Sentence Retrieval

Sentence- versus Document approaches

Question Answering

Patrick Schulze

Overview

1. Document retrieval techniques for sentence retrieval

   Vanessa Murdoch, Aspects of Sentence Retrieval, PhD Thesis, 2006

2. A more sophisticated approach for sentence retrieval

   Fernandez et al., Extending the language modeling framework for sentence retrieval to include local context, Journal of Information Retrieval, 2010

3. Are there any benefits for our project?

1. Documents versus Sentences - Goals

- Document retrieval:
  - find relevant documents regarding a certain query
- Sentence retrieval:
  - Question answering
  - Extractive summarization
  - Novelty detection
  - Opinion mining

1. Document retrieval for sentences

- Basic assumptions:
  - sentence retrieval is document retrieval
  - documents have a certain typical length
    - Aquaint collection ~ 14 sentences
    - TREC volumes 1, 2 ~ 22 sentences
    - TREC volume 3 ~ 23 sentences
    - TREC volumes 4, 5 ~ 25 sentences
  - all test collections consist of newswire articles
    - Associated press
    - Xinhua news agency
    - New york times news service

Is there a correlation between the document length and the performance of information retrieval systems?

1.1 Influence of document length

	Docs	750 bytes	500 bytes	250 bytes	Sents
Relevant	31672	99703	112544	130139	73623
Relevant and Retrieved	31672	19278	16876	13026	8639
Interpolated Recall-Precision					
at 0.00	0.497	0.421	0.379	0.327	0.268
at 0.10	0.329	0.172	0.153	0.103	0.085
at 0.20	0.270	0.119	0.092	0.061	0.052
at 0.30	0.234	0.073	0.059	0.035	0.036
at 0.40	0.201	0.048	0.042	0.028	0.028
at 0.50	0.186	0.039	0.034	0.023	0.022
at 0.60	0.162	0.023	0.021	0.012	0.007
at 0.70	0.140	0.018	0.016	0.009	0.006
at 0.80	0.126	0.015	0.013	0.007	0.004
at 0.90	0.116	0.013	0.012	0.006	0.003
at 1.00	0.109	0.012	0.010	0.005	0.003

- Setup:
  - TREC QA-track questions
  - Top 1000 documents of Aquaint corpus
  - 413 questions
  - 375 available answers in top 1000 documents
  - Answer tokens detected with regular expressions
  - Used retrieval model: Query Likelihood with Helinek-Mercer smoothing
  - k-byte fragments built of the documents
  - fragments overlap by half

Longer documents achieve a better performance.

1.2 **Term frequency** versus Query likelihood

$$w_{q_i,D} = tf_{q_i,D} \cdot idf_{q_i}$$

$$tf_{q_i,D} = \frac{c(q_i;D)}{\max_l\{c(q_l;D)\}}$$

$$idf_{q_i} = \log \frac{N}{n_{q_i}}$$

- $W_{qi,D}$ weight of the term q_i in D
- $tf_{qi,D}$ term frequency of term q_i in D
- idf_{qi} inverse document frequency of term
- $c(q_i;D)$ count of q_i in D
- $\max_l\{c(q_l;D)\}$ count of the most frequent term l in document D
- N number of documents in the collection
- n_{qi} number of documents containing term q_i

- Documents with higher term frequency are ranked higher
- idf score prevents very frequent terms from dominating the score
- **But** term-weights are determined heuristically

1.2 Term frequency versus **Query likelihood**

$$P(Q,D) = \frac{P(Q \mid D)P(D)}{P(Q)}$$

$$P(Q \mid D) \propto \prod_{i=1}^{|Q|} P(q_i \mid D)$$

$$P(q_i \mid D) = \frac{c(q_i; D)}{|D|}$$

- Q query
- D document
- P(D) initial relevance, equal over all documents
- P(Q) probability of generating the term Q
- |Q| number of terms in the query
- q_i i-th term in the query
- $c(q_i;D)$ count of the term qi in the document D

- Ranks documents by the probability the query was generated by the same distribution of terms the document is from
- Allows comparison and ranking in terms of a document model

1.2 Results

- Both models share the same disadvantages in terms of sentence retrieval
  - Relevant sentences will only contain a small number of query terms

- Examples:
  - Aquaint collections average document length: 250 words
  - Aquaint collections average sentence length : 18 words

Few term matches result in barely distinguishable relevant and non-relevant documents.

1.3 Query expansion and Relevance feedback

- Addresses problem of vocabulary mismatch
- Query expansion (Maron and Kuhns)
    - $query_{new} = query_{old}$ + new related terms

- Relevance feedback and Pseudo-relevance feedback (Lavrenko and Croft)
    - terms from known documents or clusters of related terms as terms in place of the original query
    - 2 passes:
        1. Initial retrieval using the original query
           Create a topic model of the query from the top N documents with m content terms
        2. Re-ranking of the documents with respect to the likelihood they genereated the new distribution of query terms

1.3 Experiment

	Query Likelihood	Query Expansion
Prec @ 1	.168	.074
Prec @ 5	.117	.043
Prec @ 10	.111	.044
Prec @ 15	.101	.044
Prec @ 20	.096	.040
Prec @ 1000	.037	.023
Recall	.506	.416

	Query Likelihood	Relevance Models
Prec @ 1	.168	.161
Prec @ 5	.117	.123
Prec @ 10	.111	.113
Prec @ 15	.101	.100
Prec @ 20	.096	.094
Prec @ 1000	.037	.037
Recall	.506	.506

- Setup:
  - Top 1000 documents regarding their relevance were used of Aquaint collection and the TREC collection 4,5
  - All documents were sentence segmented
  - Baseline retrieval via query likelihood to retrieve top 1000 sentences using description queries
  - Relevance assessments provided by NIST for the Novelty Task
  - Query expansion using a probabilistic dictionary of related terms (from TREC topic titles)
  - Relevance feedback using the top 50 to 75 sentences (N) with the top 75 terms (m)

Query expansion degrades the performance of sentence retrieval.

1.3 Results

- Query expansion
  - Automatic query expansion leads to mixed results
  - Most successful on poorly specified queries
- Relevance feedback
  - Non-matching terms in the query get a background probability
  - Problem of terms of a different topic in the query
    - Document retrieval mitigates the influence of documents of other topics
    - Sentence retrieval is very vulnerable to spurious query terms
- Different relevance models of DR and SR
  - DR relevance models are designed to capture the topic of a document

1.4 Smoothing

- Used in the query-likelihood model to avoid zero probabilities

$$P(Q \mid S) \approx \boxed{\prod_{i=1}^{|Q|} P(q_i \mid S)}$$

- Example: Dirichlet Smoothing

$$P(Q \mid S) = P(S) \prod_{i=1}^{|Q|} \frac{c(q_i; S) + \mu P(q_i \mid C)}{|S| + \mu}$$

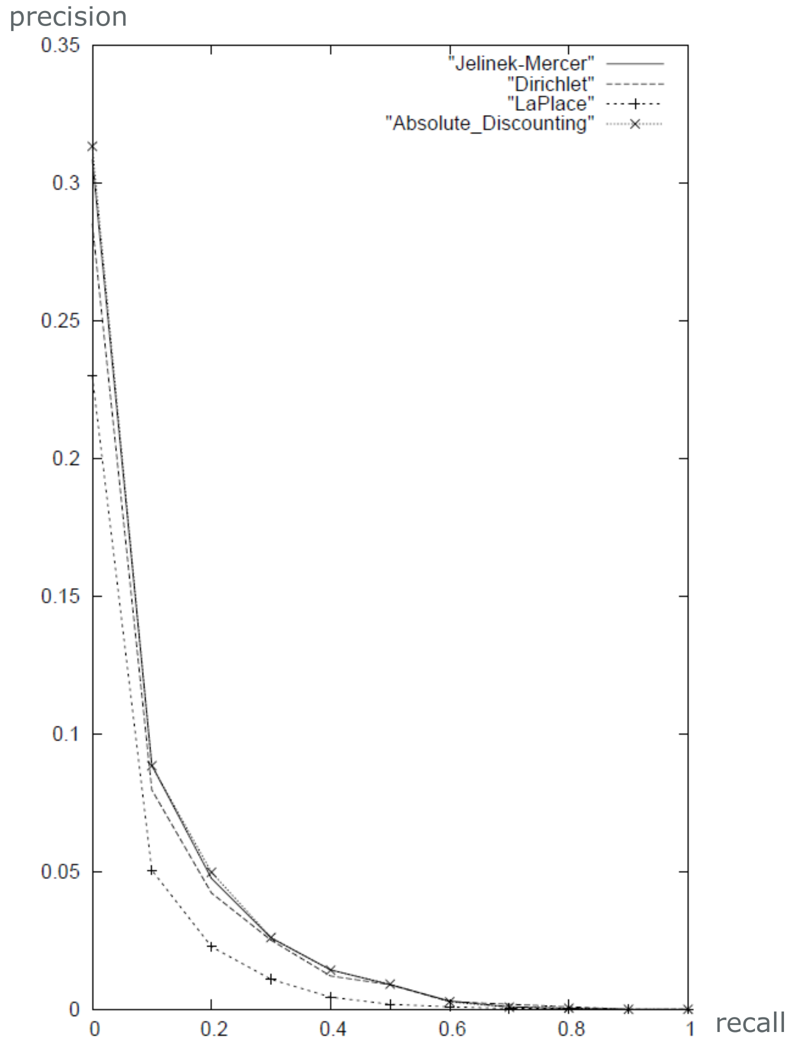
1.4 Dirichlet Smoothing

$$P(Q \mid S) = P(S) \prod_{i=1}^{|Q|} \frac{c(q_i; S) + \mu P(q_i \mid C)}{|S| + \mu}$$

- P(S) initial (constant) sentence relevance
- μ smoothing parameter > 0 (constant over all sentences of C)
- C collection consisting of all sentences
- $c(q_i;S)$ count of the term q_i in the sentence S
- P(qi|C) probability the query term q_i was generated by C (term freq.)

- Short documents in comparison to μ lead to more weight of the collection probabilities.
- Dirichlet smoothing penalizes short documents more than long ones.

1.4 Experiment



- Setup:
  - TREC novelty task
  - Designed to investigate system's abilities to locate relevant and new information relevant to a TREC topic
  - Preconditions: the topic and a set of relevant documents ordered by date
  - Systems have to idenfity sentences containing relevant and/or new information
  - 150 topic descriptions
  - Almost no difference between the smoothing techniques because of small variance in sentence lengths
  - La-Place smoothing is a bit worse because of a bad chosen smoothing parameter

Smoothing has almost no impact on the performance.

2.1 Conclusion

- Reduced document length leads to lower performance.
- Unchanged document retrieval techniques are not suitable for sentence retrieval.

- Reasons
  - Higher term counts result in higher scores without any differentiation between unique and multiple terms.
  - Compensation of vocabulary mismatches (e.g. via query expansion) assumes that expanded queries have many terms in common with the document.
  - Sentences are much more sensitive for smoothing techniques i.e. it is hard to distinguish between relevant and non-relevant information.
  - Discrepancy between the model of relevant entities
    - Documents: topics
    - Sentences: more specific information

2.2 Sentence retrieval

- State of the art retrieval method
  - term frequency – inverse sentence frequency
- More sophisticated methods were not able to outperform tf-isf
  - Natural language processing
  - Clustering
  - Query expansion
- Current assumptions do not hold because:
  - sentences are dependent and have a local context
  - relevant sentences need to be indicative of the query topic
  - relevant sentences are important in the context of the document

3. Contributions for our project

- We use document retrieval (google) for finding the right answer tokens.
- The main problem is the specification of the query.
- Since query expansion works good for documents google already uses that.
- We have no possibility of getting all documents upfront to retrieve the relevant information on a sentence-level.

FIN