Documents/Summaries as background models for update summary extraction

Maheedhar Kolla, Olga Vechtomova, Charles L.A. Clarke
IR Group
University of Waterloo

April 26, 2007

Information search

- Users going through search results often get deceived.
- Highlighting only shows that the document has the query term(s).
- Would it help if we could show them snippets that would highlight the content novel to the docs on previous results page(s)?

Proposed model:

- Overview: Find the key terms and extract the sentences.
- Which terms are key ? and which are redundant?
- Words' score in current document is discounted by the weight of the term in the background model.

$$P(w|d_{new}) = (1 - \alpha)(P(w|d)) - \alpha(P(w|\theta_{old})) \qquad (1)$$

where, $P(w|\theta_{old})$ - probability of word in background model and $(P(w|d))$ is the Maximum Likelihood estimate of the word in current document.

Maximum Likelihood Estimate

- Probability of a term based using raw term counts:

$$P_{mle}(w|d) = \frac{tf(w,d)}{\sum_{w_j} C(w_j; d)} \tag{2}$$

  where $\sum_{w_j} C(w_j; d)$ is equivalent to the length of the document.

- Drawback: less frequent words are assigned lesser weights.
- Better approximate would be to smooth with collection probabilities, using Cluster Based Document Model:

$$P(w|d) = \lambda(P_{mle}(w|d) + (1-\lambda)[\beta P(w|cl) + (1-\beta)(P(w|coll))] \tag{3}$$

  where

$$P(w|cl) = \lambda\left(\frac{tf(w,cl)}{\sum_{w^j \epsilon cl} tf(w^j; cl)}\right) + (1-\lambda)\frac{tf(w,coll)}{\sum_{w^j \epsilon V} tf(w^j, coll)} \tag{4}$$

Background Models θ_b:

- Not every recurring term redundant.
- Care should be taken to avoid topic drift.
- Which of the following make a better background model?
  - Models constructed from all previous documents.
  - Models constructed from only summaries of previous documents.

Document based model: θ_d

- Compute the term probability in each of the previously seen document(s).
- Term's weight in the background model is then the average of those probabilities.
- Some other methods may be to use the term's max probability value in the previous document set(s).
- Could use divergence based measures too.

Summaries based model: θ_{summ}

- Generic single document summaries are used for model construction.
- We used simple lexical term overlap between the sentences to rank the sentences.
- Two sentences should have at least 3 word overlap to be considered bonded.
- Top two sentences were selected from each document.
- Weight of the term in this model is probability computed using the raw counts of the term to the total length of the summaries.

Estimation of the parameters λ and β

- First, we determine the optimal values for λ and β
- Used the initial set of documents- that lacks any previous documents.
- Fixed the values of λ and β that maximize the ROUGE-2 and ROUGE-SU4 scores.
- The values were found to be 0.4 and 0.6 respectively.
- Values were fixed for experiments to study the effect of α on two background models.

Performance of the background models, at various α values

Table: ROUGE-SU4 measures for various values of α on the two different background models [λ and β constant in this case]

Model	$\alpha = 0.3$	α =0.4	α =0.5	$\alpha = 0.7$
θ_{doc}	0.10830	0.10514	0.09809	0.08279
θ_{summ}	0.10630	0.10192	0.09319	0.08449

Table: ROUGE-2 measures for various values of α on the two different background models [λ and β constant in this case]

Model	α =0.3	α =0.4	α =0.5	$\alpha = 0.7$
θ_{doc}	0.07073	0.06938	0.06089	0.04395
θ_{summ}	0.06752	0.06433	0.05652	0.04601

Discussion:

- For a given set of α β and λ, both models perform almost the same.
- Both models has similar effect on the variation of the parameter α.
- Document clusters were hand-picked - very focused on the topic.
- The values of the λ and β estimated from the first set were stable across the other summary extraction experiments.

Conclusion and Future Work:

- Proposed method to update the word probability using previously seen docs.
- Further experiments to use standard query-likelihood techniques to rank the documents.
- Induce noise documents into all sets - close approximate to the real-world problem.
- Extensive experiments using several smoothing functions.

Thank you