CONICET AGENCIA

Tuning Topical Queries through Context Vocabulary Enrichment: a Corpus-Based Approach

Carlos M Lorenzetti

cml@cs.uns.edu.ar

Ana G Maguitman

agm@cs.uns.edu.ar

Grupo de Investigación en Recuperación de Información y Gestión del Conocimiento

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial

Universidad Nacional del Sur
Av. L.N. Alem 1253
Bahía Blanca - Argentina

Introduction

Context–Based Search

Context–Based Search

Google

YAHOO!

Computers

Entertainment

Flora

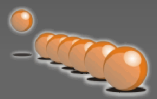
Animals

BRITISH AMERICAN TOBACCO

Consumables

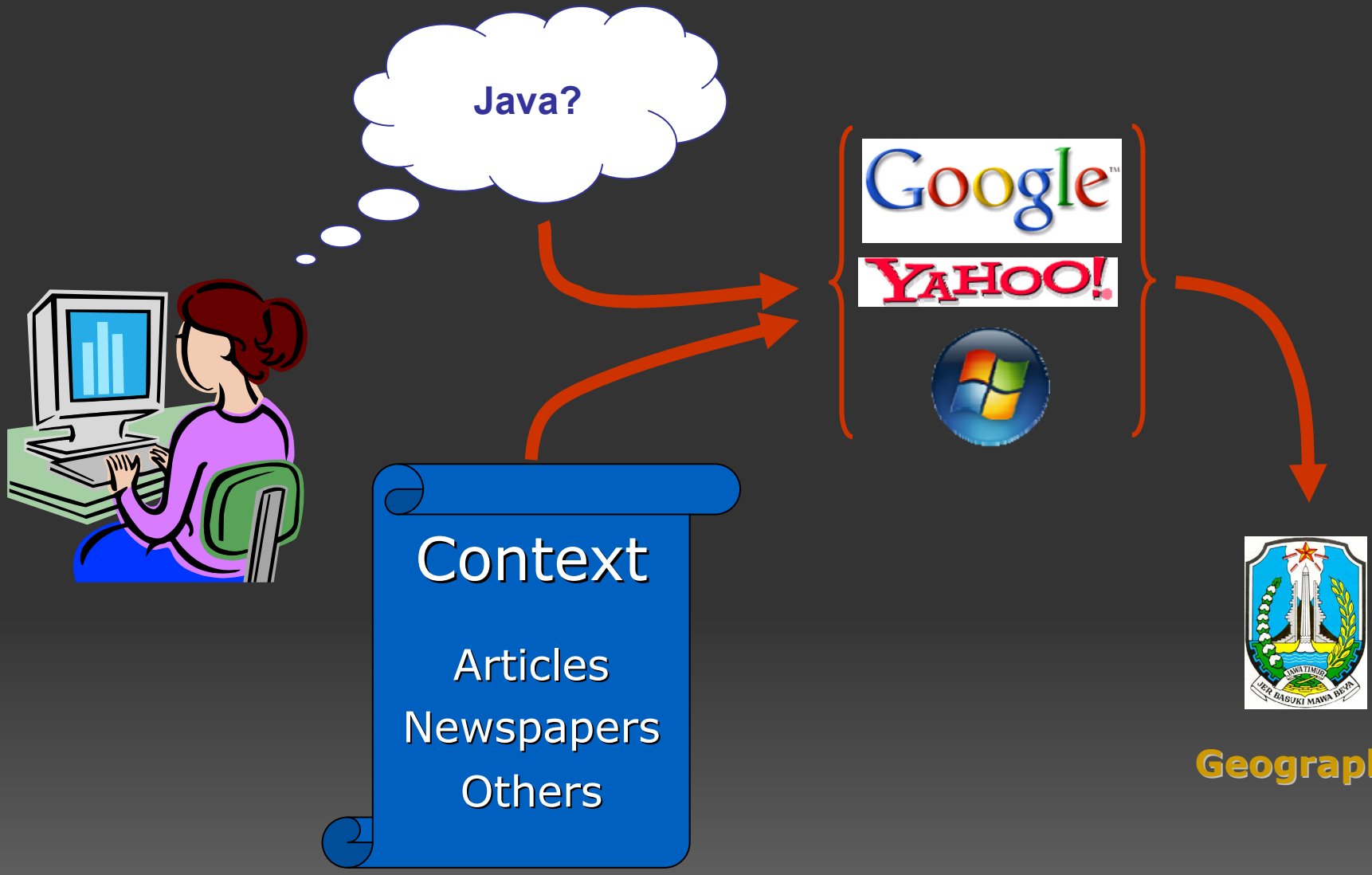
Geography

Ships



Context–Based Search

Context–Based Search



Query tuning

- Step 1: Query
- Step 2: Initial set of results
- Step 3: Relevance assessment
  - Supervised feedback
  - Unsupervised feedback
  - Semi-supervised feedback
- Step 4: Better representation
- Step 5: Revised set of results

Different Role of Terms

- *Descriptors*

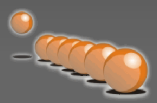
  Terms that appear often in documents related to the given topic

  What is this topic about?

- Discriminators

  Terms that appear only in documents related to the given topic

  What terms are useful to seek similar information?



Different Role of Terms

- *Descriptors*

  Terms that appear often in documents related to the given topic

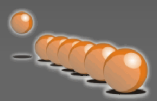
  What is this topic about?

Recall

- Discriminators

  Terms that appear only in documents related to the given topic

  What terms are useful to seek similar information?



Different Role of Terms

- *Descriptors*

  Terms that appear often in documents related to the given topic

  What is this topic about?

  → Recall

- Discriminators

  Terms that appear only in documents related to the given topic

  What terms are useful to seek similar information?

  → Precision

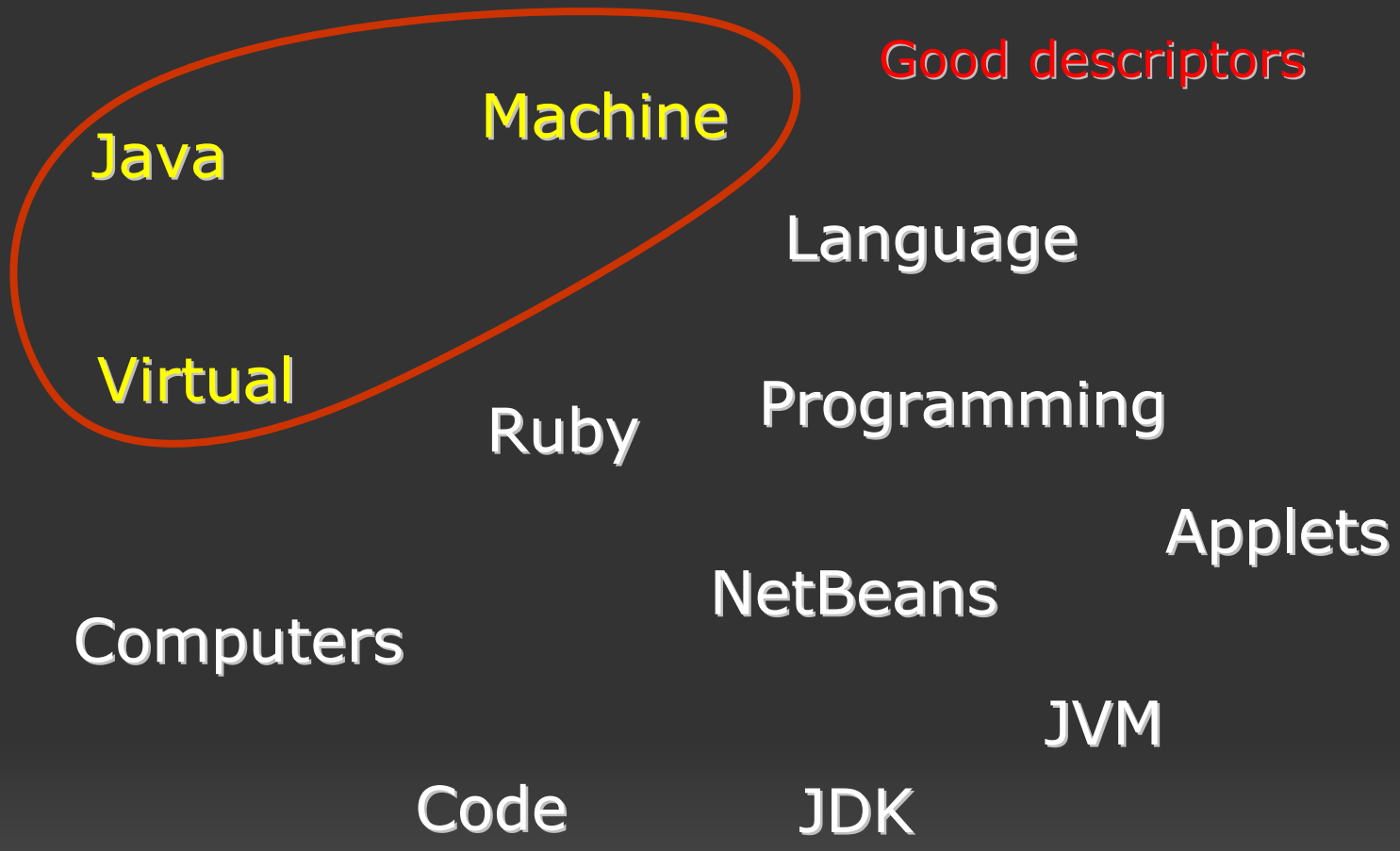
Descriptors and Discriminators Computation: an example

Descriptors and Discriminators

Java
Machine
Language
Virtual
Ruby
Programming
Applets
NetBeans
Computers
JVM
Code
JDK

Topic: Java Virtual Machine

Descriptors and Discriminators



Descriptors and Discriminators

Java

Machine

Language

Virtual

Ruby

Programming

Applets

NetBeans

Computers

JVM

JDK

Good discriminators

Code

Topic: Java Virtual Machine

Documents Descriptors and Discriminators

			H			
			(1)	(2)	(3)	(4)
Initial Context	java	4	2	5	5	2
	machine	2	6	3	2	0
	virtual	1	0	1	1	0
	language	1	0	2	1	1
	programming	3	0	2	2	0
	coffee	0	3	0	0	3
	island	0	4	0	0	2
	province	0	4	0	0	1
	jvm	0	0	2	1	0
	jdk	0	0	3	3	0

Topic: Java Virtual Machine

(1) espressotec.com
(2) netbeans.org
(3) sun.com
(4) wikitravel.org

$$\mathbf{H}[d_i, t_j] = k$$

Number of occurrences of term *j* in document *i*

Documents *Descriptors*

Initial Context		$\lambda(d_0, t_j)$
java	4	0,718
machine	2	0,359
virtual	1	0,180
language	1	0,180
programming	3	0,539
coffee	0	0,000
island	0	0,000
province	0	0,000
jvm	0	0,000
jdk	0	0,000

Topic: Java Virtual Machine

Descriptive power of a term in a document

$$\lambda(d_i, t_j) = \frac{\mathbf{H}[i, j]}{\sqrt{\sum_{k=0}^{n-1} (\mathbf{H}[i, k])^2}}$$

Documents *Discriminators*

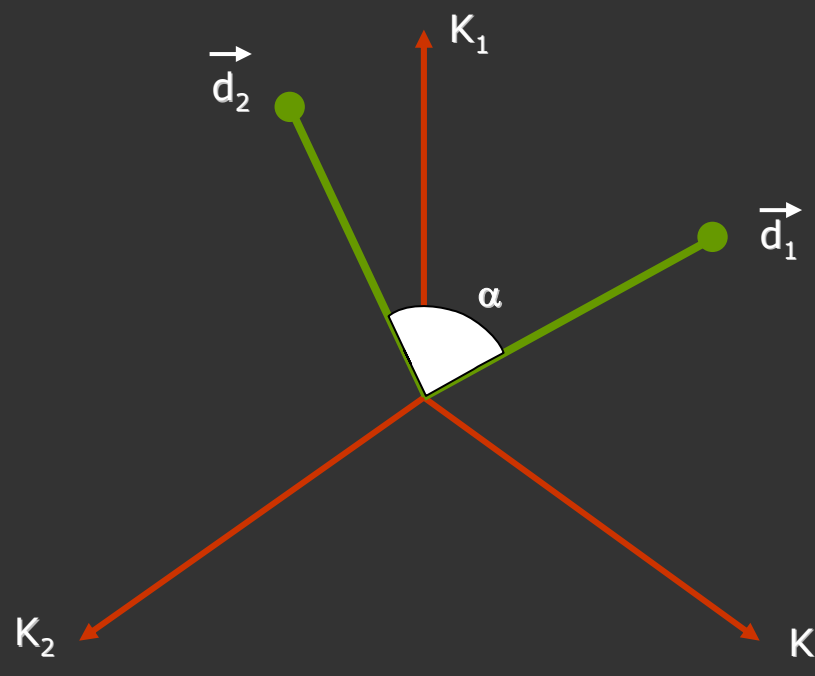
Initial Context		$\delta(t_i, d_0)$
java	4	0,447
machine	2	0,500
virtual	1	0,577
language	1	0,500
programming	3	0,577
coffee	0	0,000
island	0	0,000
province	0	0,000
jvm	0	0,000
jdk	0	0,000

Topic: Java Virtual Machine

Discriminating power of a term in a document

$$\delta(t_i, d_j) = \frac{s(\mathbf{H}^T[i,j])}{\sqrt{\sum_{k=0}^{m-1} s(\mathbf{H}^T[i,k])}}$$

Documents comparison criteria



Cosine similarity

$$\sigma(d_i, d_j) = \sum_{k=0}^{n-1} (\lambda(d_i, t_k) \cdot \lambda(d_j, t_k))$$

Documents similarity

Topics Descriptors

Initial Context		$\Lambda(d_0, t_j)$
java	4	**0,385**
machine	2	**0,158**
virtual	1	0,124
language	1	0,089
programming	3	0,064
coffee	0	0,055
island	0	0,040
province	0	0,040
jvm	0	0,032
jdk	0	0,014

Topic: Java Virtual Machine

Term descriptive power in a topic of a document

$$\Lambda(d_i, t_j) = \frac{\sum_{k=0,\ k\neq i}^{m-1} (\sigma(d_i, d_k) \cdot \lambda(d_k, t_j)^2)}{\sum_{k=0,\ k\neq i}^{m-1} \sigma(d_i, d_k)}$$

Topics Discriminators

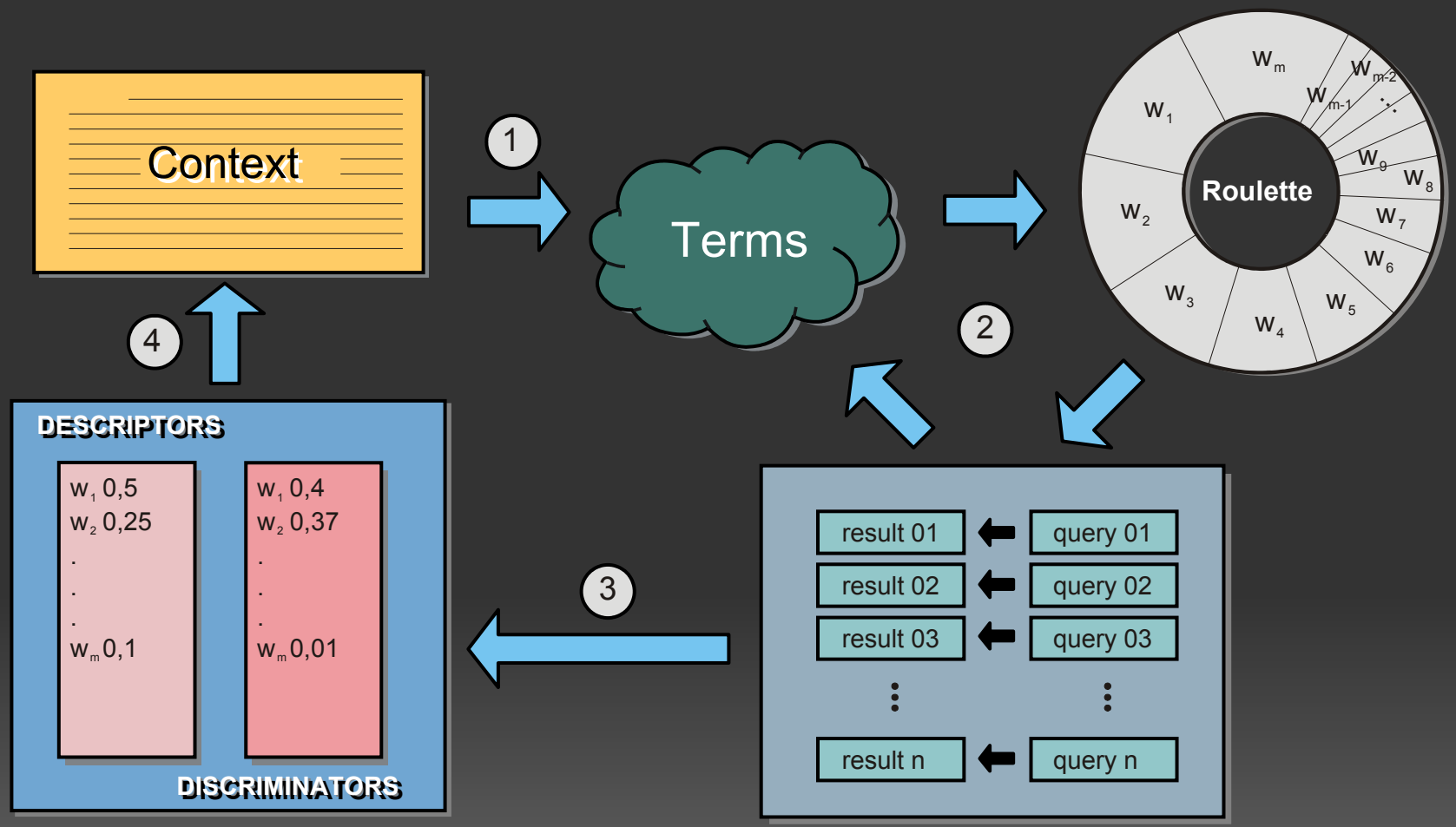
Initial Context		$\Delta(t_i, d_0)$
jvm	0	**0,848**
jdk	0	**0,848**
virtual	1	0,566
programming	3	0,566
machine	2	0,524
language	1	0,517
java	4	0,493
coffee	0	0,385
island	0	0,385
province	0	0,385

Topic: Java Virtual Machine

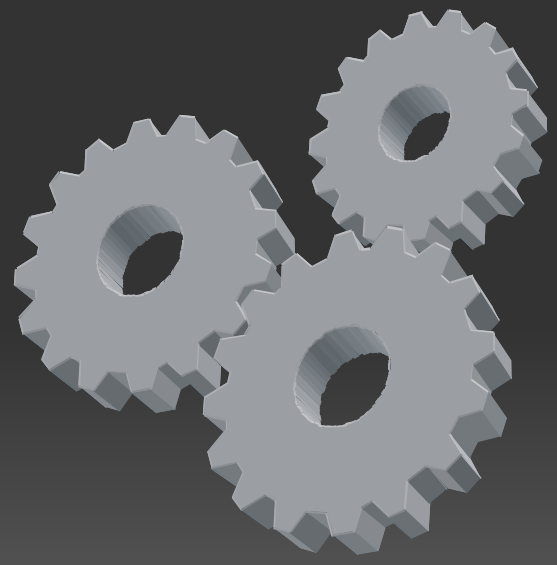
Term discriminating power in a topic of a document

$$\Delta(t_i, d_j) = \sum_{k=0,\ k \neq j}^{m-1} (\sigma(d_k, d_j) \cdot \delta(t_i, d_k)^2)$$

Proposed Algorithm



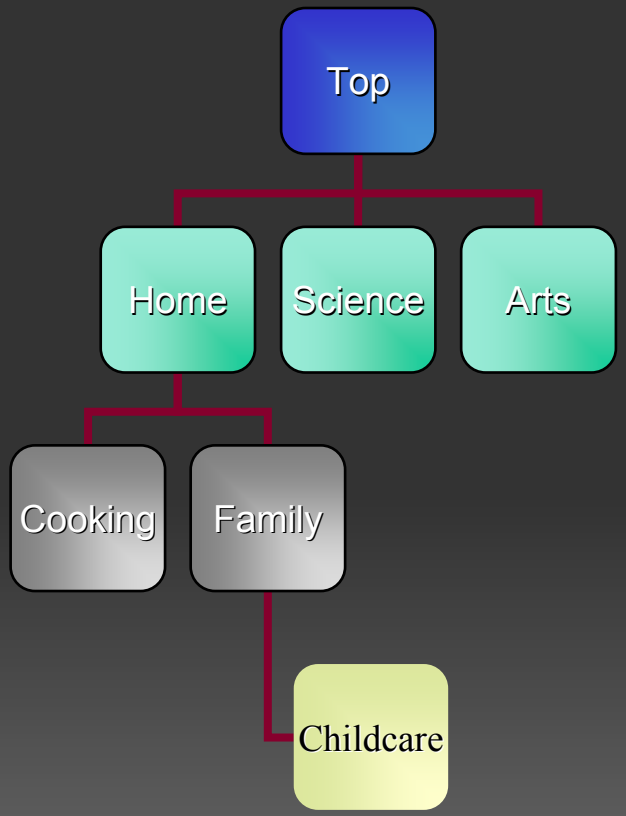
Evaluation



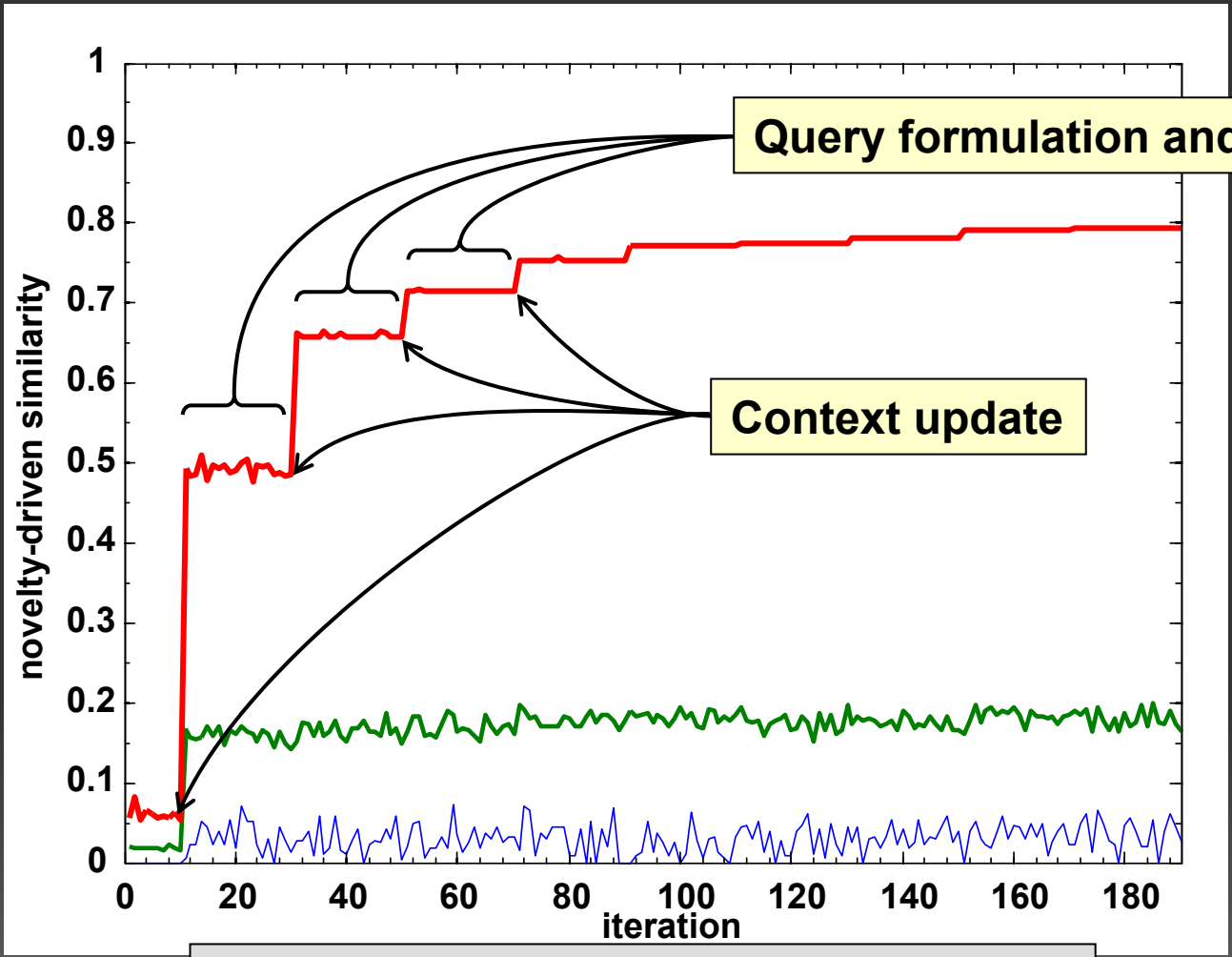
Evaluation

Local Index

- DMOZ – ODP Project
  - ~ 500 Topics
  - 3rd level of the hierarchy
  - 100 URLs at least
  - English language
  - > 350,000 pages
- Initial Context
- vs. Bo1 & Baseline
  - Novelty-Driven Similarity
  - Precision
  - Semantic Precision

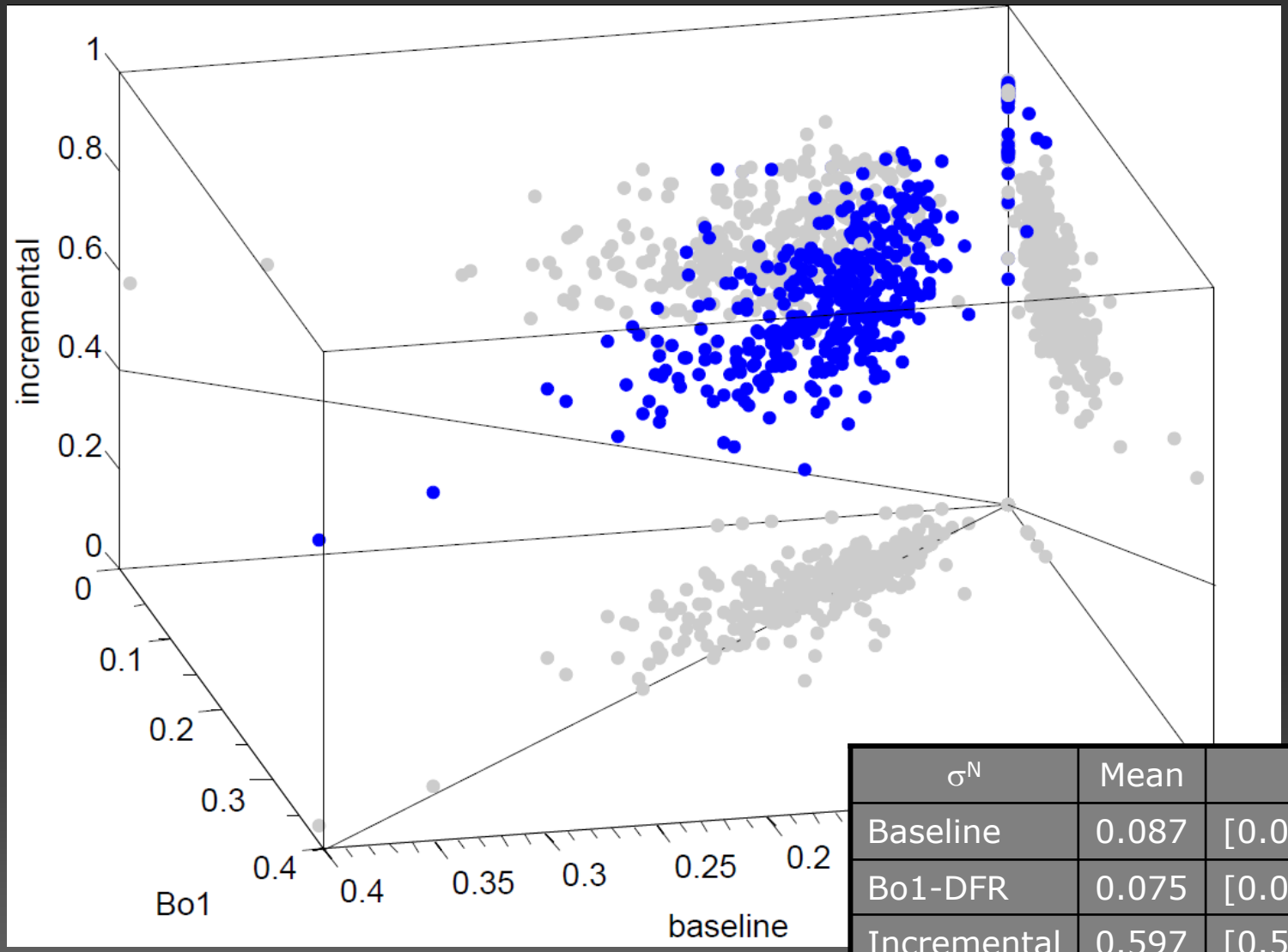


Evaluation – σ^N Similarity



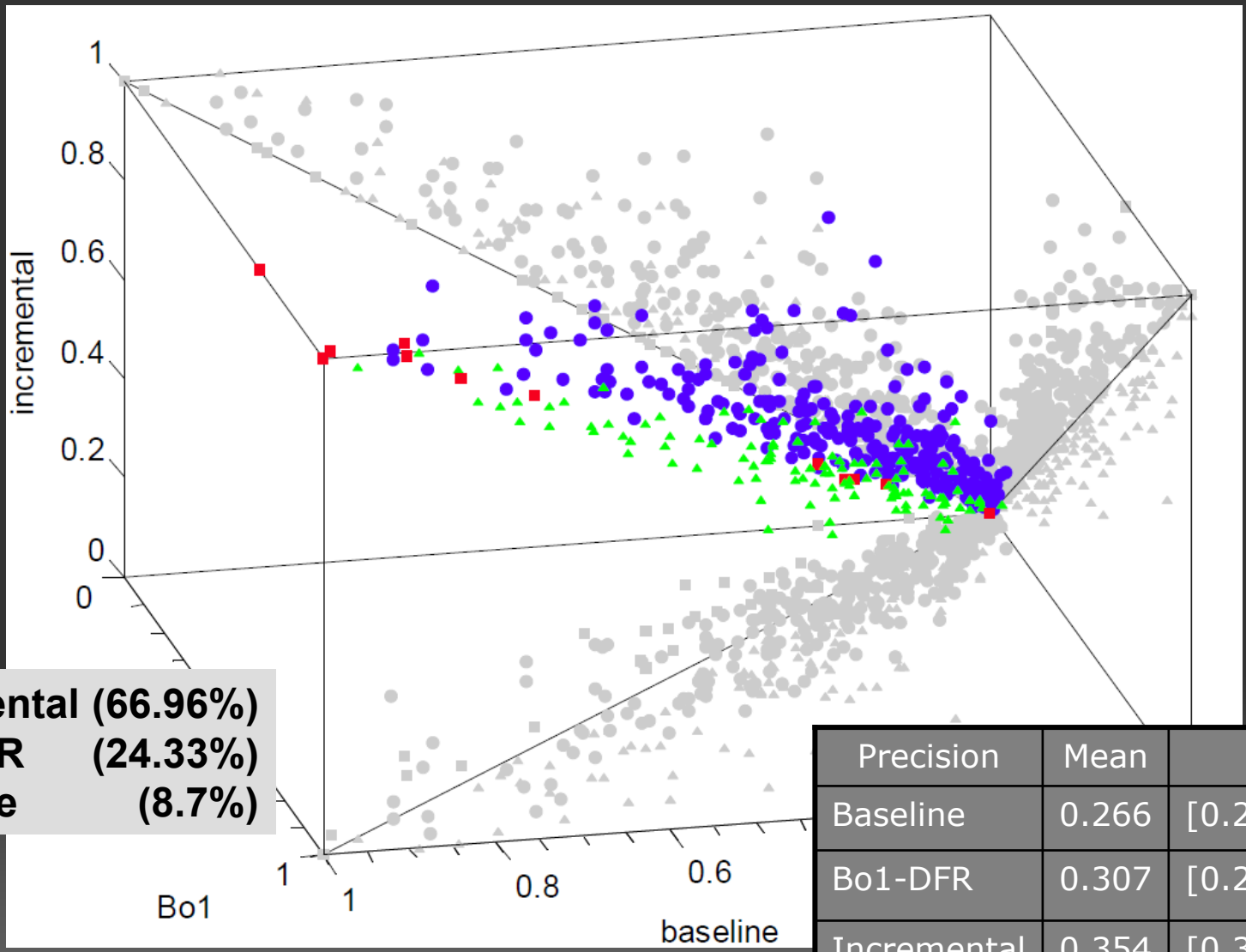
σ^N	Mean	95% CI
1st	0.0661	[0.0618; 0.0704]
best	0.5970	[0.5866; 0.6073]

Evaluation – σ^N Similarity



σ^N	Mean	95% CI
Baseline	0.087	[0.0822; 0.0924]
Bo1-DFR	0.075	[0.0710; 0.0803]
Incremental	0.597	[0.5866; 0.6073]

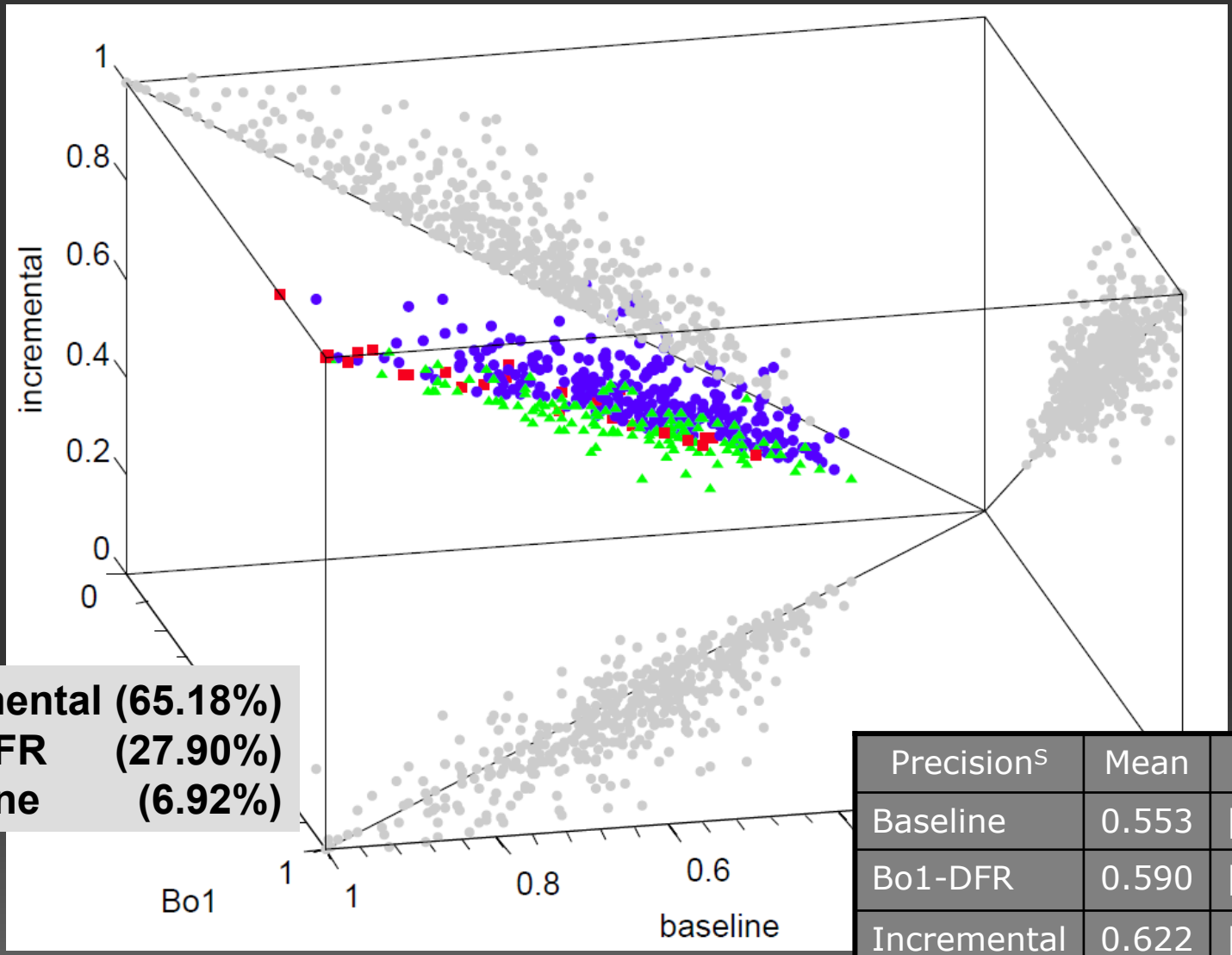
Evaluation – Precision



- Incremental (66.96%)
- Bo1-DFR (24.33%)
- Baseline (8.7%)

Precision	Mean	95% CI
Baseline	0.266	[0.2461; 0.2863]
Bo1-DFR	0.307	[0.2859; 0.3298]
Incremental	0.354	[0.3325; 0.3764]

Evaluation – Semantic Precision



- Incremental (65.18%)
- Bo1-DFR (27.90%)
- Baseline (6.92%)

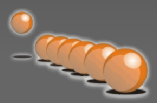
Precision^S	Mean	95% CI
Baseline	0.553	[0.5383; 0.5679]
Bo1-DFR	0.590	[0.5750; 0.6066]
Incremental	0.622	[0.6068; 0.6372]

Conclusions

- We presented an intelligent IR approach for learning *context-specific* terms.
  - Take advantage of the ***user context.***
- We have shown evaluations and the ***effectiveness*** of incremental methods.

Future Work

- Investigate different parameters.
- Develop methods to learn and adjust parameters.
- Run additional tests using other IR metrics.



Thank you!

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial

lidia.cs.uns.edu.ar

Universidad Nacional del Sur
Bahía Blanca

www.uns.edu.ar

CONICET

AGENCIA