Web Page Clustering Using Heuristic Search in the Web Graph

Ron Bekkerman

Shlomo Zilberstein

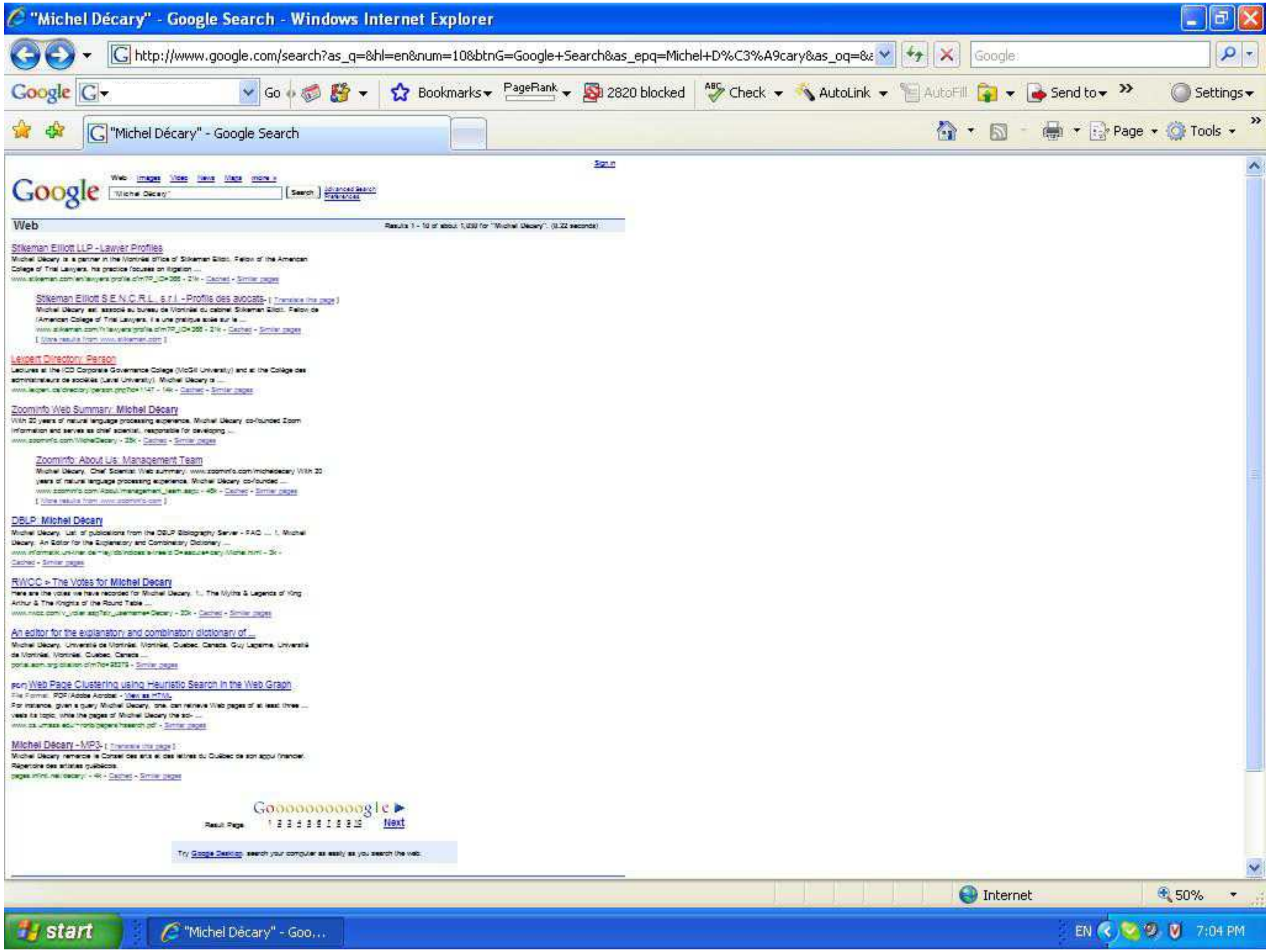
James Allan

UMASS

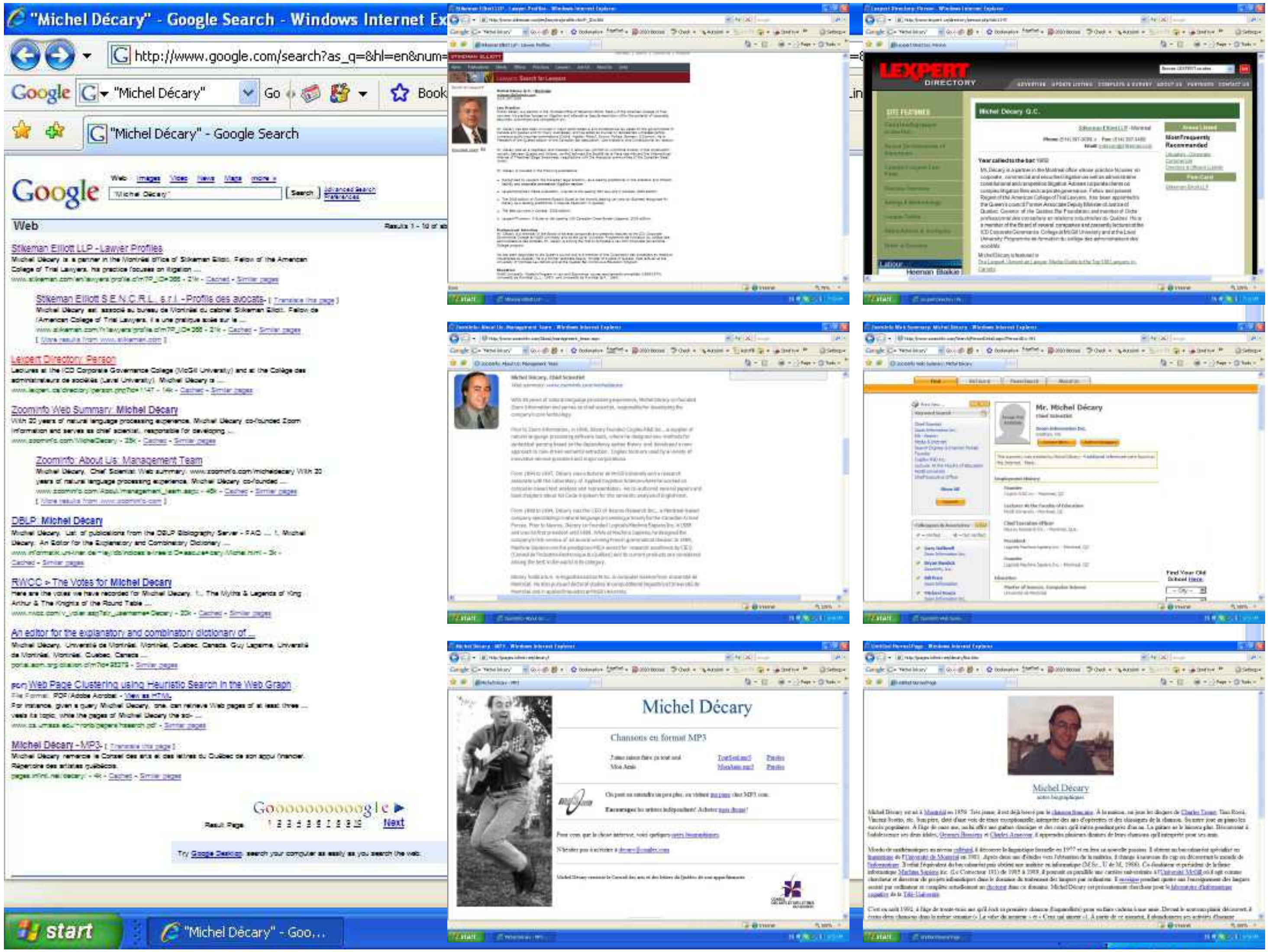
Example application

- Given a person name
- Find out *everything* about this person
  - What is available in the Web
- Possible solution:
  - Query a search engine with the person name
  - Retrieve documents
  - Cluster retrieved documents
    - Build *largest* clusters possible

Query: “Michel Décary”



Query: “Michel Décary”

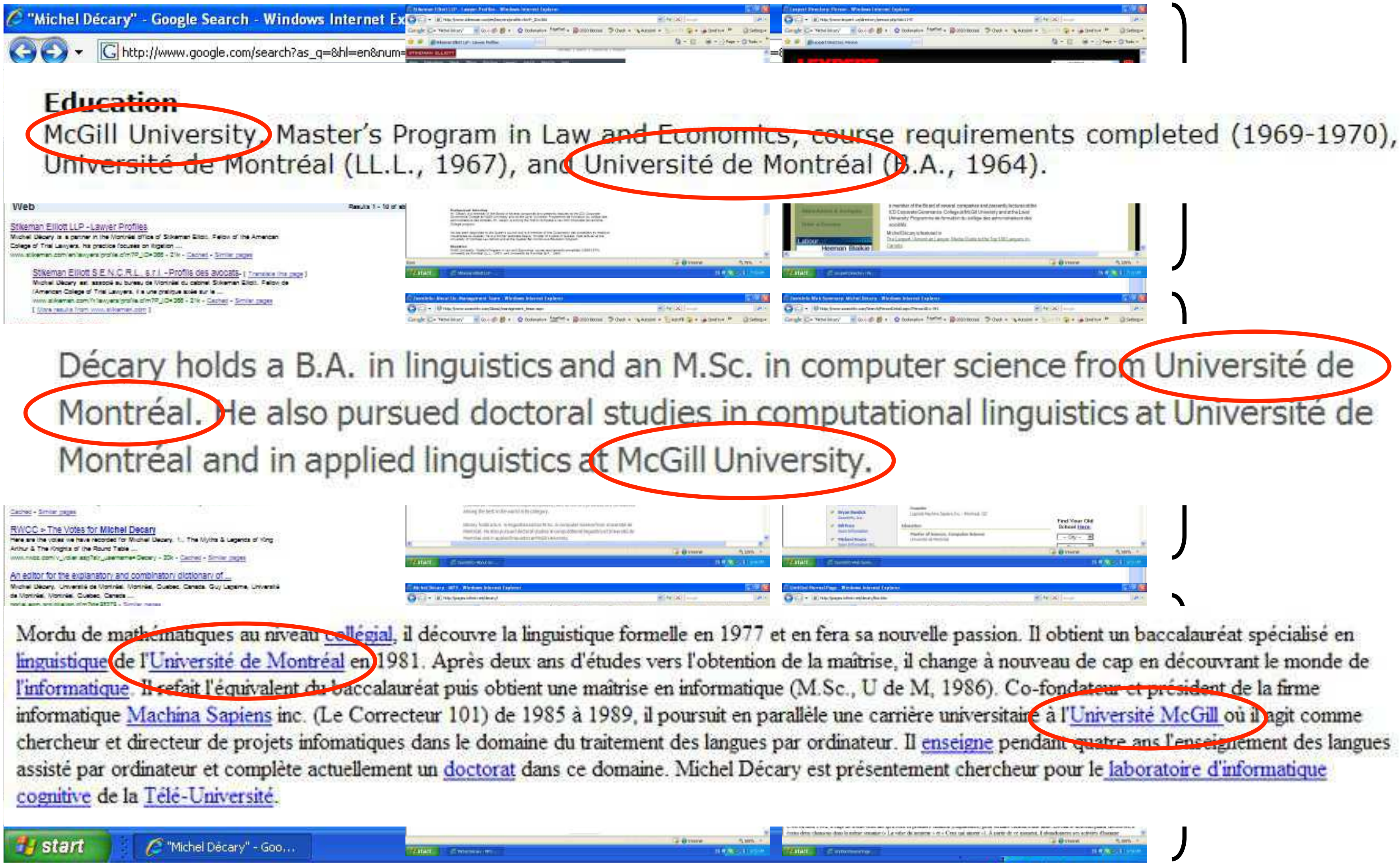


Lawyer

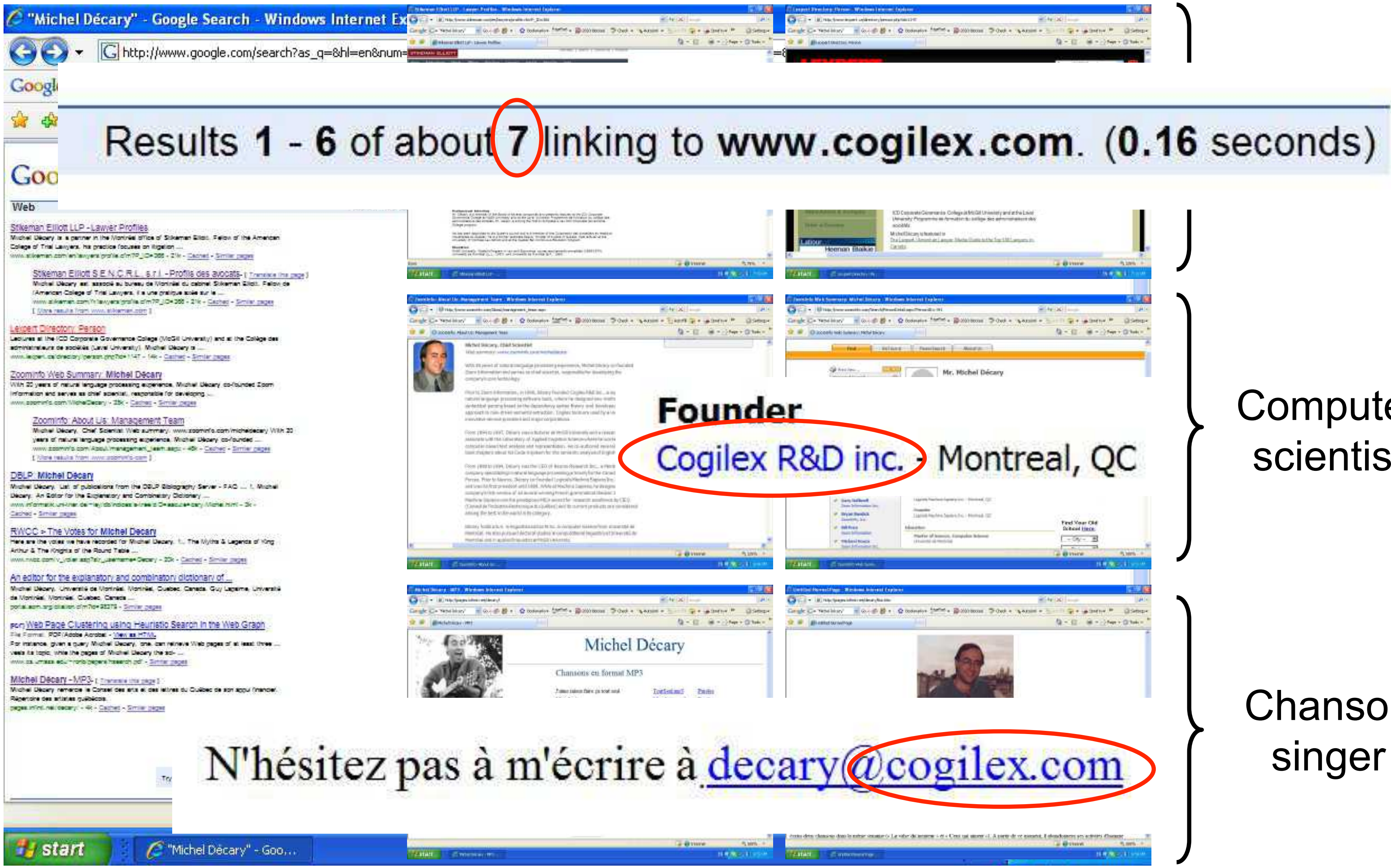
Computer scientist

Chanson singer

Query: “Michel Décary”



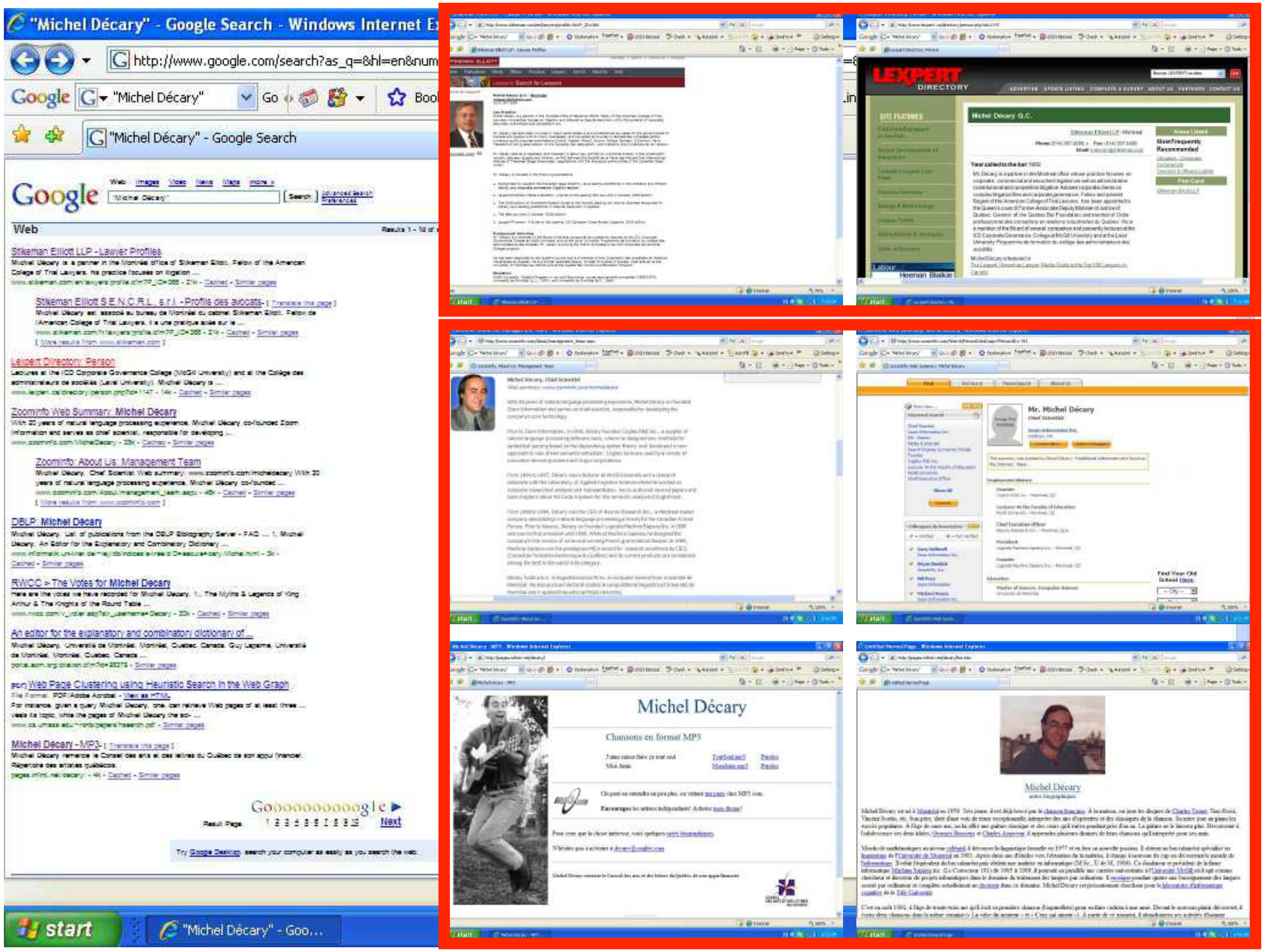
Query: “Michel Décary”



Computer scientist

Chanson singer

Query: “Michel Décary”



Lawyer

Computer scientist

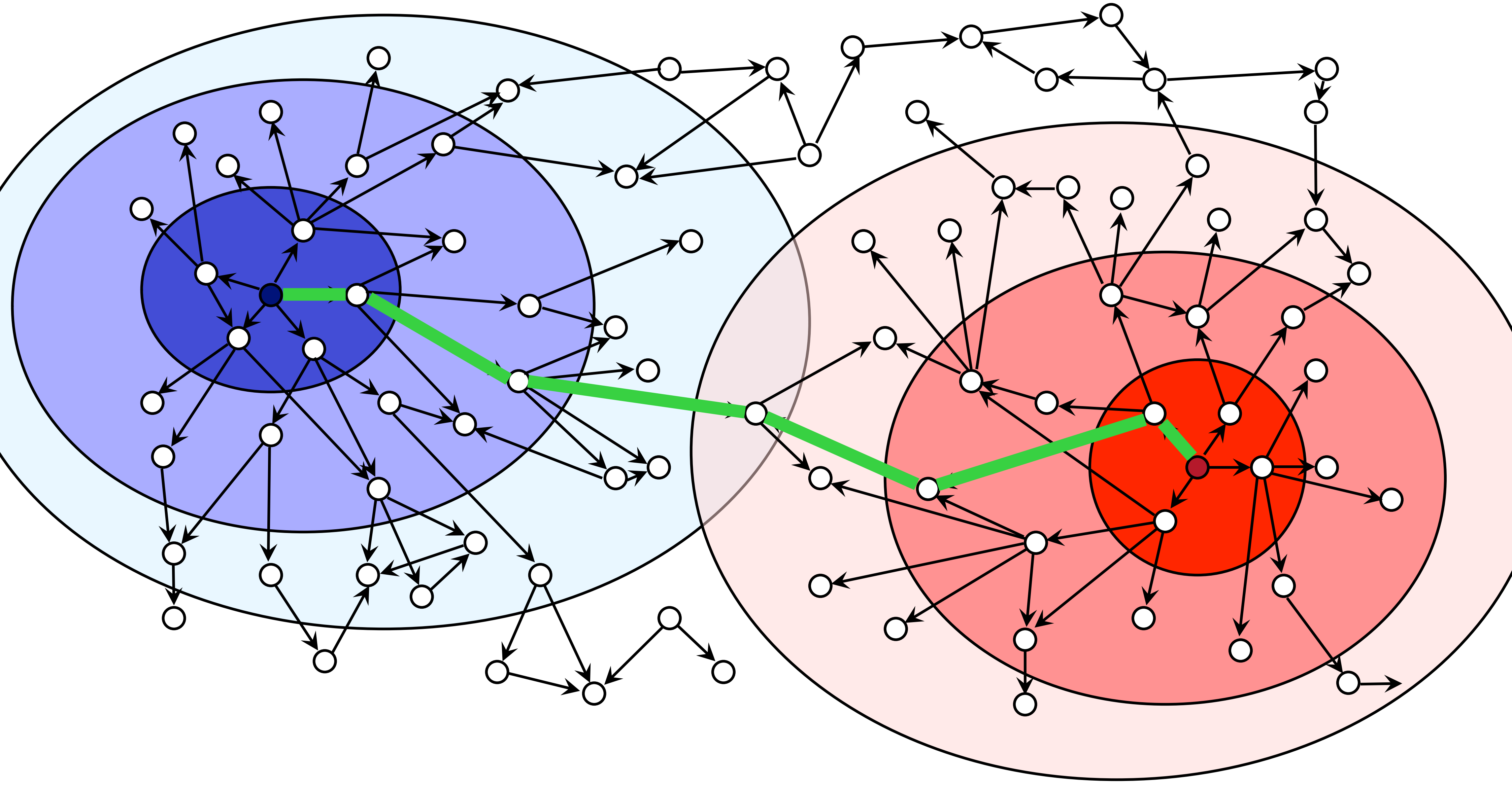
+

Chanson singer

Summary of observations

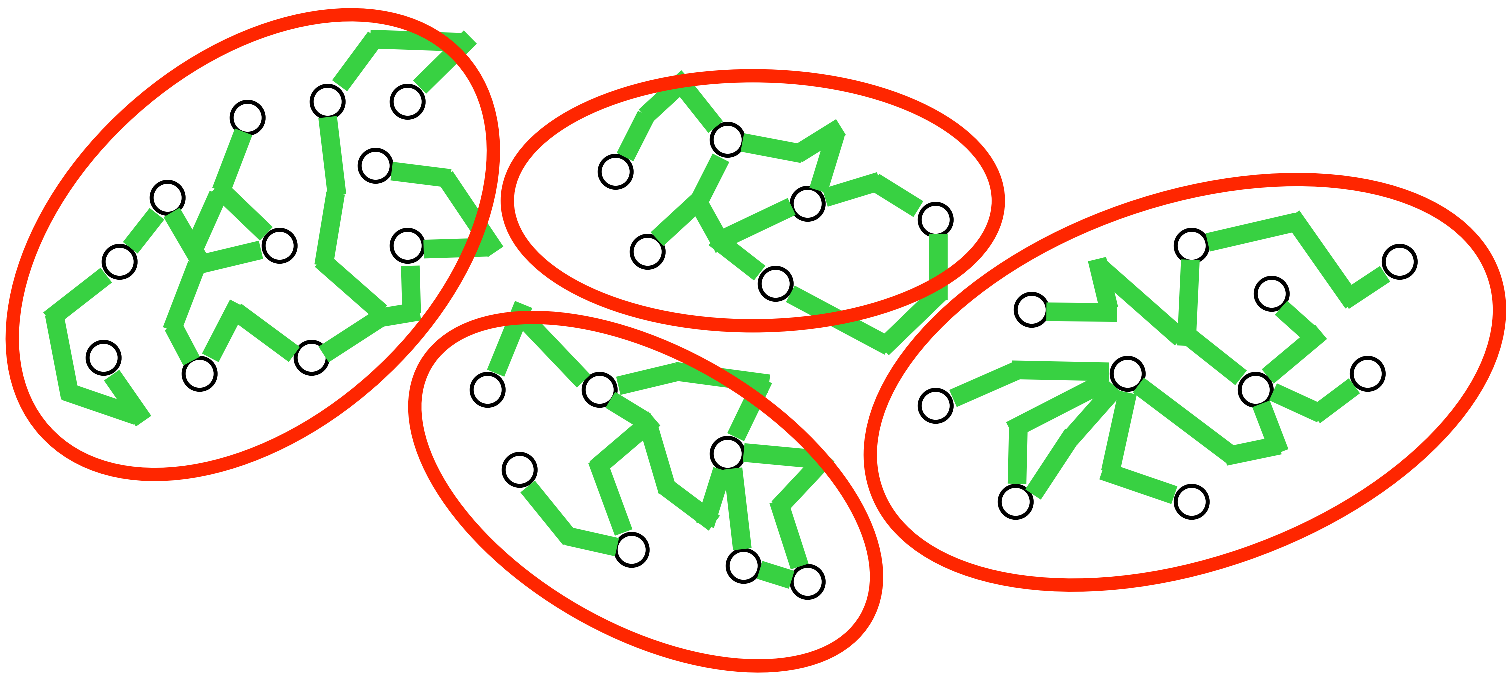
- *Topical* clustering is not enough
  - Although not to be ignored
- Web graph topology should be exploited
  - Close pages tend to be semantically related
    - There's a *reason* for hyperlinking page $A \longrightarrow$ page B
    - Be careful: arbitrary connections exist as well
  - Apply *beam search* to find close pages
    - Use heuristics to prune undesired branches

Example: breadth first search



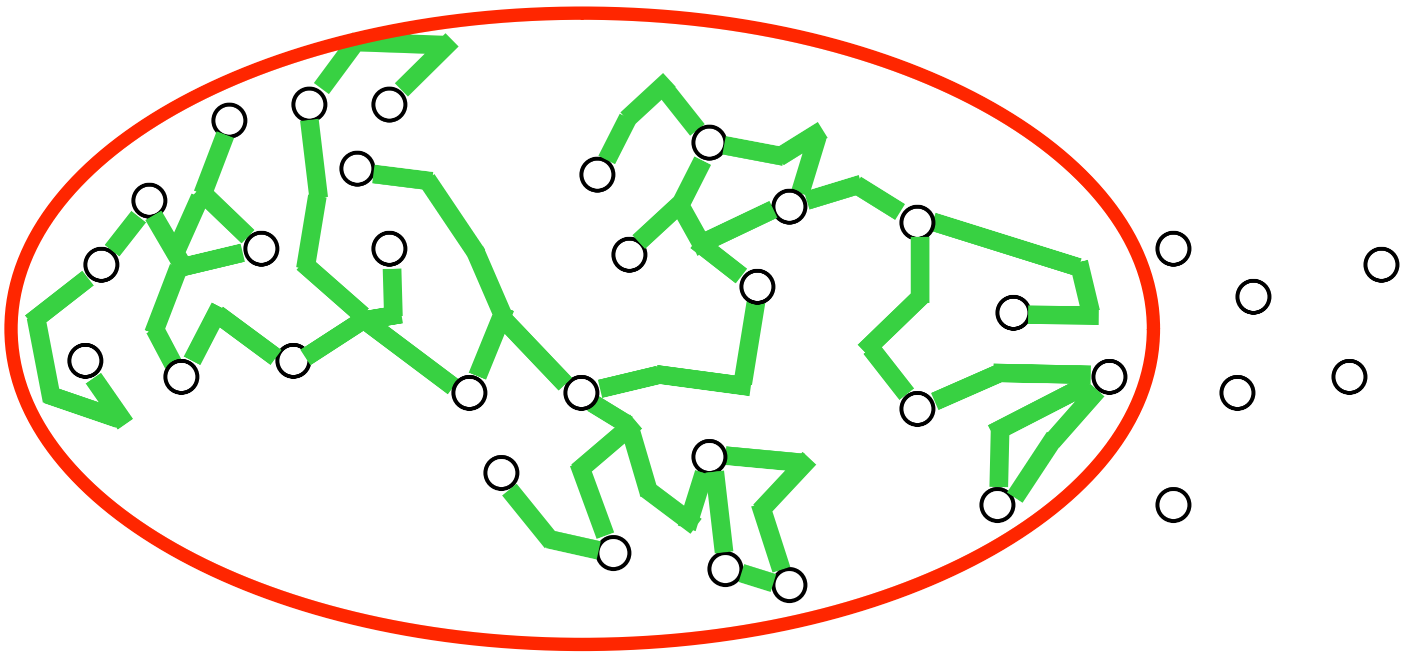
Clustering by multi-agent search

- Each page is represented by a Web agent
  - Whose task is to traverse the Web graph
  - And meet as many other agents as possible



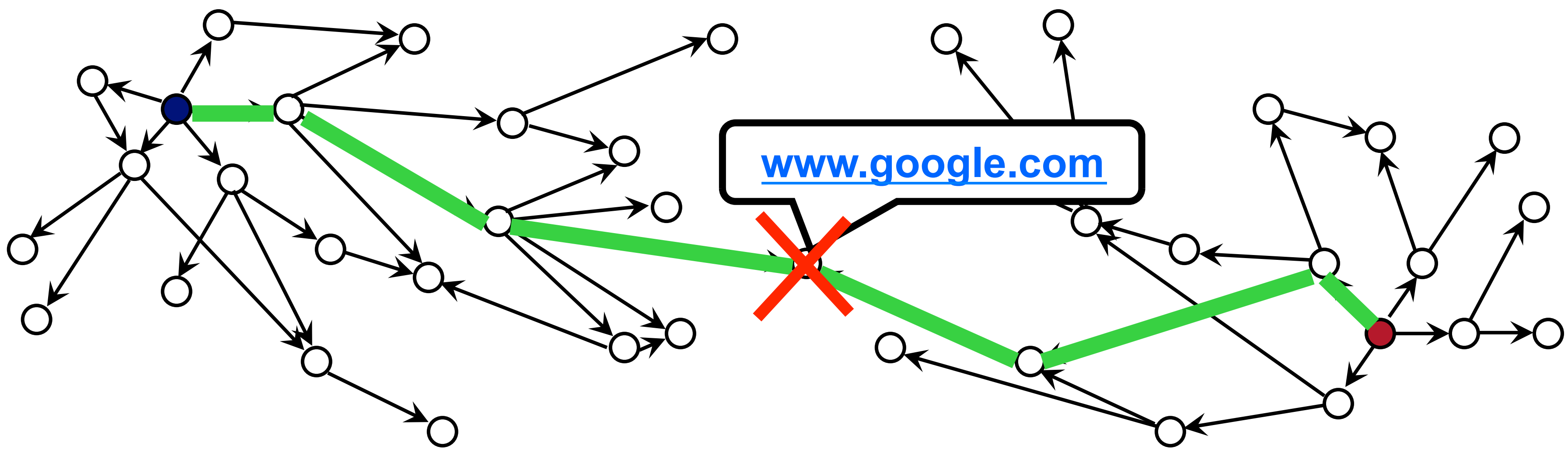
Real-world case

- The Web is tightly interconnected
  - About 70% agents meet after 3 search iterations
  - Which is clearly an undesired outcome



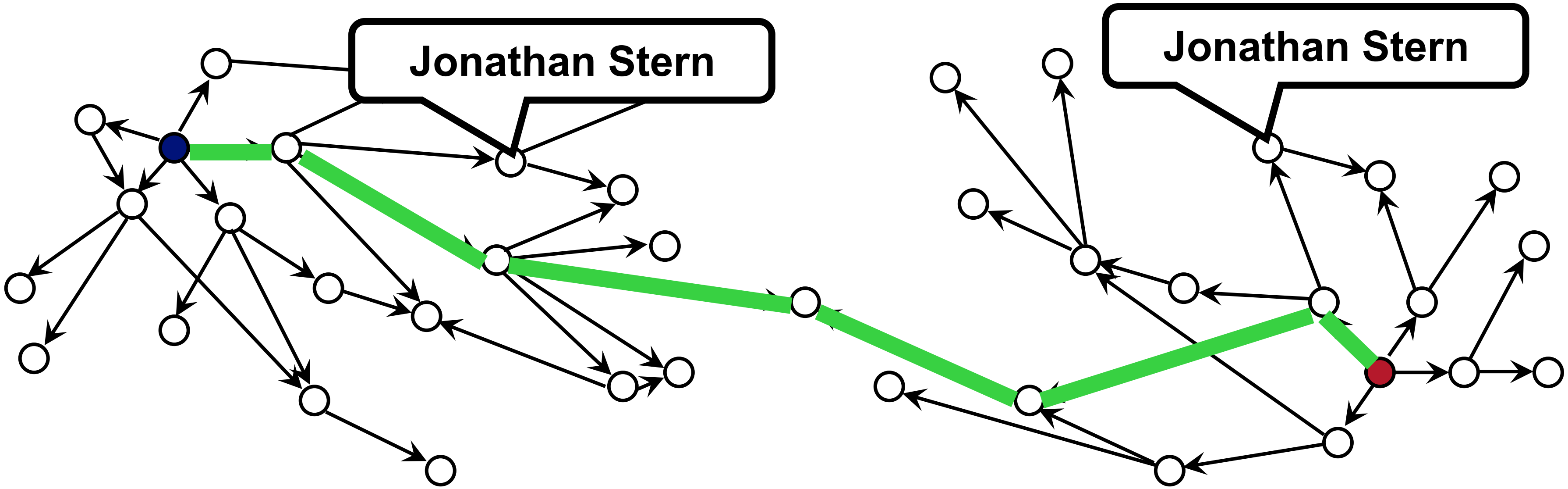
Heuristic 1

- Elimination of high-degree nodes
  - Both high in-degree and high out-degree
  - They often connect unrelated pages



Heuristic 2

- Person name sharing
  - Expanded nodes share a hyperlink
  - AND a person name (ignore too popular names)

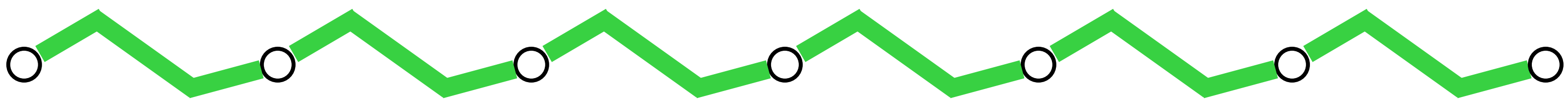


Heuristic 3

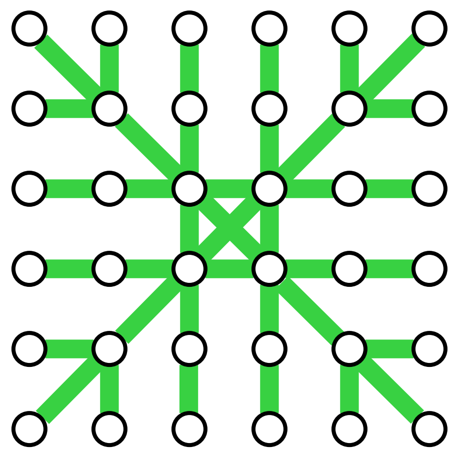
- Anchor text sharing
  - Anchor texts often summarize the content of hyperlinked pages
- Same idea as in person name sharing
  - But much simpler to implement
    - No sophisticated information extraction needed
    - Shallow parsing of HTML is enough
  - Again, ignore too frequent anchor texts
    - Like *“Contact Us”* or *“Copyright”*

Algorithmic enhancement

- Unpleasant artifact: too long connections
  - Too weak semantic relationships



- Proposed solution: keep track of cluster's diameter
  - Start with a tightly connected component
  - Add pages found within one hop



Experimentation domains

- Web appearance disambiguation

  Bekkerman & McCallum, WWW-05

  - Given pages retrieved on N people names
    - From one social network
  - Filter out pages that refer to their unrelated namesakes
- Clustering of Web search results

  Hearst & Pedersen, SIGIR-96

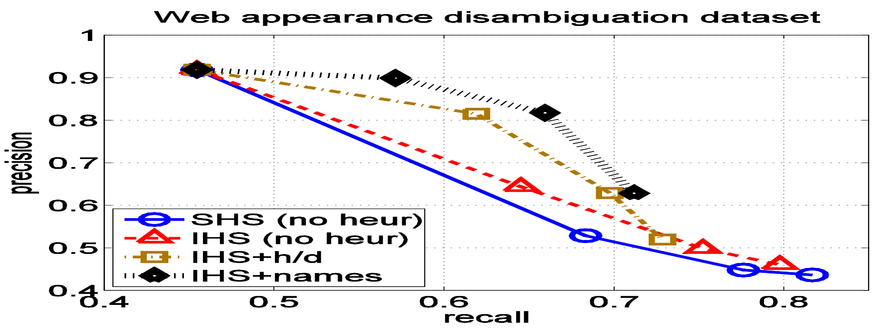
  - Represent ranked lists of retrieved documents as clusters of semantically related documents

Disambiguation dataset

- 12 names out of Melinda Gervasio's social network

Personal name	***Position***	***Pages***	***Namesakes***	***Relevant pages***
Adam Cheyer	SRI Manag	97	2	96
William Cohen	CMU Prof	88	10	6
Steve Hardt	SRI Eng	81	6	64
David Israel	SRI Manag	92	19	20
Leslie Pack Kaelbling	MIT Prof	89	2	88
Bill Mark	SRI Manag	94	8	11
Andrew McCallum	UMass Prof	94	16	54
Tom Mitchell	CMU Prof	92	37	15
David Mulford	Stanf Undergrad	94	13	1
Andrew Ng	Stanf Prof	87	29	32
Fernando Pereira	UPenn Prof	88	19	32
Lynn Voss	SRI Eng	89	26	1
	OVERALL:	***1085***	***187***	***420***

Disambiguation results



- *h/d* – high degree node elimination
- *names* – person name heuristic
- Each point: one iteration of search
  - Only 2 iterations are enough

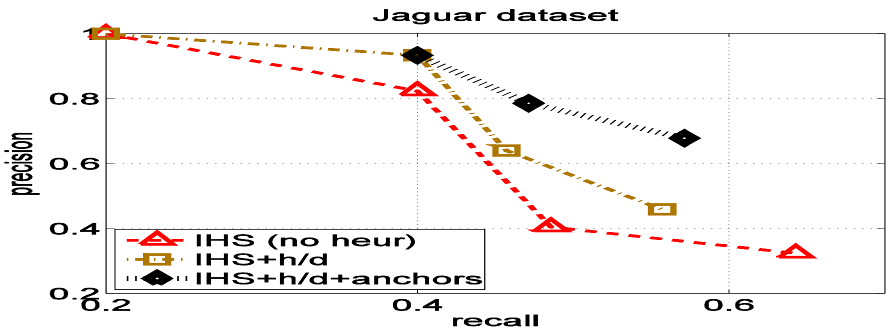
- *SHS* – sequential heuristic search (basic algorithm)
- *IHS* – incremental heuristic search
  - With the enhancement of diameter tracking

Jaguar dataset

- 100 pages retrieved on query *"Jaguar"*
  - 23 different categories!
- The task is to build 3 clusters
  - Of cars, Mac OS and wild cats

Category	***Pages***	***Category***	***Pages***	***Category***	***Pages***
Car	36	Reef lodge	2	Atari game	5
MacOS	11	Book	1	Guitar	1
Wild cat	23	Singer	2	TV channel	1
Biotech firm	2	Emulator	2	Web designer	2
Youth org	1	Cornell project	2	E-commerce firm	1
Maya culture	1	Metal band	1	Game archive	1
Resin models	1	Movie	1	Aircraft	1
Web hosting	1	Photo gallery	1	***OVERALL:***	100

Jaguar results



- SHS algorithm fails
  - 70 agents meet together
- *anchors* – anchor text heuristic
- Best performance:
  - High degree AND anchors heuristics

Topical & topological clustering

- Build topical clusters

Bekkerman & McCallum, WWW-05

- Enrich topical clusters with pages obtained by heuristic search based clustering

Web appearance disambiguation				Clustering of Web search results			
Method	***Precision***	***Recall***	***F-measure***	***Method***	***Precision***	***Recall***	***F-measure***
Topical	87.3%	71.3%	78.4%	Topical	75.0%	64.3%	69.2%
IHS (iteration 1)	89.9%	57.1%	69.9%	IHS (iteration 1)	93.3%	40.0%	56.0%
Hybrid (iteration 1)	84.5%	83.3%	**83.9%**	Hybrid (iteration 1)	77.1%	77.1%	**77.1%**
IHS (iteration 2)	81.7%	66.0%	73.0%	IHS (iteration 2)	78.6%	47.1%	58.9%
Hybrid (iteration 2)	78.5%	86.2%	82.2%	Hybrid (iteration 2)	72.7%	80.0%	76.2%

- Best performance: after one iteration of heuristic search only!

Conclusion

- First application of heuristic search to the Web graph
  - Very simple algorithms / heurstics
  - Heuristic search theory yet to be applied
    - E.g., can an *admissible* heuristic be proposed?
- Search can be performed in real time!
  - Modern search engines store the link structure of most of the Web
  - Maximum 2 search iterations are enough
    - Fully distributable in a multi-agent fashion