compose sentences into a document representation.
- Many classes of models are possible!



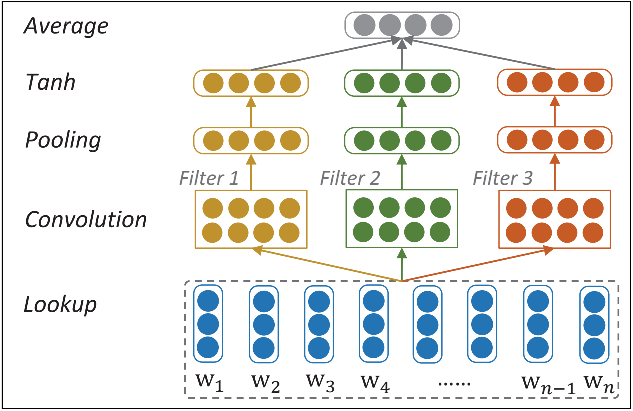
Representations for Documents

- We have to first decide how to represent sentences.
- Then, we compose sentences into a document representation.
- Many classes of models are possible!

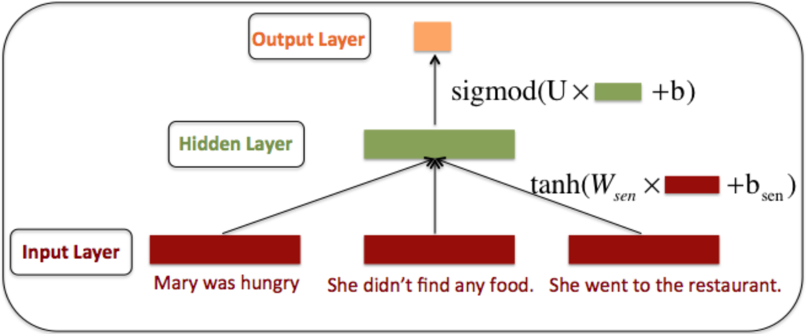


Representations for Documents

- We have to first decide how to represent sentences.
- Then, we compose sentences into a document representation.
- Many classes of models are possible!



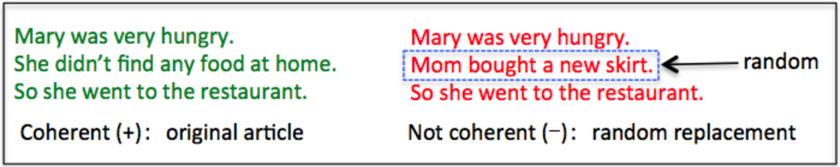
Representations for Documents



- Model is trained with a classification task in mind.
- Hidden layer learns document representation.
- We will see how this can be used for coherence modeling.

More Formally

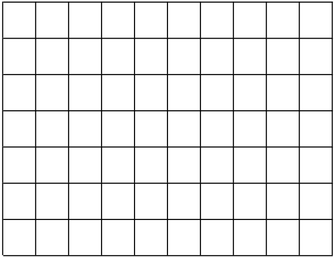
- Let clique C denote a window of sentences
- Each clique has a label $y_C \in 1$ if C coherent and 0 otherwise.
- How are labels generated?



More Formally

Mary was hungry
She didn't find any food at home
So she went to the restaurant
She had a prawn cocktail
It smelled funny
When she went home, she felt sick
She had food poisoning!

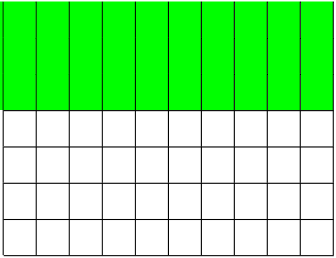
$d = 10$



More Formally

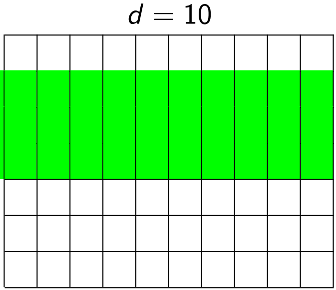
$d = 10$

Mary was hungry
She didn't find any food at home
So she went to the restaurant
She had a prawn cocktail
It smelled funny
When she went home, she felt sick
She had food poisoning!



More Formally

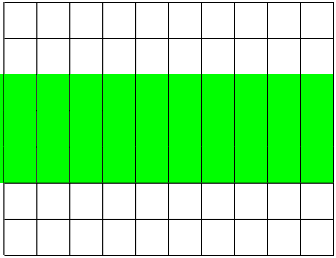
Mary was hungry
She didn't find any food at home
So she went to the restaurant
She had a prawn cocktail
It smelled funny
When she went home, she felt sick
She had food poisoning!



More Formally

Mary was hungry
She didn't find any food at home
So she went to the restaurant
She had a prawn cocktail
It smelled funny
When she went home, she felt sick
She had food poisoning!

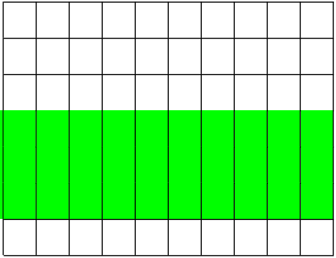
$d = 10$



More Formally

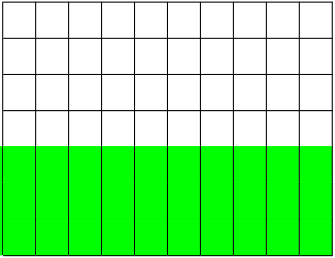
$d = 10$

Mary was hungry
She didn't find any food at home
So she went to the restaurant
She had a prawn cocktail
It smelled funny
When she went home, she felt sick
She had food poisoning!



More Formally

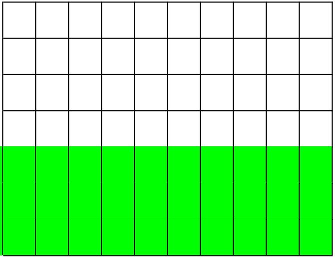
$d = 10$



Mary was hungry
She didn't find any food at home
So she went to the restaurant
She had a prawn cocktail
It smelled funny
When she went home, she felt sick
She had food poisoning!

More Formally

$d = 10$



Mary was hungry
She didn't find any food at home
So she went to the restaurant
She had a prawn cocktail
It smelled funny
When she went home, she felt sick
She had food poisoning!

More Formally

- Let $h_C = [h_{s1}, h_{s2}, \ldots, h_{sL}]$ denote concatenation of sentences.
- Each clique C takes as input a $L \times K$ vector h_C (L is the sentence window size, K dimensionality input sentence)
- Hidden layer H takes h_C as input and performs convolution using non-linear function:

$$q_C = \tanh(W_{sen} \times h_C + b_{sen})$$

- Output layer takes q_C and generates a scalar using linear function; sigmoid projects value to [0,1] probability space:

$$p(y_C = 1) = \text{sigmoid}(U^T q_C + b)$$

Training Objective

$$
\begin{aligned}
J(\Theta) \quad = \quad & \frac{1}{M} \sum_{C \in trainset} \{-y_C \log[p(y_C = 1)] \\
& - (1 - y_C) \log[1 - p(y_C = 1)]\} + \frac{Q}{2M} \sum_{\theta \in \Theta} \theta^2
\end{aligned}
$$

Training Objective

$$
\begin{aligned}
J(\Theta) &= \frac{1}{M} \sum_{C \in trainset} \{-y_C \log[p(y_C = 1)] \\
&\quad - (1 - y_C) \log[1 - p(y_C = 1)]\} + \frac{Q}{2M} \sum_{\theta \in \Theta} \theta^2
\end{aligned}
$$

- Collection of M labeled training cliques C

Training Objective

$$J(\Theta) = \frac{1}{M} \sum_{C \in trainset} \{-y_C \log[p(y_C = 1)] - (1 - y_C) \log[1 - p(y_C = 1)]\} + \frac{Q}{2M} \sum_{\theta \in \Theta} \theta^2$$

- Collection of M labeled training cliques C
- Error for one clique C with label y_C

Training Objective

$$
\begin{aligned}
J(\Theta) &= \frac{1}{M} \sum_{C \in trainset} \{-y_C \log[p(y_C = 1)] \\
&\quad - (1 - y_C) \log[1 - p(y_C = 1)]\} + \frac{Q}{2M} \sum_{\theta \in \Theta} \theta^2
\end{aligned}
$$

- Collection of M labeled training cliques C
- Error for one clique C with label y_C
- $\Theta = [W_{Recurrent}, W_{sen}, U_{sen}]$

Training Objective

$$
\begin{aligned}
J(\Theta) &= \frac{1}{M} \sum_{C \in trainset} \{-y_C \log[p(y_C = 1)] \\
&\quad - (1 - y_C) \log[1 - p(y_C = 1)]\} + \frac{Q}{2M} \sum_{\theta \in \Theta} \theta^2
\end{aligned}
$$

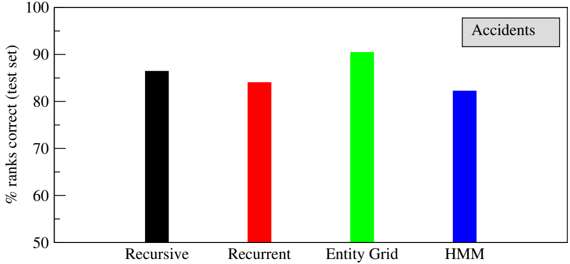
- Collection of M labeled training cliques C
- Error for one clique C with label y_C
- $\Theta = [W_{Recurrent}, W_{sen}, U_{sen}]$
- Regularization parameter

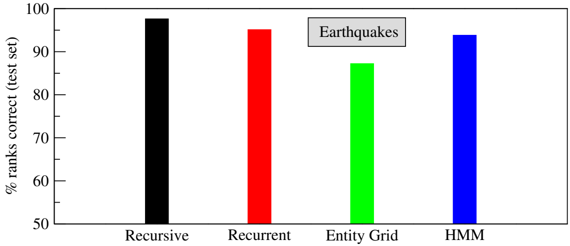
Coherence Rating

- Let S_d denote coherence score for document d
- d consists of sequence of sentences $d = \{s_1, s_2, \dots, s_{N_d}\}$
- The coherence score for a given document S_d is the probability that all cliques within d are coherent:

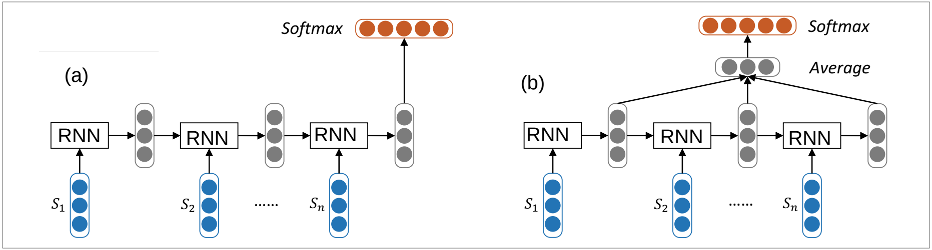
$$S_d = \prod_{C \in d} p(y_C = 1)$$

Results: Ordering





Hierarchical RNNs



Hierarchical CNNs

Document Embedding

Vectorisation

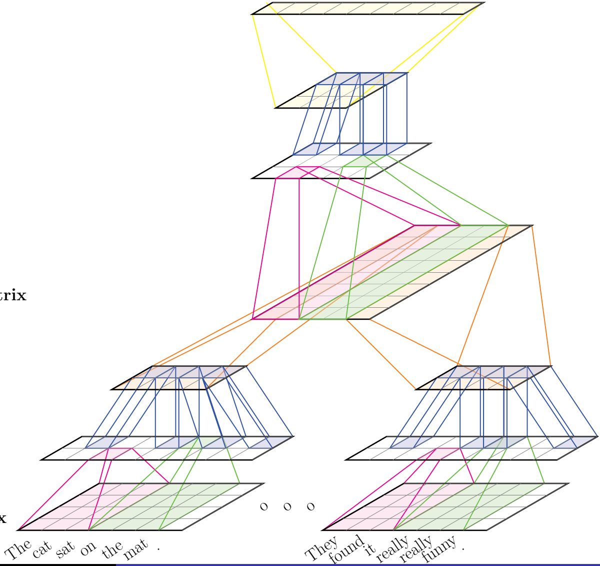
Stackable Layers
- Pooled representation
- K-max pooling
- Feature map
- Wide convolution

Sentence Embedding - Document Matrix

Vectorisation

Stackable Layers
- Pooled representation
- K-max pooling
- Feature map
- Wide convolution

Word Embedding - Sentence Matrix



Conclusions

- Techniques for sentence modeling transfer to documents
- Different classes of models depending on choice of composition model for sentences/documents
- Is it reasonable to compress the meaning of a document in a single vector?
- Choice is motivated by computational reasons.

Conclusions

- Techniques for sentence modeling transfer to documents
- Different classes of models depending on choice of composition model for sentences/documents
- Is it reasonable to compress the meaning of a document in a single vector?
- Choice is motivated by computational reasons.