Today

- Summarization (content selection, evaluation)
- Techniques: alignment, classification, rewriting

Types of Summarization

- Input: speech/text, single-/multi-document
- Output: generic/query-oriented
- Approach: domain dependent/independent, extraction/generation

What is Summarizer

- Find important information in a text
- Learn transformation rules based on training instances
- Extract certain facts from a text, and combine them into a text
Supervised Approaches

- Alignment (trivial for extraction, hard for generation)
- Feature Selection
- Classification (standard classifiers — Naive Bayes, SVM, maximum entropy, Boostexter)

Key Questions

- Content selection
- Content organization and linguistic realization
- Evaluation
### Zipf Distribution
The product of the frequency of words (f) and their rank(r) is approximately constant: \( f \times R = C \) (where C is around \( N/10 \))

![Zipf Distribution Graph]

Rank = order of words’ frequency of occurrence

### Assigning Weights
- Raw frequencies (typically with the list of stop-words)
- TF*IDF – a way to deal with the problem of the Zipf distribution
  - TF - Term frequency
  - IDF - Inverse term frequency

### Feature Selection
**Shallow Features:**
- Locational Features (in the newspaper genre, the first paragraph is a summary)
- Presence of cue words (e.g., “in conclusion”)
- Sentence length
- Number of highly weighted words in a sentence

### Word Frequency vs Resolving Power
(from van Rijsbergen, 1979) The most frequent words are not the most descriptive

![Word Frequency vs Resolving Power Graph]
Feature Selection

“Deep Features”
- Rhetorical structure based
  - RST (Marcu, 2000)
  - Domain-dependent argumentative structure
    (Teufel & Moens, 2000)
- Content-based (Barzilay & Lee, 2003)

Around 10% improvement

TF*IDF

\[ w_{ik} = T f_{ik} \times \log\left(\frac{N}{n_k}\right) \]

- \( w_{ik} \): Term \( k \) in document \( D_i \)
- \( T f_{ik} \): Frequency of term \( k \) in document \( D_i \)
- \( N \): Total number of documents in the collection \( C \)
- \( n_k \): Total number of documents in the collection \( C \) that contain \( T_k \)

Alignment Input

Amsterdam is the largest city in The Netherlands and the country's economic center. It is the official capital of The Netherlands, though The Hague is the home of the government. Tourists come to see Amsterdam's historic attractions and collections of great art. They admire the city's scenic canals, bridges, and stately old houses. Amsterdam is also famous for its atmosphere of freedom and tolerance.

City and port, western Netherlands, located on the IJsselmeer and connected to the North Sea. It is the capital and the principal commercial and financial centre of The Netherlands. To the scores of tourists who visit each year, Amsterdam is known for its historical attractions, for its collections of great art, and for the distinctive colour and flavour of its old sections, which have been so well preserved. However, visitors to the city also see a crowded metropolis beset by environmental pollution, traffic congestion, and housing shortages. It is easy to describe Amsterdam, which is more than 700 years old, as a living museum of a bygone age and to praise the eternal beauty of the centuries-old canals, the ancient patrician houses, and the atmosphere of freedom and tolerance, but the modern city is still working out solutions to the pressing urban problems that confront it. Amsterdam is the nominal capital of The Netherlands but not the seat of government, which is The Hague. The royal family, for example, is only occasionally in residence at the Royal Palace, on the square known as the Dam, in Amsterdam.

Alignment

Champollion ‘1822

Find pairs of corresponding elements
Alignment in MT

- Alignment task: Given bitext, identify units which are translations of each other.
- Units: paragraphs, sentences, phrases, words.
- Usage: first step for full translation (Brown et al), lexicography (Dagan & Church, Fung & McKeown), aid for human transaltors (Shemtov), multi-lingual IR.

Length-based Alignment

Let $D(i, j)$ be the lowest cost alignment between sentences $s_1, \ldots, s_i$ and $t_1, \ldots, t_j$.

Base: $D(0, 0) = 0$.

$$
D(i, j) = \min \begin{cases} 
D(i, j - 1) + \text{cost}(0:1 \text{ align } \phi, t_j) \\
D(i - 1, j) + \text{cost}(1:0 \text{ align } s_i, \phi) \\
D(i - 1, j - 1) + \text{cost}(1:1 \text{ align } s_i, t_j) \\
D(i - 1, j - 2) + \text{cost}(1:2 \text{ align } s_i, t_{j-1}, t_j) \\
D(i - 2, j - 1) + \text{cost}(2:1 \text{ align } s_{i-1}, s_i, t_j) \\
D(i - 2, j - 2) + \text{cost}(2:2 \text{ align } s_{i-1}, s_i, t_{j-1}, t_j) 
\end{cases}
$$
Design Choices in Alignment

Determined by a Corpus Type

- Matching predicate
- Search strategy

Matching Predicate

- Length similarity. (Gale & Church, Brown et al)
- Lexical similarity:
  - Bilingual dictionary (Wu)
  - Words with the same distribution. (Kay & Roscheisen, Fung & McKeown)
  - Cognates (Simard et al, Church, Melamed)

Methods for Overall Alignment

- Dynamic programming
- Methods based on Computational Geometry
- Signal processing Methods

Corpus Type

- Language Proximity (Monolingual vs Bilingual, technical vs lay)
- Content Proximity (comparable vs parallel)
- Matching Granularity (1:1 vs 1:5)
Computational Geometry Methods

(Melamed, 1997) Assumption: Distribution of “true points of correspondence (TPC)” satisfies certain geometric properties

- Generate all the matching points satisfying the matching predicate (over-generation)
- Find a subset of matching points that satisfies a pattern of TPC:
  - Linearity
  - Injectivity
  - Low variance of slope

Various heuristics are used to minimize the search space

Alignment for Summarization

- Always monolingual
- Seems to be trivial (use word intersection!)

Signal Processing Methods

(Fung, 1995)
Weak Similarity Function

(A) · Petersburg served as the capital of Russia for 200 years.
· For two centuries Petersburg was the capital of the Russian Empire.

(B) · The city is also the country's leading port and center of commerce.
· And yet, as with so much of the city, the port facilities are old and inefficient.

Domain-Dependent Structure-Based Alignment

(Barzilay&Elhadad, 2003) Assumption: Weak similarity function augmented with structural information
· Content Structure Induction
· Learning of Structural Mapping Rules
· Macro Alignment
· Micro Alignment

It is hard!

· Insertions, deletions, reordering
· Weak similarity function

Patterns of Mapping
Lisbon has a mild and equable climate, with a mean annual temperature of 63 degree F (17 degree C). The proximity of the Atlantic and the frequency of sea fogs keep the atmosphere humid, and summers can be somewhat oppressive, although the city has been esteemed as a winter health resort since the 18th century. Average annual rainfall is 26.6 inches (666 millimetres).

Jakarta is a tropical, humid city, with annual temperatures ranging between the extremes of 75 and 93 degree F (24 and 34 degree C) and a relative humidity between 75 and 85 percent. The average mean temperatures are 79 degree F (26 degree C) in January and 82 degree F (28 degree C) in October. The annual rainfall is more than 67 inches (1,700 mm). Temperatures are often modified by sea winds. Jakarta, like any other large city, also has its share of air and noise pollution.
Evaluation

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0%–40%</td>
<td>50%</td>
<td>25%</td>
<td>23%</td>
<td>15%</td>
</tr>
<tr>
<td>40%–70%</td>
<td>85%</td>
<td>73%</td>
<td>66%</td>
<td>86%</td>
</tr>
<tr>
<td>70%–100%</td>
<td>95%</td>
<td>95%</td>
<td>90%</td>
<td>95%</td>
</tr>
</tbody>
</table>

Semiotic-based Summarization

Assumption: In a limited domain, we know “what is important” (Radev&McKeown, 1995, Elhadad&McKeown, 2001)
- Use an information extraction system to select “important information”
- Use a semantics-to-text generation system to generate a new text

Micro-Alignment

\[
s(i, j) = \max \begin{cases} 
  s(i, j - 1) - \text{skip\_penalty} \\
  s(i - 1, j) - \text{skip\_penalty} \\
  s(i - 1, j - 1) + \text{sim}(i, j) \\
  s(i - 1, j - 2) + \text{sim}(i, j) + \text{sim}(i, j - 1) \\
  s(i - 2, j - 1) + \text{sim}(i, j) + \text{sim}(i - 1, j) \\
  s(i - 2, j - 2) + \text{sim}(i, j - 1) + \text{sim}(i - 1, j) 
\end{cases}
\]

Summarization Evaluation

- Precision/Recall or their weighted version are used
- As a baseline, people use a “lead” summary
- Human agreement is computed using Kappa
- When evaluation results matter, it is done manually (DUC competition)
  - Provides large collection of human-generated summaries
  - Outputs are evaluated manually