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

- 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).

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:

- chained
- ringed
- monolithic
- piecewise

Term repetition signals topic shift/cohesion

Example Text: “The history of algebra”

1. Algebra provides a generalization of arithmetic by using symbols, usually letters, to represent numbers. For example, it is obviously

28. In about 1100, the Persian mathematician Omar Khayyam wrote a treatise...
The most important application of Boolean algebra is in digital computing. Computer chips are made up of transistors arranged in logic gates. Each gate performs a simple logical operation. For example, an AND gate produces a high voltage electrical pulse at the output if and only if a high voltage pulse is received at both inputs \( p, q \). The computer processes the logical propositions in its program by processing electrical pulses - in the case of the AND gate, the proposition represented is \( p \land q \). A high pulse is equivalent to a truth value of "true" or binary digit 1, while a low pulse is equivalent to a truth value of "false", or binary digit 0. The design of a particular circuit or microchip is based on a set of logical statements. These statements can be translated into the symbols of Boolean algebra. The algebraic statements can then be simplified according to the rules of the algebra, and translated into a simpler circuit design.

An algebraic equation shows the relationship between two or more variables. The equation below states that the area (\( a \)) of a circle is given by the formula \( a = \pi r^2 \).

**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 $b_2 = t_{i+1}, ..., t_{i+k+1}$

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

($T$: set of all tokens)

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 $\rightarrow$ want only strong shifts
- Mid-portion with only minor topic shifts $\rightarrow$ 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 $\text{Depth} > \mu - \sigma$ (low cutoff; liberal)
  - Boundary only if $\text{Depth} > \mu - \frac{\sigma}{2}$ (high cutoff; high P, low R)
- For each gap, assign closest paragraph boundary
- Need heuristics to avoid sequence of small segments:

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

TextTiling: Output of depth scorer on “Stargazer” text


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

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

$$C := \{X\} (= \{c_1\})$$

$j := 1$

$\textbf{while } \exists c_i \in C \text{ s.t. } |c_i| > 1 \text{ do}$

$c_u := \text{arg min}_{c_v \in C} coh(c_v)$

$(c_{j+1}, c_{j+2}) = split(c_u)$

$C := C \setminus \{c_u\} \cup \{c_{j+1}, c_{j+2}\}$

$j := j + 2$

$\textbf{end}$

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 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:

<table>
<thead>
<tr>
<th>No. bound.</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>43</th>
<th>50</th>
<th>60</th>
<th>70</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P$</td>
<td>.80</td>
<td>.80</td>
<td>.73</td>
<td>.68</td>
<td>.67</td>
<td>.62</td>
<td>.60</td>
<td>.59</td>
</tr>
<tr>
<td>$R$</td>
<td>.19</td>
<td>.37</td>
<td>.51</td>
<td>.63</td>
<td>.67</td>
<td>.72</td>
<td>.83</td>
<td>.95</td>
</tr>
</tbody>
</table>
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 $\text{win}_{\text{diff}}$

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: $\text{win}_{\text{diff}}$.

- 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 $\text{win}_{\text{diff}}$

Literature

- Topic segmentation algorithms

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