Query-Focused Summarization: Summarization is Easy, Retrieval is Hard

Tal Baumel, Raphael Cohen, Jumana Nassour-Kassis, Michael Elhadad

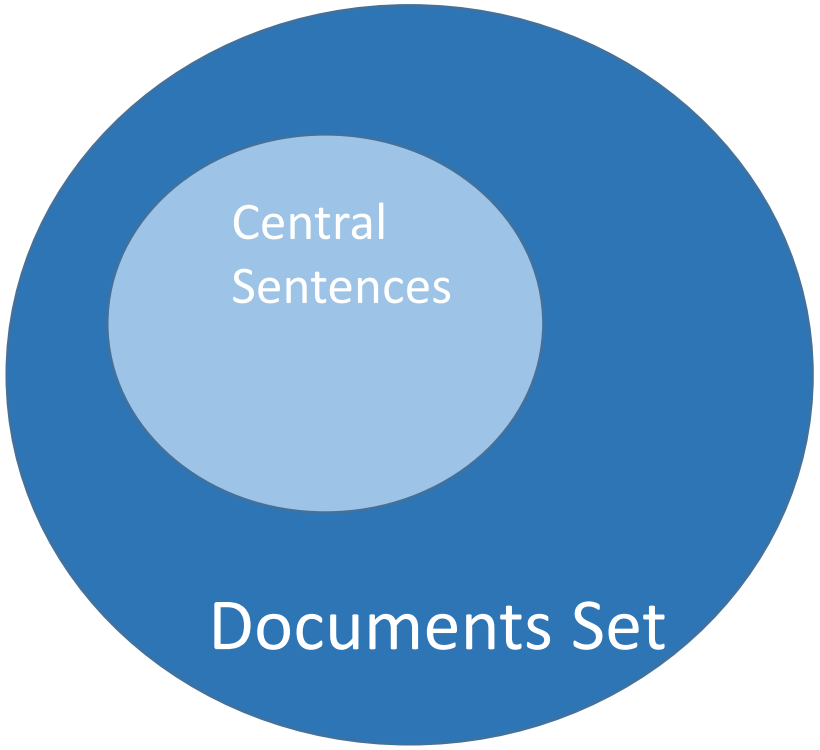
Summarization Tasks

Multi-Document Summarization (MDS)

Query-Focused Summarization (QFS)

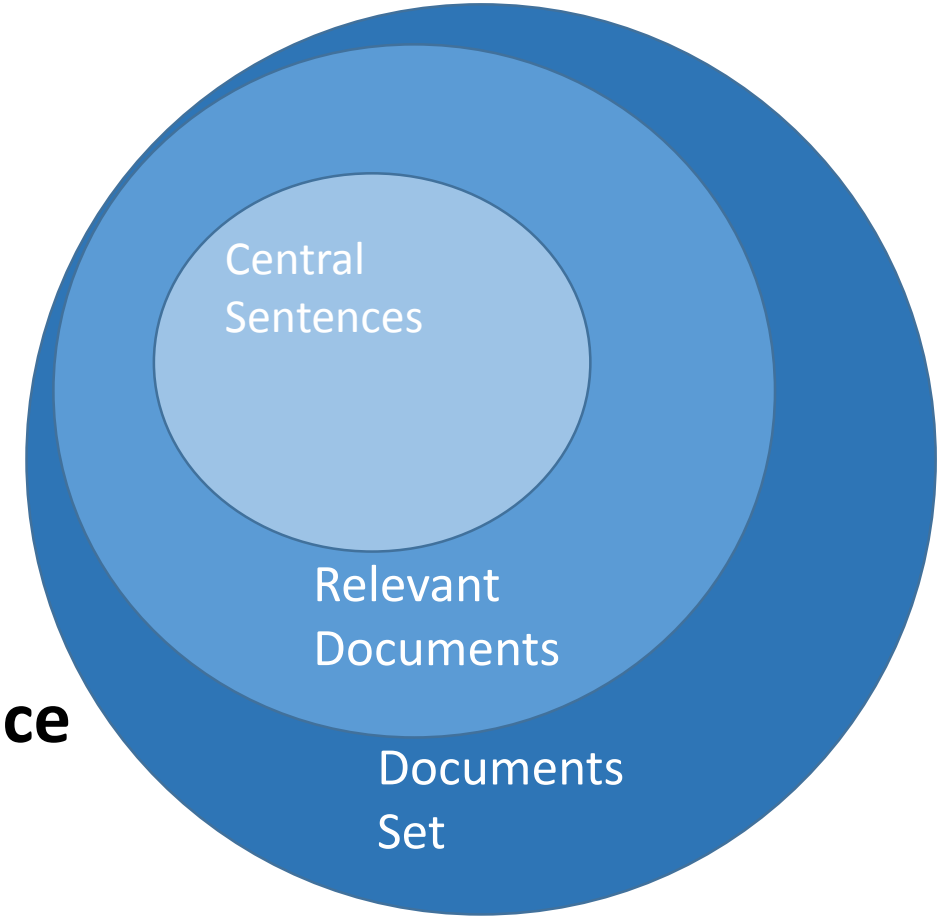
Multi-Document Summarization (MDS)

- Given a set of documents D
  - All documents from D are topically related
- Select a set of sentences S:
  - $|S| < L$
  - "Central information" from D is captured by S
  - S does not contain redundant information
- Key concepts: **Centrality, Redundancy**



Query-Focused Summarization (QFS)

- Given a set of documents D and a query Q
  - All documents from D are topically related
  - D contains information relevant to Q
- Select a set of sentences S:
  - $|S| < L$
  - S captures information in D relevant to Q
  - S does not contain redundant information
- Key concepts: centrality, redundancy, **relevance**



QFS Datasets

DUC 2005

QCFS

DUC 2005

*"The task is to synthesize from a set of **25-50 documents** a brief, well-organized, fluent answer to a **need for information** that cannot be met by just stating a name, date, quantity"*

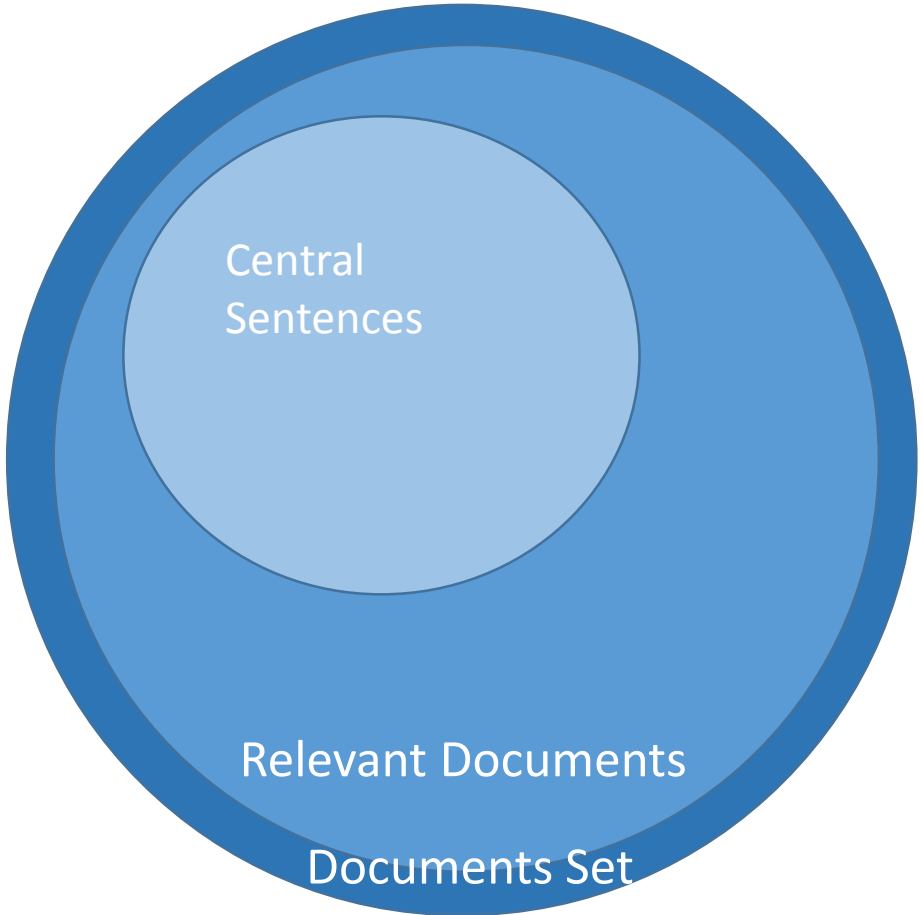
http://www-nlpir.nist.gov/projects/duc/duc2005/

DUC Query example

“Identify and describe types of organized crime that crosses borders or involves more than one country. Name the countries involved. Also identify the perpetrators involved with each type of crime, including both individuals and organizations if possible.”

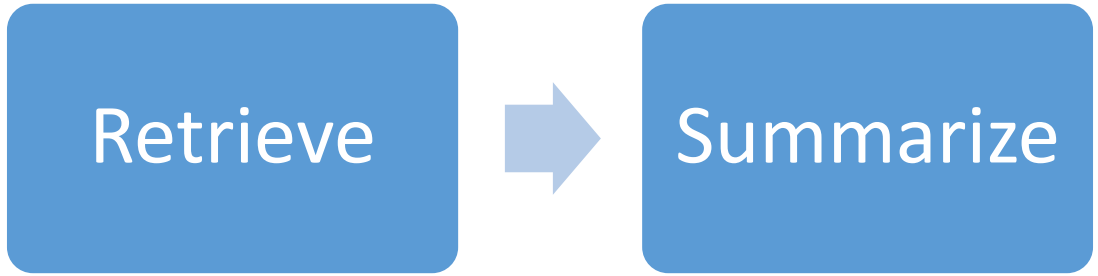
DUC 2005 Criticism (Gupta et al. 2007)

- *"it is noteworthy that the generic summarizers perform about as well as their focused counterparts."*
- Problem: **topic concentration**



How can we measure topic concentration

- Intrinsic – Test the similarity between the sentences and the query (Gupta)
- Extrinsic – Task based evaluation
  - Objective – summarization
  - Method – retrieve and summarize



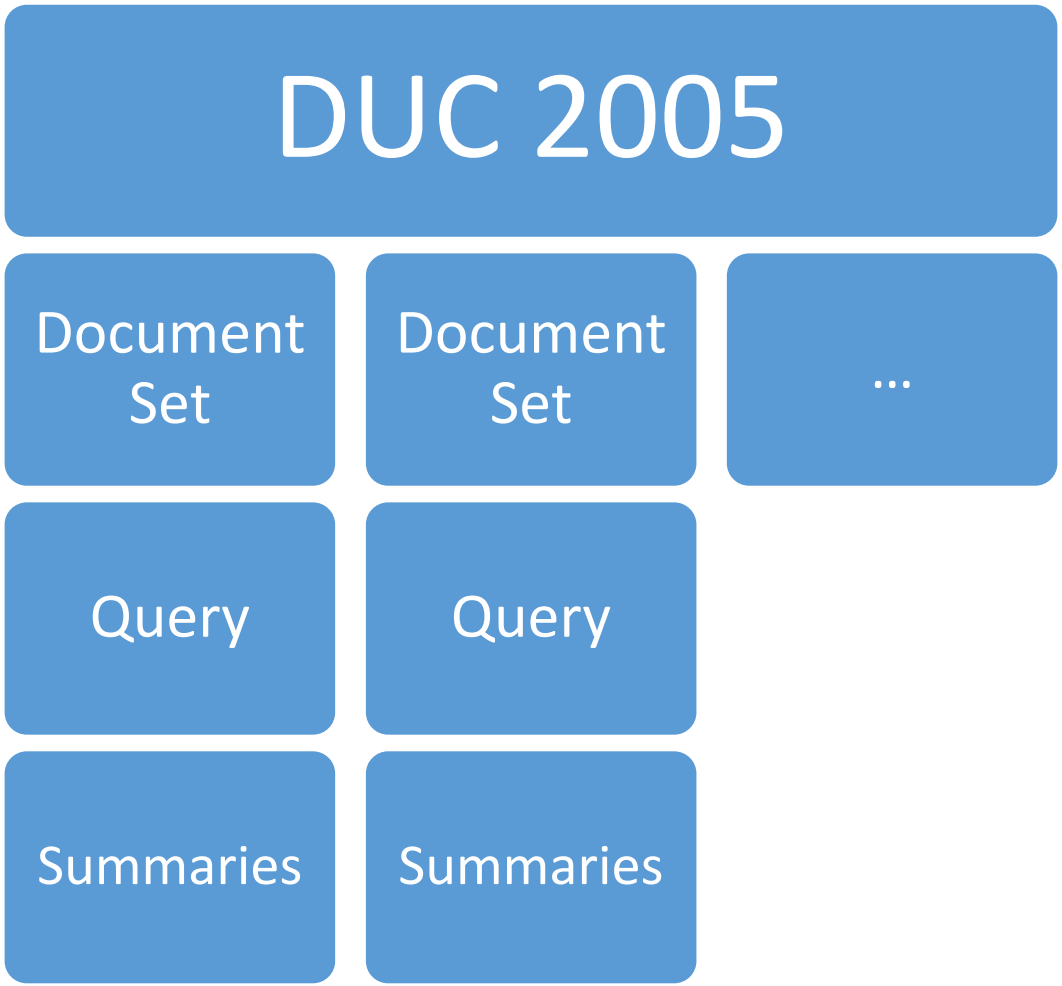
What We Found

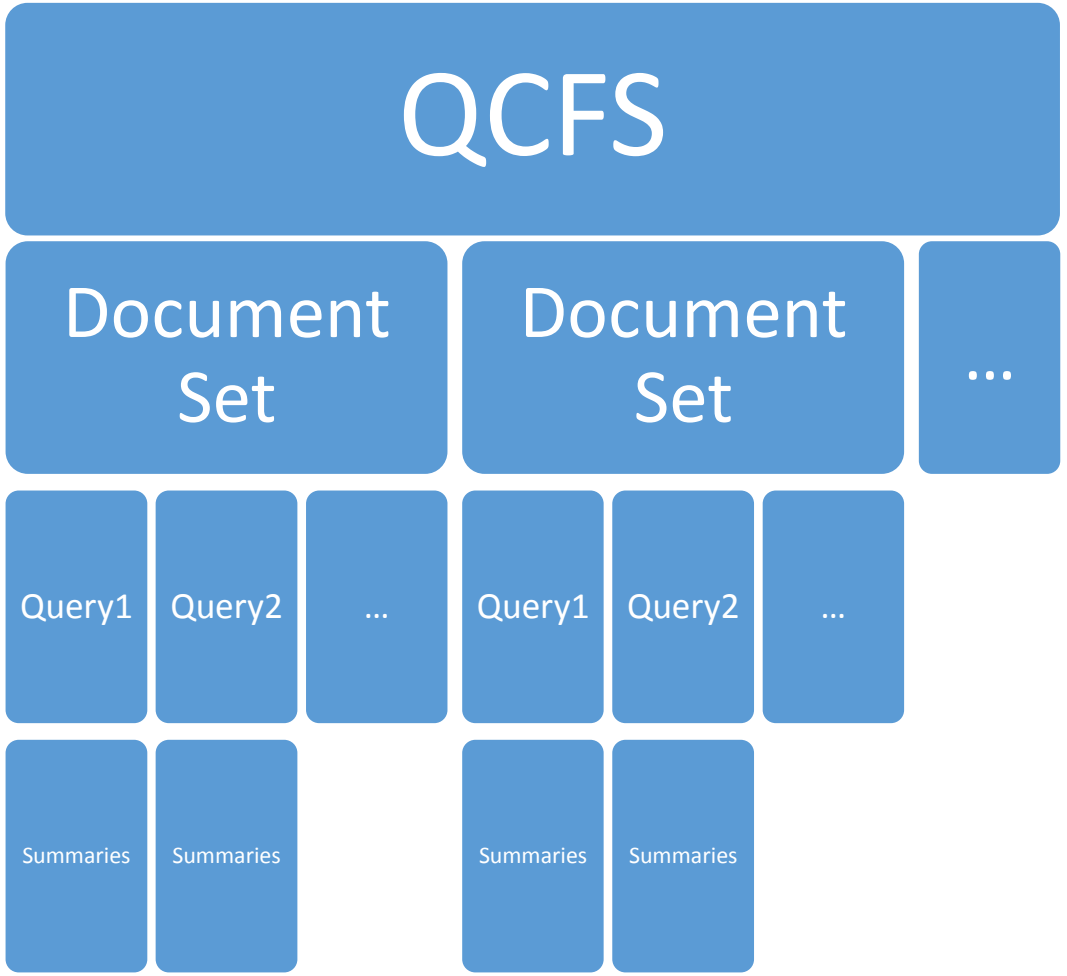
- DUC dataset is overly topic concentrated
- We introduce a new dataset for QFS
  - Not concentrated
- We analyzed the effects of retrieval on QFS performances

Query Chain Focus Summarization (QCFS) Dataset

- Created by BGU NLP Lab to capture the effect of a number consecutive queries on the user information need
- For this work we used only the 1st query in each chain

Principal Difference between the datasets



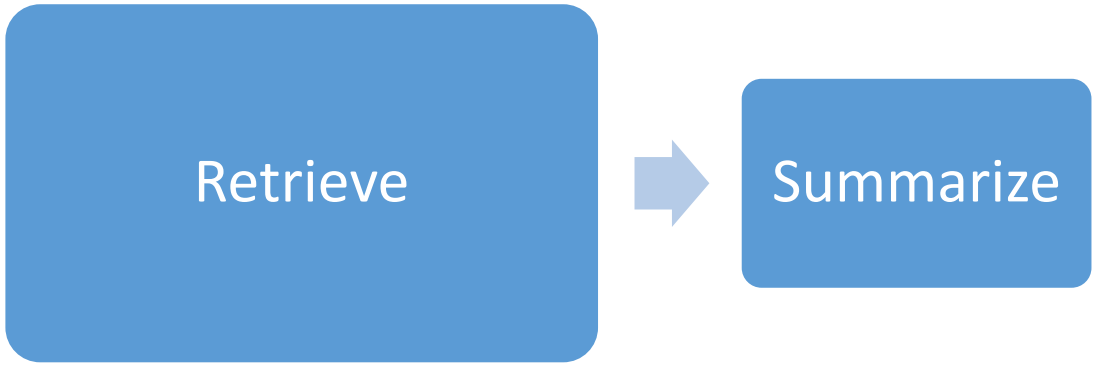


Retrieval Methods

Cosine Similarity

Relevance Model

Gold



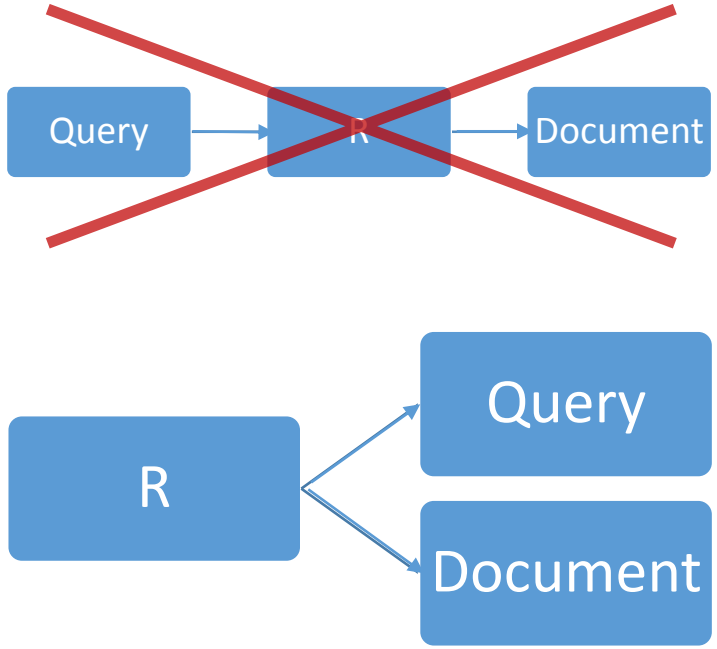
IR Method 1: Cosine Similarity

- Each document and the query is represented as a TF*IDF vector
  - $TF(t,d) = \log(f(t,d)+1)$
  - $IDF(t,D) = \log\left(\frac{N}{|d \in D : t \in d|}\right)$
  - Each dimension represents a term with $TF(t,d) * IDF(t,D)$
- We rank the documents by the cosine similarity of their TF*IDF vector to the query TF*IDF vector
  - $similarity(V,U) = \frac{V \cdot U}{|V| * |U|} = \frac{\sum_{i=1}^{n} V_i * U_i}{\sqrt{\sum_{i=1}^{n} V_i^2}\sqrt{\sum_{i=1}^{n} U_i^2}}$

Salton 1972

IR Method 2: Relevance Model

- Assume that both the query and the sentences are generated from a latent relevance model R and N is the model for non relevant document
- $Score(Doc) = \frac{p(Doc|R)}{p(Doc|N)} \sim \prod_{w \in Doc} \frac{p(w|R)}{p(w|N)}$
- R is estimated using documents relevant to the query (in our case high cosine similarity score)



Lavrenco et al., 2001

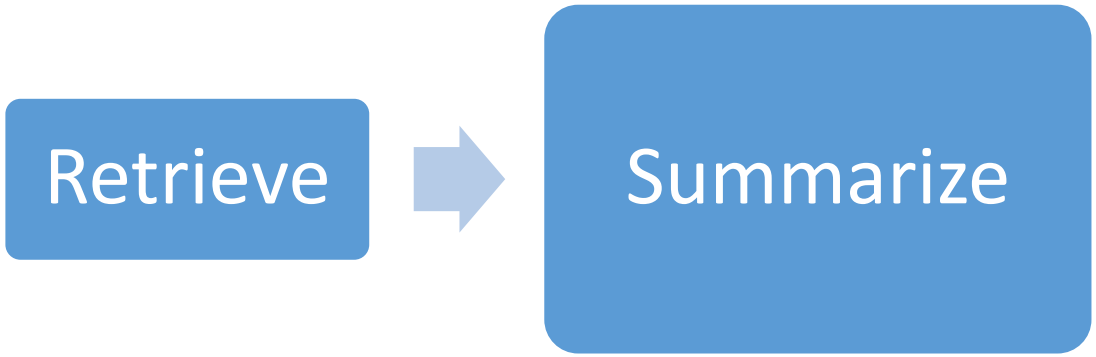
IR Method 3: Gold Baseline

- Use manual summaries as oracle
- Rank passages by their similarity to the manual summaries

Summarization Methods

KLSum

Biased LexRank



KLSum

- KLSum is a generic summarization method
- Objective: minimize the KL-divergence between the summary and document set N-gram distribution

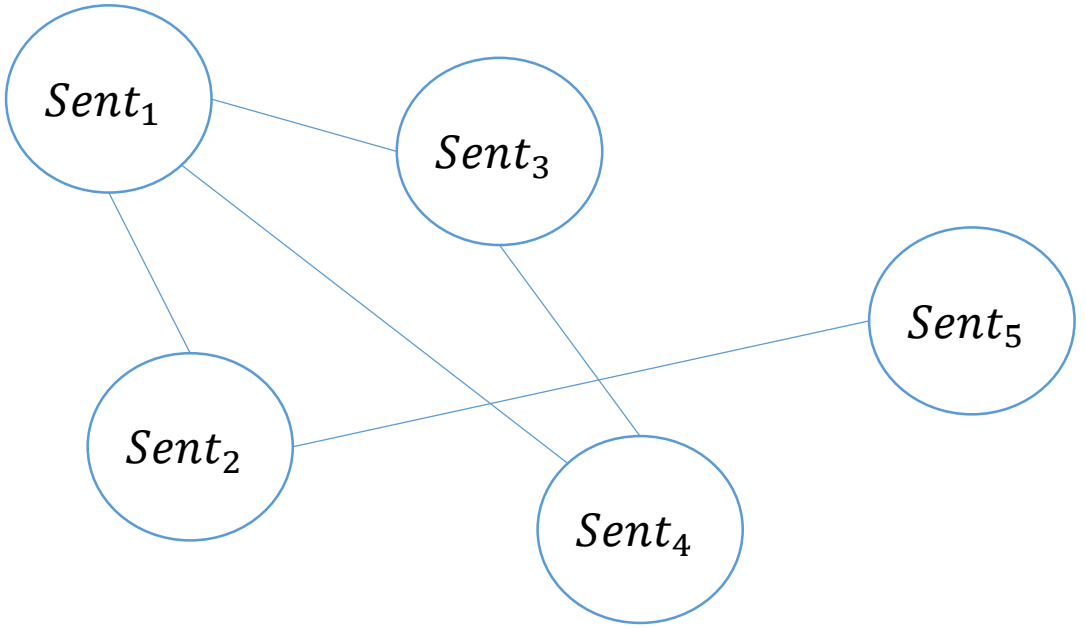
$$D_{KL}(P||Q) = \sum_i \ln\left(\frac{P(i)}{Q(i)}\right) P(i)$$

$$\underset{S \subseteq D}{\operatorname{argmin}}(D_{KL}(S||D))$$

Haghighi and Vanderwende 2009

LexRank

- LexRank: graph-based measure of **centrality**
  - Sentences represented as nodes
  - Pairwise sentence similarity → edge weight
  - PageRank algorithm identifies central nodes



Erkan and Radev 2004

Biased LexRank QFS Baseline

- A QFS variant of LexRank
- In Biased LexRank a specialized damping vector is used (instead of uniformly distributed vector) to give certain nodes higher ranking
- In QFS settings the damping of each node is determined by its similarity to query
- Integrates query similarity to each sentence node in the graph using a variant of “Relevance Model”
- Achieves state of the art results on QFS DUC

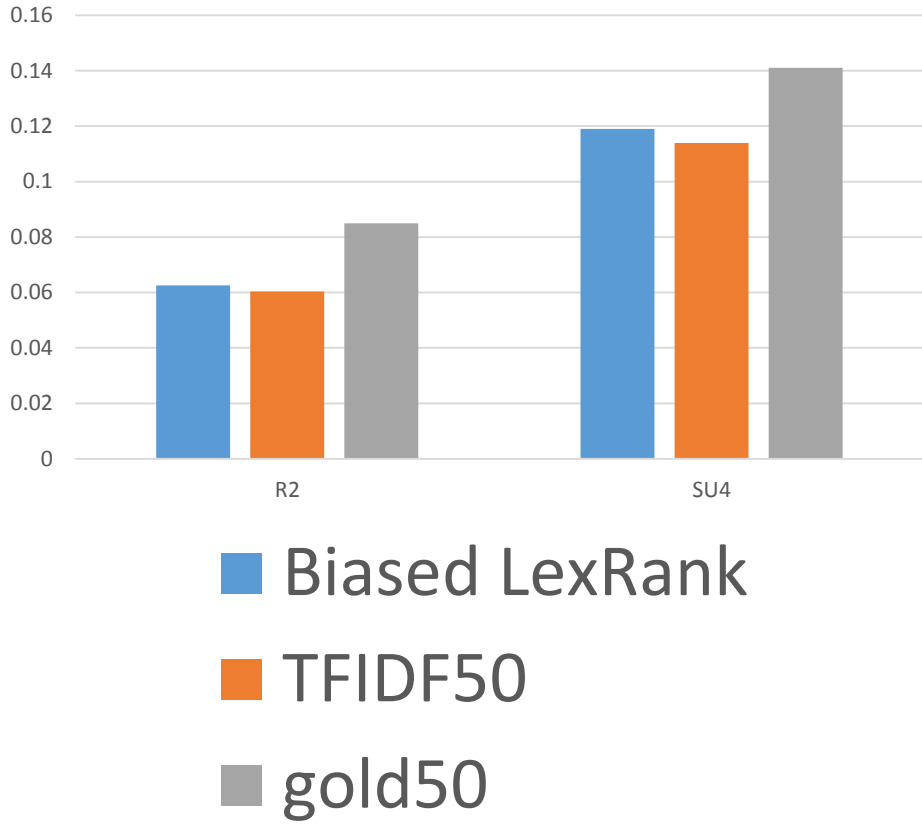
Otterbacher et al., 2009

Experimental Results

Dataset Comparison of Topic Concentration

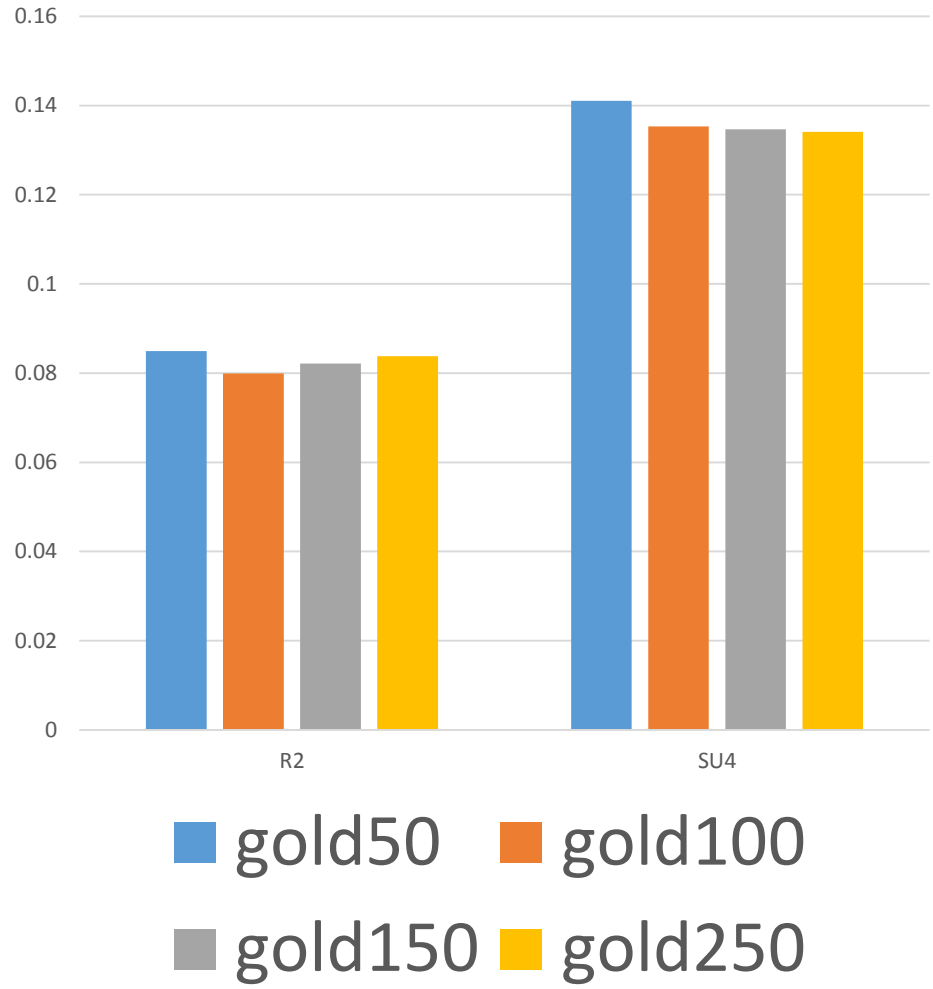
DUC 2005: R+S vs. Specialized QFS

- Rouge results R2/SU4
- QFS specialized Biased LexRank does not outperform Retrieve + Summarize (Generic KLSum)
- Gold retrieval improves performance only slightly (insignificantly)



DUC 2005: Effect of Retrieval Selectivity

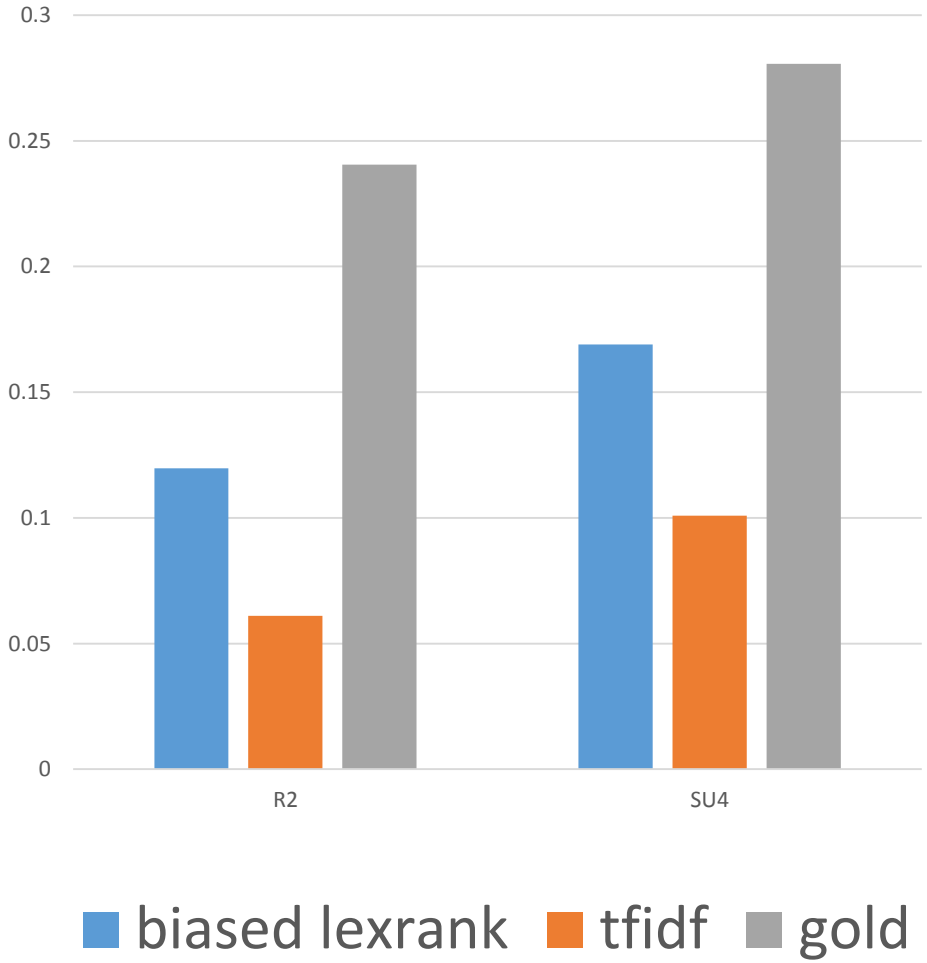
- Method: Gold Retrieve + Generic KLSum
- ROUGE performance not affected by number of passage retrieved!



QCFS: R+S vs. Specialized QFS

- QFS specialized Biased LexRank is significantly better than Retrieve + Summarize (Generic KLSum)
- Gold retrieval improves performance significantly.

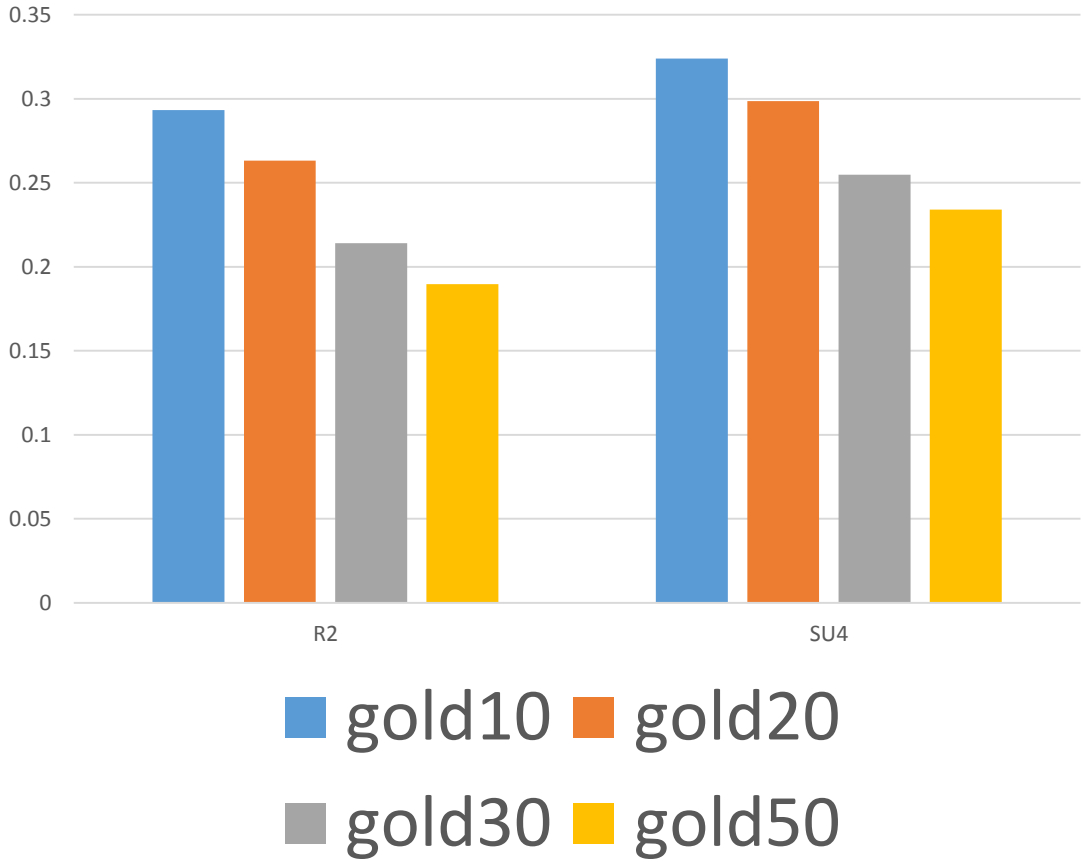
Gold Retrieval dominates



QCFS: Effect of Retrieval Selectivity

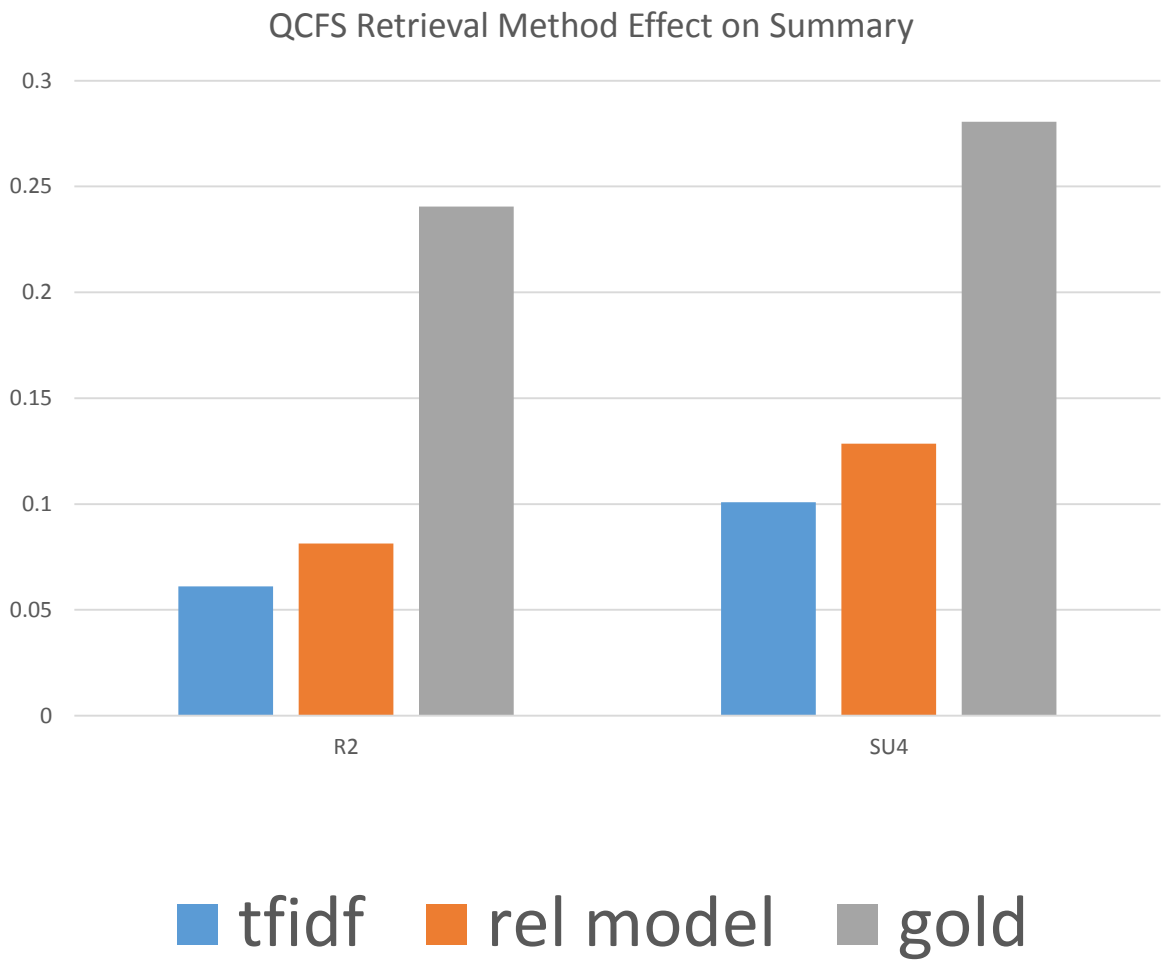
- The less selective retrieval, the worse ROUGE.

Retrieval precision affects summarization performance.



QCFS: Effect of Retrieval Method

- Compare different retrieval methods:
  - TF*IDF
  - Relevance model
  - Gold
- Retrieval method quality affects ROUGE score significantly



Retrieval Sensitive KLSum

Better retrieval yields better results

Improving Summarization With Retrieval

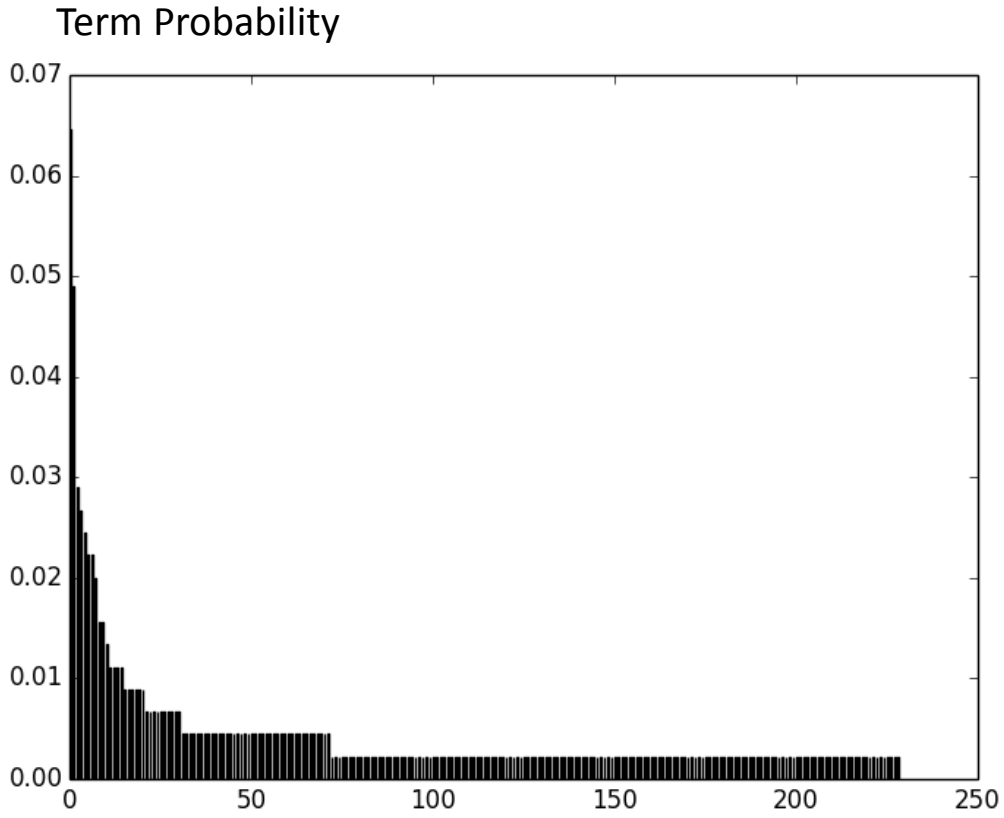
- We found that retrieval effects the summary quality
- We developed RelSum - a version of KLSum that is sensitive to relevance scores

RelSum: Retrieval sensitive KLSum

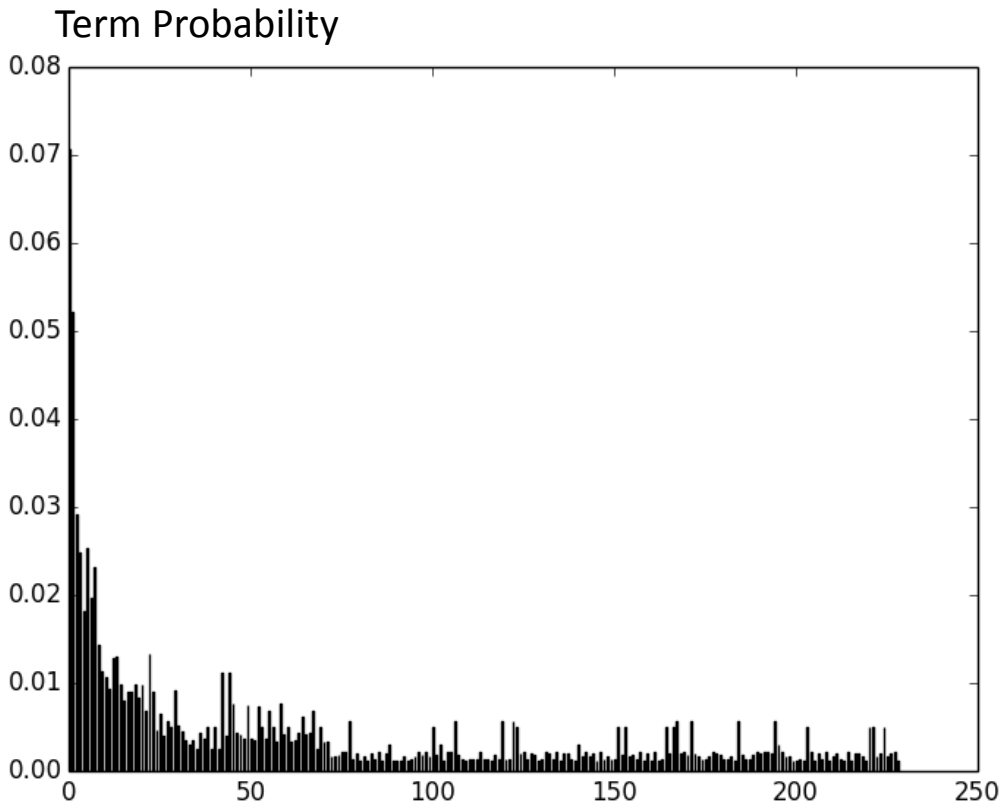
- We changed the language model of KLSum from N-gram distribution to a hierarchical model that incorporates document relevance to the query
  - First normalize relevance score to be a distribution
  - $P(w) = \sum_{d \in C} rel(d) * freq(w, d)$

Doc_1 Relevance Score	Doc_2 Relevance Score	Doc_3 Relevance Score
• w_1 freq • w_2 freq • …	• w_1 freq • w_2 freq • …	• w_1 freq • w_2 freq • …

Effects of the Hierarchical Model on Word Probability



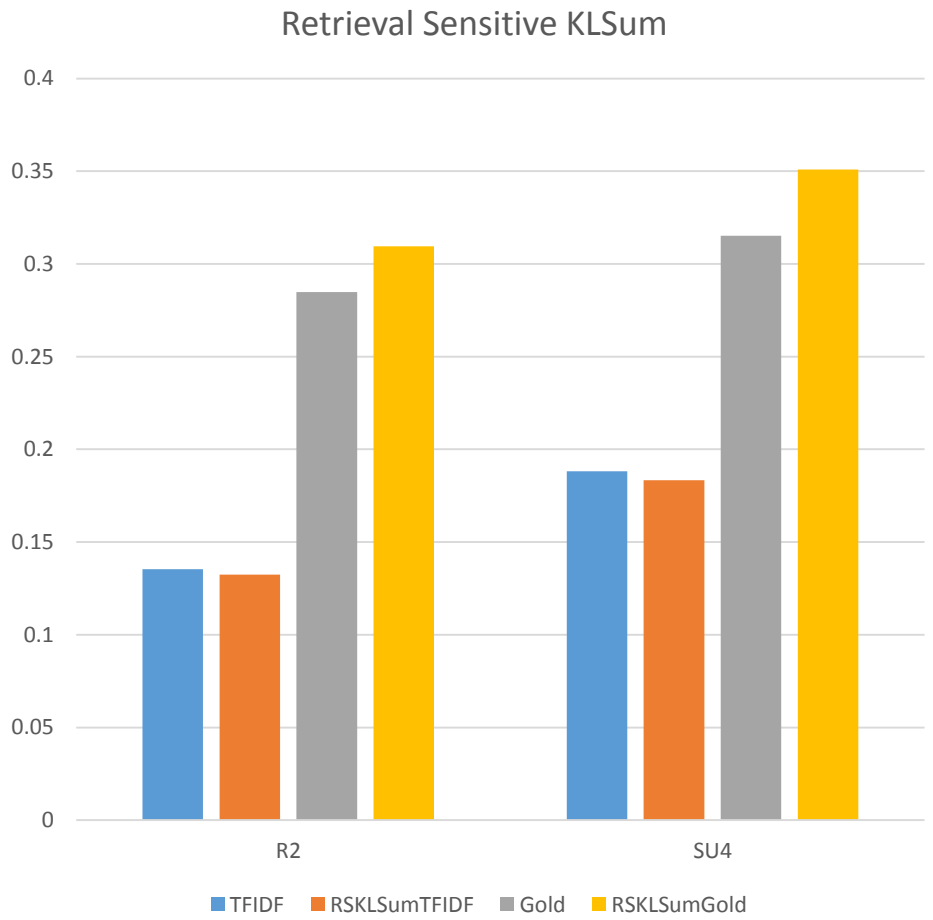
Unigram Distribution



Hierarchical Model Distribution

RelSum: Results on QCFS

- When using “bad” retrieval (TF*IDF + cosine similarity) our method performed similarly to (retrieve + KLSum).
- For gold retrieval method, RelSum performed significantly better than (retrieve + KLSum).



Conclusions

- DUC2005 is too topic concentrated to test QFS
- QCFS dataset can be used as a QFS dataset
- Achieve state-of-the-art QFS ROUGE scores using (Retrieval + Generic Multi-Document Summarization)
- Introduced RelSum that outperforms KLSum when given good retrieval

Questions?

Thank You!