Semantic Service Retrieval based on Natural Language Querying and Semantic Similarity

TECHNISCHE UNIVERSITÄT DARMSTADT

Richard Eckart de Castilho
Iryna Gurevych

Our Goal

TECHNISCHE UNIVERSITÄT DARMSTADT

Discover a web service offering a particular functionality

1. Treat technical **web-service descriptions** as **natural language artifacts**

2. Use **information retrieval** for **service discovery**

3. Use **semantic methods** to bridge the **vocabulary gap**

Examples

TECHNISCHE UNIVERSITÄT DARMSTADT

Queries

- *... service to get the author of the novel.*

Service Description

book person service

book person

person

person

book book
is titled title
has type book type
written by author

person

person person

Examples taken from the OWLS-TC 3rev1 benchmark

What is a Web Service?

TECHNISCHE UNIVERSITÄT DARMSTADT

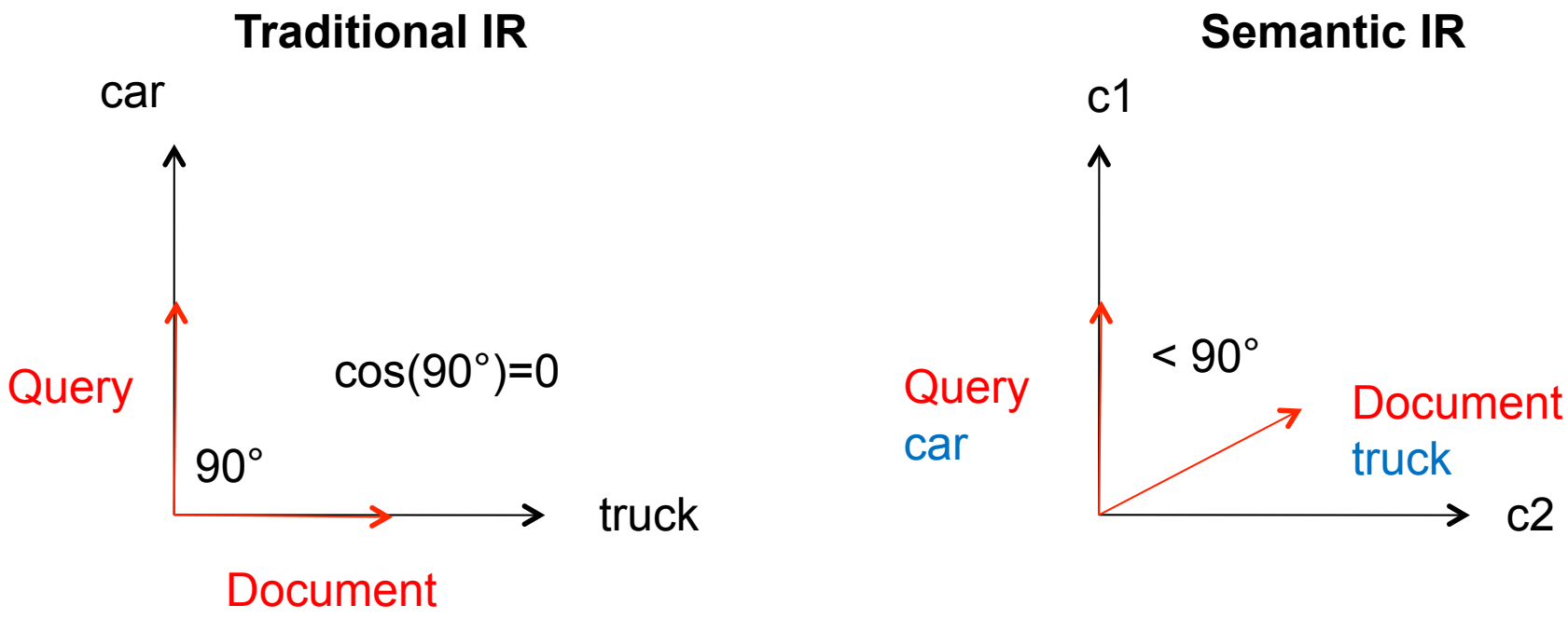
- Reusable and self-describing software component
- Supports remote method invocation
- Supports web standards
  - Description: e.g. Web Service Description Language (WSDL)
  - Invocation: e.g. REST or SOAP

Examples

- Currency converter
- Search engines
- Travel reservation systems
- Online shop
- Digital library
- Spam filter

What makes IR semantic?

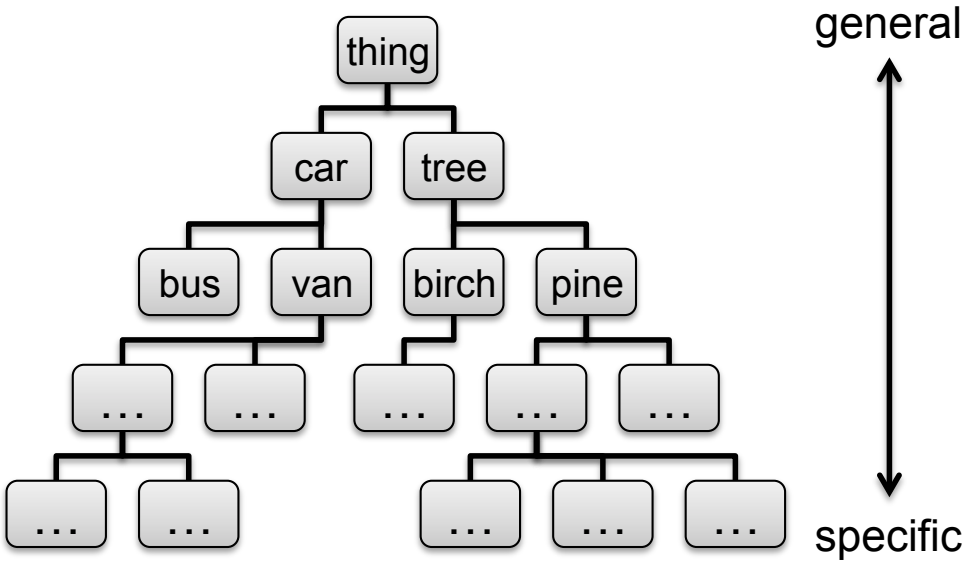
- Traditional IR: No match if query and document use different terms
- Semantic IR: match if query terms are similar to document terms



Knowledge-based Models

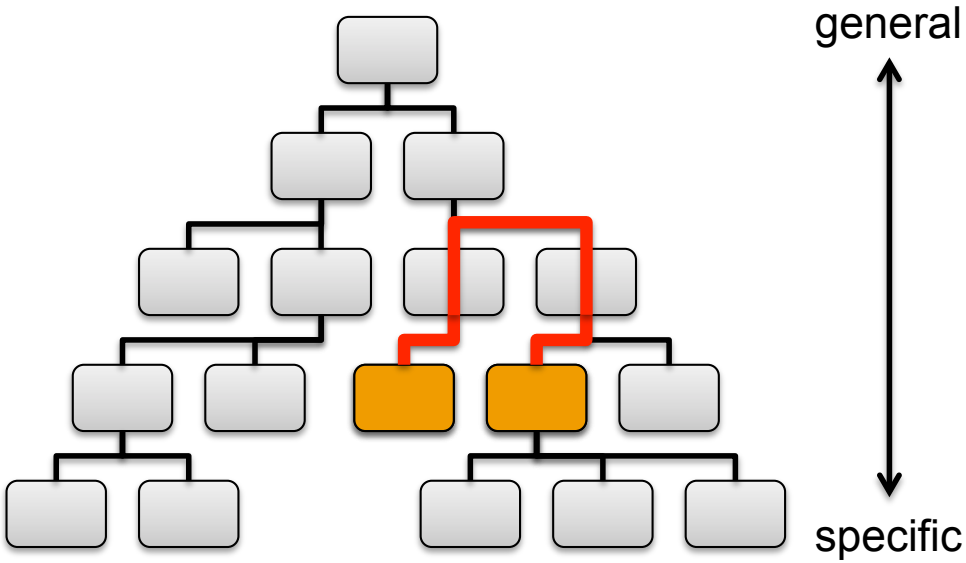
- Explicitly modeled knowledge
- Hypernym/hyponom relation

- Examples
  - WordNet
  - EuroWordNet
  - …



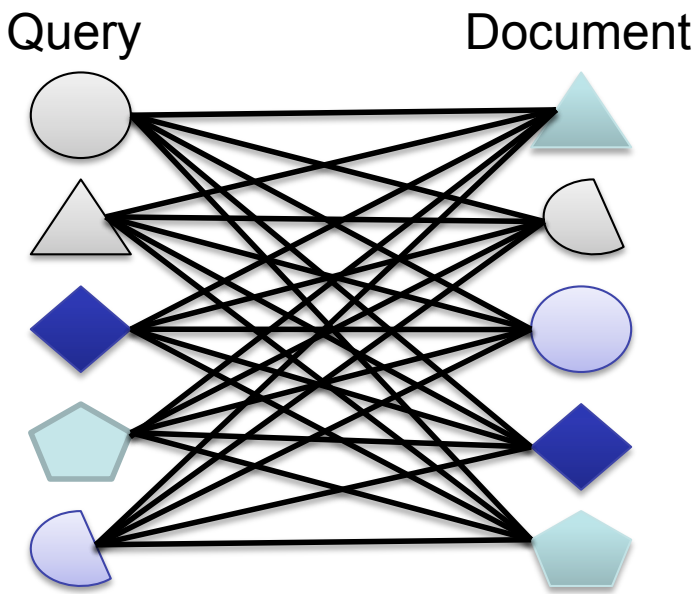
Semantic Similarity using Knowlegde-base

- Term similarity
- Calculate similarity using path between two concepts



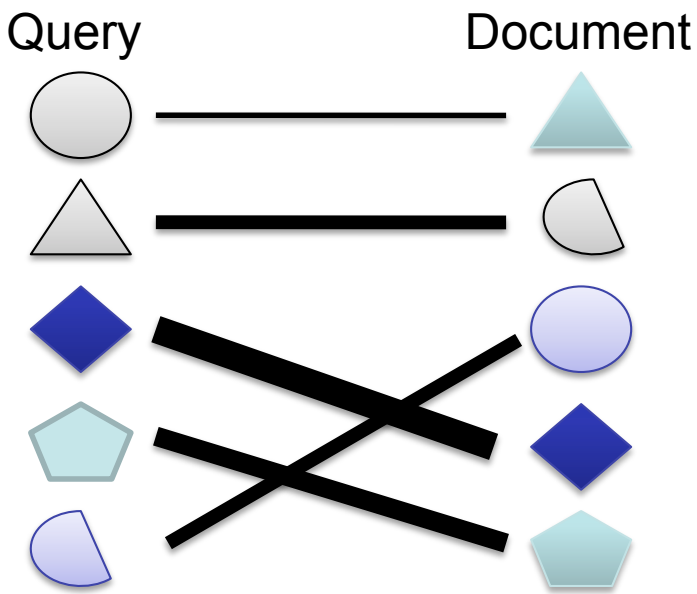
Aggregation of Term Similarities

- Two sets of terms A, B
- Calculate similarity for each pair of terms ($O(n^2)$)
- Various aggregation strategies
  - Max, average, median, geometric mean…
  - Bi-partite graph matching



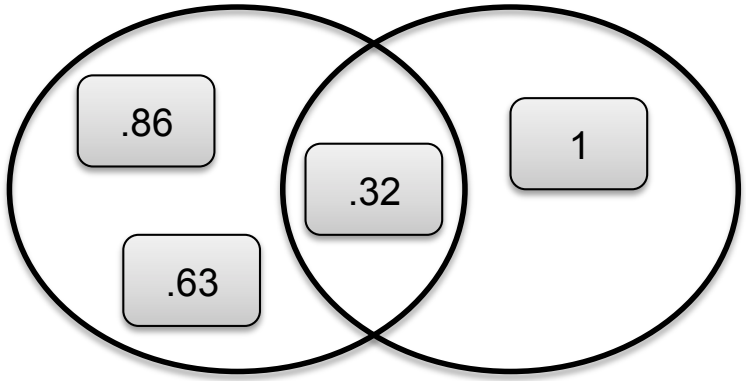
Weighted Bi-Partite Graph Matching

- Two sets of terms A, B
- Each element of A may be associated to at most one element of B
- Finds best semantic mapping between concepts in query and document



Knowledge-based Fuzzy Set Model

- Intention
  - Use a knowledge-base
  - Avoid calculating pair-wise term similarities
  - Pre-calculate semantic representation of documents
  - Create semantic representation of query on-the-fly

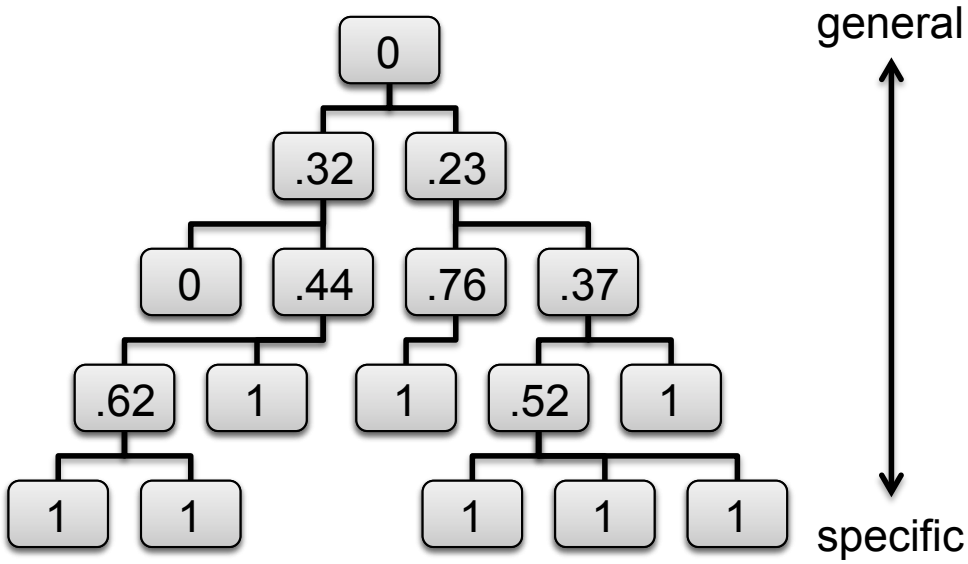


- Introducing: *Semantic Context*
  - Fuzzy set of terms
  - Start with a base set of terms
  - Assign weight 1.0 to each term
  - Add all hypernyms of the terms
  - Assign semantic relatedness between term and its hypernyms as weight
  - Compare query and document using set similarity measure (Dice)

Intrinsic Information Content

Seco et al., 2004

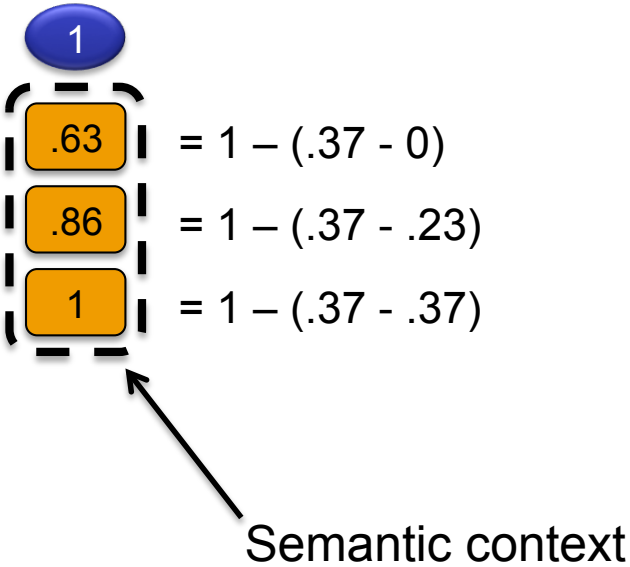
- Information content (IC)
  - "Specificity" of a concept
- Intrinsic information content (IIC)
  - Number of hyponyms compared to size of knowledge base

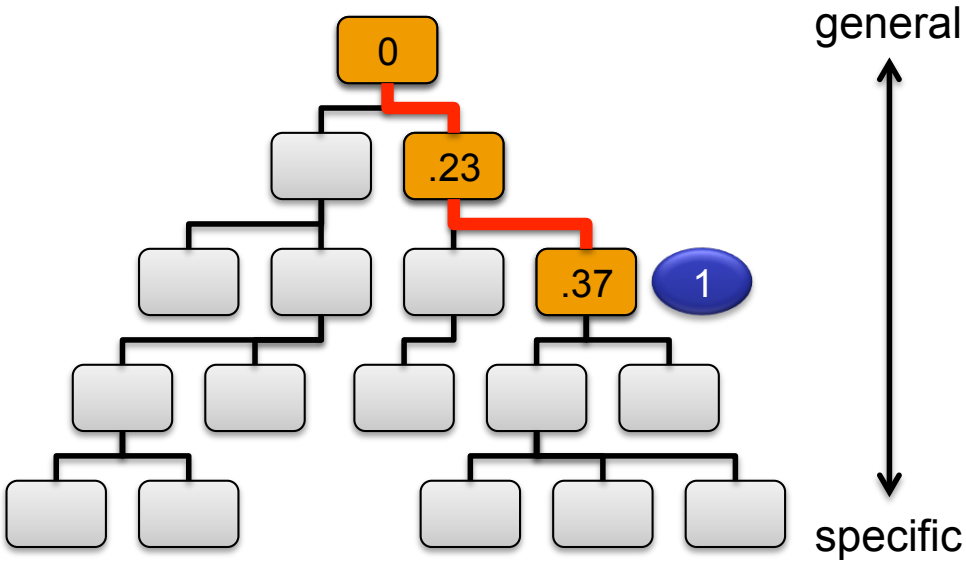


|K| = 17

$$IIC(c) = 1 - \frac{\log\left(\left|h(c)\right| + 1\right)}{\log\left|K\right|}$$

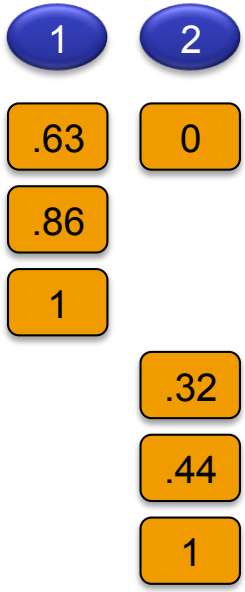
Semantic Context of a Term

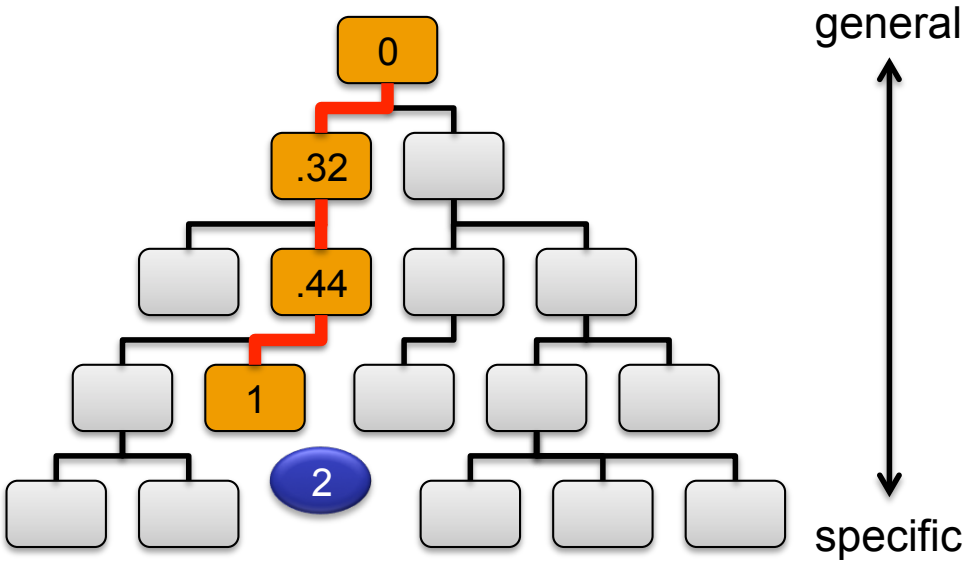




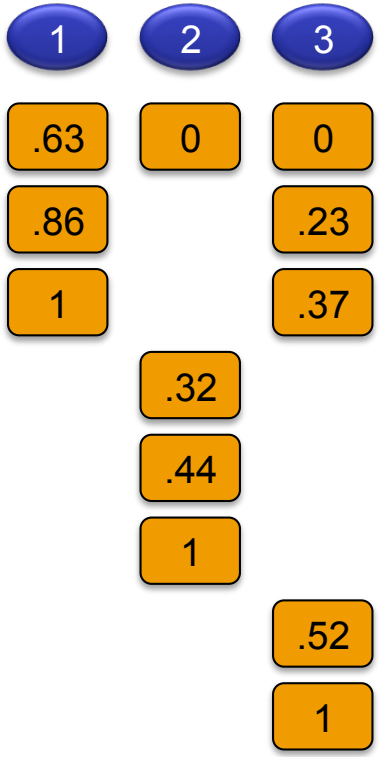
- Hypernyms used to create the semantic context of a term
- Information content determines relatedness between term and hypernym

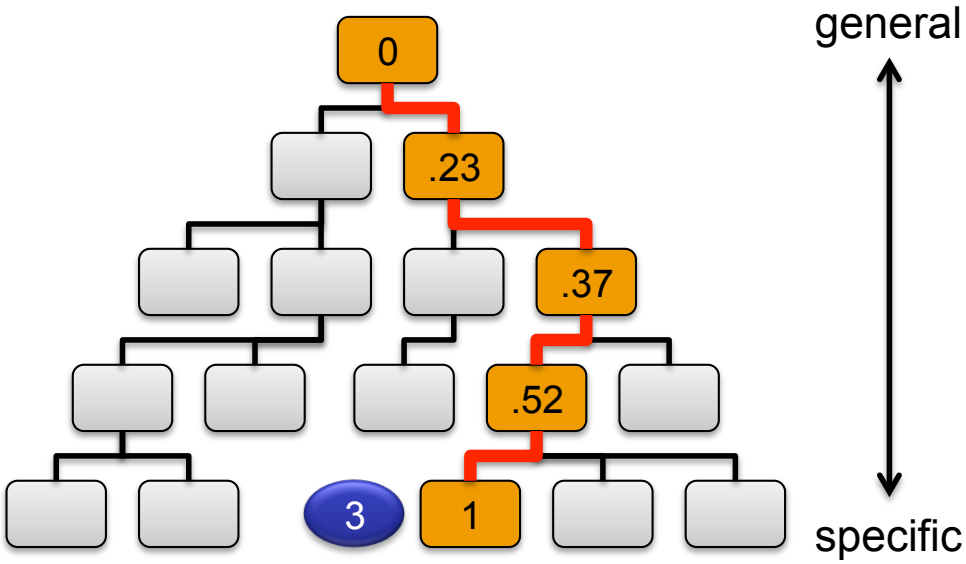
Semantic Context of a Set of Terms





Semantic Context of a Set of Terms

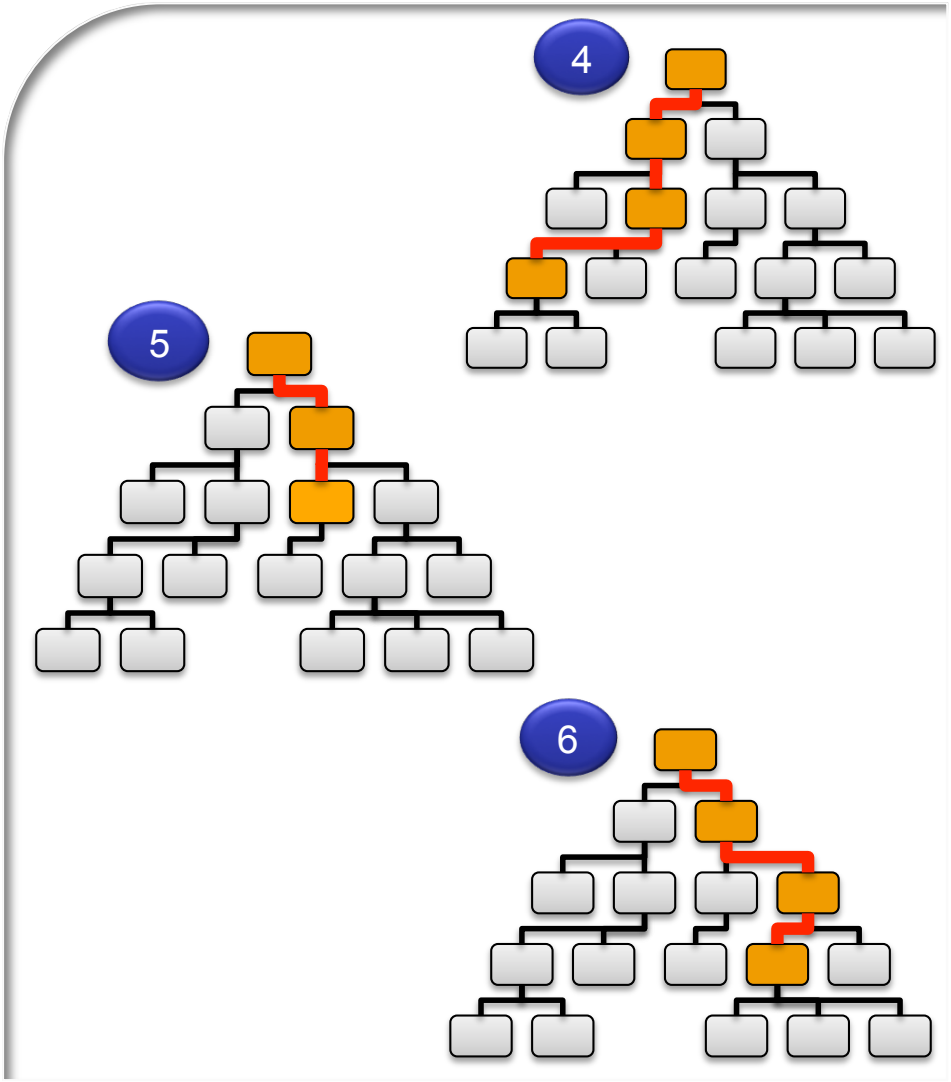




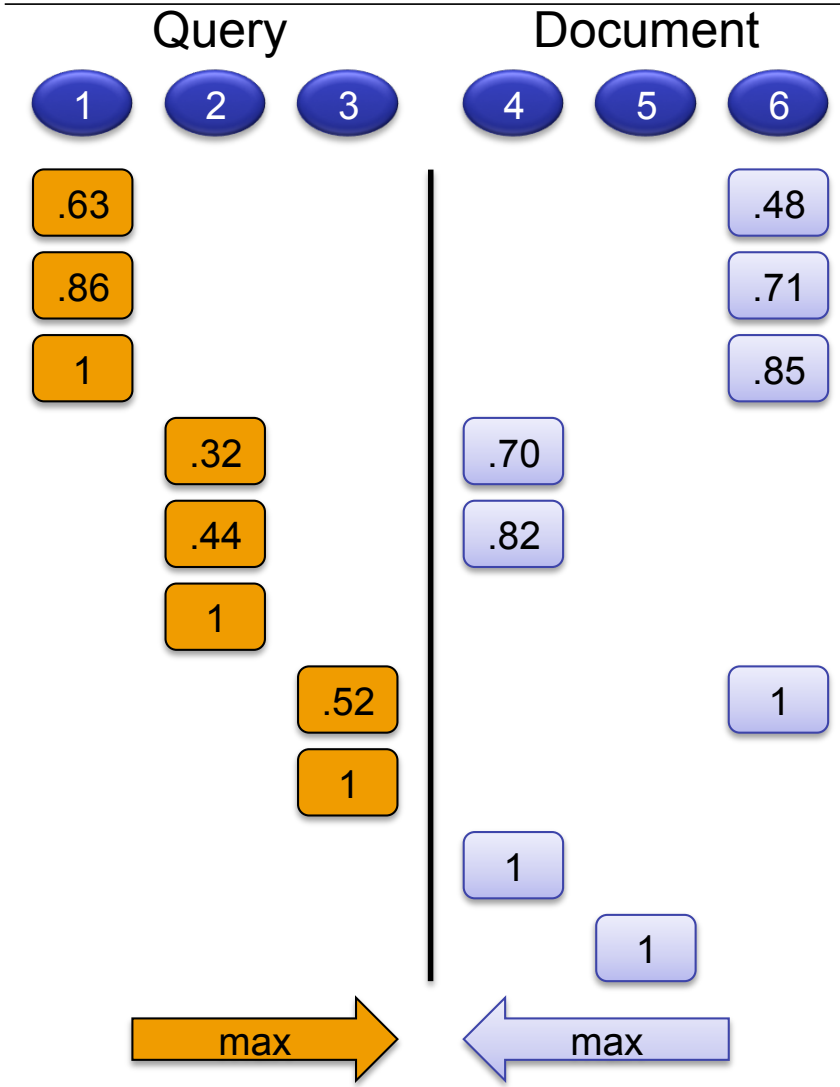
Semantic Context of two Sets of Terms

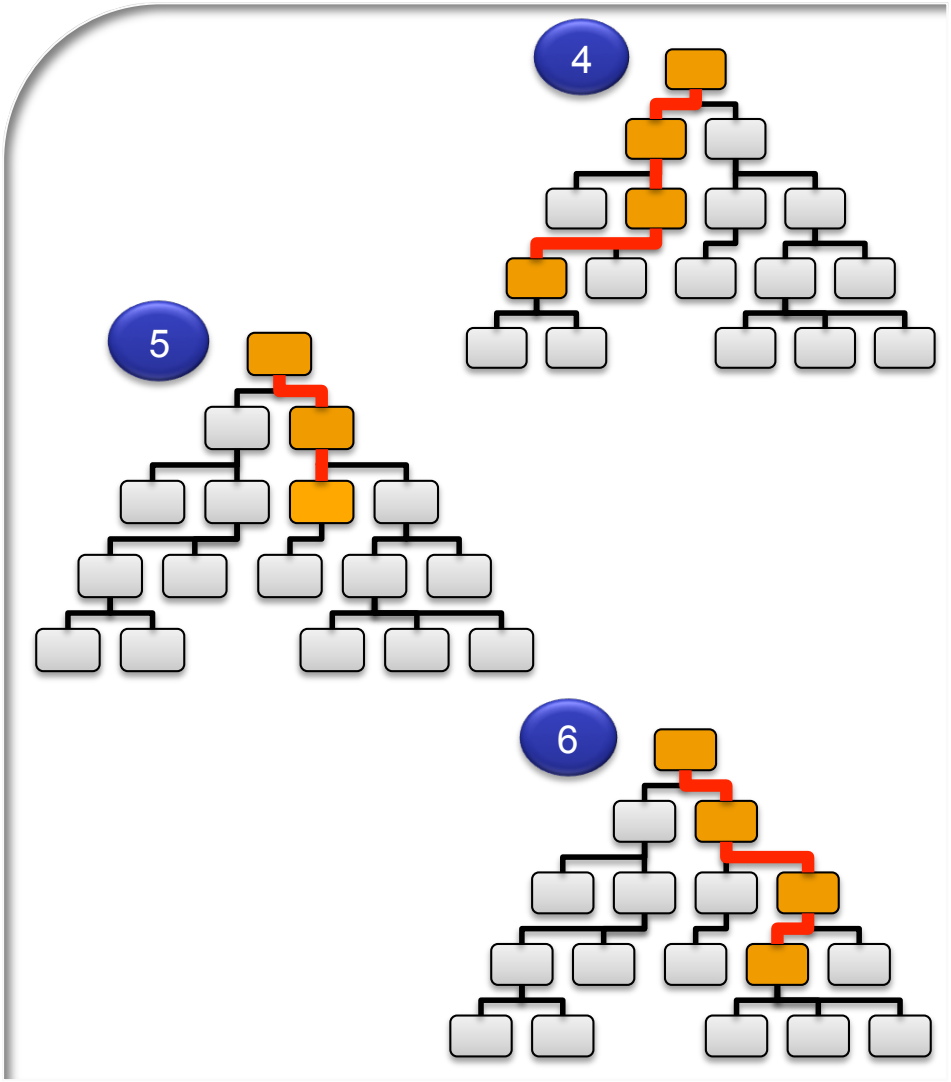
TECHNISCHE UNIVERSITÄT DARMSTADT

Query			Document		
1	2	3	4	5	6
.63	0	0	.38	.24	.48
.86		.23		.47	.71
1		.37			.85
	.32		.70		
	.44		.82		
	1				
		.52			1
		1			
			1		
				1	
max			max		

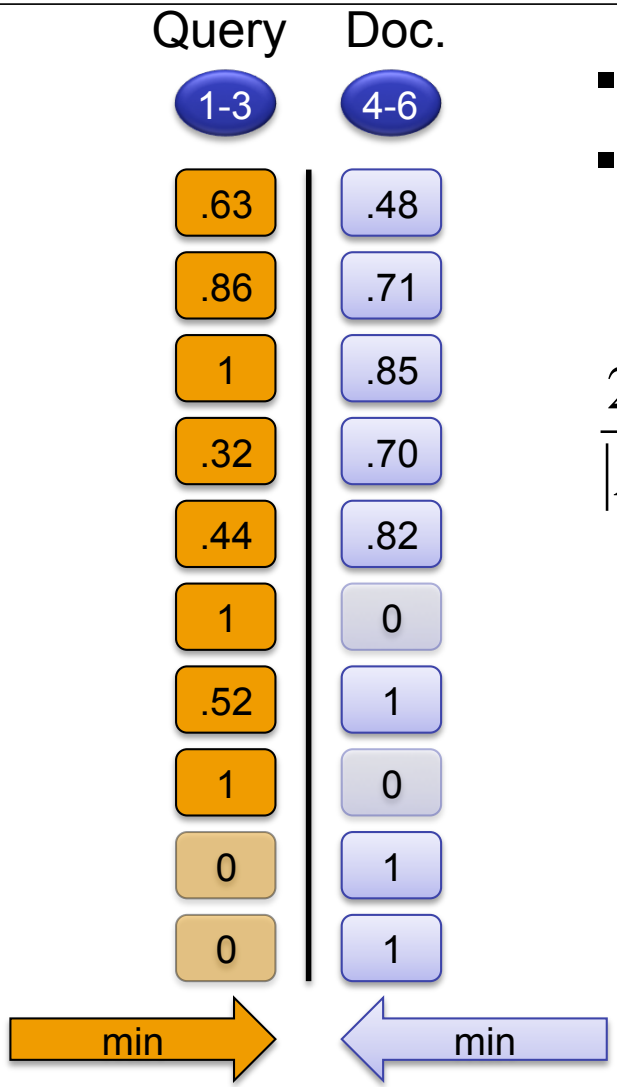


Merging the Semantic Contexts





Calculating Similarity of Semantic Contexts



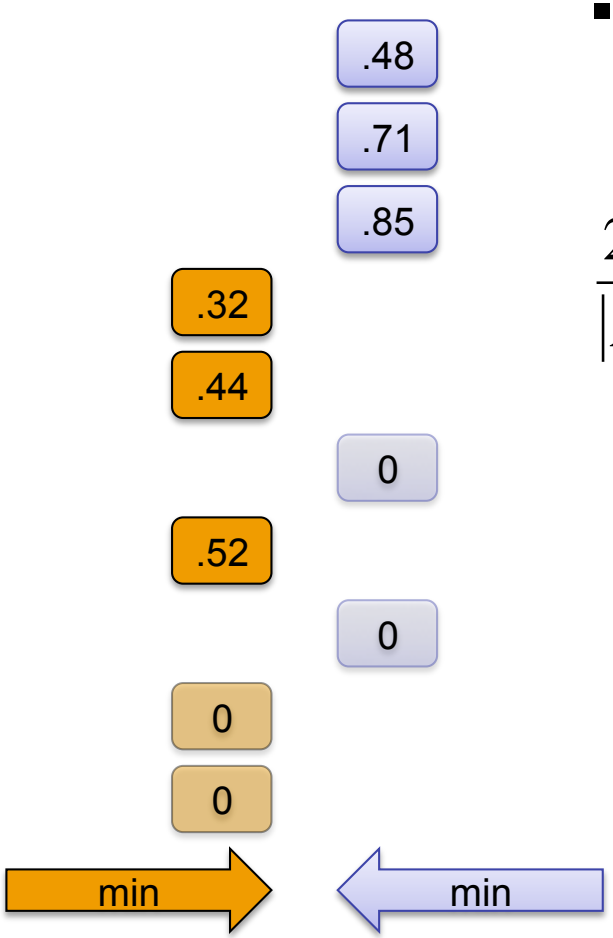
- Dice coefficient to calculate similarity
- Dice generalizes to fuzzy sets
  - Intersection operator -> *min,* |A| -> sum

$$\frac{2|(A \cap B)|(x)}{|A|(x) + |B|(x)} = \frac{2\sum \min[A(x), B(x)]}{\sum A(x) + \sum B(x)}$$

1-3 $\sum A(x) = 5.77$

4-6 $\sum B(x) = 6.56$

Calculating Similarity of Semantic Contexts



- Dice coefficient to calculate similarity
- Dice generalizes to fuzzy sets
  - Intersection operator -> *min,* |A| -> sum

$$\frac{2|(A \cap B)|(x)}{|A|(x) + |B|(x)} = \frac{2\sum \min[A(x), B(x)]}{\sum A(x) + \sum B(x)}$$

1-3 $\sum A(x) = 5.77$

4-6 $\sum B(x) = 6.56$

$$2\sum[A(x), B(X)] = 6.64$$

$$\frac{6.64}{5.77 + 6.56} = 0.53$$

similarity score

Evaluated Models

- Baseline (stemmed)
  - Apache Lucene
- Path-based models (lemmatized)
  - Resnik (1995)
  - Jiang & Conrath (1997)
  - Lin (1998)
- Statistical (lemmatized)
  - Explicit Semantic Analysis (ESA) by Gabrilovich & Markovitch (2007)
  - Latent Semantic Analysis (LSA) by Landauer et al. (1998)

Evaluation Data Set

TECHNISCHE UNIVERSITÄT DARMSTADT

- OWLS-TC 3rev1 benchmark for semantic service retrieval
  http://www.semwebcentral.org/frs/?group id=89
  - 29 queries
  - 999 service descriptions (WSDL)
  - 3584 judgements
  - Used in Semantic Service Selection Contest

- Semantic resources
  - WordNet for Path-based models
  - WordNet for Fuzzy Set Model
  - Wiktionary for ESA
  - LSA index created on evaluation data set itself

- Evaluated on Mean Average Precision (MAP)

Term Similarity using Weighted Bi-Partite Graph Matching for Aggregation

TECHNISCHE UNIVERSITÄT DARMSTADT

Model	inner product	MAP
Lucene (baseline)	Cosine	.6558
Jiang & Conrath	-	.4477
Lin	-	.4772
Resnik	-	.5314
Fuzzy Set Model	Dice	.6372
ESA	Avg. Prod.	.6515
LSA	Cosine	.6815

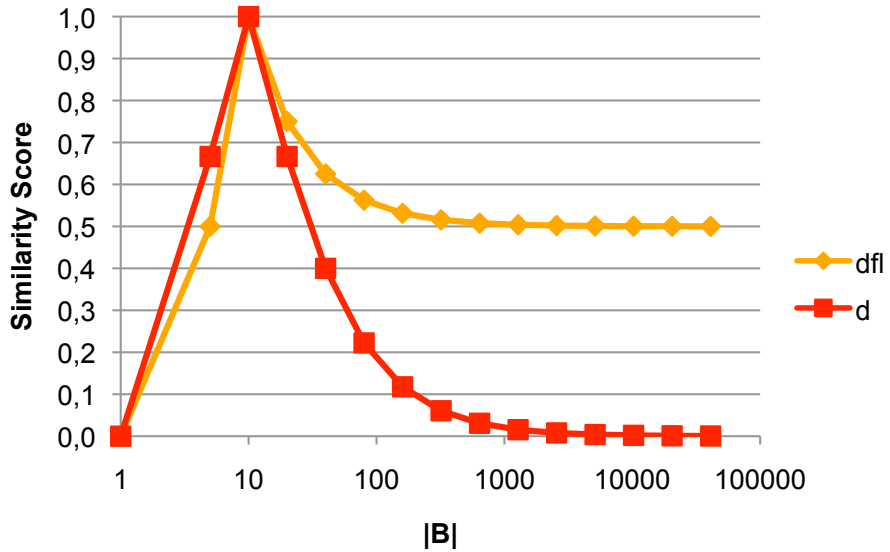
Modified Dice Coefficient

- Document is good match when all query terms are found
- Document is bad match when some query items are missed
- Dice heavily penalizes if document contains more terms than query

- Modified dice d_{fl}
  - Penalty if document does not contain all query terms
  - Never drops below 0.5 if document contains all query terms

$$c = \frac{|A|(x)}{\max\left(|A|(x),|B|(x)\right)}$$

$$dfl\left(A(x),B(x)\right) = \frac{(1+c)|(A \cap B)|(x)}{2|A|(x)}$$



Similarity score for |A|=10 and |A ∩ B| = min(|A|,|B|)

Modified Dice Coefficient

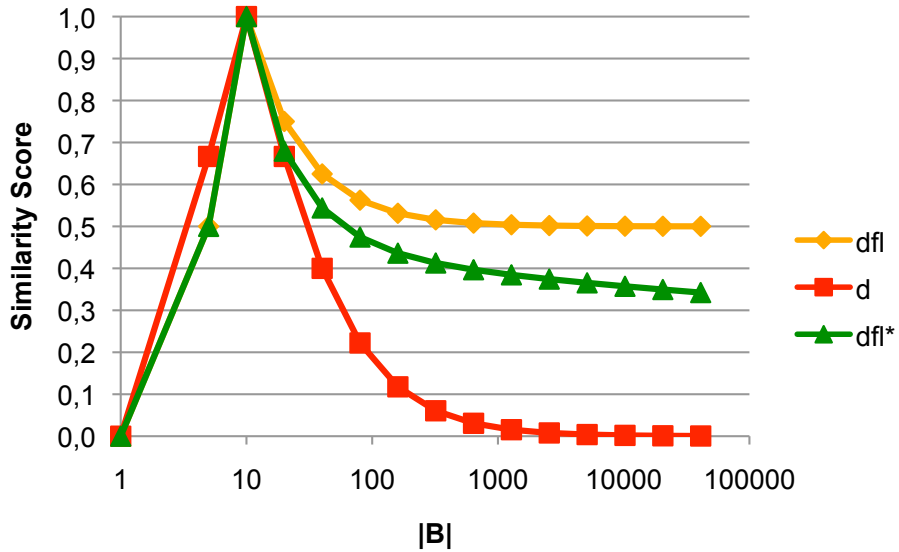
- Document is good match when all query terms are found
- Document is bad match when some query items are missed
- Dice heavily penalizes if document contains more terms than query

- Modified dice d_{fl}*
  - Penalty if document does not contain all query terms
  - Score drops slowly with document length if all query terms are present

$$m = \max\left(|A|(x), |B|(x)\right)$$

$$c = \frac{|A|(x)}{m}$$

$$dfl^*\left(A(x), B(x)\right) = \frac{(1+c)|(A \cap B)|(x)}{2\left(|A|(x) + \log(m - |A|(x) + 1)\right)}$$



Similarity score for |A|=10 and |A ∩ B| = min(|A|,|B|)

Term Set Similarity using Vector/Set Aggregation

TECHNISCHE UNIVERSITÄT DARMSTADT

Model	inner product	MAP
Lucene (baseline)	Cosine	.6558
ESA	Avg. Prod.	.5579
Fuzzy Set Model	Dice	.6037
LSA	Cosine	.6822

Combination of Semantic and Non-Semantic Models

TECHNISCHE UNIVERSITÄT DARMSTADT

- Use non-semantic model (s_{bow}) in addition to semantic model (s_{sem})
  - Apache Lucene
- Calculate final score using linear combination

$$w \times s_{sem}(Q,D) + (1 - w) \times s_{bow}(Q,D)$$

- Parameter *w* controls balance between semantic and non-semantic

Term Set Similarity Combined with Non-Semantic Model

Model	inner product	MAP	w
Lucene (baseline)	Cosine	.6558	-
ESA	Avg. Prod.	.6884	.8
LSA	Cosine	.7012	.4
Fuzzy Set Model	d_{fl}^*	.7033	.4

Conclusion

- Successful queried technical service descriptions using natural language
- Vocabulary gap addressed using semantic models

- Fuzzy set model
  - Improved knowledge-based model
  - Semantic document representations can be pre-calculated
  - Outperformed path-based models
  - In combination with Lucene on par with LSA

- Fuzzy set model seems a good option if a knowledge-base is available

Outlook

- Fuzzy set model yields good results on this data set

- Re-run the experiments
  - With a different data set
  - With additional knowledge bases (e.g. ontologies)
  - Use a separate training set to determine w
  - More meaningful combination of knowledge-base and TF/IDF statistics

TECHNISCHE UNIVERSITÄT DARMSTADT

Thank you!

Questions?

Thanks to all my colleagues at UKP!

Federal Ministry of Economics and Technology

THESEUS
Neue Technologien für das Internet der Dienste