# Lecture 10: Discourse Segmentation

Lexical Semantics and Discourse Processing
MPhil in Advanced Computer Science

Simone Teufel

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

February 23, 2011











## Reading

- Hearst, M. *Computational Linguistics*, 2007.
- Jurafsky and Martin, chapter 21.1





## 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)
- Reasons for not simply using paragraph or section boundaries:
  - Stark (1988) found not all paragraph boundaries reflect topic shifts
  - Paragraph conventions genre-dependent
  - Sections often too large

## Example

Pennicillin is a group of beta-lactam antibiotics used in the treatment of bacterial infections caused by susceptible, usually Gram-positive, organisms. The discovery of penicillin is usually attributed to Scottish scientist Sir Alexander Fleming in 1928. Fleming noticed a halo of inhibition of bacterial growth around a contaminant blue-green mold Staphylococcus plate culture. Fleming concluded that the mold was releasing a substance that was inhibiting bacterial growth and lysing the bacteria. Common adverse drug reactions associated with the use of the penicillins include: diarrhoe, nausea, rash, urticaria, and/or superinfection (including candidiasis).





## Example

Pennicillin is a group of beta-lactam antibiotics used in the treatment of bacterial infections caused by susceptible, usually Gram-positive, organisms. The discovery of penicillin is usually attributed to Scottish scientist Sir Alexander Fleming in 1928. Fleming noticed a halo of inhibition of bacterial growth around a contaminant blue-green mold Staphylococcus plate culture. Fleming concluded that the mold was releasing a substance that was inhibiting bacterial growth and lysing the bacteria. Common adverse drug reactions associated with the use of the penicillins include: diarrhoe, nausea, rash, urticaria, and/or superinfection (including candidiasis).





## Applications

- Text Summarisation
- Information Retrieval
- Hypertext display



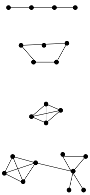


## Factors for Detecting Topic Shifts

- **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:**
  - Measure word overlap between sentences; define different topological structures (Skorochod'ko 1979)
  - New vocabulary terms (Youmans, 1991)
  - Sliding Window; word repetition (**TextTiling**; Hearst 1994, 1997)
  - Maximise density in **dotplots** (Reynar, 1994, 1998; Choi, 2000)
  - Probabilistic model (Beeferman, Berger, Lafferty, 1999)

## Text Structure Types (Skorochod'ko 1972)

Compute word overlap between sentences and look at distribution of highly connected sentences:








## 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
---------------------------------------------------------------------------------------------------------
14       form     1     111 1   1                               1 1    1    1       1     1     1    1
 8  scientist                     11            1   1               1          1      1  1
 5      space  11    1    1                                                              1
25       star     1              1                                       11 22  111112   1 1  1   11 1111
 5     binary                                                            11  1           1
 4    trinary                                                             1   1          1
 8 astronomer  1                 1                                       1  1          1   1     1  1
 7      orbit    1                     1                                    12    1 1
 6       pull                            2    1 1                                  1  1
16     planet    1    1        11              1          1                   21  11111
 7     galaxy     1                                                     1               1  11      1
 4      lunar             1  1       1       1
19       life  1  1  1                               1   11 1  11  1         1               1 1     1 111
27       moon         13  1111   1 1 22 21  21 21   21          11 1
 3       move                                          1   1   1
 7  continent                                          2 1 1 2 1
 3  shoreline                                                12
 6       time                        1                 1  1  1     1
 3      water                                     11            1
 6        say                                   1 1        1        11               1
 3    species                                         1  1  1
---------------------------------------------------------------------------------------------------------
Sentence:         05   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80   85   90
---------------------------------------------------------------------------------------------------------
```





## 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
  - Lexical chains (CL article)
  - 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

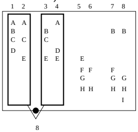
- Pseudo-sentences consist of $w$ tokens (including stop words). Typical $w$=20
- Blocks consist of $k$ pseudo-sentences (blocks should approx. paragraphs; often $k$ = 6–10, but $k = 2$ in example)
- Sliding window of 2 blocks
- Compute and plot one or more scores at break between blocks
  - $2kw$ tokens 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)

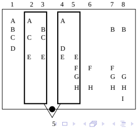
## TextTiling: Minimal block similarity signals boundary

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

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

($T$: set of all tokens)





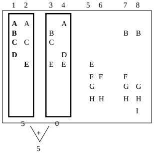


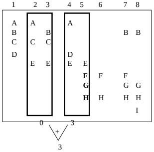


## TextTiling: Max. in new vocab. items signals boundary

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

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



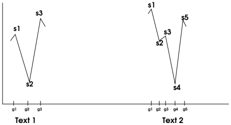






## TextTiling: Relative Depth

- 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)





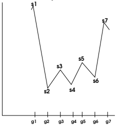



## Cohesion is relative

- 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
- Need heuristics to avoid sequence of small segments:



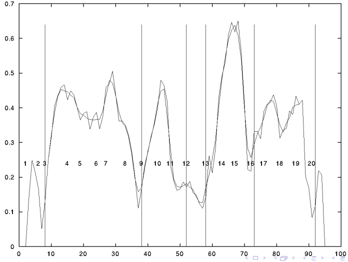

Text 3

  - Do not assign close adjacent segment boundaries; 3 pseudosentences must intervene





## TextTiling: Output of depth scorer on "Stargazer" text

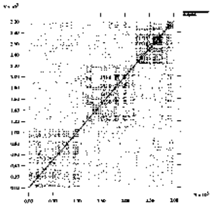







## Alternative Segmentation Algorithms: Reynar (1998)

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

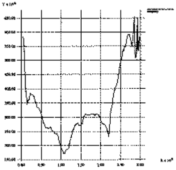






## Alternative Segmentation Algorithms: Reynar (1998)

- 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**

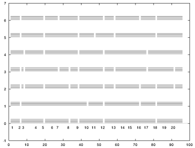
This is a greedy algorithm!





## Evaluation: How to define 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 | |





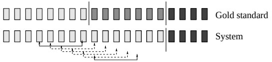
## Evaluation by detecting document boundaries

- 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

- Problems with precision and recall
  - Trade-off between P and R; F-measure hard to interpret here
  - Insensitive to near misses
- $P_k$ measure (Beeferman et al. 1999)
  - Set $k$ to half the average segment size, compute penalties via a moving window of length $k$ (here: $k$=4)
  - If the two ends of the probe are in the same segments, add 1
  - Divide by number of measurements taken; $P_k$ is in [0..1]








## $P_k$ and *win_diff*

Problems with $p_k$ (Prevner and Hearst 2002):

- False negatives penalised more than false positives
- False positives within $k$ sentences of true boundaries not penalised
- Sensitive to variations in segment size
- Near-miss error penalised too much

→ Counter-suggestion: *Win_diff*.

- For each position of the probe, compare true number of segment boundaries falling into this interval($r_i$) with algorithm's number of boundaries ($a_i$)
- If $r_i \neq a_i$, assign penalty of $|r_i - a_i|$
- Divide by $N - k$ (number of measurements taken)





## Summary

- TextTiling
  - Score cohesion
  - Score depth and assign boundaries
- Alternative algorithms
- Evaluation
  - Definition of reference segmentation
  - Metrics $p_k$ and *win_diff*.





## Literature

- **Topic segmentation algorithms**
  - M. Hearst, "Multi-paragraph segmentation of expository text", ACL 1994.
  - **Marti Hearst, "TextTiling: Segmenting Text into Multi-paragraph subtopic passages", Computational Linguistics, 23(1) , 1997**
  - J. Reynar, "An automatic method of finding topic boundaries", ACL 1994.
- **Evaluation Issues**
  - Prevner and M. Hearst: "A critique and improvement of an evaluation metric for text segmentation", Computational Linguistics, 28(1), 2002