Information Ordering for Text Generation

Computational Models of Discourse
Presented by: Henock Tilahun
Information Ordering for Text Generation

- Papers to be presented:
  
Overview

  ● Motivation
  ● Computing CF lists, CPs and CBs
  ● Computing the classification rate
  ● Performance of M.NO CB
  ● Discussion
Motivation

- Answering the questions:
  - How successful Centering’s constraints are on their own in generating a felicitous text structure? *By developing an approach to text structuring purely based on Centering*
  - Which of these constraints represent the most promising candidates for text structuring?
- Assumed approach characteristics:
  - A search based approach text structuring
  - Most straight way of using centering by defining a centering-based metric of coherence
Computing CF lists, CPs and CBs

- **Utterance**: what is annotated as a finite unit in the GNOME corpus.

  (a) 144 is a torc. (b) Its present arrangement, twisted into three rings, may be a modern alteration; (c) it should probably be a single ring, worn around the neck. (d) The terminals are in the form of goats’ heads.

- **Grammatical functions (gf)** together with linear order used for CF ranking

- CP is the referent of the first NP within the unit that is annotated as a subject for gf.
Computing CF lists, CPs and CBs

- According to gf, the CP of (a) is referent of ne410, “144”

  <unit finite='finite-yes' id='u210'>
  <ne id="ne410" gf="subj">144</ne>
  is
  <ne id="ne411" gf="predicate">a torc</ne> </unit>.

- Ranking of CFs other than the CP is defined as:
  obj > iobj > other

- CFs with the same gf are ranked according to the linear order of the corresponding NPs in the utterance.
Computing CF lists, CPs and CBs

- (a) 144 is a torc.
  \[ \text{cb} = \{\text{undefined}\}, \text{cf} = \{144, \text{torc}\}, \text{cp} = \{144\} \]

- (b) Its present arrangement, twisted into three rings, may be a modern alteration;
  \[ \text{cb} = \{144\}, \text{cf} = \{144, \text{present arrangement, modern alteration}\}, \text{cp} = \{\text{present arrangement}\} \]

<table>
<thead>
<tr>
<th>U</th>
<th>CF list: {CP, other CFs}</th>
<th>CB</th>
<th>Transition</th>
<th>CHEAPNESS ( \text{CB}<em>n = \text{CP}</em>{n-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>{de374, de375}</td>
<td>n.a.</td>
<td>n.a.</td>
<td>n.a.</td>
</tr>
<tr>
<td>(b)</td>
<td>{de376, de374, de377}</td>
<td>de374</td>
<td>RETAIN</td>
<td>+</td>
</tr>
<tr>
<td>(c)</td>
<td>{de374, de379}</td>
<td>de374</td>
<td>CONTINUE</td>
<td>*</td>
</tr>
<tr>
<td>(d)</td>
<td>{de380, de381, de382}</td>
<td>-</td>
<td>NOCB</td>
<td>+</td>
</tr>
</tbody>
</table>

Output of scripts developed by poesio et al (2004)
Computing transitions

- NOCB -> $CF_n$ and $CF_{n-1}$ don’t have any common center

<table>
<thead>
<tr>
<th>SALIENCE:</th>
<th>COHERENCE:</th>
<th>COHERENCE*:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$CB_n = CB_{n-1}$</td>
<td>$CB_n = CB_{n-1}$</td>
<td>$CB_n \neq CB_{n-1}$</td>
</tr>
<tr>
<td>or NOCB in $CF_{n-1}$</td>
<td>or NOCB in $CF_{n-1}$</td>
<td>or NOCB in $CF_{n-1}$</td>
</tr>
<tr>
<td>SALIENCE*:</td>
<td>$CB_n \neq CP_n$</td>
<td>RETAIN</td>
</tr>
<tr>
<td></td>
<td>SMOOTH-SHIFT</td>
<td>ROUGH-SHIFT</td>
</tr>
</tbody>
</table>

- Continue > Retain > Smooth shift > Rough shift
- CHEAPNESS -> $CB_n = CP_{n-1}$
Centering based metrics of coherence

- M.NOCB (baseline for the experiment)
  - Classify each ordering of prepositions according to the number of NOCB it contains
  - Take the ordering with the fewest node
- M.CHEAP: in favour of fewest violation of cheapness
- M.BFP: in favor of highest number of continues
- M.KP: sums up the NOCBs with violations of CHEAPNESS, COHERENCE and SALIENCE
System for Evaluating Entity Coherence (SEEC) is program used to compare metrics.

Basis for Comparison (BfC), B, text translated to sequence of CF lists. (Gold standard)

Classification rate (V) of Metric M = the ability of M to produce B as the output
Computing the classification rate

- First search possible orderings and divide them into *better*, *equal* or *worse* than GS according to M.
- For the selections scoring equal, **half** of the orderings will have better chances than GS to be selected.
- **The fewer** the members of the set of better scoring orderings, the **better** the chances of B to be the chosen output.
Computing the classification rate

- Classification rate:
  \[ v(M, B) = \text{Better}(M) + \frac{\text{Equal}(M)}{2} \]

- \( M_y \) is a more suitable candidate than \( M_x \) for generating \( B \), if \( v(M_y, B) \) is smaller than \( v(M_x, B) \)

- Generalizing across many GSs \( B_1, \ldots, B_m \) from corpus \( C \):
  - Average classification rate:
    \[ Y(M, C) = \frac{v(M, B_1) + \cdots + v(M, B_m)}{m} \]

- The performance of \( M \) on \( m \) GSs from \( C \) in terms of average classification \( Y \).
Comparing M.NOCB with other metrics

- Corpus used: 20 museum labels from GNOME corpus
- Result:

<table>
<thead>
<tr>
<th>Pair</th>
<th>M.NOCB</th>
<th>p</th>
<th>Winner</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lower</td>
<td>greater</td>
<td>ties</td>
</tr>
<tr>
<td>M.NOCB vs M.CHEAP</td>
<td>18</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>M.NOCB vs M.KP</td>
<td>16</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>M.NOCB vs M.BFP</td>
<td>12</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>N</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Ties = cases where the classification rate of the two metrics is the same.
- P = reports whether the difference in the number of GSs is significant
Performance of M.NOCB

- The average classification rate, $Y$, of M.NOCB indicates whether it is likely to arrive at the GS:
  - For this study $Y = 19.95\%$
  - This means the GS is close to the top 20% of alternative orderings.
Two questions addressed in this paper for the first time:

- Which of the centering notions are most relevant to the text structuring task?
- To which extent Centering on its own be useful for this purpose?

Your opinion?
Overview

  - Introduction
  - Methodology
    - Evolutionary Algorithm
    - Operators
  - Result
  - conclusion
Introduction

- Evolution of entity-based models of coherence:
  - McKeown, 1985: uses predefined schemata to describe the structure of a text
  - Mann and Thompson, 1987: uses Rhetorical Structure Theory (RST) as a domain-independent framework for text structuring
  - Grosz et al. 1995: entity-based models of text coherence such as Centering Theory
Introduction

- Kibble and power (2000) define Centering Theory in terms of four principles:
  - **Continuity**: adjacent clauses have at least one referent in common.
  - **Coherence**: maintain the same Cb in successive clauses.
  - **Cheapness**: CBn = CPn-1
  - **Salience**: CBn = CPn
- These principles weighted equally
- Evaluation metric for entity based coherence = \( \sum \) number of times that each candidates structure violates each of the underlying principle of CT
Introduction

- Principle of continuity performs better than the others.
- The principle says "Each utterance in the discourse refers to at least one entity in the utterance that precedes it"
- Karamanis (2001): experiment based solely on the principle of continuity:
  - Generate all possible ordering by permuting the facts in the original ordering
  - Record the number of violation of the principle of continuity and compare it with the score of the original ordering (gold standard)
  - Classify them better, equal or worse than the gold standard based on the score.
Introduction

- Result:
  - 90% worse
  - 9% equal
  - 1% better than the gold standard
- Replacing the principle of continuity with other metrics of entity-based coherence results:
  - 41% worst
  - 15% equal
  - 44% better than gold standard
- This shows principle of continuity permits only a limited number of possible orderings to score better than the original structure.

What can you generalize from this result?
Introduction

- This search is impractical due to the factorial complexity of exhaustive search.
- For one text consisting of 12 facts, there are $12!$ Orderings.
- **Stochastic approach** extends this for large input with more efficient search space.
Methodology

- Input = unordered set of facts that correspond to the underlying semantics of possible text.
- Goal = to find an ordering of all the facts that maximizes its entity-based coherence.
- The input represents the output of content determination phase of NLG.
- NLG Pipeline architecture:
  - Content determination
  - Discourse planning
  - Sentence aggregation
  - Lexicalization
  - Referring expression generation
  - Surface realization
Text used are short descriptions of archaeological artifacts.

Towards the end of the archaic period, coins were used for transactions. This particular coin, which comes from that period, is a silver stater from Croton, a Greek Colony in South Italy. On both the obverse and the reverse side there is a tripod (vessel standing on three legs), Apollo’s sacred symbol. Dates from between 530-510 BC.
Methodology

- The unordered set of facts that will be the semantic input to the system is:
  1. use-coins(archaic-period)
  2. creation-period(ex5, archaicperiod)
  3. madeof(ex5, silver)
  4. name(ex5, stater)
  5. origin(ex5, croton)
  6. concept-description(croton)
  7. exhibit-depicts({ex5, sides}, tripod)
  8. concept-description (tripod)
  9. symbol (tripod, apollo)
 10. dated (ex5, 530-510bc)
Evolutionary algorithm

- The task of generation doesn’t necessarily require a global optimum.
- what is needed is:
  - A text that is coherent enough to be understood
  - Anytime algorithm: which can terminate at anytime and to yield the best result
- These characteristics belongs to Evolutionary Algorithms.
Evolutionary algorithm

- $t = 0$
- Initialize population $P(t)$ with $n$ random orderings of the given facts.
- Evaluate $P(t)$ and rank/select $P(t+1)$
- While optimal solution not found or $t < \text{maximum iterations}$
  - Do
    - Evolve $P(t)$ with mutation and/or crossover operations
    - Evaluate $P(t)$ and rank/select $P(t+1)$
    - $t := t + 1$
  - End While
Evolutionary algorithm

- The selection process uses:
  - **Roulette-wheel algorithm**: which selects candidate solution with probability proportional to their fitness values.
  - **Elitist strategy**: where a small percentage of the fittest individuals are always copied over unevolved to the next generation.
- Fitness function assign a higher score to more continuous texts.
- The theoretical global maximum score for \( n \) input semantic facts is \( n - 1 \)
Operators

- **Mutation:**
  - Generating completely random permutation
  - Random *swapping* of two facts
  - Random *repositioning* of a fact

- **Crossover:**
  - Combining of subsequences from two orderings $x$ and $y$
**Result (Preliminary experiment)**

- Preliminary experiment to find the most promising choice of parameters for EA.

  - The result obtained using crossover with three different values for elitist ratio:
    - Elitist ratio = 0.0
    - 0.1
    - 0.2
Result (Preliminary experiment)

- Result obtained using Permute with three elitist ratio:
  - Elitist ratio = 0.0
  - Elitist ratio = 0.1
  - Elitist ratio = 0.2

- Permute perform worse than crossover because completely random permutation is a highly non-local move in search space.
Result (main experiment)

- No of iteration = 4000
- Population size = 50
- Elitist ratio = 0.2
- Operator = crossover

<table>
<thead>
<tr>
<th>Text name</th>
<th>$n$ facts</th>
<th>Target</th>
<th>Mean</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>stater</td>
<td>10</td>
<td>7</td>
<td>6.482</td>
<td>8.0</td>
</tr>
<tr>
<td>tetradrachm</td>
<td>10</td>
<td>8</td>
<td>7.602</td>
<td>9.0</td>
</tr>
<tr>
<td>drachma</td>
<td>11</td>
<td>9</td>
<td>8.384</td>
<td>10.0</td>
</tr>
<tr>
<td>kouro</td>
<td>18</td>
<td>13</td>
<td>14.022</td>
<td>17.0</td>
</tr>
<tr>
<td>amphora</td>
<td>20</td>
<td>17</td>
<td>15.328</td>
<td>19.0</td>
</tr>
<tr>
<td>hydria</td>
<td>23</td>
<td>20</td>
<td>16.783</td>
<td>20.8</td>
</tr>
</tbody>
</table>

Table 1: Results of the main experiment
Result (main experiment)

Mean and maximum population scores for hydria
Conclusion

- It is **not complete solution** for text structuring, rather experimenting the feasibility of using stochastic search.
- Even though the optimal solutions are quite rare, this approach manages to **reach global optimum**
- **Not** all the resulting surface texts are quite coherent.
Thank you for your Attention

Your opinion?