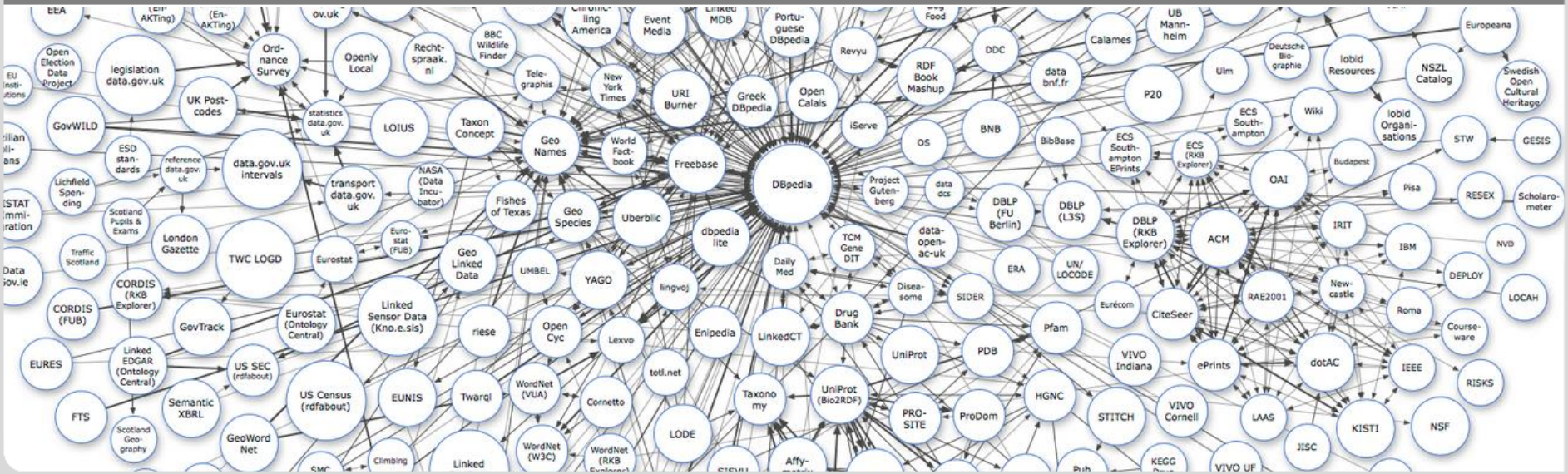
TRM – Learning Dependencies between Text and Structure with Topical Relational Models

Veli Bicer[1], Thanh Tran[2], Yongtao Ma[2] and **Rudi Studer**[2]

[1]IBM Research, Smart Cities Technology Center, Dublin, Ireland
[2]Institute AIFB, Karlsruhe Institute of Technology, Germany

12th ISWC, Sydney, October 2013



Agenda

- Introduction
  - Motivation
  - RDF Data
  - State-of-the-Art
- Topical Relational Model (TRM)
  - Overview
  - Template-based Construction
  - TRM Output
- Evaluation
- Conclusion

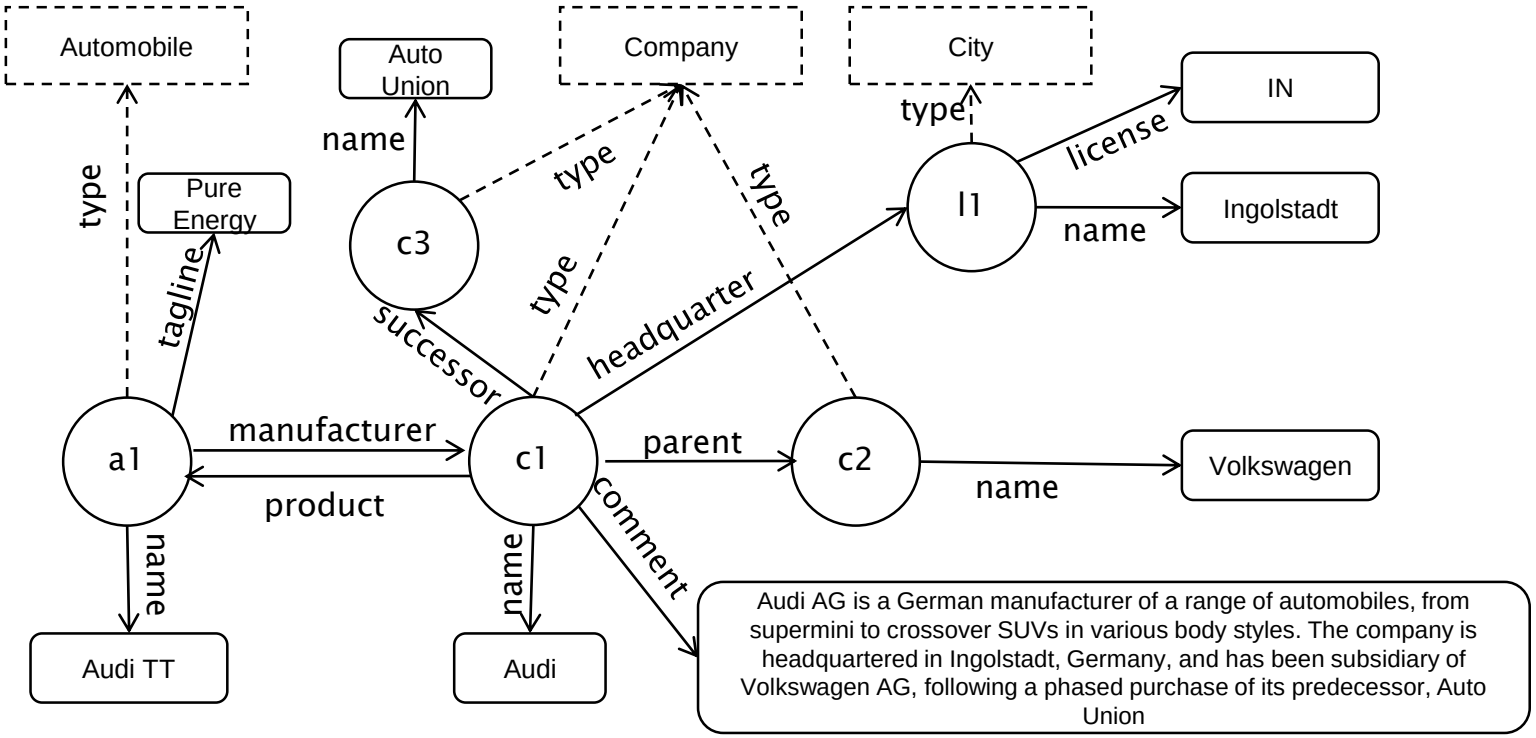
Agenda

- **Introduction**
  - **Motivation**
  - **RDF Data**
  - **State-of-the-Art**
- Topical Relational Model (TRM)
  - Overview
  - Template-based Construction
  - TRM Output
  - Applications
- Evaluation
- Conclusion

Motivation

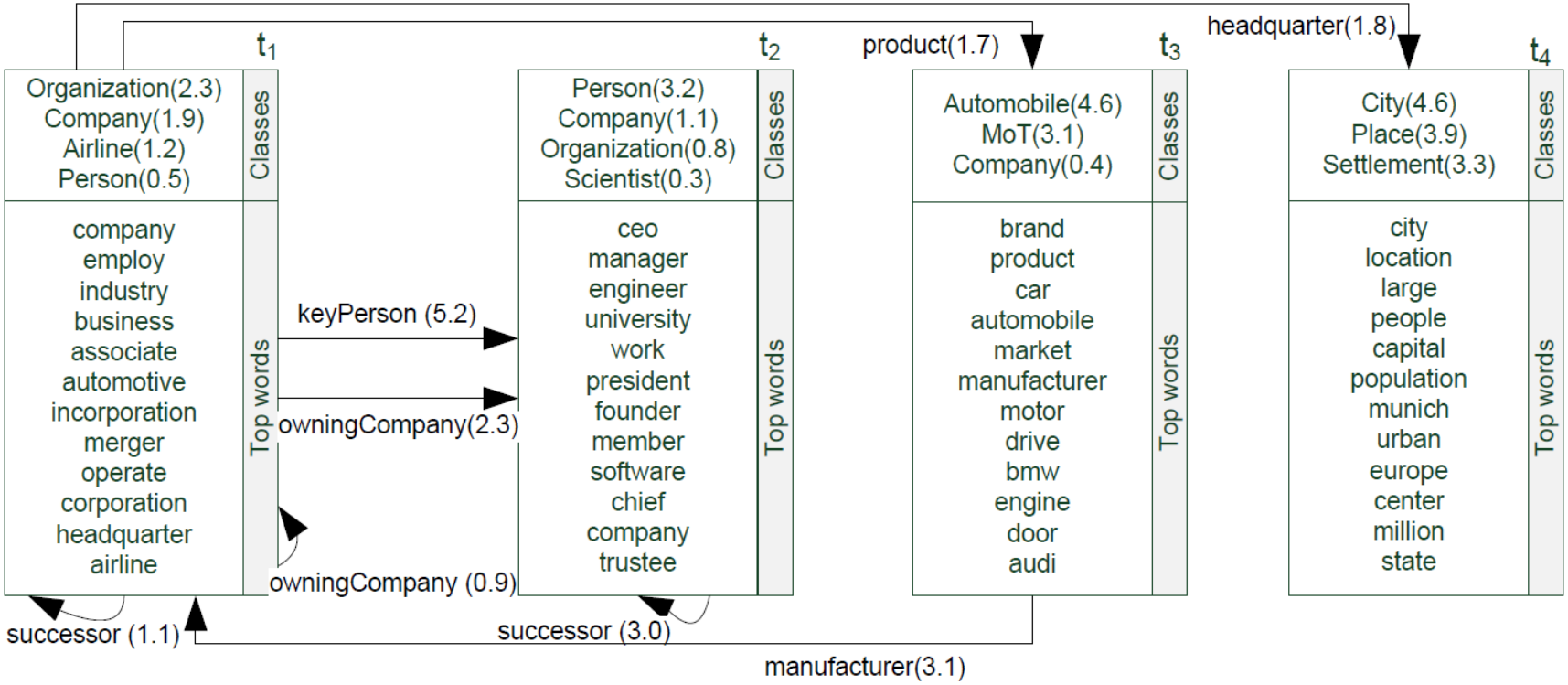
- Bridging the gap between structured and unstructured data:
  - Text-rich structured data is emerging on the Web in the form of RDF
  - Characteristics of RDF data
    - structured data in the form of entities, classes, and relations
    - unstructured data associated with attributes
- Topic Models
  - Model topics as distribution over words
  - Various models considering relational data
    - E.g., social networks, citations, Web links
  - Unique characteristics of RDF are not taken into account
- Statistical Relational Learning
  - Learning probabilistic models from relational data
  - Handling text is a problem

RDF Data



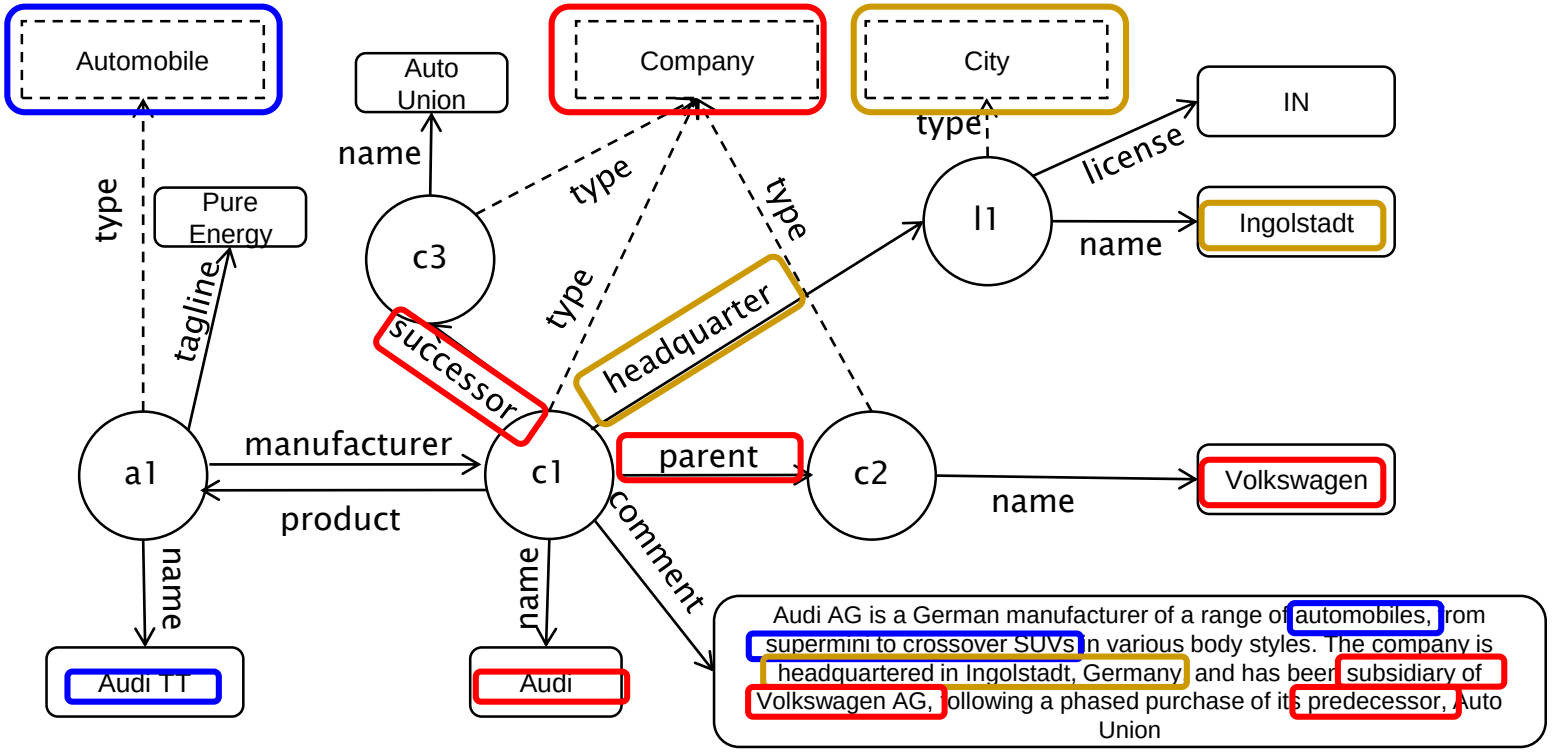
- RDF graph: Vertices denote **resources** and text **values**, connected via **relations** and **attributes**
- **Semi-structured:** Containing both structured and unstructured data

Topics to be learned



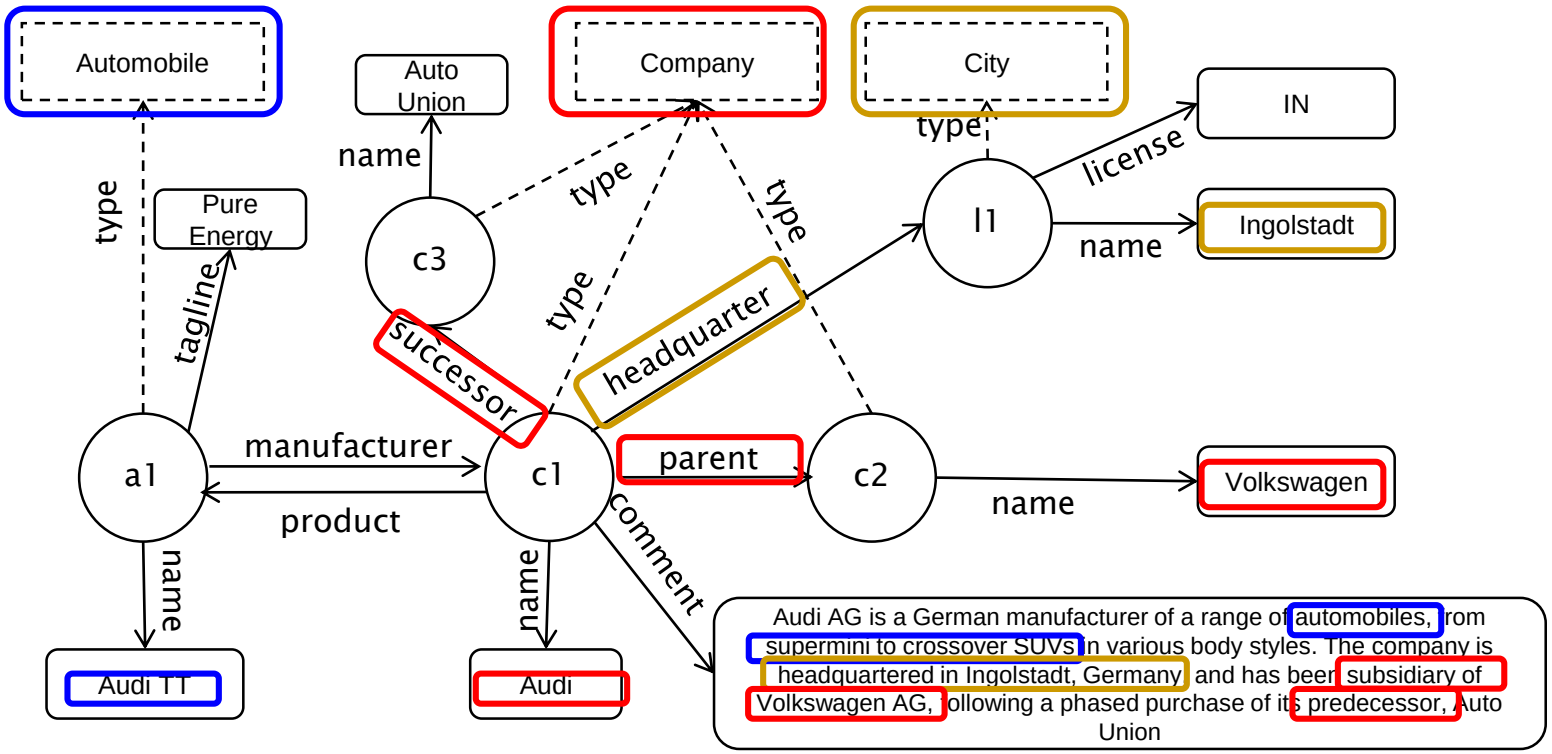
- Topics have to cover words, classes and relations

RDF Data



- RDF graph: Vertices denote **resources** and text **values**, connected via **relations** and **attributes**
- **Semi-structured:** Containing both structured and unstructured data
- **Highly heterogeneous:** Many classes and relations each of which has varying effects on different topics
- **Sparseness:** Different structure elements have sparse correlations to the topics in the text

RDF Data



- RDF graph: Vertices denote **resources** and text **values**, connected via **relations** and **attributes**
- **Semi-structured:** Containing both structured and unstructured data

-**Highly heterogeneous:** Many classes and relations each of which has varying effects on different topics

-**Sparseness:** Different structure elements have sparse correlations to the topics in the text

-**Challenges:**

-**How to learn topics which better reflect RDF structures**

-**How to learn correlations between topics and structure taking heterogeneity and sparseness into account**

State-of-the-art

- Supervised Topic Models
  - sLDA: word co-occurance and multi-label documents
  - Type-LDA: focus on relation extraction in NLP
- Topic Models for homogeneous networks
  - RelLDA, Pairwise Link-LDA: words and links in hypertext
  - Nubbi, Author-Topic models, Multi-relational TMs
- Topic Models for heterogeneous networks
  - Topic Model with Biased Propagation (TMBP)
    - Limits in capturing complex correlations between structure and topics
    - Sparsity not well addressed
- Statistical Relational Learning
  - MLN: FOL rules for text
  - C^3: SVM based on textual features (e.g. Jaccard)

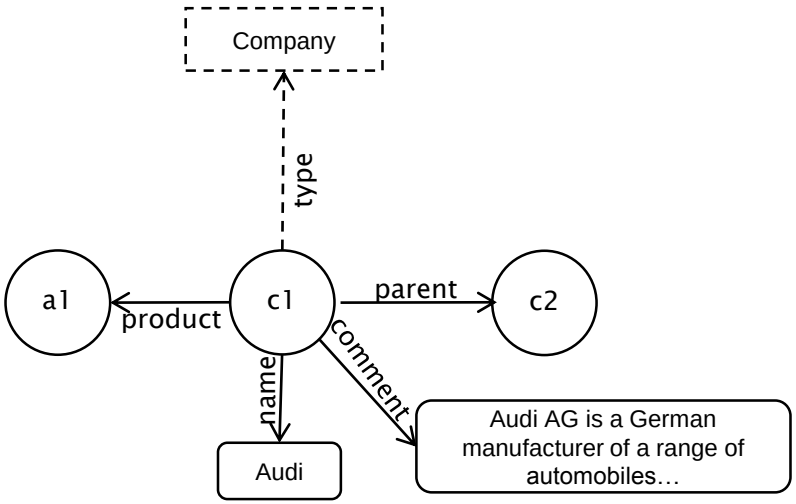
Agenda

- Introduction
  - Motivation
  - RDF Data
  - State-of-the-Art
- **Topical Relational Model (TRM)**
  - **Overview**
  - **Template-based Construction**
  - **TRM Output**
- Evaluation
- Conclusion

Topical Relational Models

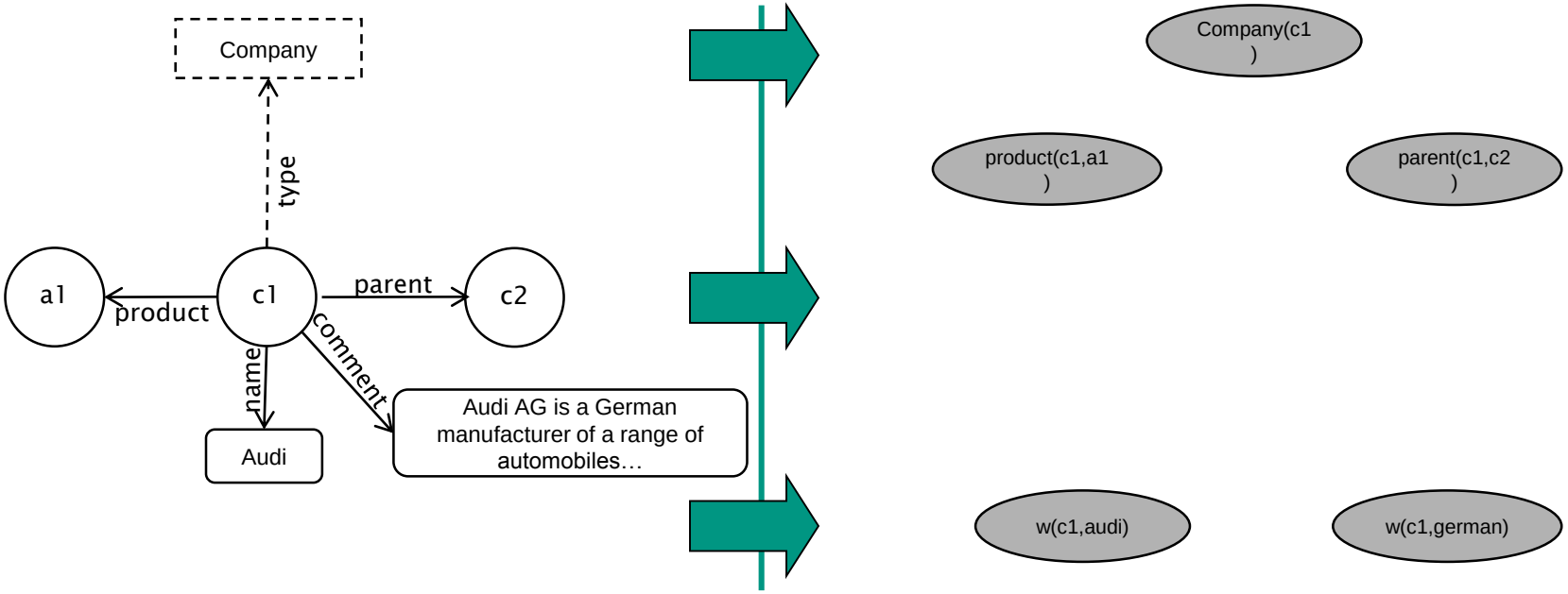
- A model to learn a set of topics and their correlations to the classes, relations and texts from RDF
- Topics as a low-dimensional representation of text in an RDF graph
  - Biased towards the structure
    - e.g. words such as '*employ*' or '*merger*' correlate with classes such as *Company*
- Correlations between the class (relation) and the topics are captured as a vector (matrix) of weights
  - e.g. the topics which assign high probabilities to words '*employ*' or *'merger'* have high weights in the vector of the class *Company*
- Learned using a variational Bayesian EM
  - Model is intractable for exact inference

Template-based Construction



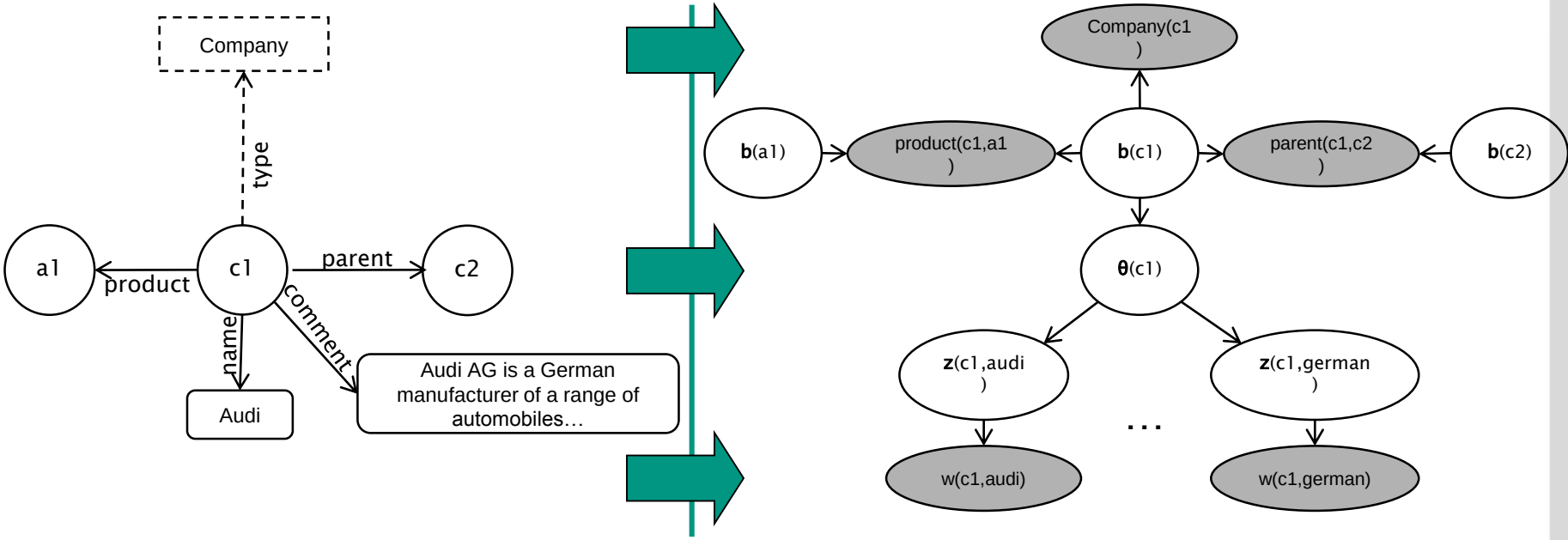
- TRM uses an RDF graph as a template to construct a ground BN

Template-based Construction



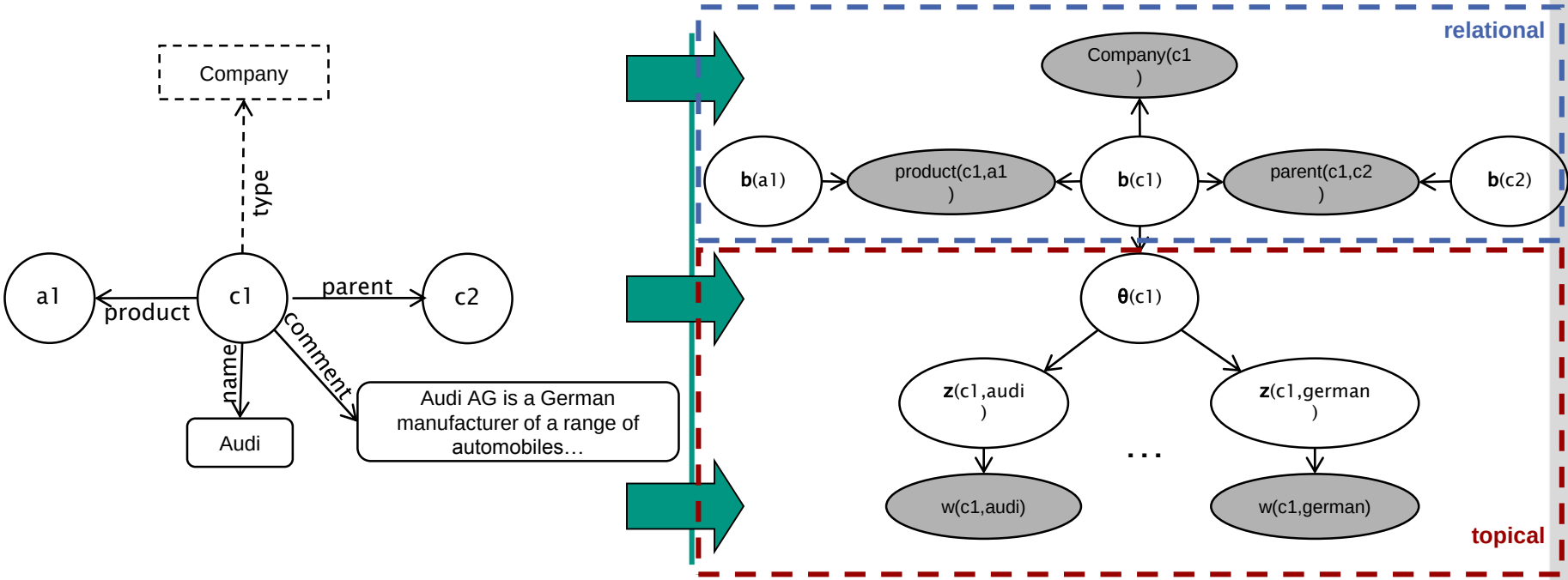
- TRM uses an RDF graph as a template to construct a ground BN
- **Three types of templates for observed variables:** Classes, relations, and words
- Random variables are instantiated for each resource from the template

Template-based Construction



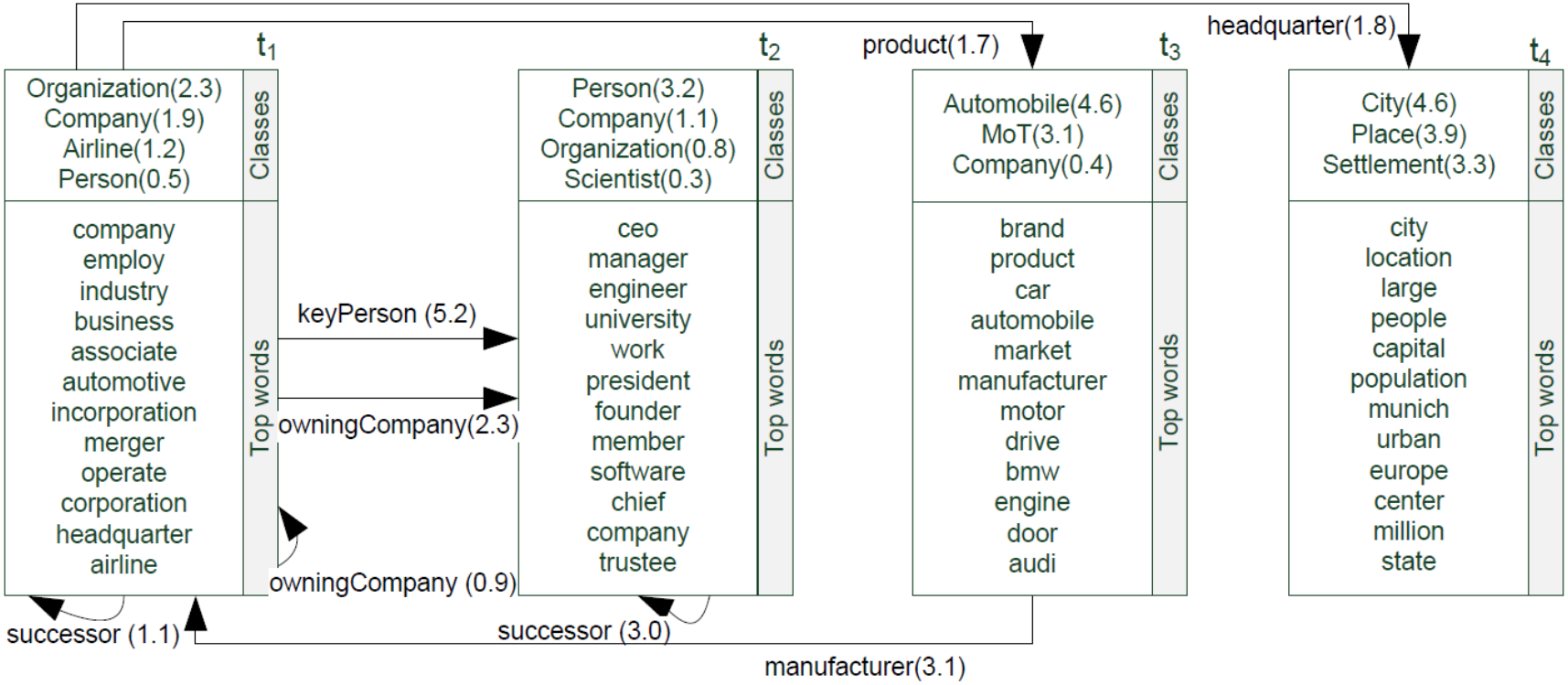
- TRM uses an RDF graph as a template to construct a ground BN
- **Three types of templates for observed variables:** Classes, relations, and words
- Random variables are instantiated for each resource from the template
- Hidden topic-related variables:
  - Topic indicator vector **b**
  - Topic proportions **θ**
  - Topic-word assignments **z**

Template-based Construction

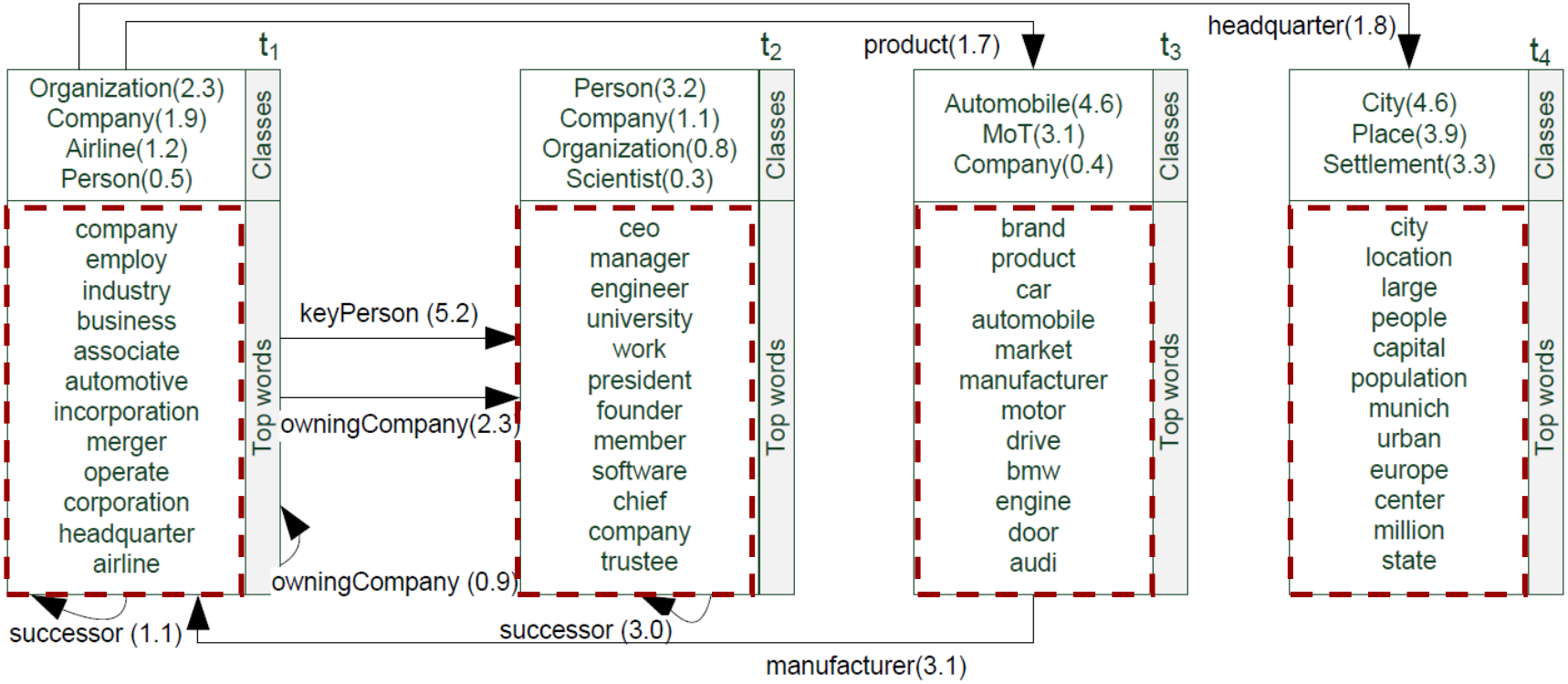


- TRM uses an RDF graph as a template to construct a ground BN
- **Three types of templates for observed variables:** Classes, relations, and words
- Random variables are instantiated for each resource from the template
- Hidden topic-related variables:
  - Topic indicator vector **b**
  - Topic proportions **θ**
  - Topic-word assignments **z**
- The ground BN can be considered as two parts (topical, relational) managed by **b vector**

TRM Output


t1
t2
t3
t4
Organization(2.3)
Company(1.9)
Airline(1.2)
Person(0.5)
Classes
company
employ
industry
business
associate
automotive
incorporation
merger
operate
corporation
headquarter
airline
Top words
Person(3.2)
Company(1.1)
Organization(0.8)
Scientist(0.3)
ceo
manager
engineer
university
work
president
founder
member
software
chief
company
trustee
Automobile(4.6)
MoT(3.1)
Company(0.4)
brand
product
car
automobile
market
manufacturer
motor
drive
bmw
engine
door
audi
City(4.6)
Place(3.9)
Settlement(3.3)
city
location
large
people
capital
population
munich
urban
europe
center
million
state
keyPerson (5.2)
owningCompany(2.3)
owningCompany (0.9)
product(1.7)
headquarter(1.8)
successor (1.1)
successor (3.0)
manufacturer(3.1)

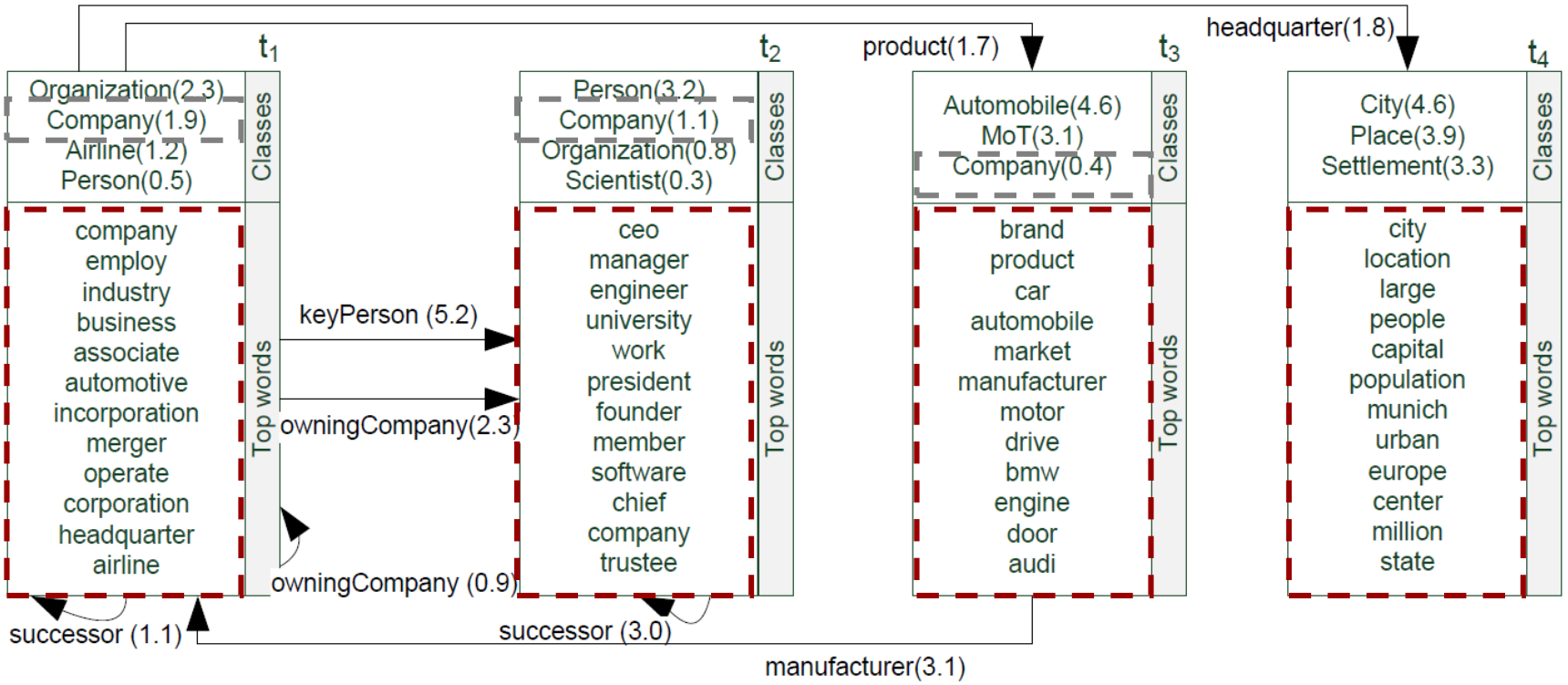
TRM Output



-TRM discovers the topics biased towards the structure

-Topics are multinomial distributions over the words in the vocabulary

TRM Output

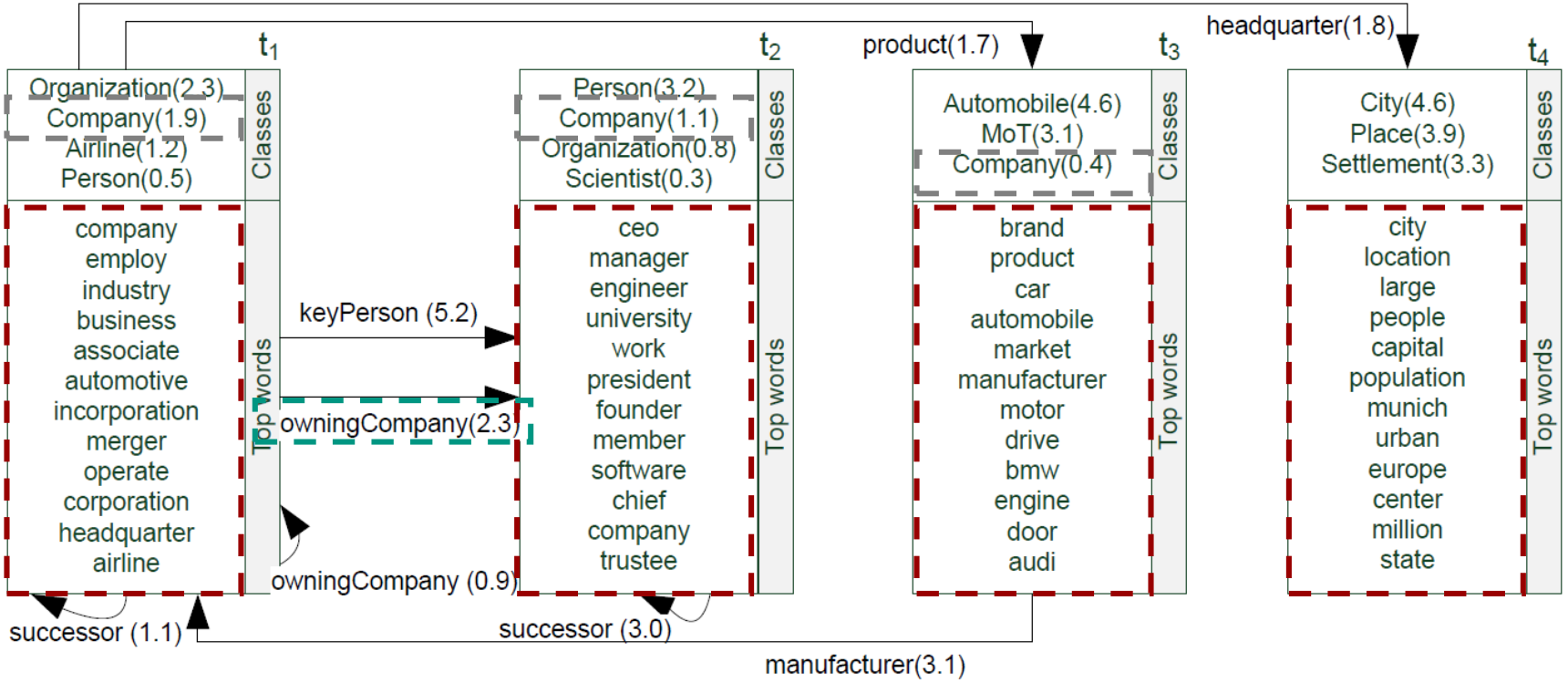


-TRM discovers the topics biased towards the structure

-Topics are multinomial distributions over the words in the vocabulary

-TRM captures the correlations between those topics and the classes via the weights in its class-topic parameter

TRM Output



-TRM discovers the topics biased towards the structure

-Topics are multinomial distributions over the words in the vocabulary

-TRM captures the correlations between those topics and the classes via the weights in its class-topic parameter

-The weight of observing a relation between any two topics is encoded in its relation-topic parameter (by taking the direction of the relation into account)

Agenda

- Introduction
  - Motivation
  - RDF Data
  - State-of-the-Art
- Topical Relational Model (TRM)
  - Overview
  - Template-based Construction
  - TRM Output
- **Evaluation**
- Conclusion

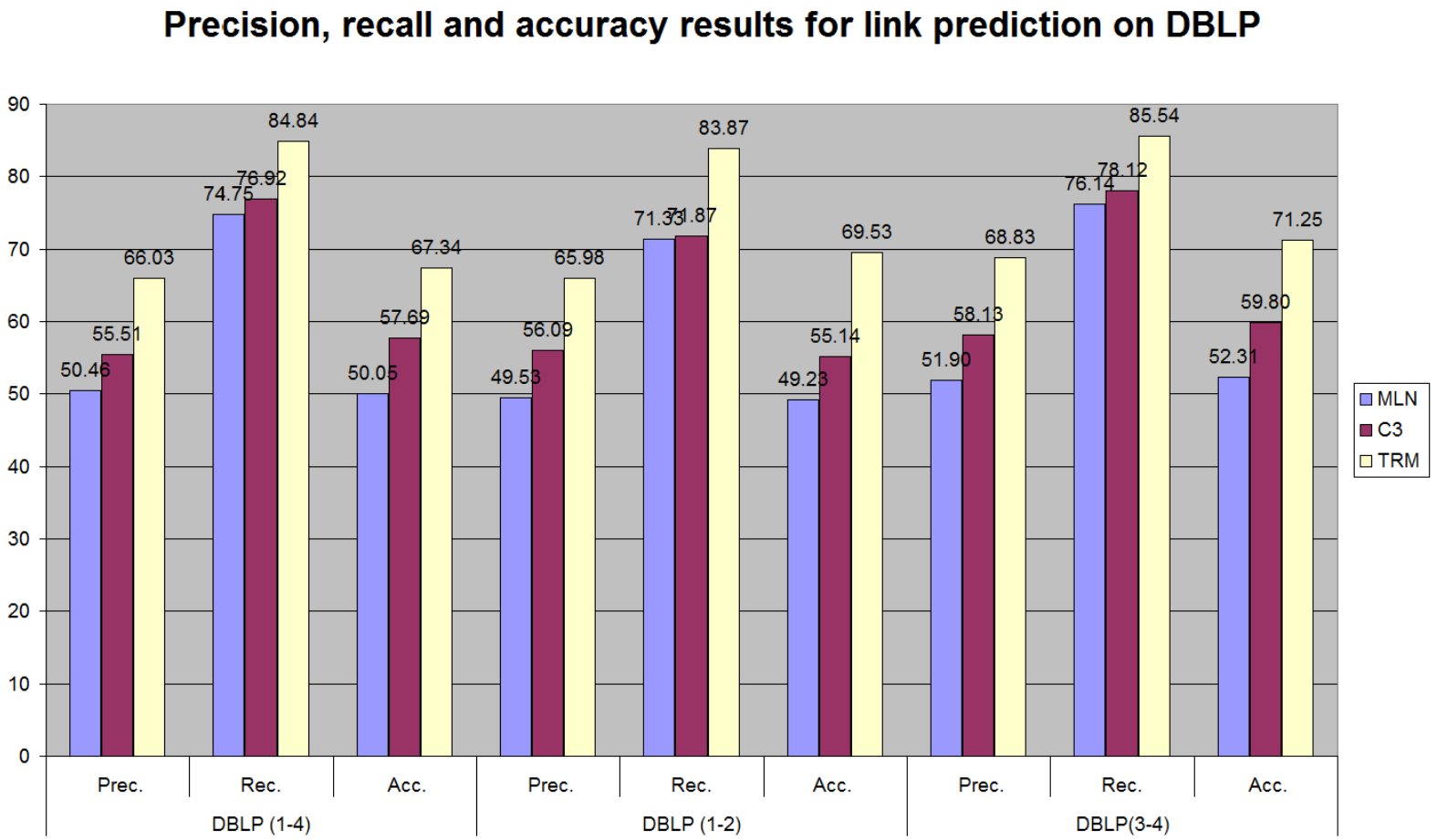
Evaluation

- Datasets:
  - DBLP: Authors, conferences and their relations to papers
    - 28,569 papers, 28,702 authors and 20 conferences
    - The abstract and title of the papers are treated as textual data
  - DBpedia: Movie domain
    - 20,094 entities described by 112 distinct classes and 49 different types of relations
    - All attribute values are treated as textual data
- Tasks:
  - **Link Prediction**
  - Object Clustering

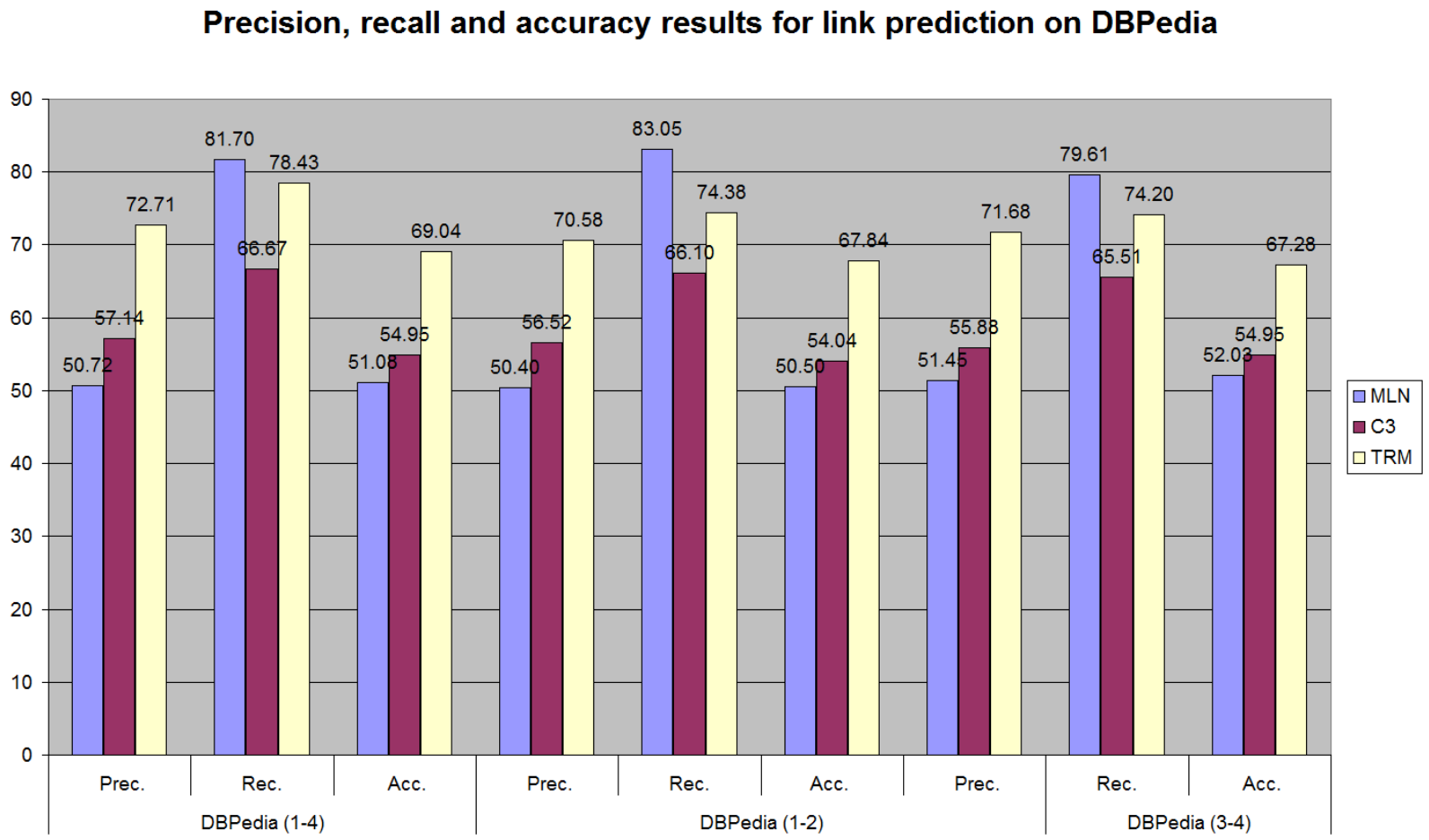
Evaluation

- Link Prediction (LP)
  - Predicting ***author*** relations between papers and authors in DBLP
  - *Predicting* ***starring*** relations between movies and actors in DBpedia
- Baselines:
  - MLN [Richardson and Domingos, 2006]
  - C^3 [Namata et al., 2011]

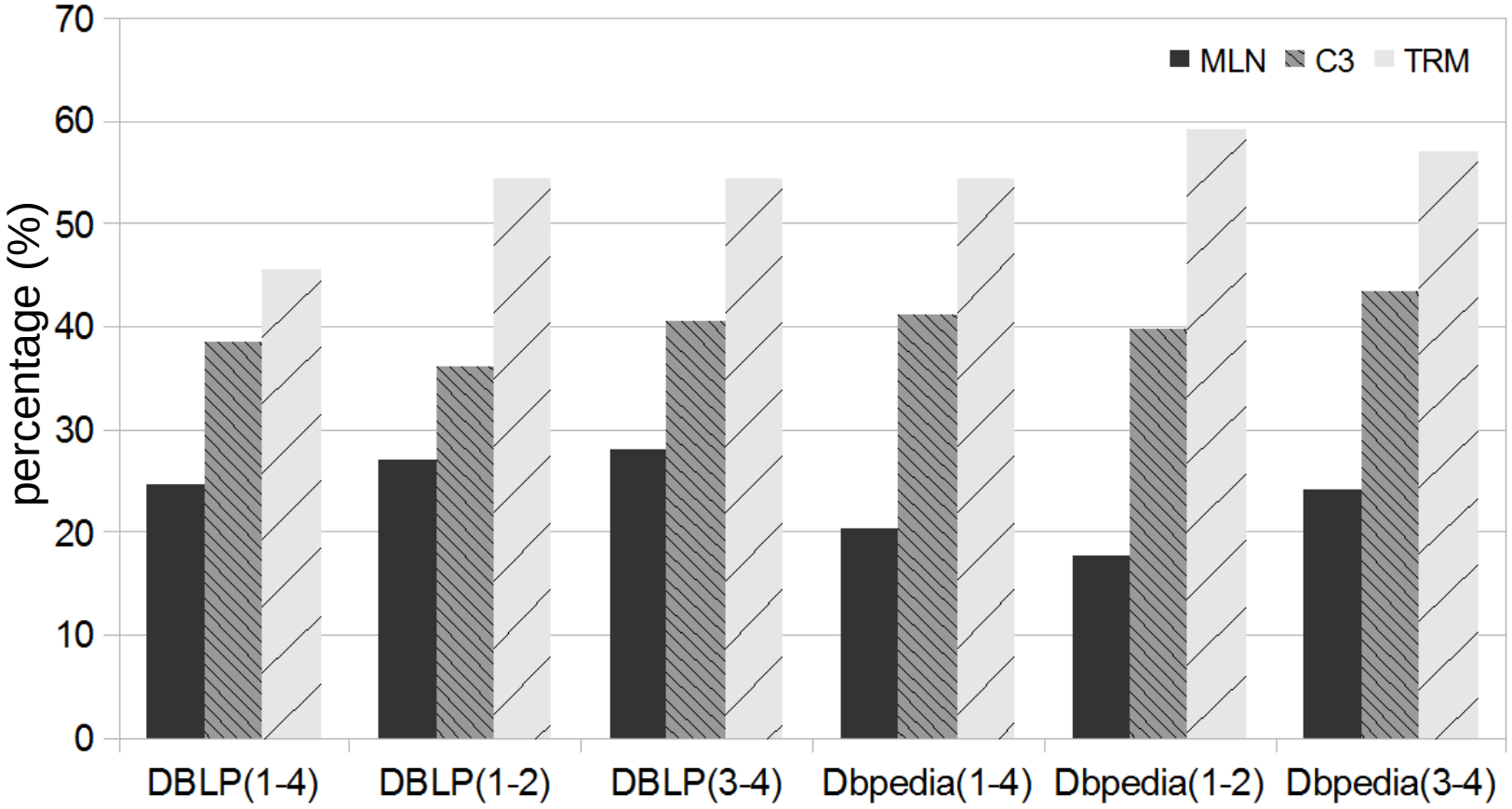
Evaluation


Precision, recall and accuracy results for link prediction on DBLP
90
80
70
60
50
40
30
20
10
0
50.46
55.51
66.03
74.75
76.92
84.84
50.05
57.69
67.34
49.53
56.09
65.98
71.33
71.87
83.87
49.23
55.14
69.53
51.90
58.13
68.83
76.14
78.12
85.54
52.31
59.80
71.25
Prec.
Rec.
Acc.
DBLP (1-4)
DBLP (1-2)
DBLP(3-4)
MLN
C3
TRM

Evaluation


Precision, recall and accuracy results for link prediction on DBPedia
90
80
70
60
50
40
30
20
10
0
50.72
57.14
72.71
81.70
66.67
78.43
51.08
54.95
69.04
50.40
56.52
70.58
83.05
66.10
74.38
50.50
54.04
67.84
51.45
55.88
71.68
79.61
65.51
74.20
52.03
54.95
67.28
Prec.
Rec.
Acc.
DBPedia (1-4)
DBPedia (1-2)
DBPedia (3-4)
MLN
C3
TRM

Evaluation



True negative rate for DBLP and DBpedia

Agenda

- Introduction
  - Motivation
  - RDF Data
  - State-of-the-Art
- Topical Relational Model (TRM)
  - Overview
  - Template-based Construction
  - TRM Output
- Evaluation
- **Conclusion**

Conclusion

- TRM provides an effective model for text-rich RDF data
- TRM captures dependencies between words in textual and structured data
- Compared to existing TM approaches, TRM is more effective in exploiting structure information

- Application Opportunities
  - Link prediction, object clustering
  - Estimating the result size of hybrid SPARQL queries
  - Keyword search on RDF data: Estimating the most probable connections between entities given the query

- Future work:
  - Extension of TRM to richer generative models, such as time-varying and hierarchical topic models

Applications

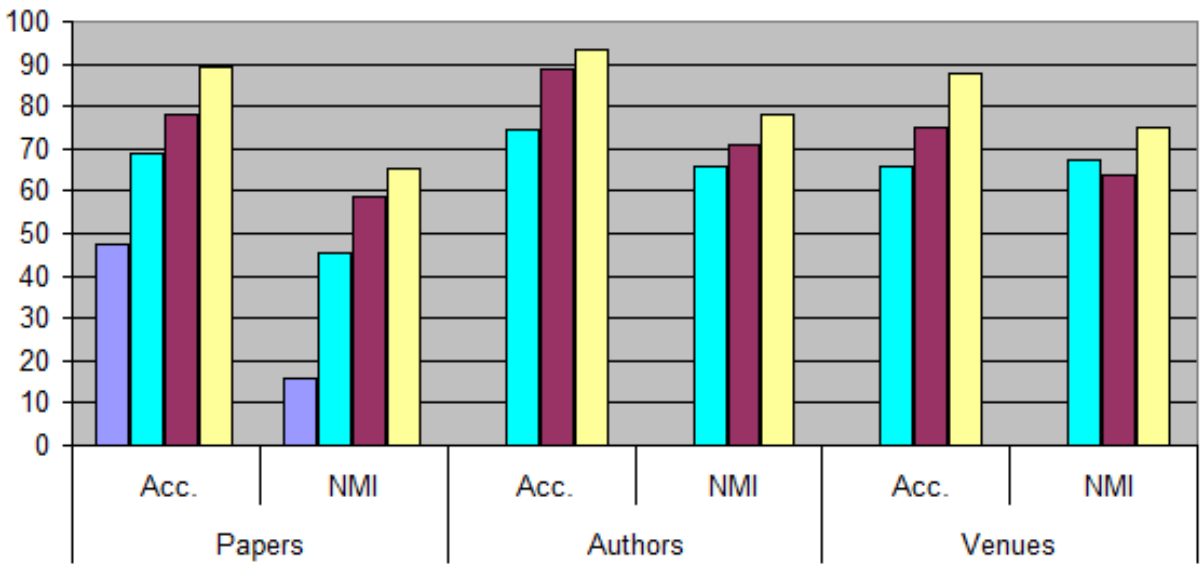
- Selectivity Estimation on RDF [Wagner et al. 2014]
  - Estimating the result size of a hybrid SPARQL query, i.e. query with structural and textual predicates
  - TRM provides a uniform data synopsis to summarize the text-rich structured data
    - estimation is based on using TRM output of topic distributions, class-topic and relation-topic parameters
- Keyword Search on RDF Data
  - State-of-the-art: Keywords are mapped to RDF elements and connections are discovered among those elements over the schema [Tran et al., ICDE 2009]
  - TRM provides a probabilistic schema capturing the most probable connections given the query

Evaluation

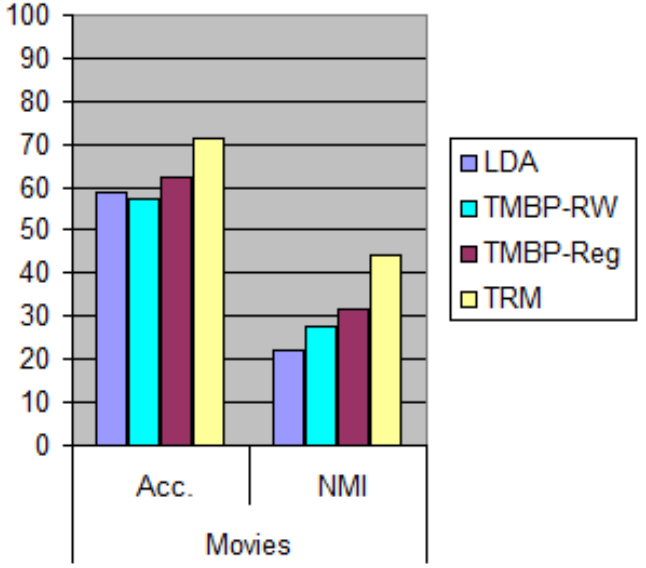
- Object Clustering
  - **DBPedia**:
    Clustering movie entities into different types
    - e.g. American, British, German movies
  - **DBLP**:
    Use the six labels in DBLP representing various computer science fields as clusters
    - cluster paper, author and venue entities
- Baselines:
  - LDA, TMBP-RW, TMBP-Reg

Evaluation

Accuracy and Normalized Mutual Information (NMI) results for object clustering



DBLP



DBPedia