L113 Word Meaning and Discourse Understanding

Session 6: Coherence in Text

Simone Teufel

Natural Language and Information Processing (NLIP) Group
UNIVERSITY OF CAMBRIDGE
Simone.Teufel@cl.cam.ac.uk

2012/2013

Coherence in Text

Coherence:

- is a property of well-written texts;
- makes them easier to read and understand;
- ensures that sentences are meaningfully related;
- and that the reader can work out what the linguistic expressions mean.

A coherent text is

- thematically organized;
- temporally organized;
- rather than a random concatenation of sentences.

Automatic models of coherence

Two uses for models of coherence:

- Discourse segmentation: Detecting breaks in text where coherence is relatively low
  - Useful in text summarisation, information retrieval, hypertext display. . .
- Judgement of textual coherence:
  - Can help rank the quality of alternative potential output texts/summaries in NLG/summarisation
  - Automatic grading of student essays

Topic Segmentation: The task

- Segment text into non-hierarchical, non-overlapping zones which contain the same subtopic
- Equivalent definition: Detect subtopic shifts (changes of subtopic)
- Can't we simply use paragraph or section boundaries?
  - Stark (1988) found not all paragraph boundaries reflect topic shifts
  - Paragraph conventions genre-dependent
  - Sections often too large

Factors for Detecting Topic Shifts

Linguistic factors:

- Adverbial clauses, prosodic markers (Brown and Yule)
- Cue phrases (Passonneau and Litman, Beeferman et al., Manning), e.g. *oh, well, so, however, ...*
- Pronoun resolution
- Tense and aspect (Webber)

Lexical (co-occurrence) patterns:

- Word overlap or lexical chain overlap between sentences (Skorochod'ko 1979; Hearst 1994, 1997)
- New vocabulary terms (Youmans, 1991)
- Maximise density in dotplots (Reynar, 1994, 1998; Choi, 2000)
- Probabilistic model (Beeferman, Berger, Lafferty, 1999)

Star Gazer Text Structure

Para	Subtopics
1-3	Intro – the search for life in space
4-5	The moon's chemical composition
6-8	How early earth-moon proximity shaped the moon
9-12	How the moon helped life evolve on earth
13	Improbability of the earth-moon system
14-16	Binary/trinary star systems make life unlikely
17-18	The low probability of non-binary/trinary systems
19-20	Properties of earth's sun that facilitate life
21	Summary

Term repetition signals topic shift/cohesion

```
---------------------------------------------------------------------------------------------------------
Sentence:    05   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80   85   90   95
---------------------------------------------------------------------------------------------------------
    form     1     111 1   1                               1 1    1    1       1     1     1    1
scientist                   11            1   1               1          1      1  1
    space11    1    1                                                              1
     star   1              1                                       11 22  111112   1 1  1   11 1111    1
   binary                                                           11  1           1                   1
  trinary                                                            1   1          1                   1
astronomr1                 1                                        1  1           1   1     1  1
    orbit  1                     1                                      12    1 1
     pull                          2    1 1                                 1  1
   planet  1    1        11              1          1                   21  11111                   1   1
   galaxy   1                                                     1               1  11      1          1
    lunar           1  1       1       1
     life1  1  1                               1   11 1  11  1         1               1 1     1 111 1 1
     moon       13  1111   1 1 22 21  21 21   21          11 1
     move                                        1   1   1
continent                                        2 1 1 2 1
shoreline                                              12
     time                     1                 1  1  1     1                                          1
    water                                   11            1
      say                                  1 1        1        11               1
  species                                          1  1  1
---------------------------------------------------------------------------------------------------------
Sentence:    05   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80   85   90   95
---------------------------------------------------------------------------------------------------------
```

Example Text: “The history of algebra”

1 Algebra provides a generalization of arithmetic by using symbols,
2 usually letters, to represent numbers. For example, it is obviously
...
28 In about 1100, the Persian mathematician Omar Khayyam wrote a treatise...
...

51 Boolean algebra is the algebra of sets and of logic. It uses symbols
52 to represent logical statements instead of words. Boolean algebra was
53 formulated by the English mathematician George Boole in 1847. Logic
54 had previously been largely the province of philosophers, but in his
55 book, The Mathematical Analysis of Logic, Boole reduced the whole of
56 classical, Aristotelian logic to a set of algebraic equations. Boole's
57 original notation is no longer used, and modern Boolean algebra now
58 uses the symbols of either set theory, or propositional calculus.
59 Boolean algebra is an uninterpreted system - it consists of rules for
60 manipulating symbols, but does not specify how the symbols should be
61 interpreted. The symbols can be taken to represent sets and their
62 relationships, in which case we obtain a Boolean algebra of
63 sets. Alternatively, the symbols can be interpreted in terms of
64 logical propositions, or statements, their connectives, and their
65 truth values. This means that Boolean algebra has exactly the same
66 structure as propositional calculus.

Example Text: “The history of algebra”

67 The most important application of Boolean algebra is in digital
68 computing. Computer chips are made up of transistors arranged in logic
69 gates. Each gate performs a simple logical operation. For example, an
70 AND gate produces a high voltage electrical pulse at the output r if
71 and only if a high voltage pulse is received at both inputs p, q. The
72 computer processes the logical propositions in its program by
73 processing electrical pulses - in the case of the AND gate, the
74 proposition represented is p q r. A high pulse is equivalent to a
75 truth value of "true" or binary digit 1, while a low pulse is
76 equivalent to a truth value of "false", or binary digit 0. The design
77 of a particular circuit or microchip is based on a set of logical
78 statements. These statements can be translated into the symbols of
79 Boolean algebra. The algebraic statements can then be simplified
80 according to the rules of the algebra, and translated into a simpler
81 circuit design.
...

82 An algebraic equation shows the relationship between two or more
83 variables. The equation below states that the area (a) of a circle
...

Topic segments by word distribution

```
--------------------------------------------------------------------------------------------------------
Line   :         05   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80   85   90
--------------------------------------------------------------------------------------------------------

          the        1 211   11  21 21 111 111 11  1  121  11 111 112  1 11 1 1 1  1113  1 112 12111 1
      algebra    1   1        1       1    1 1 1        1  1  1       1 1  1    1           11
      century      1 1 1      1      1      1 1         1  1
   arithmetic 1  1     1    1              1
 mathematical      1   1     1      1    1      11                1 1
       number  1  1         1           1
       symbol 1           1                            1        1       1211               1
        Boole                                                   111  11 1  1  1 1           1
        logic                                                   111 11       1   11  1    1
          set                                                           1  1 1
     computer                                                                   2   1
         gate                                                                     11   1
        pulse                                                                      11 111
     variable                                                                                  1  1
     equation                   11     1   1      11  1      1       1                         11 11 1
--------------------------------------------------------------------------------------------------------
Line   :         05   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80   85   90
--------------------------------------------------------------------------------------------------------
```

- “the” non-distinctive, but “algebra” also non-distinctive!
- Segment from 51 to 66 about “Boole” and “logic”
- Segment from 67 to 81 about “gates”, “computers” and “Boole”
- Initial segments more general (“century”, “mathematics”)

TextTiling: The algorithm

Preprocessing: separate texts into pseudo-sentences *w* tokens long

- Score cohesion b/w pseudo-sentences
- Compare several metrics:
  - Word overlap
  - Vocabulary introduction
  - Vector space distance (not in CL article)
- Find local minima in plot of neighbouring pseudo-sentences scores (“depth scoring”)
- Project boundary onto nearest paragraph boundary

TextTiling Algorithm: Shifting window

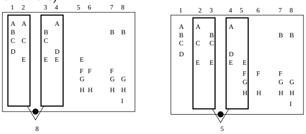
- Pseudo-sentences consist of w tokens (including stop words). Typical w=20
- Blocks consist of k pseudo-sentences (blocks should approx. paragraphs; often $k = 6$–10, but $k = 2$ in example)
- Sliding window of 2 blocks
- Compute and plot one or more scores at break between blocks
  - $2kw$ tokens are compared at a time
- Blocks shift one pseudo-sentence at a time
  - You get as many data points as there are pseudo-sentences
  - Each pseudo-sentence occurs in $2k$ calculations
  - Create two vectors from each block; use non-stoplist-tokens (stemmed)

TextTiling: Minimal block similarity signals boundary

Score: non-normalized inner product of frequencies $w_{j,b}$ of terms t_j in left and right term vector $b_1 = t_{i-k}, \dots t_i$ and $b_2 = t_{i+1}, \dots t_{i+k+1}$

$$score(i) = \sum_{j=0}^{|T|} w_{j,b_1} w_{j,b_2}$$

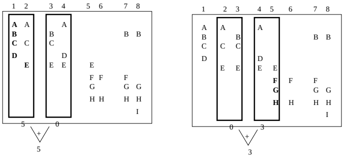
(T: set of all tokens)



TextTiling: Max. in new vocab. items signals boundary

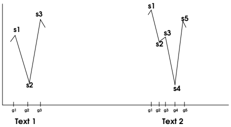
- Score is the sum of new words in left and right block:

$$score(i) = NumNewTerms(b_1) + NumNewTerms(b_2)$$



TextTiling: Relative Depth

- Use relative, not absolute, depth score: $Depth(g_i) = |s_{i-1} - s_i| + |s_{i+1} - s_i|$ (with s_{i-1} and s_{i+1} surrounding local maxima; cf. Text 1)

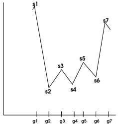


Cohesion is relative

- Introductions have many topic shifts → want only strong shifts
- Mid-portion with only minor topic shifts → want also weaker shifts
- Additional **low pass filter** (Text 2): $\frac{s_{i-1}+s_i+s_{i+1}}{3}$ (because $s_1 - s_2$ should contribute to score at g_4)

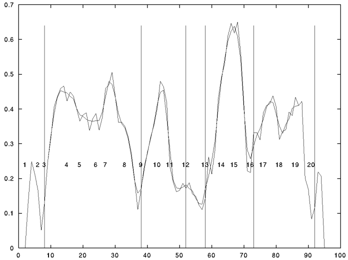
TextTiling: Boundary determination

- Sort depth scores, determine boundaries:
  - Boundary if *Depth* $> \mu - \sigma$ (low cutoff; liberal)
  - Boundary only if *Depth* $> \mu - \frac{\sigma}{2}$ (high cutoff; high P, low R)
- For each gap, assign closest paragraph boundary
- Do not assign close adjacent segment boundaries; 3 pseudosentences must intervene, to avoid sequence of small segments:



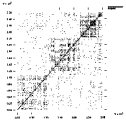
Text 3

TextTiling: Output of depth scorer on "Stargazer" text



Alternative Segmentation Algorithms: Reynar (1998)

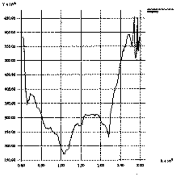
Use Church's (1993) dotplot method (e.g. on the following three concatenated WSJ articles):



- If a word appears both in word positions x and y, then plot (x,x), (x,y), (y,y), (y,x) → the diagonal is always dark
- Dark squares along diagonal indicate regions with many shared words

Alternative Segmentation Algorithms: Reynar (1998)

- Maximise density of regions within squares along the diagonal:
- Density $D = \frac{N}{x^2}$
- x: length of a square (in words); N: number of points in square
- Use divisive clustering to insert boundaries



Hierarchical clustering: divisive (TopDown) clustering

Given: a set $X = x_1, ...x_n$ of objects;
Given: a function $coh : \mathcal{P} \rightarrow \mathcal{R}$
Given: a function $split : \mathcal{P}(X) \times \mathcal{P}(X)$

$C := \{X\}(= \{c_1\})$
$j := 1$
while $\exists c_i \in C$ s.t. $|c_i| > 1$ **do**
  $c_u := \arg\min_{c_v \in C} coh(c_v)$
  $(c_{j+1}, c_{j+2}) = split(c_u)$
  $C := C \setminus \{c_u\} \cup \{c_{j+1}, c_{j+2}\}$
  $j := j + 2$
end

This is a greedy algorithm!

A probabilistic model for topic segmentation

Linear combination of two models:

- A short-range model: trigram language model $P_{tri}(w_2|w_0 w_1)$
- A long-range model:
  - Determine trigger pairs (s,t) with high Mutual Information (59,936 pairs found)
  - If s has occurrend within the past 500 words, then the probability of t is boosted by a certain factor ($e^{\lambda(s,t)}$, which is estimated by a method called iterative scaling).

Examples of trigger pair boosting (long-range LM)

s	t	$e^{\lambda_{s,t}}$
residues	carcinogens	2.3
Charleston	shipyards	4.0
microscopic	cuticle	4.1
defense	defense	8.4
tax	tax	10.5
Kurds	Ankara	14.8
Vladimir	Gennady	19.6
Steve	Steve	20.7
education	education	22.2
insurance	insurance	23.0
Pulitzer	prizewinning	23.6
Yeltsin	Yeltsin	23.7
sauce	teaspoon	27.1
flower	petals	32.2
picket	scab	103.1

Lexical Chains, Revisited

Consider these two summaries.

Summary A

Britain said he did not have diplomatic immunity. The Spanish authorities contend that Pinochet may have committed crimes against Spanish citizens in Chile. Baltasar Garzon filed a request on Wednesday. Chile said, President Fidel Castro said Sunday he disagreed with the arrest in London.

Summary B

Former Chilean dictator Augusto Pinochet, was arrested in London on 14 October 1998. Pinochet, 82, was recovering from surgery. The arrest was in response to an extradition warrant served by a Spanish judge. Pinochet was charged with murdering thousands, including many Spaniards. Pinochet is awaiting a hearing, his fate in the balance. American scholars applauded the arrest.

Two Summaries

Summary A

Britain said he did not have diplomatic immunity. The Spanish authorities contend that Pinochet may have committed crimes against Spanish citizens in Chile. Baltasar Garzon filed a request on Wednesday. Chile said, President Fidel Castro said Sunday he disagreed with the arrest in London.

Summary B

Former Chilean dictator Augusto Pinochet, was arrested in London on 14 October 1998. Pinochet, 82, was recovering from surgery. The arrest was in response to an extradition warrant served by a Spanish judge. Pinochet was charged with murdering thousands, including many Spaniards. Pinochet is awaiting a hearing, his fate in the balance. American scholars applauded the arrest.

Lexical Chains

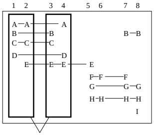
	LC1	LC2	LC3	LC4
S1	Pinochet	arrested		
S2	Pinochet			
S3		arrest, extradition	warrant, served	Spanish
S4	Pinochet		charged	Spaniards
S5	Pinochet		hearing	
S6		arrest		

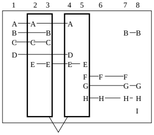
Reminder: Lexical Chains

- Sequence of related words in text, spanning short (adjacent sentences) or longer distances (entire text)
- Originally due to Halliday and Hasan (1976)
- First CL application in Morris and Hearst (1991).
  - Allowed lexical relations: identity, synonymy, hyponymy, siblings
- **Claim (here):** they capture (some of) the cohesive structure of the text
- (This is on top of the old claim that they provide the right context for WSD – which we know from session 2)

Texttiling with Lexical Chains

- Hearst (1997) also considers lexical chains as a scoring method in the sliding window method (TextTiling).
- High number of lexical chains spannig over a window gap should indicate high coherence.
- However, results are disappointing.

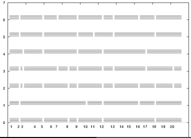




Evaluation of Topic Segmentation

Defining a gold standard

- Hearst (1997): "group opinion" amongst human annotators (3 out of 7)
- 12 magazine articles
- Humans find boundaries at 39% of "allowed" places (paragraph boundaries only)
- Baseline: randomly assign 39% of boundaries



Evaluation: precision and recall

- Measure precision and recall, in comparison to group opinion
- Precision tells us about false positives, recall about false negatives

	Tiling (VocabIntro)	Tiling (Lexical)
High cutoff	P=.58, R=.64	P=.71, R=.59
Low cutoff	P=.52, R=.78	P=.66, R=.75
Judges	P=.83, R=.71	
Baseline	P=.50, R=.51	

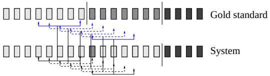
Evaluation by detecting document boundaries

- Create pseudo document by gluing unrelated documents together; measure how well the original document boundaries are found.
- This evaluation method violates a major assumption of the task:
  - It assumes article boundaries are by definition stronger shifts than within-article subtopic shifts
  - Algorithms is penalized for finding within-article subtopic shifts
- Evaluation of TextTiling on 44 WSJ articles glued together:

No. bound.	10	20	30	40	**43**	50	60	70
P	.80	.80	.73	.68	**.67**	.62	.60	.59
R	.19	.37	..51	.63	**.67**	.72	.83	.95

Evaluation Metrics for Topic Segmentation

- But: precision and recall are insensitive to near misses
- Improvement: P_k measure (Beeferman et al. 1999)
  - Compute penalties via two probes (moving windows)
  - One probe checks gold standard, the other checks system text
  - Probe width k: half the avg. segment size (here k=4)
  - If both probes have their ends in the same segment or if both probes have their ends in different ones, score 1 point
  - Divide by number of measurements taken; P_k is in [0..1]



Win$_{diff}$ – Improvement over P_k

- *Win*$_{diff}$ (improvement over P_k; Prevner and Hearst 2002): the ends of the two probes must cover the same number of boundaries (0 or more) to score a point.

Entity-based Coherence: Barzilay and Lapata 2005

- Coherence as a model of sequences of entity types in text
- Assume we know whether two linguistic expression **co-refer**, i.e., talk about the same entity (more about this in session 7)
- Observations from discourse theory:
  - The way entities are introduced and discussed influences coherence (Grosz et al 1995).
  - Salience of entities is related to where in the sentence they occur (Sidner, 1992).
  - Frequency, syntactic position, pronominalisation are relevant coherence properties.

The Entity Grid

1 Former Chilean dictator Augusto Pinochet, was arrested in London on 14 October 1998.
2 Pinochet, 82, was recovering from surgery.
3 The arrest was in response to an extradition warrant served by a Spanish judge.
4 Pinochet was charged with murdering thousands, including many Spaniards.
5 He is awaiting a hearing, his fate in the balance.
6 American scholars applauded the arrest.

The Entity Grid

1 Former Chilean dictator Augusto Pinochet$_S$, was arrested in London$_X$ on 14 October$_X$ 1998.
2 Pinochet$_S$, 82, was recovering from surgery$_X$.
3 The arrest$_S$ was in response$_X$ to an extradition warrant$_X$ served by a Spanish judge$_S$.
4 Pinochet$_O$ was charged with murdering thousands$_O$, including many Spaniards$_O$.
5 Pinochet$_S$ is awaiting a hearing$_O$, his fate$_X$ in the balance$_X$.
6 American scholars$_S$ applauded the arrest$_O$.

The Entity Grid

1 Pinochet$_S$ London$_X$ October$_X$
2 Pinochet$_S$ surgery$_X$
3 arrest$_S$ response$_X$ warrant$_X$ judge$_O$
4 Pinochet$_O$ thousands$_O$ Spaniards$_O$
5 Pinochet$_S$ hearing$_O$ Pinochet$_X$ fate$_X$ balance$_X$
6 scholars$_S$ arrest$_O$

The Entity Grid

	Pinochet	London	October	Surgery	Arrest	Extradition	Warrant	Judge	Thousands	Spaniards	Hearing	Fate	Balance	Scholars
1	S	X	X	–	–	–	–	–	–	–	–	–	–	–
2	S	–	–	X	–	–	–	–	–	–	–	–	–	–
3	–	–	–	–	S	X	X	O	–	–	–	–	–	–
4	O	–	–	–	–	–	–	–	O	O	–	–	–	–
5	S	–	–	–	–	–	–	–	–	–	O	X	X	–
6	–	–	–	–	O	–	–	–	–	–	–	–	–	S

Columns: entities; lines: sentences

Entity Transitions

Definition

A local entity transition is a sequence $\{\mathbf{S}, \mathbf{O}, \mathbf{X}, -\}^n$ that represents entity occurrences and their syntactic roles in n adjacent sentences.

Feature Vector Notation

Each grid x_{ij} for document d_i is represented by a feature vector:

$$\Phi(x_{ij}) = (p_1(x_{ij}), p_2(x_{ij}), \ldots, p_m(x_{ij}))$$

m: number of entity transitions (predefined)
$p_t(x_{ij})$: probability of transition t in grid x_{ij}

Entity Transitions

Example (transitions of length 2)

	SS	SO	SX	S–	OS	OO	OX	O–	XS	XO	XX	X–	–S	–O	–X	––
d_1	0	0	0	.03	0	0	0	.02	.07	0	0	.12	.02	.02	.05	.25
d_2	0	0	0	.02	0	.07	0	.02	0	0	.06	.04	0	0	0	.36
d_3	.02	0	0	.03	0	0	0	.06	0	0	0	.05	.03	.07	.07	.29

Linguistic Dimensions

Coreference: Talking about the same entity

- Entities are coreferent if they have (roughly) the same surface form.
- Coreference resolution systems exist (cf. session 7)

Syntax: Does syntactic knowledge matter?

- Use four categories {**S**, **O**, **X**, –}.
- Or just two {**X**, –}.

Salience: Are some entities more important than others?

- Discriminate between salient (frequent) entities and the rest.
- Collect statistics separately for each group.

Learning a Ranking Function

Training Set

Ordered pairs (x_{ij}, x_{ik}), where x_{ij} and x_{ik} represent the same document d_i, and x_{ij} is more coherent than x_{ik} (assume $j > k$).

- Source document and permutations of its sentences.
- Original order **assumed coherent.**
- Given k documents, with n permutations, obtain $k \cdot n$ pairwise rankings for training and testing.
- Two corpora, Earthquakes and Accidents, 100 texts each.

Sentence 1 Sentence 2 Sentence 3 Sentence 4	>	Sentence 2 Sentence 3 Sentence 4 Sentence 1	>	Sentence 4 Sentence 3 Sentence 2 Sentence 1	>	Sentence 2 Sentence 1 Sentence 4 Sentence 3

Learning a Ranking Function

Goal

Find a parameter vector $\vec{w}$ such that:

$$\vec{w} \cdot (\Phi(x_{ij}) - \Phi(x_{ik})) > 0 \; \forall j, i, k \text{ such that } j > k$$

$\vec{w}\Phi(xij)$ is a ranking score, such that the violations of pairwise rankings in the training set are minimised.
Use Support Vector Machines (SVMs, Joachims; 2002) to solve this constraint optimization problem.

Results

Model	Earthquakes	Accidents
Coreference+Syntax+Salience+	**87.2**	**90.4**
Coreference+Syntax+Salience−	88.3	90.1
Coreference+Syntax−Salience+	86.6	88.4**
Coreference−Syntax+Salience+	83.0**	89.9
Coreference+Syntax−Salience−	86.1	89.2
Coreference−Syntax+Salience−	82.3**	88.6*
Coreference−Syntax−Salience+	83.0**	86.5**
Coreference−Syntax−Salience−	81.4**	86.0**

Evaluation metric: % correct ranks in test set.
**: sig. different from Coreference+Syntax+Salience+

Results

- Entity-based model outperforms LSA.
- Linguistically poorer models generally worse.
- Omission of coreference causes performance drop.
- Syntax and Salience have more effect on Accidents corpus.
- In summary, is robust and learns appropriate ranking function.

BUT:

- Entity grid doesn't contain lexical information.
- Doesn't contain a notion of global coherence.
- Can't model multi-paragraph text.

Literature

Topic segmentation algorithms:

- Jurafsky and Martin, chapter 21.1
- Hearst, "Multi-paragraph segmentation of expository text", ACL 1994.
- Hearst, "TextTiling: Segmenting Text into Multi-paragraph subtopic passages", Computational Linguistics, 23(1) , 1997
- Reynar, "An automatic method of finding topic boundaries", ACL 1994.
- Beeferman, Berger, Lafferty, "Statistical Models for Text Segmentation", Machine Learning, 1999

Literature, ctd.

Evaluation Issues:

- Prevner and M. Hearst: "A critique and improvement of an evaluation metric for text segmentation", Computational Linguistics, 28(1), 2002

Entity-based Coherence:

- Barzilay and Lapata, 2005. Modelling Local Coherence: An Entity-based approach. *Proceedings of ACL*.
- Barzilay and Lapata, 2008. Modeling Local Coherence: An Entity-Based Approach. *Computational Linguistics*, 34:1, 1-34.