

Application of neural networks to discourse coherence.

Chris Kedzie & Dwayne Campbell

February 18, 2015

Overview

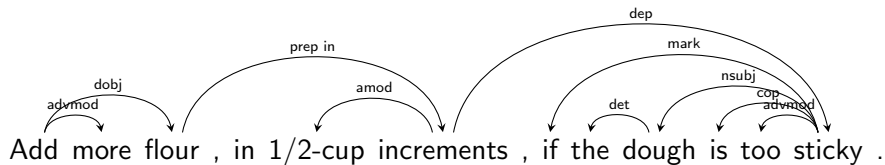
- 1 Introduction
- 2 Discourse in Linguistics
- 3 Task Definition
- 4 Computational Models
- 5 Proposal

Discourse Coherence

- 1 Introduction
- 2 Discourse in Linguistics
- 3 Task Definition
- 4 Computational Models
- 5 Proposal

Add more flour , in 1/2-cup increments , if the dough is too sticky .

Structure in NLP



- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.

- ① Start kneading the dough with both hands, primarily using the palms of your hands.
- ② Add more flour, in 1/2-cup increments, if the dough is too sticky.
- ③ Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.

- ① Start kneading the **dough** with both hands, primarily using the palms of your hands.
- ② Add more flour, in 1/2-cup increments, if the **dough** is too sticky.
- ③ Once the **dough** is a cohesive mass, remove the **dough** from the board and scrape up any left over dry bits.

- ① Start kneading the dough with both hands, primarily using the palms of your hands.
- ② Add more **flour**, in 1/2-cup increments, if the dough is too sticky.
- ③ Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.

- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.

- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 1 Start kneading the dough with both hands, primarily using the palms of your hands.

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- ! Place dough in oven and set for 400 degrees.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.

Discourse Structure is everywhere!

The screenshot shows the Wikipedia page for 'Artificial neural network'. The main content area displays a table of contents with the following structure:

- 1 Background
- 2 History
 - 2.1 Improvements since 2006
 - 2.2 Successes in pattern recognition contests since 2009
- 3 Models
 - 3.1 Network function
 - 3.2 Learning
 - 3.2.1 Choosing a cost function
 - 3.3 Learning paradigms
 - 3.3.1 Supervised learning
 - 3.3.2 Unsupervised learning
 - 3.3.3 Reinforcement learning
 - 3.4 Learning algorithms
- 4 Employing artificial neural networks
- 5 Applications
 - 5.1 Real-life applications
 - 5.2 Neural networks and neuroscience
 - 5.2.1 Types of models
- 6 Neural network software
- 7 Types of artificial neural networks
- 8 Theoretical properties
 - 8.1 Computational power
 - 8.2 Capacity
 - 8.3 Convergence
 - 8.4 Generalization and statistics
- 9 Controversies
 - 9.1 Training issues
 - 9.2 Hardware issues
 - 9.3 Practical counterexamples to criticisms
 - 9.4 Hybrid approaches
- 10 Gallery
- 11 See also
- 12 References
- 13 Bibliography
- 14 External links

At the bottom of the main content area, there is a link for 'Background [edit]'.

The right sidebar contains several categorized lists:

- Problems**
 - Classification • Clustering • Regression
 - Anomaly detection • Association rules
 - Reinforcement learning
 - Structured prediction • Feature learning
 - Online learning
 - Semi-supervised learning • Grammar induction
- Supervised learning (classification • regression)**
 - Decision trees • Ensembles (Bagging, Boosting, Random forest) • k-NN • Linear regression • Naive Bayes
- Neural networks** • Logistic regression
 - Perceptron
 - Support vector machine (SVM) • Relevance vector machine (RVM)
- Clustering**
 - BIRCH • Hierarchical • k-means • Expectation-maximization (EM) • DBSCAN • OPTICS • Mean-shift
- Dimensionality reduction**
 - Factor analysis • CCA • ICA • LDA • NMF • PCA • t-SNE
- Structured prediction**
 - Graphical models (Bayes net, CRF, HMM)
- Anomaly detection**
 - k-NN • Local outlier factor
- Neural nets**
 - Autoencoder • Deep learning • Multilayer perceptron • RNN • Restricted Boltzmann machine • SOM • Convolutional neural network
- Theory**
 - Bias-variance dilemma • Computational learning theory • Empirical risk minimization • PAC learning • Statistical learning • VC theory
- Machine learning portal**
- Computer science portal**

Discourse Coherence

- 1 Introduction
- 2 Discourse in Linguistics**
- 3 Task Definition
- 4 Computational Models
- 5 Proposal

- 1 Describe the relations that hold between discourse units.

- 1 Describe the relations that hold between discourse units.
- 2 Varying levels of granularity.

- 1 Describe the relations that hold between discourse units.
- 2 Varying levels of granularity.
- 3 Varying linguistic theories.

Discourse Parsing is Hard

- 1 High ambiguity.

Discourse Parsing is Hard

- ① High ambiguity.
- ② Annotation is complex.

Discourse Coherence

- 1 Introduction
- 2 Discourse in Linguistics
- 3 Task Definition**
- 4 Computational Models
- 5 Proposal

Task Definition

Let h be a human authored document.

Task Definition

Let h be a human authored document.

Let $perm(h)$ be the set of all permutations of h .

Task Definition

Let h be a human authored document.

Let $perm(h)$ be the set of all permutations of h .

Learn a function $rank$,

s.t. $rank(h) > rank(p)$ for all $p \in \{d \in perm(h) : d \neq h\}$.

Discourse Coherence

- 1 Introduction
- 2 Discourse in Linguistics
- 3 Task Definition
- 4 Computational Models**
- 5 Proposal

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.

- 1 Start kneading the **dough** with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the **dough** is too sticky.
- 3 Once the **dough** is a cohesive mass, remove the **dough** from the board and scrape up any left over dry bits.

- ① dough, *direct object*
- ② Add more flour, in 1/2-cup increments, if the **dough** is too sticky.
- ③ Once the **dough** is a cohesive mass, remove the **dough** from the board and scrape up any left over dry bits.

- ① dough, *direct object*
- ② dough, *subject*
- ③ Once the **dough** is a cohesive mass, remove the **dough** from the board and scrape up any left over dry bits.

- 1 dough, *direct object*
- 2 dough, *subject*
- 3 dough, *subject; direct object*

dough

Role. 1 *S*

Role. 2 *O*

Role. 3 *S*

Entity Grid

	dough	flour
Role. 1	<i>S</i>	—
Role. 2	<i>O</i>	<i>O</i>
Role. 3	<i>S</i>	—

Entity Grid

	dough	flour	hands
Role. 1	<i>S</i>	–	<i>X</i>
Role. 2	<i>O</i>	<i>O</i>	–
Role. 3	<i>S</i>	–	–

Entity Grid

	dough	flour	hands	...
Role. 1	<i>S</i>	–	<i>X</i>	...
Role. 2	<i>O</i>	<i>O</i>	–	...
Role. 3	<i>S</i>	–	–	...

Entity Grid

	dough	flour	hands	...
Role. 1	S	—	X	...
Role. 2	O	O	—	...
Role. 3	S	—	—	...

$$\mathbf{x} = [p(O|S) \quad]$$

Entity Grid

	dough	flour	hands	...
Role. 1	S	—	X	...
Role. 2	O	O	—	...
Role. 3	S	—	—	...

$$\mathbf{x} = [p(O|S), p(S|S \rightarrow O), \dots]$$

Entity Grid

	dough	flour	hands	...
Role. 1	S	—	X	...
Role. 2	O	O	—	...
Role. 3	S	—	—	...

$$\mathbf{x} = [p(O|S), p(S|S \rightarrow O), \dots]$$

Learn a linear ranking function using this vector representation, e.g.,

$$\mathbf{w} \cdot \mathbf{x}_h > \mathbf{w} \cdot \mathbf{x}_p$$

where \mathbf{x}_h and \mathbf{x}_p are derived from human authored and randomly permuted documents respectively.

Neural Networks: Li and Hovy, 2014

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 4 Lightly flour the board and continue kneading for 3 more minutes.
- 5 All hail Zuul!
- 6 Continue to knead for another 3 minutes, remembering to dust your board with flour when necessary.

Neural Networks: Li and Hovy, 2014

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 4 Lightly flour the board and continue kneading for 3 more minutes.
- 5 All hail Zuul!
- 6 Continue to knead for another 3 minutes, remembering to dust your board with flour when necessary.

Neural Networks: Li and Hovy, 2014

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 4 Lightly flour the board and continue kneading for 3 more minutes.
- 5 All hail Zuul!
- 6 Continue to knead for another 3 minutes, remembering to dust your board with flour when necessary.

$$p(y_2 = \text{coherent} | s_1, s_2, s_3) = .85$$

Neural Networks: Li and Hovy, 2014

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 4 Lightly flour the board and continue kneading for 3 more minutes.
- 5 All hail Zuul!
- 6 Continue to knead for another 3 minutes, remembering to dust your board with flour when necessary.

$$p(y_3 = \text{coherent} | s_2, s_3, s_4) = .87$$

Neural Networks: Li and Hovy, 2014

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 4 Lightly flour the board and continue kneading for 3 more minutes.
- 5 All hail Zuul!
- 6 Continue to knead for another 3 minutes, remembering to dust your board with flour when necessary.

$$p(y_4 = \text{coherent} | s_3, s_4, s_5) = .48$$

Neural Networks: Li and Hovy, 2014

- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 4 Lightly flour the board and continue kneading for 3 more minutes.
- 5 All hail Zuul!
- 6 Continue to knead for another 3 minutes, remembering to dust your board with flour when necessary.

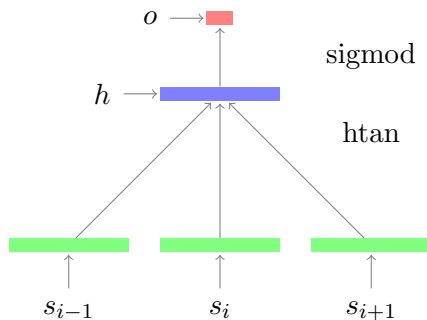
$$p(y_5 = \text{coherent} | s_4, s_5, s_6) = .12$$

Neural Networks: Li and Hovy, 2014

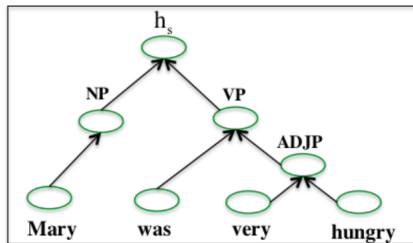
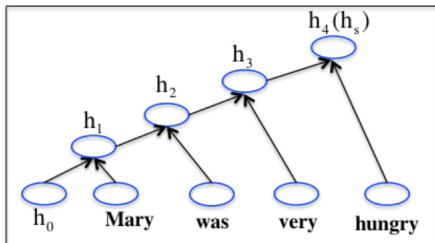
- 1 Start kneading the dough with both hands, primarily using the palms of your hands.
- 2 Add more flour, in 1/2-cup increments, if the dough is too sticky.
- 3 Once the dough is a cohesive mass, remove the dough from the board and scrape up any left over dry bits.
- 4 Lightly flour the board and continue kneading for 3 more minutes.
- 5 All hail Zuul!
- 6 Continue to knead for another 3 minutes, remembering to dust your board with flour when necessary.

$$p(y_6 = \text{coherent} | s_7, s_6, s_7) = .85$$

Neural Networks: Li and Hovy, 2014



Sentence Representation



Discourse Coherence

- 1 Introduction
- 2 Discourse in Linguistics
- 3 Task Definition
- 4 Computational Models
- 5 Proposal**

Convolutional Neural Network

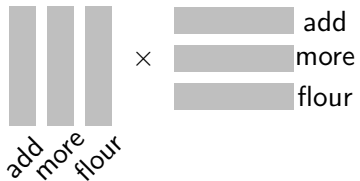
- RNN's don't propagate important features efficiently
- Hierarchical NN's require a parse (expensive)

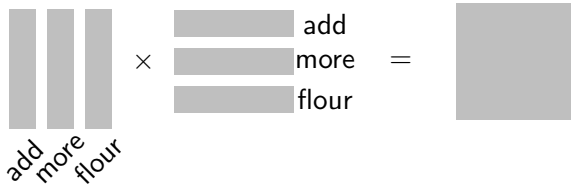
Convolutional Neural Network

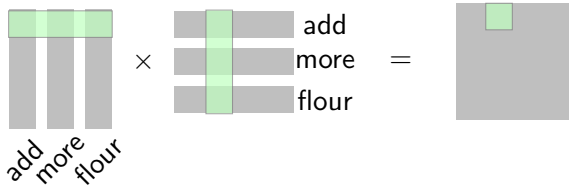
- RNN's don't propagate important features efficiently
- Hierarchical NN's require a parse (expensive)
- Perhaps there is a convolutional approach...



add
more
flour

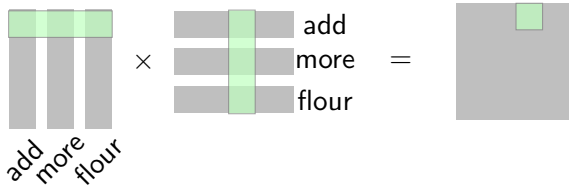






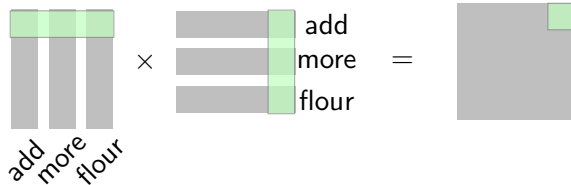
Sentence Kernel





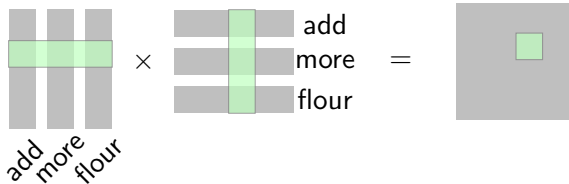
Sentence Kernel





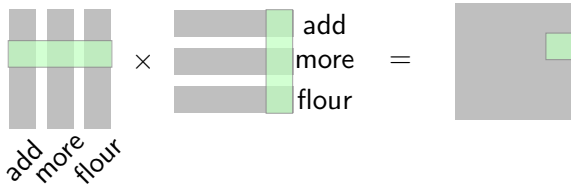
Sentence Kernel





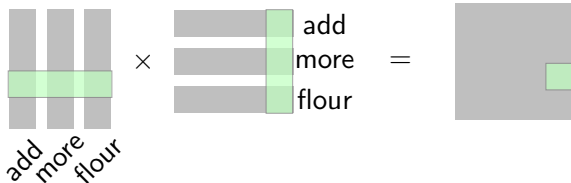
Sentence Kernel





Sentence Kernel





Sentence Kernel



The End