

Text Segmentation Using Affinity Propagation

Anna Kazantseva and Stan Szpakowicz
University of Ottawa
{ankazant,szpak}@site.uottawa.ca

June 6, 2011

Plan of the talk

- Motivation
- State-of-the-art in text segmentation
- Task definition and the proposed solution
 - Background
 - Affinity Propagation for clustering
 - Logic behind the derivations
- The algorithm
- Evaluation
- Conclusions and future work

We need text segmentation...

... whenever we want a simple (or simplified) view of a text's structure:

- speech transcripts – they have no clear organisation;
- long documents: books, in particular novels.

In many NLP tasks which work with documents, it helps to have *some* idea of the structure of a document. Examples of such tasks:

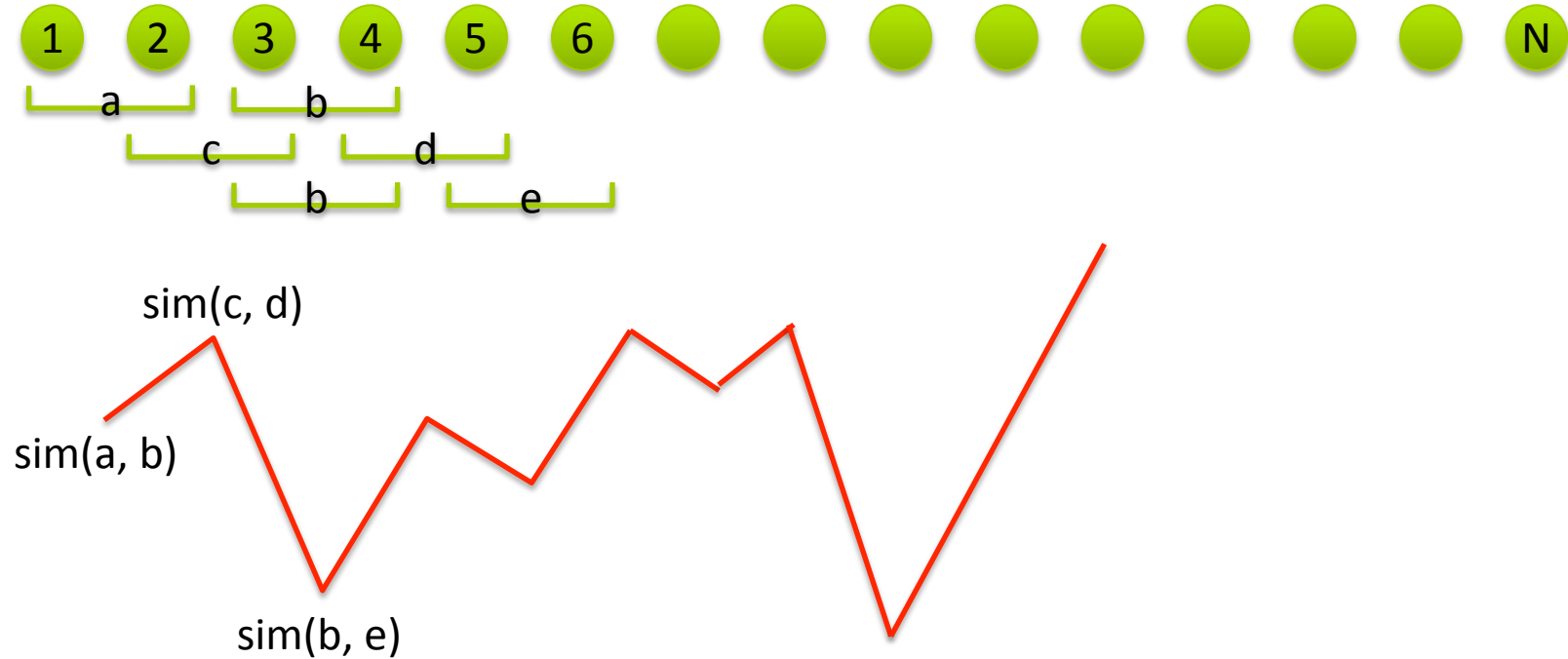
- text summarization,
- co-reference resolution,
- question answering.

Text segmentation: state-of-the-art

- Text segmentation has been around for two decades, but somehow has not become a hot topic.
- The basic idea is compellingly simple: when topic changes, so does the vocabulary.

Text segmentation: state-of-the-art

TextTiling (Hearst 1997)



Segmentation: state-of-the-art (cont.)

- Local models: Hidden Markov Models or Conditional Random Fields.
- **Shortcomings:** such models are easily thrown off by short digressions. That is because they see only one sentence back.

Graph-based segmentation

Minimum Cut Segmenter (Malioutov & Barzilay 2006): cut the document graph in a way which maximizes the number of connections within segments and minimizes the number of connections between segments.

Shortcomings:

- There is no intuitive interpretation of what the segment is about.
- The objective function used is not necessarily best for all document types.

Bayesian Segmentation

Eisenstein and Barzilay (2008):

- each sentence in the document is modeled as a draw from a multinomial language model associated with a segment;
- segmentation points are selected so as to maximize the probability of observing the whole sequence of sentences.

Shortcomings: unlike similarity-based models, the model cannot easily be extended to incorporate sources of information other than word repetition.

Desirable qualities for a segmenter

- Unsupervised
- Globally-informed (or at least less local)
- Extendable
- Should provide an idea of what the segment is about

Proposed Solution

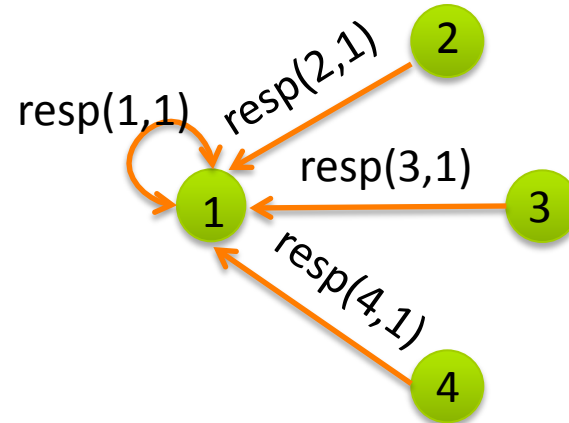
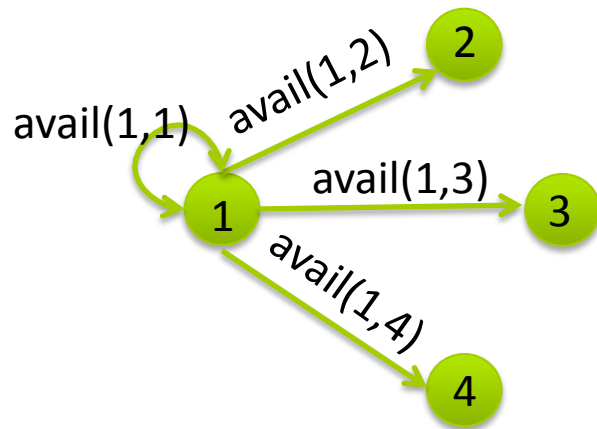
- Cast segmentation as a constrained clustering problem.
- Modify Affinity Propagation (AP) clustering algorithm to perform segmentation.

What is AP and why use it?

- It is an algorithm for similarity-based clustering.
- Objective function: maximize net similarity between all data points and their respective cluster centers.
- Inputs:
 - a matrix of **pairwise similarities** between points,
 - for each data point, a priori **preference** to be selected as a cluster center.
- Outputs: cluster assignments and cluster centers (exemplars).
- Complexity: at most $O(N^2)$ memory and time.

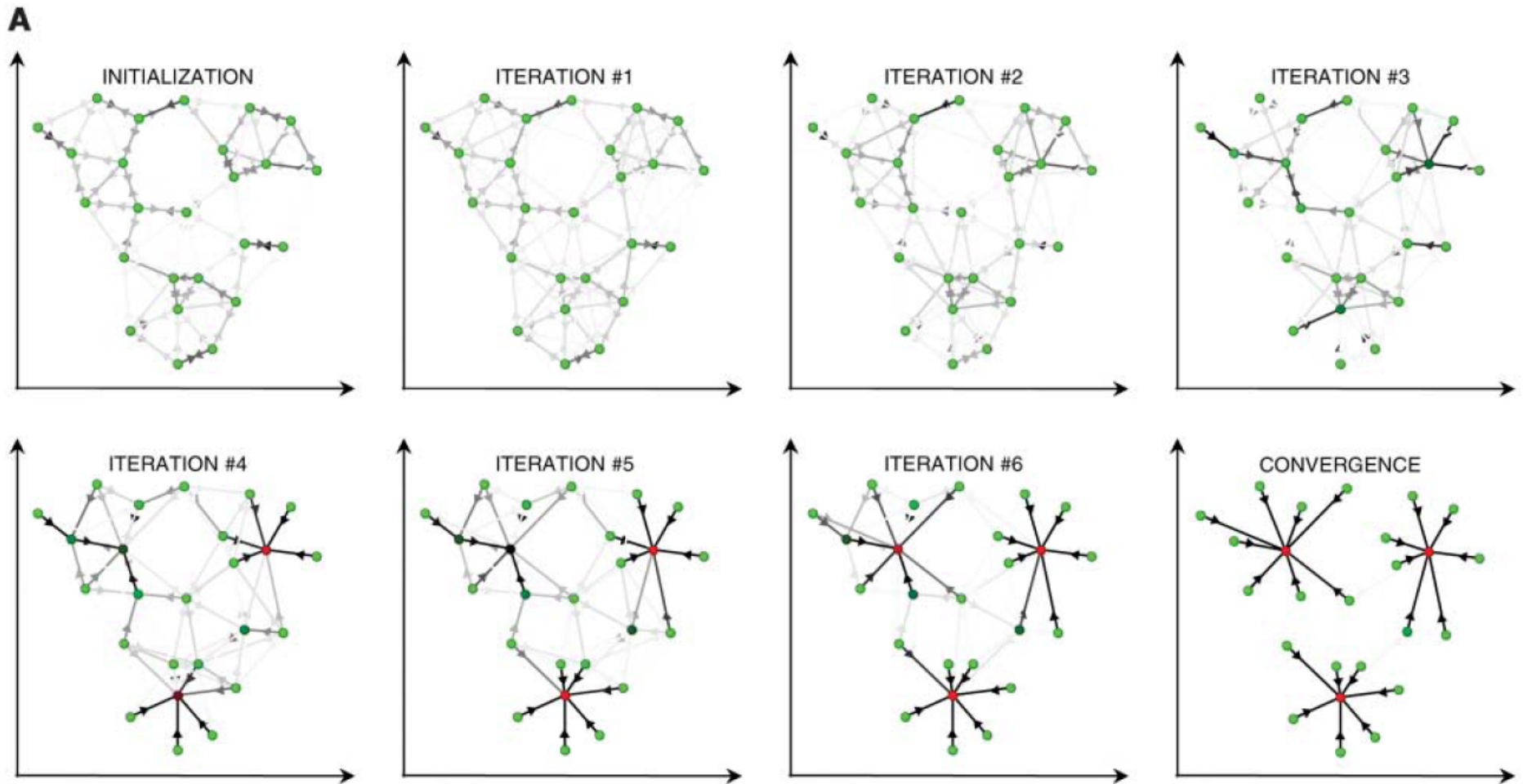
Affinity Propagation

Availability: how likely is it that the sender is the exemplar for the receiver, given evidence from all other data points?



Responsibility: how likely is it that the receiver is the exemplar for the sender, given evidence from all other potential exemplars?

Affinity Propagation: Example



Affinity Propagation: pseudocode

Input: a set of pairwise similarities $\{s(i,k)\}$ and a set of self-similarities $\{s(k,k)\}$ indicating *a priori* belief how likely a point is to be an exemplar

Initialization: set availabilities to 0

Repeat: send responsibility and availability messages until convergence

$$a(k,k) \leftarrow \sum_{i \text{ s.t. } i \neq k} \max\{0, r(i',k)\}$$

$$a(i,k) \leftarrow \min\left\{0, r(k,k) + \sum_{i' \text{ s.t. } i' \notin \{i,k\}} \max\{0, r(i',k)\}\right\}$$

$$r(i,k) \leftarrow s(i,k) - \max_{k' \text{ s.t. } k' \neq k} \{a(i,k') + s(i,k')\}$$

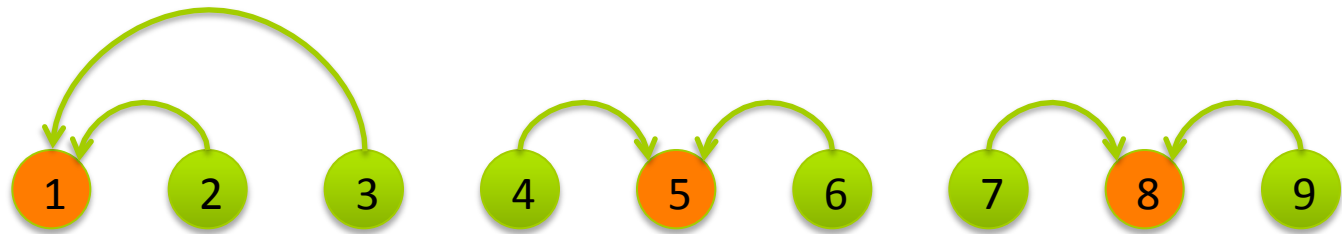
Output: cluster assignments

Affinity Propagation for Segmentation

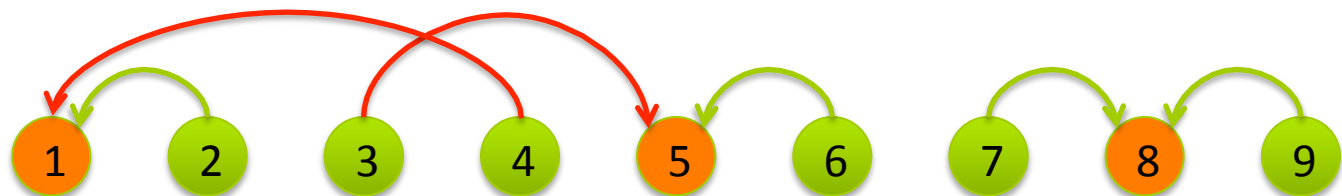
- Adjacency constraint:

- $segment_center(i + 1) \geq segment_center(i)$

- Good:*



- Bad:*



- So, what we essentially need to do is change the equations for availability message.

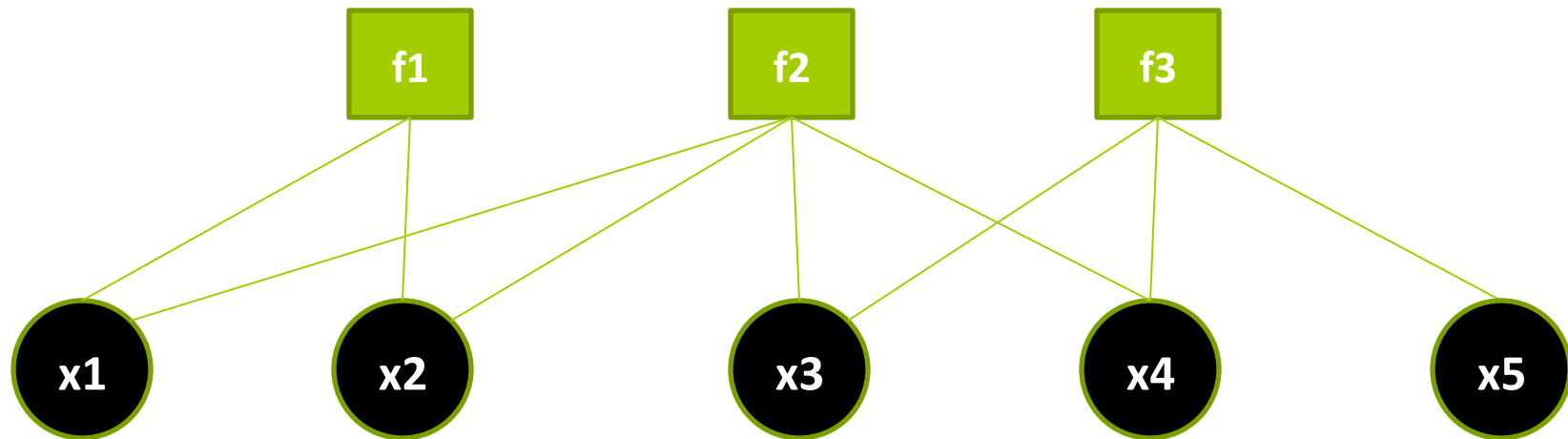
Factor graphs and the max-sum algorithm



- Useful when we need to maximize the value of a global function which can be approximated by a sum of local functions.
- Find values that maximize functions of the form:

$$F(x) = \sum_s f(x_s)$$

Factor graphs

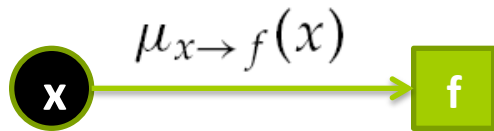
$$F(x_1, x_2, x_3, x_4, x_5) = f_1(x_1, x_2) + f_2(x_1, x_2, x_3, x_4) + f_3(x_3, x_4, x_5)$$



It is a bi-partite graph with two types of nodes: variable nodes () and function nodes ().

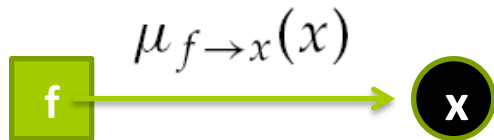
The max-sum algorithm

- To find the maximizing configuration of the global function, all nodes send messages.
- From variable node to function node:



$$\mu_{x \rightarrow f}(x) = \sum_{\{l | f_l \in \text{ne}(x) \setminus f\}} \mu_{f_l \rightarrow x}(x)$$

- From function node to variable node:



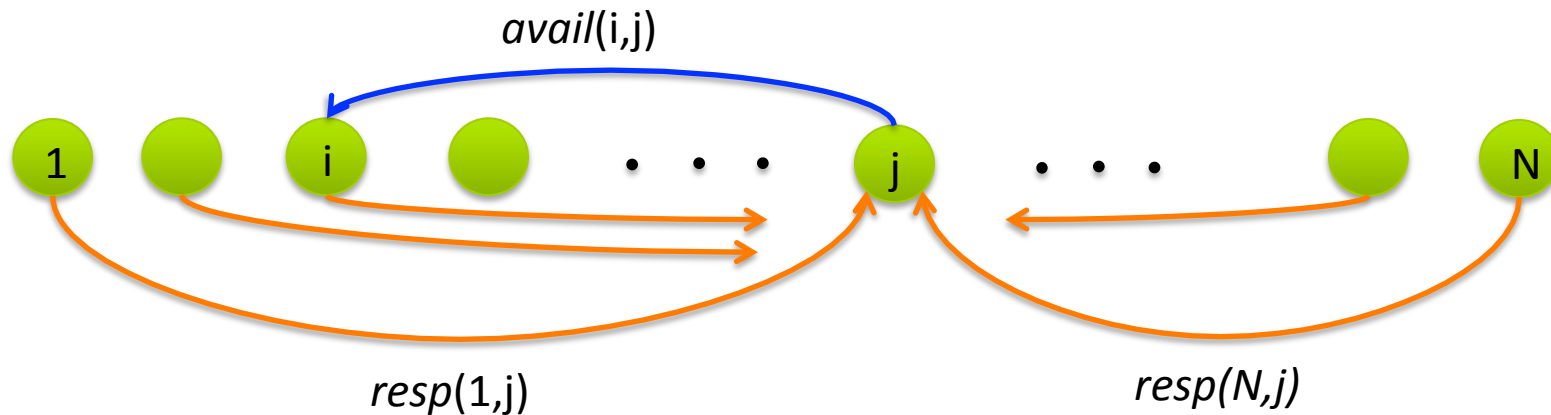
$$\mu_{f \rightarrow x}(x) = \max_{x_1, \dots, x_M} \left[f(x, x_1, \dots, x_m) + \sum_{\{m | x_m \in \text{ne}(f) \setminus x\}} \mu_{x_m \rightarrow f}(x_m) \right]$$

New availability messages: a short version

- Availability messages capture the difference in the likelihood of two hypotheses.
 - **H1:** Sentence j (the sender) is the segment center for sentence i (the receiver);
 - **H2:** Sentence j (the sender) is NOT the segment center for sentence i (the receiver).
- To compute the availability message, we very simply compute the likelihood of H1 and of H2, and take the difference.

Re-deriving availability messages (1)

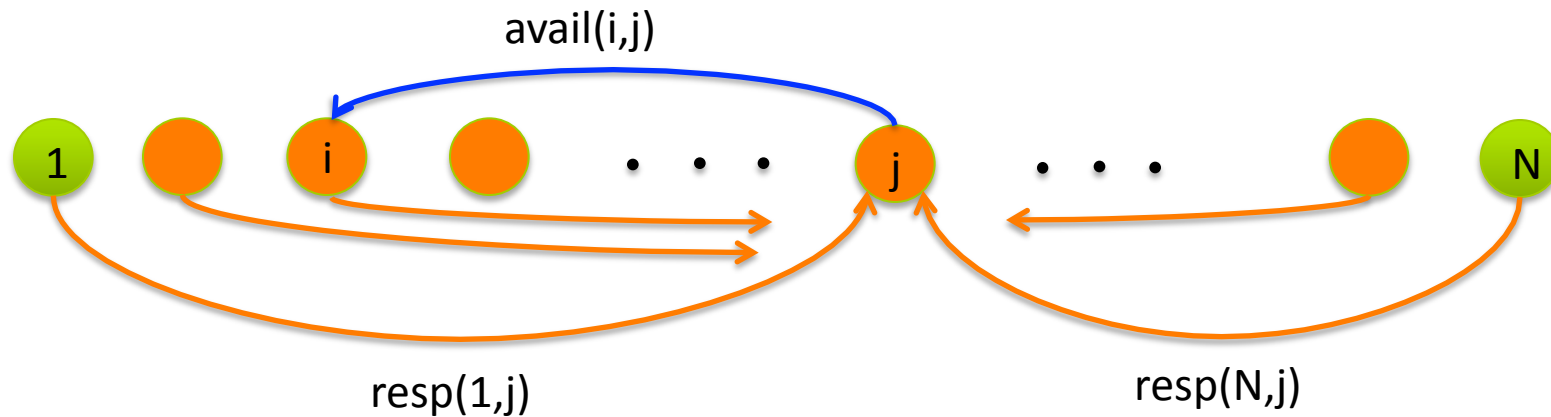
H1: sentence j is the segment center for sentence i .



What are possible valid configurations for H1?

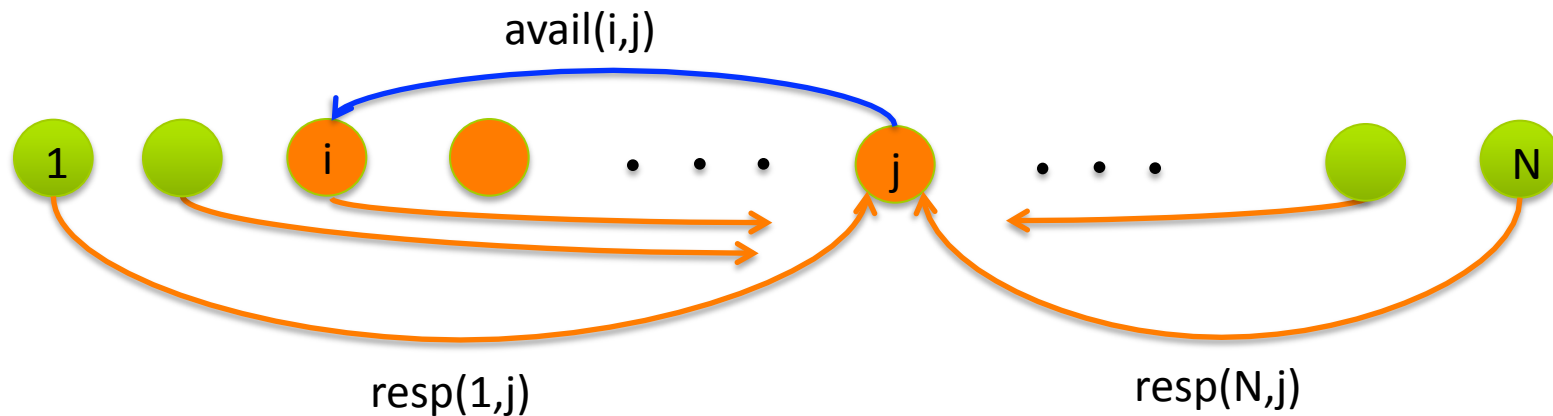
Re-deriving availability messages (2)

Our segment can look like this:



Re-deriving availability messages (3)

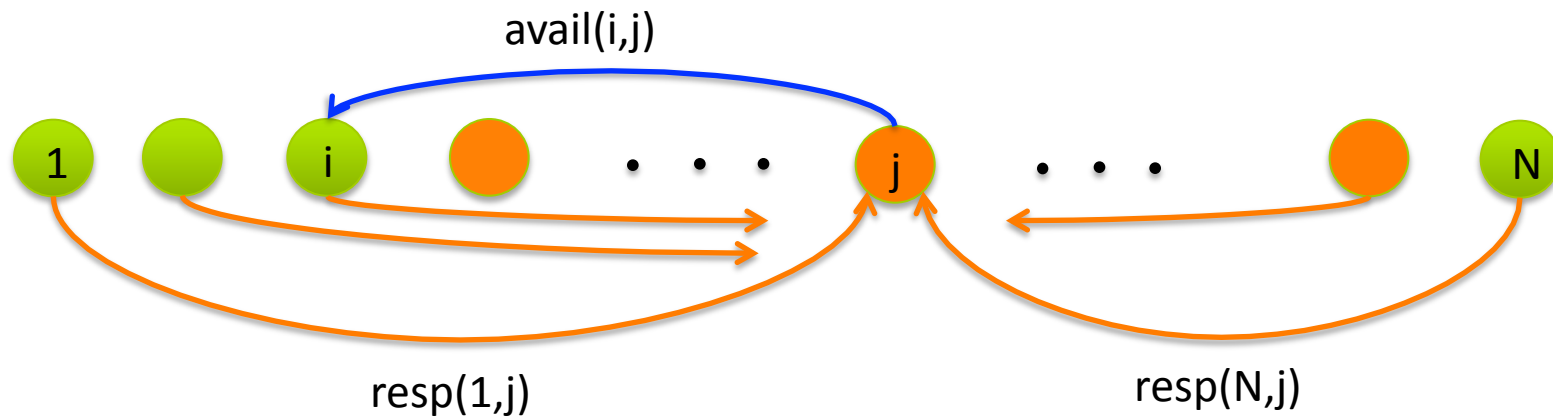
Or like this:



Using responsibility messages and the Max-Sum algorithm for factor graphs from the previous iteration, we will find the most likely configurations.

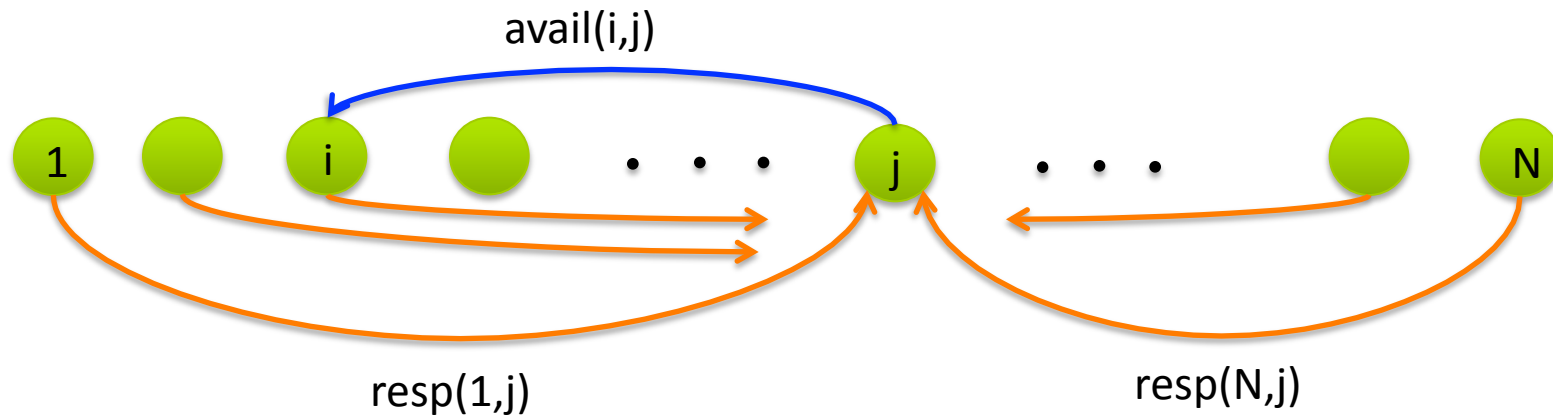
Re-deriving availability messages (4)

H2: sentence j is NOT the segment center for sentence i .



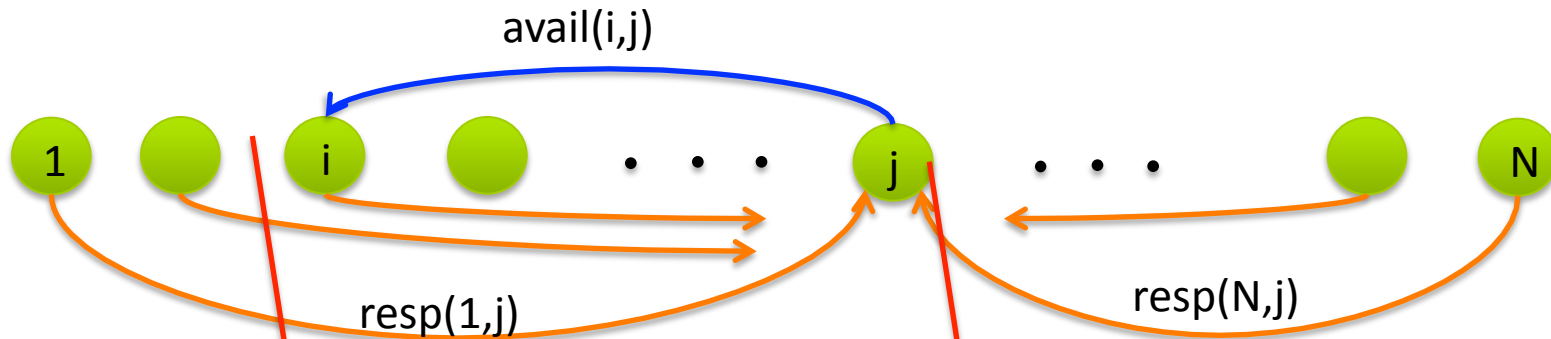
Re-deriving availability messages (5)

H2: sentence j is NOT the segment center for sentence i .



Re-deriving availability messages (6)

We will compare the likelihoods of valid configurations for H1 and H2 and get the new availability message (for $i < j$):



$$\max[$$

$$\max[\max_{f=1}^{i-1} \sum_{d=f}^{i-1} \rho_{dj}, 0] + \sum_{s=i+1}^j \rho_{sj} + \max[\max_{k=j+1}^N \sum_{l=k}^N \rho_{lj}, 0],$$

$$\max[\max_{f=1}^{i-1} \sum_{d=f}^{i-1} \rho_{dj}, 0] + \min(\min_{f=i+1}^{j-1} \sum_{l=f}^{j-1} \rho_{ej}, \sum_{v=i+1}^{j-1} \rho_{vj})$$

$$]$$

Affinity Propagation for Segmentation

2: **initialization**: set all availabilities to 0: $\forall i, j : \alpha_{ij} = 0$

3: **repeat**

4: iteratively update responsibilities and availabilities

5:

$$\forall i, j : \rho_{ij} = s(i, j) - \max_{k \neq j} (s(i, k) - \alpha_{ik})$$

6:

$$\forall i, j : \alpha_{ij} \begin{cases} = \max[\max_{f=1}^{j-1} (\sum_{d=f}^{j-1} \rho_{dj}), 0] + \max[\max_{k=j+1}^N (\sum_{l=j+1}^k \rho_{lj}), 0] & \text{if } i = j \\ = \min[\max[\max_{f=1}^{i-1} \sum_{d=f}^{i-1} \rho_{dj}, 0] + \sum_{s=i+1}^j \rho_{sj} + \max[\max_{k=j+1}^N \sum_{l=k}^N \rho_{lj}, 0], \\ \max[\max_{f=1}^{i-1} \sum_{d=f}^{i-1} \rho_{dj}, 0] + \min(\min_{f=i+1}^{j-1} \sum_{l=f}^{j-1} \rho_{ej}, \sum_{v=i+1}^{j-1} \rho_{vj})] & \text{if } i < j \\ = \min[\max[\max_{f=1}^{j-1} \sum_{d=f}^{j-1} \rho_{dj}, 0] + \sum_{s=j}^{i-1} \rho_{sj} + \max[\max_{k=i+1}^N \sum_{l=k}^N \rho_{lj}, 0], \\ \max[\max_{k=i+1}^N \sum_{l=i+1}^k \rho_{lj}, 0] + \min(\min_{k=j+1}^{i-1} \sum_{m=k+1}^{i-1} \rho_{mj}, \sum_{s=j+1}^{i-1} \rho_{sj})] & \text{if } i > j \end{cases}$$

7: **until** until convergence

8: compute the final configuration of variables: $\forall i, j$ j is the exemplar for i iff $\rho_{ij} + \alpha_{ij} > 0$

9: **output**: exemplar assignments

Complexity

- In practice, we have some idea of average and maximum segment length, so we slide a window through document and the similarity matrix is sparse.
- Memory: $O(MN)$. N is the number of sentences, M is the window size.
- Time: in each iteration we need to send $O(MN)$ messages.
 - The cost of computing each message is negligible because we do not need to compute all of it each time.

Does it work?

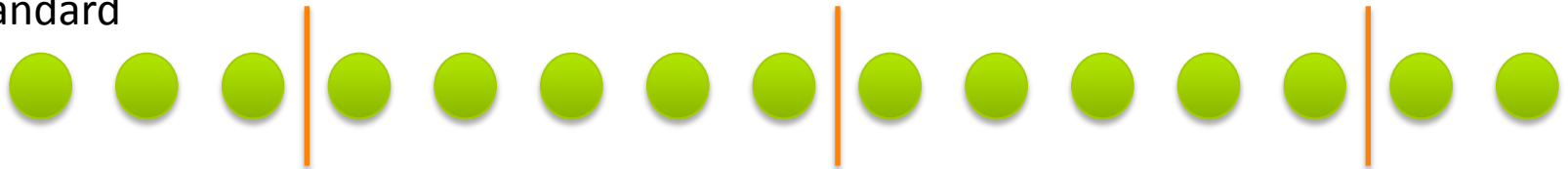
- Baselines: two state-of-the art segmenters.
 - Minimum cut segmenter (Malioutov and Barzilay 2006)
 - Bayesian segmenter (Eisenstein and Barzilay 2008)
- Datasets:
 - AI lecture transcripts (20 for testing + 3 for development)
 - Chapters from medical textbooks: finding sections (221 + 5)
 - Novels: finding chapter breaks in novels (82 + 3)
- Similarity metric: cosine similarity (stop words removed, using tf.idf weighing and smoothing).

Evaluation Metric

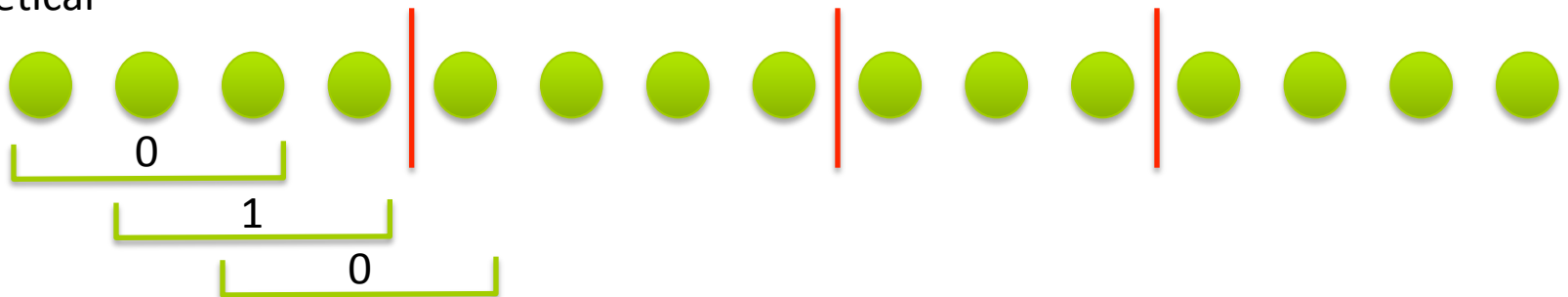
WindowDiff (Pevzner and Hearst 2002)

$$WindowDiff(ref, hyp) = \frac{1}{N - k} \sum_{i=1}^{N-k} (|num_ref_breaks - num_hyp_breaks|)$$

Gold standard



Hypothetical



Results

	Lectures (19)	Medical books (221)	Novels (82)
MinCutSeg	0.437	0.382	0.381
BayesSeg	0.443	0.353	0.377
APS	0.404	0.371	0.350

Contributions and shortcomings

- Contributions
 - A new segmentation algorithm which finds segment breaks as well as segment centers.
 - Performs quite well, especially on the data where it is feasible to expect descriptive segment centers.
 - The code is available.
- Shortcomings
 - Performance is OK, but not as good as one would wish. It may be due to unsuitable similarity metric (easy to correct) or not the best objective function (hard to correct).

Future work

- Smarter similarity metric.
- Hierarchical segmentation.
- Evaluate using a more discriminative metric.

Thank you!

References

- Delbert Dueck. 2009. *Affinity Propagation: Clustering Data by Passing Messages* . University of Toronto Ph.D. thesis.
- Jacob Eisenstein and Regina Barzilay. [*"Bayesian Unsupervised Topic Segmentation"*](#), EMNLP 2008.
- Brendan Frey. Affinity Propagation Tutorial. (slides)
- Brendan J. Frey and Delbert Dueck . 2007. *Clustering by Passing Messages Between Data Points*. Science 315, 972–976.
- Inmar E. Givoni and Brendan J. Frey. 2009. *A Binary Variable Model for Affinity Propagation*. Neural Computation, Vol 21, issue 6, pp 1589-1600.
- Marti Hearst. TextTiling: Segmenting Text into Multi-Paragraph Subtopic Passages, *Computational Linguistics* , 23 (1), pp. 33-64, March 1997
- Anna Kazantseva and Stan Szpakowicz. 2011. Linear Text Segmentation Using Affinity Propagation. EMNLP 2011, Edinburgh, to appear.
- Frank R Kschischang, Brendan J Frey and Hans-A Loeliger. 2001. *Factor graphs and the sum-product algorithm* . IEEE Transactions on Information Theory, Vol 47, No 2, pp 498-519, February 2001.
- Igor Malioutov, Regina Barzilay. 2006. [*Minimum Cut Model for Spoken Lecture Segmentation*](#). COLING-ACL 2006, pp. 9-16.
- Pevzner, L., and Hearst, M., A Critique and Improvement of an Evaluation Metric for Text Segmentation, *Computational Linguistics*, 28 (1), March 2002, pp. 19-36.