SEMANTIC INDEXING (ENTITY LINKING)

Jelena Jovanović

Email: jeljov@gmail.com

Web: http://jelenajovanovic.net

Overview

- Main concepts
  - Named Entity Recognition
  - Semantic Indexing / Entity Linking
  - Topic detection
  - Cognition as a Service
- Application examples
- Combined use of ML and knowledge bases (KB) for Semantic indexing

MAIN CONCEPTS

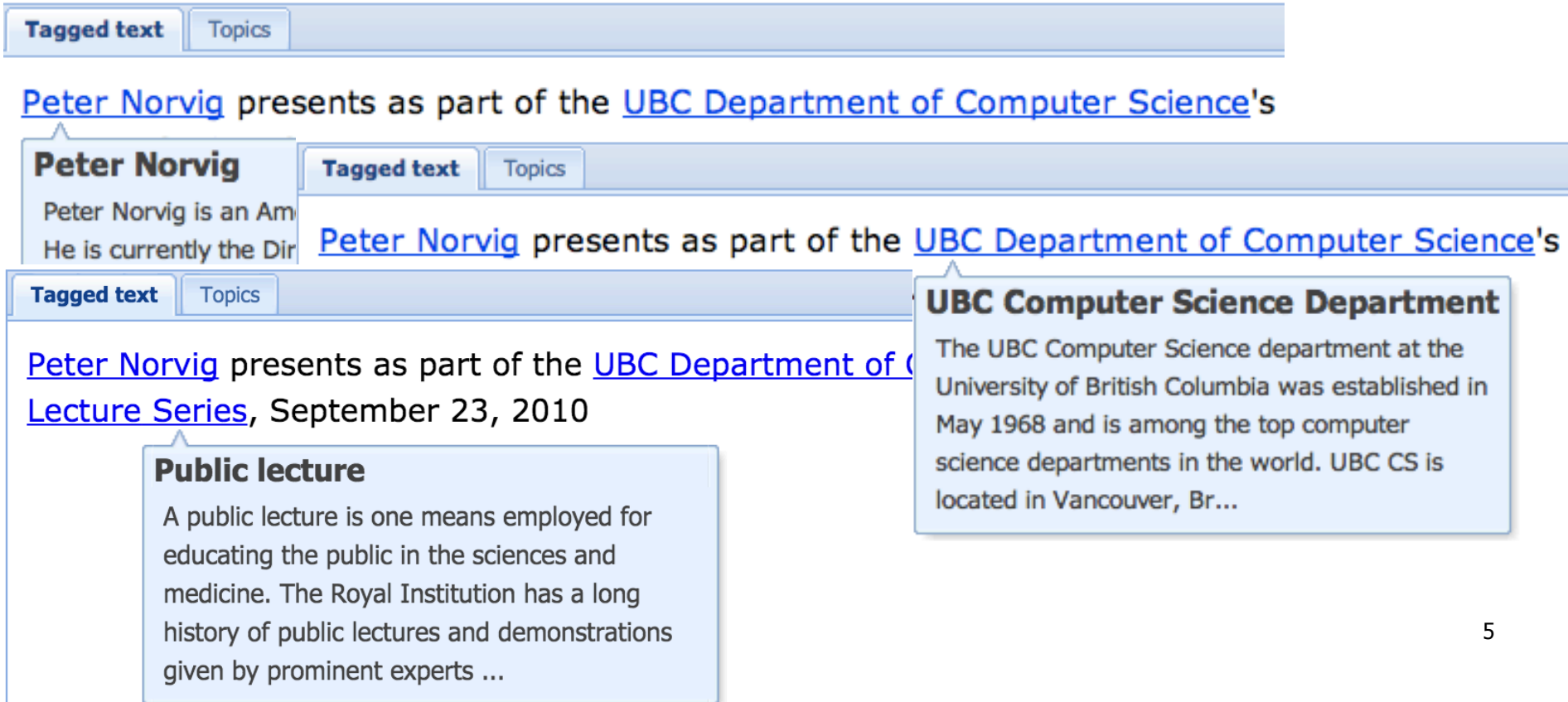
ENTITY RECOGNITION IN THE TEXT

- *Named Entity Recognition* (*NER*)
- Entities can be of different types: person, organization, location, date, currency and the like
- Example:

> Peter Norvig [PER] presents as part of the UBC Department of Computer Science's [ORG] Distinguished Lecture Series, September 23, 2010 [DATE].

SEMANTIC INDEXING

- Semantic Indexing, Entity Linking
- Semantic indexing = NER + Disambiguation
- Disambiguation = unambiguously determining the meaning of the recognized entities



TOPICS / CONCEPTS DETECTION

- Identification of the main topics / concepts of the given piece of text
- Has lot of resemblance to semantic indexing; still, differences do exist:
  - Topics/concepts are not associated with individual words/ phrases in the text, but with the text as a whole
  - After topics have been identified, they are sorted based on the estimated relevancy for the given text

TOPICS / CONCEPTS DETECTION

Example:

"The Blue Jasmine star accepted the accolade from last year's Best Actor, Daniel Day Lewis, for her performance as deeply troubled socialite Jeanette 'Jasmine' Francis in Woody Allen's acclaimed film."

Detected concepts (as Wikipedia entities):

- http://en.wikipedia.org/wiki/Film
- http://en.wikipedia.org/wiki/Celebrity
- http://en.wikipedia.org/wiki/Academy_Award_for_Best_Actor
- http://en.wikipedia.org/wiki/Woody_Allen
- http://en.wikipedia.org/wiki/Daniel_Day-Lewis
- http://en.wikipedia.org/wiki/Socialite

Cognition As A Service

- There are more and more services / tools for named entity recognition, topics/concepts detection and semantic indexing
- Common feature of all these services / tools:
  - Combined use of ‘machine intelligence’ and collected human knowledge, that is, machine learning techniques and huge knowledge bases

COGNITION AS A SERVICE

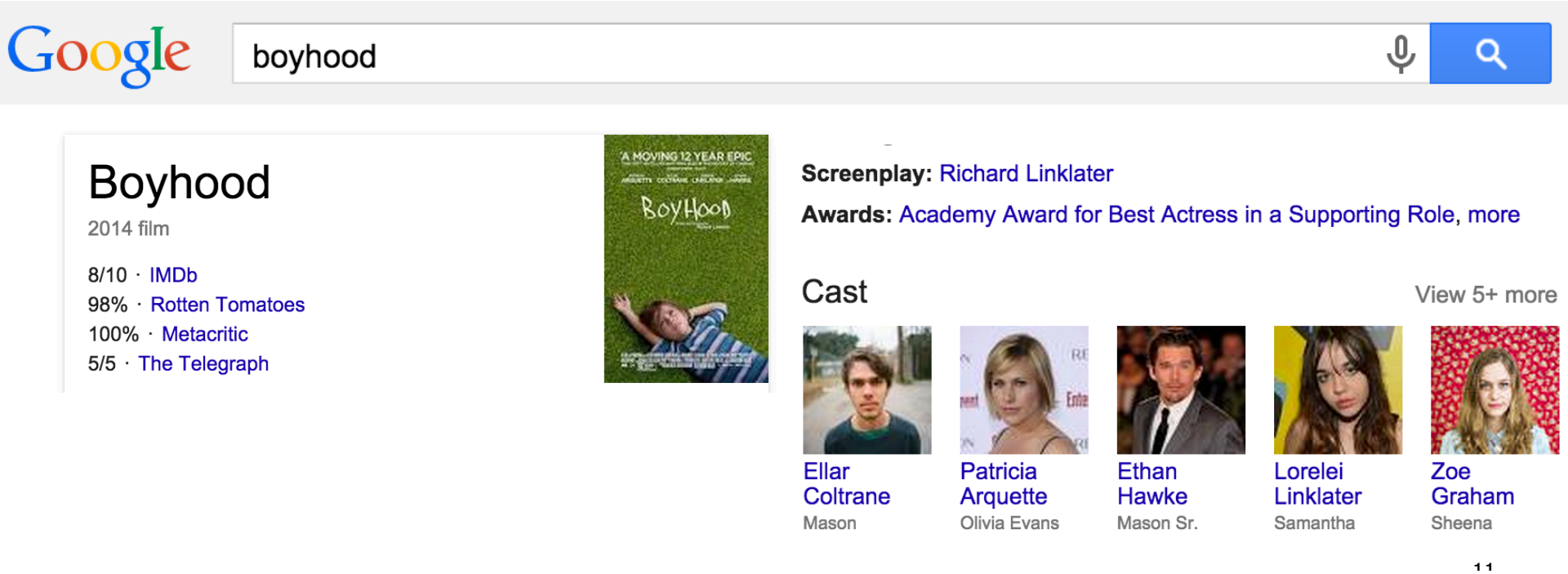
Examples:

- Alchemy API (http://www.alchemyapi.com/)
- TextRazor (http://www.textrazor.com/)
- Textwise (http://textwise.com/)
- OpenCalais (http://www.opencalais.com/)
- Dandelion API (https://dandelion.eu/)
- TagMe (http://tagme.di.unipi.it/)
- Wikipedia Miner (http://wikipedia-miner.cms.waikato.ac.nz/)
- …

Application examples

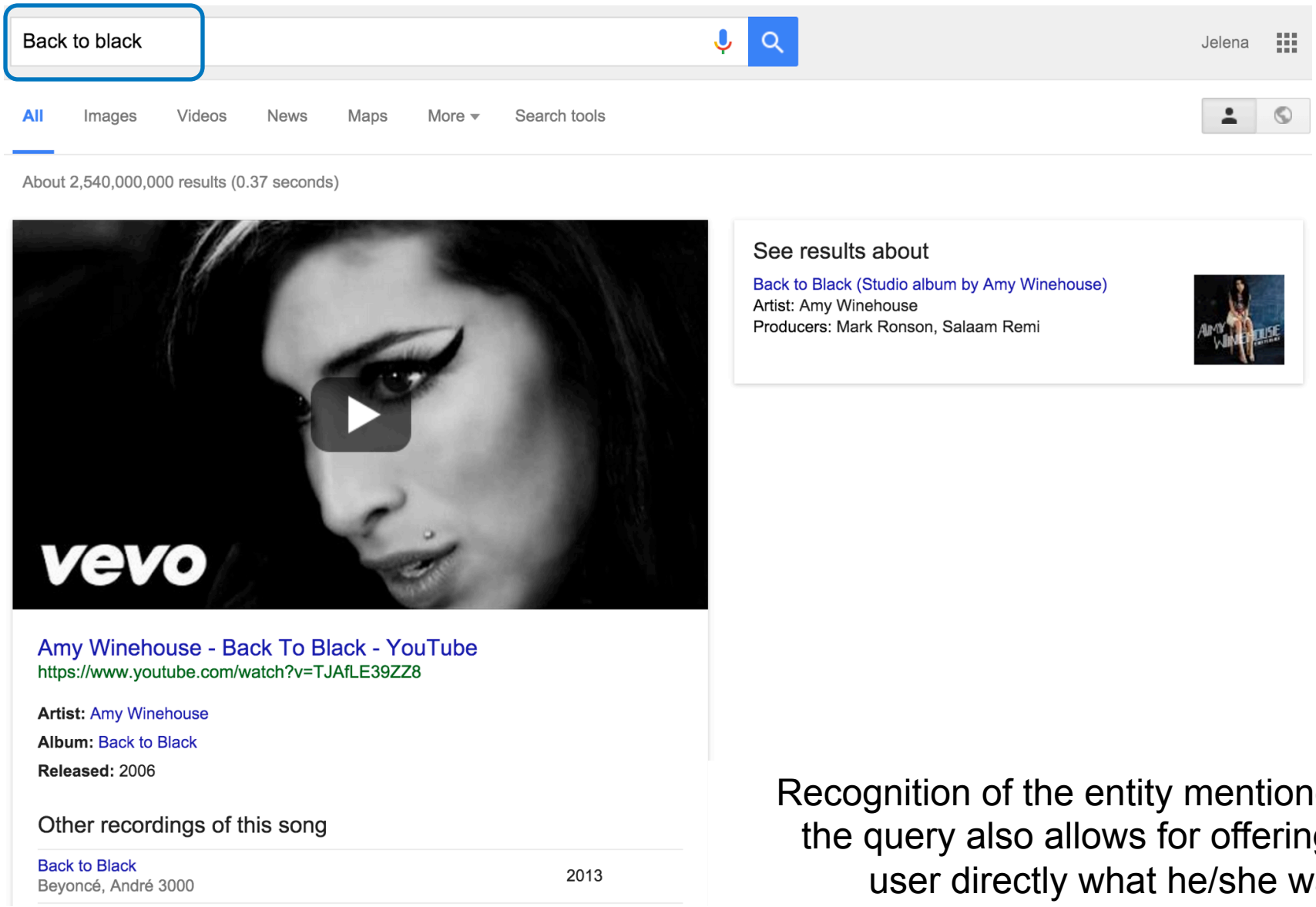
ADVANCED SEARCH

- Over 50% of all the Web search queries refer to some entity (person, movie, song, music group, city, …)*
- Recognition of the entity mentioned in the query allows for recommendation of similar and/or related entities the user might be interested in



* Pound et al. 2010. Ad-hoc object retrieval in the web of data. In: Proceedings of the 19th International Conference on World Wide Web. pp. 771-780. ACM, New York, NY, USA

ADVANCED SEARCH



Recognition of the entity mentioned in the query also allows for offering the user directly what he/she wants, in this case, play the song

Business Analytics

Example: RavenPack News Analytic

- http://www.ravenpack.com/
- Extraction of entities from news articles: companies, brands, products,…
- Extraction of geo-politic and major economic events, as well as events relevant for individual companies and brands
- Extracted pieces of information serve as input for business analytics, in particular, *business rules engine*

SOCIAL MEDIA MONITORING

Reputation management

- Objective: timely detection of public writings that might affect the reputation of a person or an organization
- How it is done: automated analysis of textual content exchanged over online social networks and social media, to
  - detect mentions of relevant entities: persons, companies, brands, products
  - detect the sentiment expressed about the identified entities

- Examples:
  - Reputation.com (http://reputation.com/)
  - Rankur (https://rankur.com/)
  - Trackur (http://www.trackur.com/)

ONLINE ADVERTISING

Example: ADmantX (http://www.admantx.com/)

- Analyzes the content of a Web page to extract pieces of information relevant for the selection / recommendation of ads for the given page
- Extracts:
  - entities (persons, locations, companies, brands,…),
  - emotions expressed in the text,
  - topics covered in the text (coarse and fine-grained)

Combined use of ML and Knowledge Bases (KB) for Semantic Indexing

SEMANTIC INDEXING

- Combined use of supervised m. learning (classification) and knowledge stored in Web-based KBs
- Most frequently used KBs: Wikipedia, Freebase, DBpedia
- Specific (additional) advantage of these approaches: they allow for easier creation of annotated corpora required for training ML models

DIFFICULTIES WITH THE TRAINING SET CREATION

Supervised ML approaches to entity recognition / linking are dependent on the availability of large annotated corpora

An example illustrating the kind of text annotations required for training a supervised ML model:

Unlike <PERSON>Robert</PERSON>, <PERSON>John Briggs Jr </PERSON> contacted <ORGANIZATION>Wonderful Stockbrockers Inc </ORGANIZATION> in <LOCATION>New York</LOCATION> and instructed them to sell his <QUANTITY>100</QUANTITY> shares in <ORGANIZATION>Acme</ORGANIZATION>

Obviously, preparation of a training dataset (corpus) is a laborious task…

EASIER CREATION OF THE TRAINING SET

Large Web-based KBs greatly facilitate the creation of training datasets

For instance, if Wikipedia is used as a KB:

- Each term that has an embedded Wikipedia link is treated as a potential entity; we'll refer to such terms as *anchors*
- Each *anchor* provides several training instances:
  - one positive example: link destination (linked Wikipedia page), that is, the "true" meaning of the given anchor in the given context
  - several negative examples: all other potential destinations, i.e., all other possible meanings of the considered anchor

Creation of a training dataset by making use of the Wikipedia's internal links – an illustration

For the term (anchor) *tree,* there are 26 possible destinations (i.e., meanings); this results in 1 positive and 25 negative examples for training the model

Depth-first search

From Wikipedia, the free encyclopedia

Depth-first search (**DFS**) is an algorithm for traversing or searching a tree tree structure or graph. One starts at the root (selecting some node as the root in the graph case) and explores as far as possible along each branch before backtracking.

Formally, DFS is an uninformed search that progresses by expanding the first child node of the search tree that appears and thus going deeper and deeper until a goal node is found, or until it hits a node that has no children. Then the search backtracks, returning to the most recent node it hadn't finished exploring. In a non-recursive implementation, all freshly expanded nodes are added to a LIFO stack for exploration.

sense	commonness	relatedness
Tree	92.82%	15.97%
Tree (graph theory)	2.94%	59.91%
Tree (data structure)	**2.57%**	**63.26%**
Tree (set theory)	0.15%	34.04%
Phylogenetic tree	0.07%	20.33%
Christmas tree	0.07%	0.0%
Binary tree	0.04%	62.43%
Family tree	0.04%	16.31%
…		

In this way, starting from, for instance, 500 Wikipedia articles one can generate a training set of >50,000 instances

SEMANTIC INDEXING: THE MAIN STEPS

1) *Entity spotting and candidate selection:* identification of terms that might represent entities in the text (entity-mentions), and selection of candidate entities from a KB for each entity-mention

2) *Disambiguation*: selection of the "best" entity, from the set of candidate entities, for each entity-mention*;*

3) *Filtering*: pruning the results with the aim of eliminating irrelevant entities

ENTITY SPOTTING & CANDIDATE SELECTION

- The objective of this phase is twofold:
  - to identify entity-mentions in the input text, i.e., the parts of the text (single words or phrases) that might represent entities;
  - to identify a set of candidate entities from a KB (e.g., Wikipedia or DBpedia) for each entity-mention

ENTITY SPOTTING & CANDIDATE SELECTION

- Example

"They performed Kashmir, written by Page and Plant.
Page played unusual chords on his Gibson."

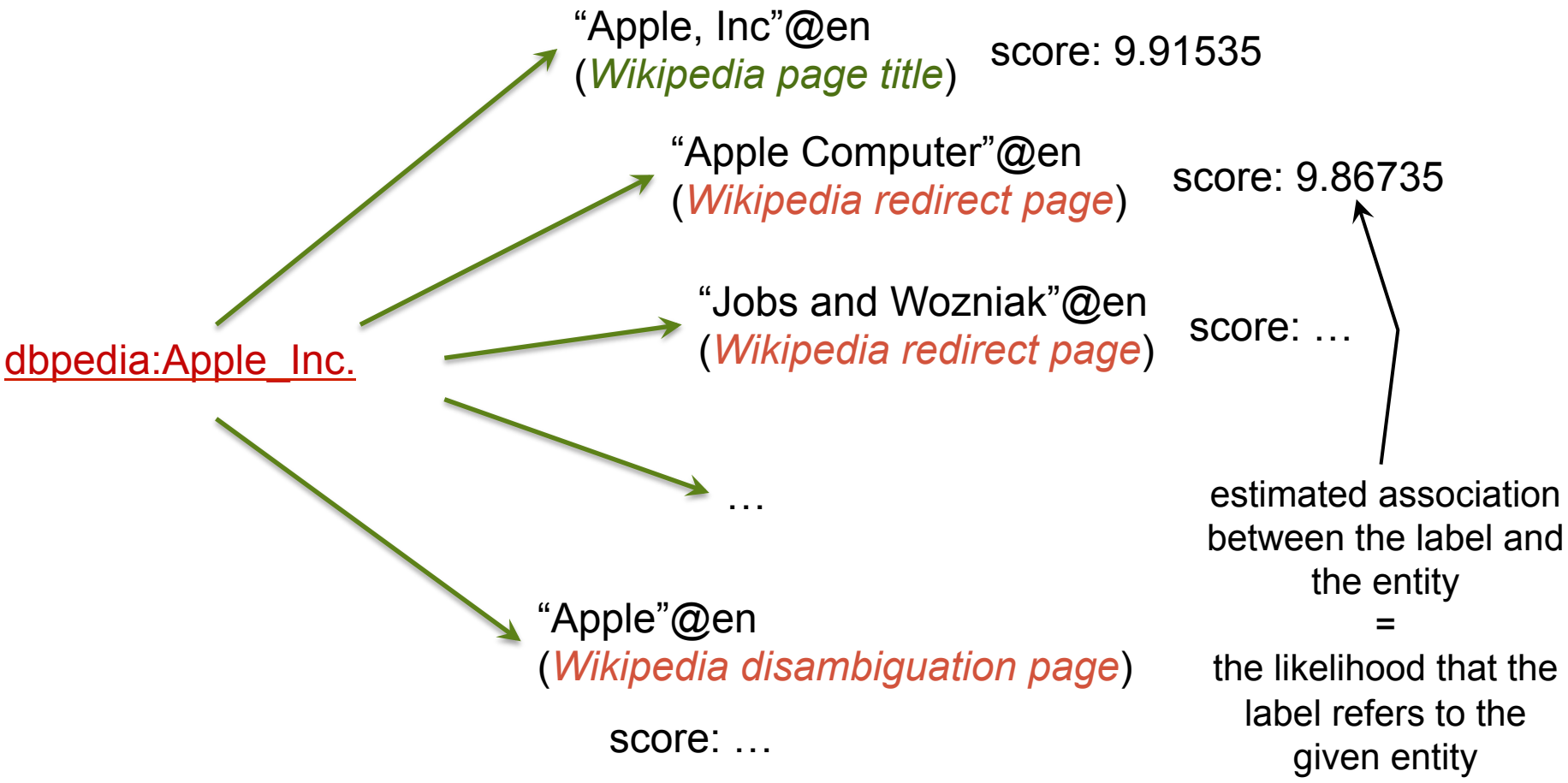
Candidates:
dbpedia:Kashmir – a valley between Pakistan, India and Ladakh
dbpedia:Kashmir_(band) – a Danish rock band
dbpedia:Kashmir_(song) – 1975 song by rock band Led Zeppelin
dbpedia:Kashmir,_Iran – a village in Iran
…

Example taken from: Hoffart J. et al. (2011). Robust disambiguation of named entities in text. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '11), 782-792.

ENTITY SPOTTING & CANDIDATE SELECTION

- Typically, the tasks of this phase are performed as *dictionary look-up* tasks
  - a dictionary is typically created through the extraction of entity labels and descriptions from a specific knowledge base
  - Wikipedia and DBpedia are often used as the source of labels and descriptions
  - dictionary entries might be enriched with statistics computed over the content of the knowledge base
    - E.g., the relevancy of certain label for certain entity

EXAMPLE: DBPEDIA LEXICALIZATION DATASET



Dataset is available at: http://dbpedia.org/Lexicalizations

DISAMBIGUATION PHASE

- The objective: for each entity-mention, select the entity/entities that properly reflect(s) the semantics of the mention
  - the selection is done from, often numerous, candidate entities identified in the previous phase

- Continuing with the same example:

  "They performed Kashmir, written by Page and Plant.
  Page played unusual chords on his Gibson."

dbpedia:Kashmir – a valley between Pakistan, India and Ladakh
dbpedia:Kashmir_(band) – a Danish rock band
dbpedia:Kashmir_(song) – 1975 song by rock band Led Zeppelin
dbpedia:Kashmir,_Iran – a village in Iran
…

DISAMBIGUATION PHASE

Different kinds of approaches for completing this task:

- Popularity-based (mention-entity) prior
- Context-based approach
- Collective disambiguation

POPULARITY-BASED (MENTION-ENTITY) PRIOR

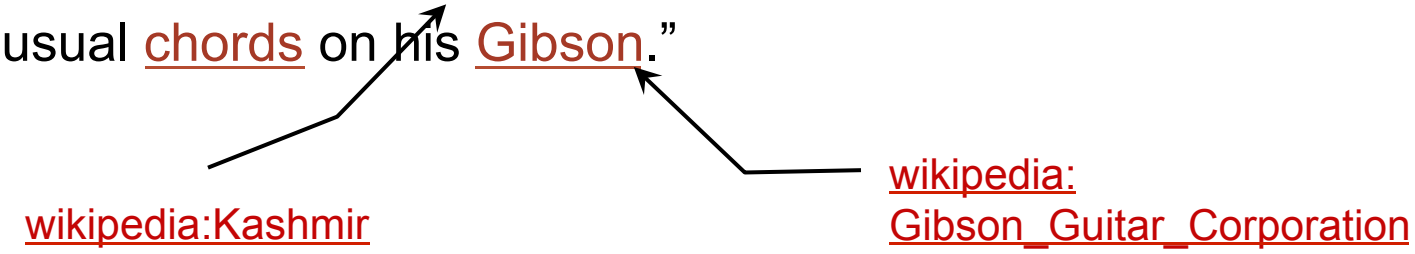
This approach consists of choosing the most prominent entity for a given mention

- E.g., the entity for which the given mention most frequently serves as the anchor text in Wikipedia

POPULARITY-BASED (MENTION-ENTITY) PRIOR

Continuing with the same example

"They performed Kashmir, written by Page and Plant. Page played unusual chords on his Gibson."



In Wikipedia,

- "Gibson" is primarily associated with the Gibson guitar corporation and only marginally with the other 24 possible meanings; however,
- "Kashmir" is predominantly associated with the Kashmir region (90.91% of all the occurrences), and rarely refers to the Led Zeppelin song (5.45%)

POPULARITY-BASED (MENTION-ENTITY) PRIOR

- This is a simple, but often erroneous approach; therefore, it is often used in combination with other approaches
- Errors occur due to the lack of proper attention to
  - the mention's context, and
  - the theme of the overall text

An example illustrating the error that tends to occur if only popularity (commonness) is considered

Depth-first search

From Wikipedia, the free encyclopedia

Depth-first search (**DFS**) is an algorithm for traversing or searching a tree tree structure or graph. One starts at the root (selecting some node as the root in the graph case) and explores as far as possible along each branch before backtracking.

Formally, DFS is an uninformed search that progresses by expanding the first child node of the search tree that appears and thus going deeper and deeper until a goal node is found, or until it hits a node that has no children. Then the search backtracks, returning to the most recent node it hadn't finished exploring. In a non-recursive implementation, all freshly expanded nodes are added to a LIFO stack for exploration.

sense	commonness	relatedness
Tree	92.82%	15.97%
Tree (graph theory)	2.94%	59.91%
Tree (data structure)	**2.57%**	**63.26%**
Tree (set theory)	0.15%	34.04%
Phylogenetic tree	0.07%	20.33%
Christmas tree	0.07%	0.0%
Binary tree	0.04%	62.43%
Family tree	0.04%	16.31%
…		

DISAMBIGUATION: CONTEXT-BASED APPROACH

- Frequently used approach for entity disambiguation
- Relies on a comparison of the context of an entity-mention, and the context of each of the candidate entities
- Typically,
  - context of a mention is the sentence it appears in; context of an entity is its description in the KB
  - context is represented as a bag-of-words, and the comparison is done using some similarity measure
    - frequently used metrics: Cosine similarity, weighted Jaccard coefficient, Wikipedia link-based measures

DISAMBIGUATION: CONTEXT-BASED APPROACH

"They performed Kashmir, written by Page and Plant. Page played unusual chords on his Gibson."

bag-of-words →

perform
Kashmir
write
Page
Plant
play
chord
...

http://en.wikipedia.org/wiki/Kashmir_(song)
…was written by Jimmy Page and Robert Plant…
…performed by the band at almost every concert…

bag-of-words →

write
Jimmy
Page
Robert
Plant
perform
band
concert
...

similarity

+ 15 more candidate entities

http://en.wikipedia.org/wiki/Kashmir
…northwestern region of the Indian subcontinent.…
…became an important center of Hinduism and later of Buddhism…

bag-of-words →

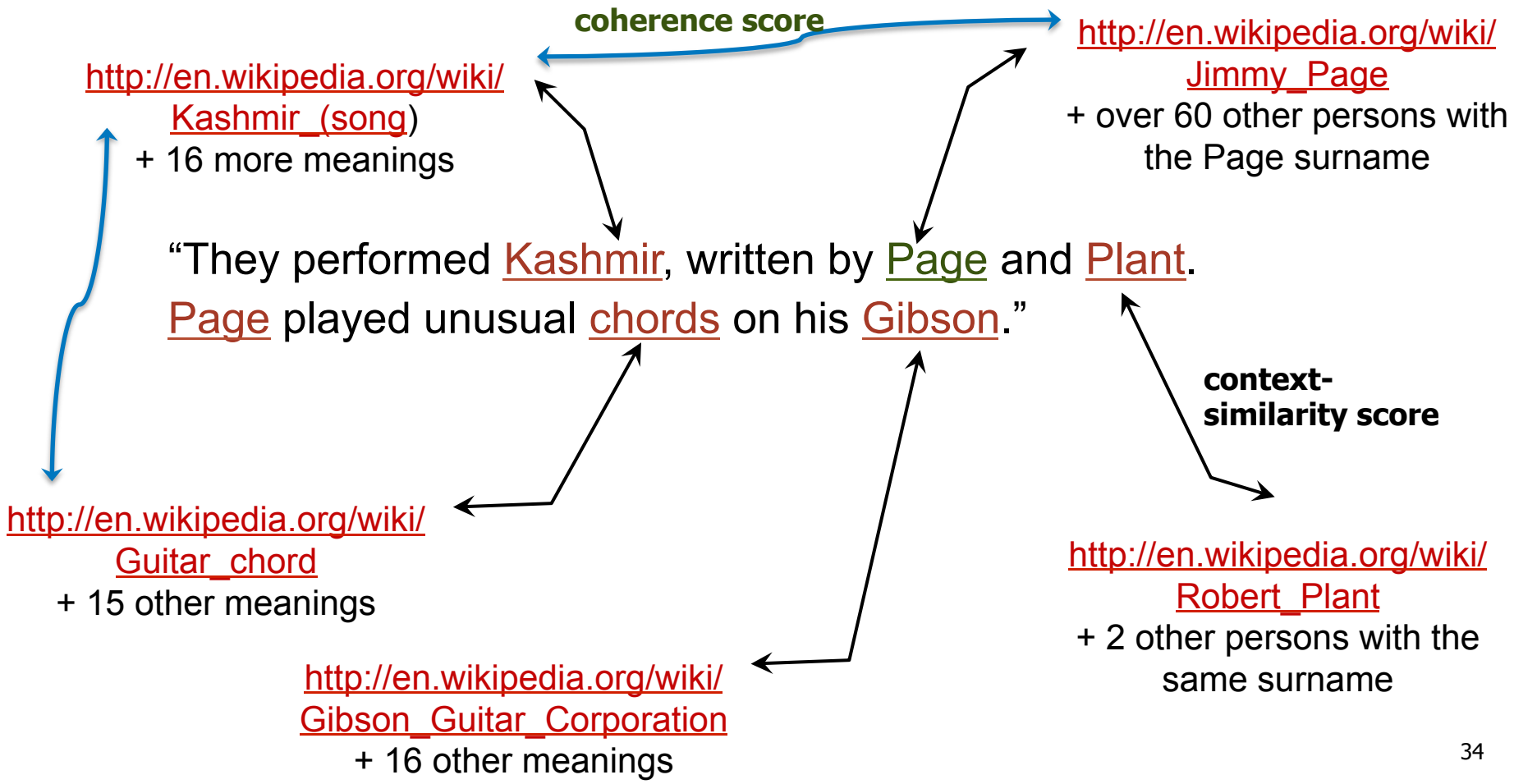
northwest
region
India
subcontinent
center
Hinduism
Buddhism
...

COLLECTIVE DISAMBIGUATION

- Consists of jointly/simultaneously disambiguating multiple mentions in the input text
- An extension of the context-based approach:
  - *context similarity scores* of each mention-entity pair are combined with the *coherence scores* of the target entities
  - coherence is defined as semantic relatedness of entities

COLLECTIVE DISAMBIGUATION

Continuing with the previous example:



COLLECTIVE DISAMBIGUATION

- Performs well when
  - there is a sufficiently large number of mentions in the input text, *and*
  - the entities form a thematically homogeneous set
- Errors tend to occur when
  - the text covers multiple, unrelated or weakly related topics
  - mentions are associated with entities that can form more than one coherent topic; example:

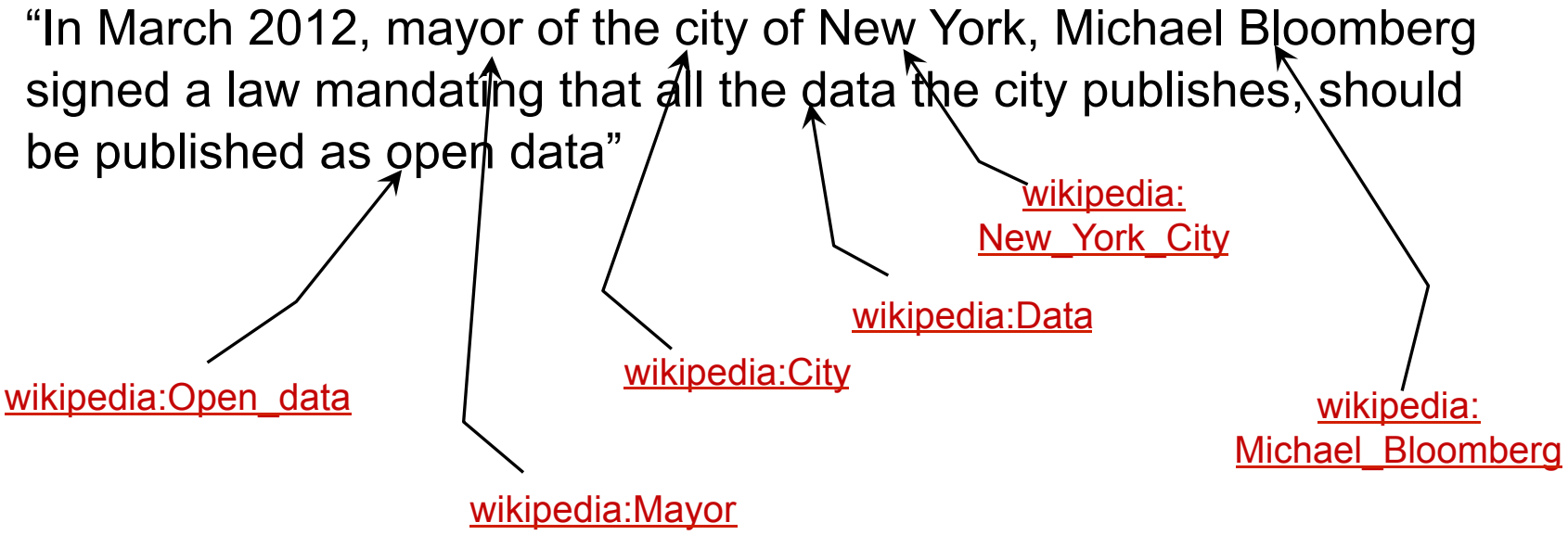
    "Real Madrid and Barcelona edge out Manchester and Chelsea to secure trials for Argentine wonder-kid"

    These mentions might be associated with two coherent sets of entities: locations (cities) and football clubs

FILTERING PHASE

- The objective is to remove results (entities) that are expected not to be of interest to the user
  - e.g., overly general entities or those that are only marginally related to the main topic of the text

- Example



Performance of today's tools for semantic indexing

Text type: news articles

	AlchemyAPI	dataTXT	DBpedia	Lupedia	Textrazor	THD	Yahoo!	Zemanta
p	**70.63**	39.20	26.93	57.98	49.21	32.50	61.24	35.58
r	14.05	**54.93**	42.21	29.90	51.66	40.10	9.65	7.78
f	23.43	45.75	32.88	39.45	**50.41**	35.90	16.68	12.77

p – precision; r – recall; f – F measure

(note dataTXT evolved into commercial service DandelionAPI)

Source: Rizzo et al. 2014. Benchmarking the Extraction and Disambiguation of Named Entities on the Semantic Web, 9th Edition of the Language Resources and Evaluation Conference (LREC'14)

Performance of today's tools for semantic indexing

Text type: Twitter messages

	AlchemyAPI	dataTXT	DBpedia	Lupedia	Textrazor	THD	Yahoo!	Zemanta
p	**72.22**	22.11	13.99	37.37	30.69	23.54	60.68	35.54
r	3.91	34.74	29.70	11.13	**34.89**	23.98	10.68	10.08
f	7.42	27.02	19.02	17.15	**32.65**	23.76	18.16	15.70

p – precision; r – recall; f – F measure

Source: Rizzo et al. 2014. Benchmarking the Extraction and Disambiguation of Named Entities on the Semantic Web, 9th Edition of the Language Resources and Evaluation Conference (LREC'14)

WIKILINKS CORPUS

- The largest public dataset for training supervised ML algorithms for the task of recognizing Wikipedia entities in the text
- URL: http://www.iesl.cs.umass.edu/data/wiki-links
- Some basic facts about this corpus:
  - 10 million annotated Web pages
  - 3 million Wikipedia entities
  - 40 million uniquely identified entity mentions
  - published by Google Research on March 8, 2013.
- Read more about this dataset in the following article: Learning from Big Data: 40 Million Entities in Context

Freebase Annotations of Social Media Content

- Google Freebase Annotations of TREC KBA 2014 Stream Corpus
  - TREC – Text Retrieval Conference
  - KBA – Knowledge Base Acceleration
- URL: http://trec-kba.org/data/fakba1/index.shtml
- Some basic facts about this corpus:
  - 394M documents with at least one entity annotated
  - 9.4 billion entity mentions with links to Freebase
  - annotation was performed automatically and is imperfect
  - based on manually inspected random sample, it is estimated that:
    - ~9% of the mentions may be linked to an incorrect Freebase entity
    - ~8% of the mentions that should be assigned Freebase entity are missed

(Anonymous) questionnaire for your critique, comments, suggestions:

http://goo.gl/cqdp3I