Construction and Querying of Large-scale Knowledge Bases

Xiang Ren\textsuperscript{1}  Yu Su\textsuperscript{2}  Xifeng Yan\textsuperscript{2}

University of Southern California\textsuperscript{1}
University of California, Santa Barbara\textsuperscript{2}
Tutorial website:
http://xren7.web.engr.illinois.edu/tutorial-cikm17.html

Slides, code, datasets, references
Turning Unstructured Text Data into Structures

Unstructured Text Data (account for ~80% of all data in organizations)

Structures

Knowledge & Insights

(Chakraborty, 2016)
Reading the reviews: From Text to Structured Facts

This **hotel** is my favorite **Hilton property** in **NYC**! It is located right on 42nd street near **Times Square**, it is close to all subways, **Broadways shows**, and next to great restaurants like **Junior’s Cheesecake**, **Virgil’s BBQ** and many others.

--- TripAdvisor

Structured Facts

1. “Typed” entities
2. “Typed” relationships
Why Text to Structures?

Structured Search & Exploration

- SQL
  - City contains "ist"
  - Category equals "Friends"
  - Birthday on 09/04/2000, Age = 30
  - Lastname equals "plugins"
- SPARQL
  - Is active

Graph Mining & Network Analysis

Pattern / Association Rule Mining

Structured Feature Generation

Input Space

Feature Space
A Product Use Case: Finding “Interesting Hotel Collections”

Technology Transfer to TripAdvisor

Features for “Catch a Show” collection
1. broadway shows
2. beacon theater
3. broadway dance center
4. broadway plays
5. david letterman show
6. radio city music hall
7. theatre shows

Features for “Near The High Line” collection
1. high line park
2. chelsea market
3. highline walkway
4. elevated park
5. meatpacking district
6. west side
7. old railway

Grouping hotels based on structured facts extracted from the review text

**Prior Art: Extracting Structures with Repeated Human Effort**

Extraction Rules
Machine-Learning Models

Structured Facts
- Broadways shows
- NYC
- Times square
- hotel
- Hilton property

Labeled data
- Text Corpus

... We had a room facing *Times Square* and a room facing the *Empire State Building*, The location is close to everything and we love ...

This hotel is my favorite Hilton property in NYC! It is located right on 42nd street near Times Square, it is close to all subways, Broadways shows, and next to many great ...
This Tutorial: Effort-Light StructMine

- Enables **quick** development of applications over various corpora
- Extracts **complex** structures without introducing human error
Effort–Light StructMine: Where Are We?

A Review of Previous Efforts

Human labeling effort

Supervised learning systems

Hand-crafted Systems

UCB Hearst Pattern, 1992
NYU Proteus, 1997

Weakly-supervised learning systems

CMU NELL, 2009 - present
UT Austin Dependency Kernel, 2005
IBM Watson Language APIs

Distantly-supervised Learning Systems

Stanford CoreNLP, 2005 - present
Stanford DeepDive, MIML-RE 2012 - present
UW FIGER, MultiR, 2012

Effort-Light StructMine
(WWW’15, KDD’15, KDD’16, EMNLP’16, WWW’17, …)

Feature engineering effort
“Distant” Supervision: What Is It?

“Matchable” structures: entity names, entity types, typed relationships ...

Freely available!
- Common knowledge
- Life sciences
- Art ...

Rapidly growing!

Number of Wikipedia articles

Human crowds

(Mintz et al., 2009), (Riedek et al., 2010), (Lin et al., 2012), (Ling et al., 2012), (Surdeanu et al., 2012), (Xu et al., 2013), (Nagesh et al., 2014), ...

Learning with Distant Supervision: Challenges

1. Sparsity of “Matchable”
   - Incomplete knowledge bases
   - Low-confidence matching

2. Accuracy of “Expansion”
   - For “matchable”: Are all the labels assigned accurately?
   - For “un-matchable”: How to perform inference accurately?

(Ren et al., KDD’15)

It is my favorite city in the United States. The United States needs a new strategy to meet this challenge.
**Effort-Light StructMine: Contributions**

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Solution Idea</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sparsity of “Matchable”</td>
<td>Effective expansion from “matchable” to “un-matchable”</td>
</tr>
<tr>
<td>Accuracy of “Expansion”</td>
<td>Pick the “best” labels based on the context (for both “matchable” and “un-matchable”)</td>
</tr>
</tbody>
</table>

Harness the “data redundancy” using graph-based joint optimization.

It is my favorite city in the **United States**.

The **United States** needs a new strategy to meet this challenge.
Effort-Light StructMine: Methodology

1. Text corpus
2. Data-driven text segmentation (SIGMOD’15, WWW’16)
3. Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17)
4. Partially-labeled corpus
5. Entity names & context units
6. Knowledge bases

- Structures from the remaining unlabeled data
Effort-Light StructMine: Methodology

- **Text corpus**
- **Data-driven text segmentation** (SIGMOD’15, WWW’16)
- **Entity names & context units**
- **Structures from the remaining **unlabeled** data**
- **Learning Corpus-specific Model** (KDD’15, KDD’16, EMNLP’16, WWW’17)
- **Partially-labeled corpus**

Knowledge bases:
- Wikipedia
- Wikidata
- Freebase
- DBpedia
- Yago

Closed-world Assumption vs Open-world Assumption
Effort-Light StructMine: Methodology

Data-driven text segmentation (SIGMOD'15, WWW'16)

Entity names & context units

Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

Knowledge bases

Text corpus

Structures from the remaining unlabeled data

Typing of Entities and Relations

VS

Meta Pattern-Based Attribute Mining
Transformation in Information Search

Desktop search

Mobile search

“Which hotel has a roller coaster in Las Vegas?”

Lengthy Documents? Direct Answers!

Answer: New York-New York hotel

Surge of mobile Internet use in China

Source: China Internet Network Information Center
Application: Facebook Entity Graph

People, Places, and Things

Facebook’s knowledge graph (entity graph) stores as entities the users, places, pages and other objects within the Facebook.

Connecting

The connections between the entities indicate the type of relationship between them, such as friend, following, photo, check-in, etc.
Structured Query: RDF + SPARQL

Triples in an RDF graph

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barack_Obama</td>
<td>parentOf</td>
<td>Malia_Obama</td>
</tr>
<tr>
<td>Barack_Obama</td>
<td>parentOf</td>
<td>Natasha_Obama</td>
</tr>
<tr>
<td>Barack_Obama</td>
<td>spouse</td>
<td>Michelle_Obama</td>
</tr>
<tr>
<td>Barack_Obama Sr.</td>
<td>parentOf</td>
<td>Barack_Obama</td>
</tr>
</tbody>
</table>

SPARQL query

```
SELECT ?x WHERE {
  Barack_Obama_Sr. parentOf ?y .
  ?y parentOf ?x .
}
```

Answer

```
<Malia_Obama>
<Natasha_Obama>
```
Why Structured Query Falls Short?

<table>
<thead>
<tr>
<th>Knowledge Base</th>
<th># Entities</th>
<th># Triples</th>
<th># Classes</th>
<th># Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freebase</td>
<td>45M</td>
<td>3B</td>
<td>53K</td>
<td>35K</td>
</tr>
<tr>
<td>DBpedia</td>
<td>6.6M</td>
<td>13B</td>
<td>760</td>
<td>2.8K</td>
</tr>
<tr>
<td>Google Knowledge Graph*</td>
<td>570M</td>
<td>18B</td>
<td>1.5K</td>
<td>35K</td>
</tr>
<tr>
<td>YAGO</td>
<td>10M</td>
<td>120M</td>
<td>350K</td>
<td>100</td>
</tr>
<tr>
<td>Knowledge Vault</td>
<td>45M</td>
<td>1.6B</td>
<td>1.1K</td>
<td>4.5K</td>
</tr>
</tbody>
</table>

* as of 2014

• It’s more than large: High heterogeneity of KBs

• *If it’s hard to write SQL on simple relational tables, it’s only harder to write SPARQL on large knowledge bases*
  • Even harder on automatically constructed KBs with a massive, loosely-defined schema
Certainly, You Do Not Want to Write This!

“find all patients diagnosed with eye tumor”

```
WITH Traversed (cls, syn) AS (
    (SELECT R.cls, R.syn
    FROM XMLTABLE ('Document("Thesaurus.xml")
    /terminology/conceptDef/properties
    [property/name/text()="Synonym" and
    property/value/text()="Eye Tumor"]
    /property[name/text()="Synonym"]/value’
    COLUMNS
    cls CHAR(64) PATH ’./parent::*/parent::*
    /parent::*/name’,
    tkt CHAR(64) PATH’.’) AS R)
UNION ALL
(SELECT CH.cls, CH.syn
FROM Traversed PR,
XMLTABLE ('Document("Thesaurus.xml")
/terminology/conceptDef/definingConcepts/
concept[./text()=$parent]/parent::*/parent::*/
properties/property[name/text()="Synonym"]/value’
PASSING PR.cls AS ”parent”
COLUMNS
cls CHAR(64) PATH ’./parent::*/
    parent::*/parent::*/name’,
syn CHAR(64) PATH’.’) AS CH))
SELECT DISTINCT V.*
FROM Visit V
WHERE V.diagnosis IN
    (SELECT DISTINCT syn FROM Traversed)
```

“Semantic queries by example”, Lipyeow Lim et al., EDBT 2014
Schema-agnostic KB Querying

"Barack Obama Sr. grandchildren"

**Keyword query:** query like search engine

**Graph query:** add a little structure

"Who are Barack Obama Sr.’s grandchildren?"

**Natural language query:** like asking a friend

<Barack Obama Sr., Malia Obama>

**Query by example:** Just show me examples
Tutorial Outline

• Introduction

• Part I: Effort–Light StructMine
  • Tea break at 3:00pm

• Part II: Schema-agnostic KB Querying

• Summary & Future Directions
Construction and Querying of Large-scale Knowledge Bases

Part I: Effort-Light StructMine for Knowledge Base Construction
Effort-Light StructMine: Overview

Data-driven text segmentation (SIGMOD'15, WWW'16)

Entity names & context units

Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

Knowledge bases

Text corpus

Structures from the remaining unlabeled data

Data-driven text segmentation (SIGMOD’15, WWW’16)

Entity names & context units

Data-driven text segmentation (SIGMOD’15, WWW’16)

Entity names & context units

Partially-labeled corpus

Knowledge bases

Structures from the remaining unlabeled data

Data-driven text segmentation (SIGMOD’15, WWW’16)

Entity names & context units

Partially-labeled corpus

Knowledge bases

Structures from the remaining unlabeled data

Entity Recognition and Coarse-grained Typing (KDD’15)

Fine-grained Entity Typing (KDD’16)

Joint Entity and Relation Extraction (WWW’17)

Corpus to Structured Network: The Roadmap
Corpus to Structured Network: The Roadmap

- Entity Recognition and Coarse-grained Typing (KDD’15)
- Fine-grained Entity Typing (KDD’16)
- Joint Entity and Relation Extraction (WWW’17)

Data-driven text segmentation (SIGMOD’15, WWW’16)

Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17)

Entity names & context units

Partially-labeled corpus

Knowledge bases

Text corpus

Structures from the remaining unlabeled data
Recognizing Entities of Target Types in Text

The best BBQ I’ve tasted in Phoenix! I had the pulled pork sandwich with coleslaw and baked beans for lunch. The owner is very nice. ...

The best BBQ I’ve tasted in Phoenix! I had the pulled pork sandwich with coleslaw and baked beans for lunch. The owner is very nice. ...

food
location
person
Traditional Named Entity Recognition (NER) Systems

- Heavy reliance on corpus-specific human labeling
- Training sequence models is slow

A manual annotation interface

e.g., (McMallum & Li, 2003), (Finkel et al., 2005), (Ratinov & Roth, 2009), ...
Weak-Supervision Systems: Pattern-Based Bootstrapping

- Requires manual seed selection & mid-point checking

Seeds for Food
- Pizza
- French Fries
- Hot Dog
- Pancake
...

Patterns for Food
the best <X> I’ve tried in their <X> tastes amazing...

Systems:
- CMU NELL
- UW KnowItAll
- Stanford DeepDive
- Max-Planck PROSPERA
...

e.g., (Etzioni et al., 2005), (Talukdar et al., 2010), (Gupta et al., 2014), (Mitchell et al., 2015), ...

Seed entities and corpus

Annotate corpus using entities

Generate candidate patterns

Score candidate patterns

Select Top patterns

Apply patterns to find new entities
Leveraging Distant Supervision

1. **Detect** entity names from text
2. **Match** name strings to KB entities
3. **Propagate** types to the un-matchable names

<table>
<thead>
<tr>
<th>ID</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td><em>Phoenix</em> is my all-time favorite dive bar in <em>New York City</em>.</td>
</tr>
<tr>
<td>S2</td>
<td>The best <em>BBQ</em> I’ve tasted in <em>Phoenix</em>.</td>
</tr>
<tr>
<td>S3</td>
<td><em>Phoenix</em> has become one of my favorite bars in <em>NY</em>.</td>
</tr>
</tbody>
</table>

(Lin et al., 2012), (Ling et al., 2012), (Nakashole et al., 2013)
Current Distant Supervision: Limitation I

1. Context-agnostic type prediction
   - Predict types for each mention regardless of context

2. Sparsity of contextual bridges

<table>
<thead>
<tr>
<th>ID</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td><em>Phoenix</em> is my all-time favorite dive bar in <em>New York City</em>.</td>
</tr>
<tr>
<td>S2</td>
<td>The best <em>BBQ</em> I’ve tasted in <em>Phoenix</em>.</td>
</tr>
<tr>
<td>S3</td>
<td><em>Phoenix</em> has become one of my favorite bars in <em>NY</em>.</td>
</tr>
</tbody>
</table>
Current Distant Supervision: Limitation II

1. Context-agnostic type prediction

2. Sparsity of contextual bridges
   - Some relational phrases are infrequent in the corpus → ineffective type propagation

<table>
<thead>
<tr>
<th>ID</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>Phoenix is my all-time favorite dive bar in New York City.</td>
</tr>
<tr>
<td>S3</td>
<td>Phoenix has become one of my favorite bars in NY.</td>
</tr>
</tbody>
</table>
**ClusType**: Data-Driven Entity Mention Detection

- **Significance** of a merging between two sub-phrases

\[
\rho_{x}(S_1, S_2) = \frac{v(S_1 \oplus S_2) - N \frac{v(S_1)}{N} \frac{v(S_2)}{N}}{\sqrt{v(S_1 \oplus S_2)}} \cdot I_x(S_1 \oplus S_2)
\]

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>(J*)N*</td>
<td>support vector machine</td>
</tr>
<tr>
<td>VP</td>
<td>tasted in, damage on</td>
</tr>
<tr>
<td>VW*(P)</td>
<td>train a classifier with</td>
</tr>
</tbody>
</table>

Good Concordance
ClusType: Data-Driven Entity Mention Detection

- **Significance** of a merging between two sub-phrases

\[
\rho_x(S_1, S_2) = \frac{v(S_1 \oplus S_2) - N \frac{v(S_1)}{N} \frac{v(S_2)}{N}}{\sqrt{v(S_1 \oplus S_2)}} \cdot I_X(S_1 \oplus S_2)
\]

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>(J*)N*</td>
<td>support vector machine</td>
</tr>
<tr>
<td>VP</td>
<td>tasted in, damage on</td>
</tr>
<tr>
<td>VW*(P)</td>
<td>train a classifier with</td>
</tr>
</tbody>
</table>

The best **BBQ** I’ve **tasted in Phoenix**! I **had** the **pulled pork sandwich with coleslaw** and **baked beans** for lunch. ... This **place** serves up the best **cheese steak sandwich in west of Mississippi**.
My Solution: **ClusType** (KDD’15)

### ID | Segmented Sentences
--- | ---
S1 | *Phoenix* is my all-time favorite dive bar in *New York City*. 
S2 | The best *BBQ* I’ve *tasted* in *Phoenix*. 
S3 | *Phoenix* has become one of my favorite bars in *NY*. 

### Putting two sub-tasks together:
1. Type label propagation
2. Relation phrase clustering
Type Propagation in ClusType

Smoothness Assumption
If two objects are similar according to the graph, then their type labels should be also similar.

(Belkin & Partha, NIPS’01), (Ren et al., KDD’15)
Relation Phrase Clustering in ClusType

• Two relation phrases should be grouped together if:
  1. Similar string
  2. Similar context
  3. Similar types for entity arguments

“Multi-view” clustering

(Ren et al., KDD’15)
## ClusType: Comparing with State-of-the-Art Systems (F1 Score)

<table>
<thead>
<tr>
<th>Methods</th>
<th>NYT</th>
<th>Yelp</th>
<th>Tweet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern (Stanford, CONLL’14)</td>
<td>0.301</td>
<td>0.199</td>
<td>0.223</td>
</tr>
<tr>
<td>SemTagger (U Utah, ACL’10)</td>
<td>0.407</td>
<td>0.296</td>
<td>0.236</td>
</tr>
<tr>
<td>NNPLB (UW, EMNLP’12)</td>
<td>0.637</td>
<td>0.511</td>
<td>0.246</td>
</tr>
<tr>
<td>APOLLO (THU, CIKM’12)</td>
<td>0.795</td>
<td>0.283</td>
<td>0.188</td>
</tr>
<tr>
<td>FIGER (UW, AAAI’12)</td>
<td>0.881</td>
<td>0.198</td>
<td>0.308</td>
</tr>
<tr>
<td><strong>ClusType (KDD’15)</strong></td>
<td><strong>0.939</strong></td>
<td><strong>0.808</strong></td>
<td><strong>0.451</strong></td>
</tr>
</tbody>
</table>

- **vs. bootstrapping**: context-aware prediction on “un-matchable”
- **vs. label propagation**: group similar relation phrases
- **vs. FIGER**: no reliance on complex feature engineering

**NYT**: 118k news articles (1k manually labeled for evaluation); **Yelp**: 230k business reviews (2.5k reviews are manually labeled for evaluation); **Tweet**: 302 tweets (3k tweets are manually labeled for evaluation)

**Precision** ($P$) = \( \frac{\#\text{Correctly-typed mentions}}{\#\text{System-recognized mentions}} \), **Recall** ($R$) = \( \frac{\#\text{Correctly-typed mentions}}{\#\text{ground-truth mentions}} \), **F1 score** = \( \frac{2(P \times R)}{(P + R)} \)
Corpus to Structured Network: The Roadmap

Entity Recognition and Coarse-grained Typing (KDD’15) → Fine-grained Entity Typing (KDD’16) → Joint Entity and Relation Extraction (WWW’17)

Text corpus → Data-driven text segmentation (SIGMOD’15, WWW’16) → entity names & context units

Structures from the remaining *unlabeled* data

Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

Knowledge bases
Donald Trump spent 14 television seasons presiding over a game show, NBC’s The Apprentice.
Current Distant Supervision: Context-Agnostic Labeling

- Inaccurate labels in **training data**
- **Prior work:** all labels are “perfect”

<table>
<thead>
<tr>
<th>ID</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>Donald Trump spent 14 television seasons presiding over a game show, NBC’s The Apprentice</td>
</tr>
</tbody>
</table>

**S1: Donald Trump**

**Entity Types:** person, artist, actor, author, businessman, politician

Entity types from knowledge base:

- person
- location
- organization
- politician
- artist
- businessman
- author
- actor
- singer

Entity: Donald Trump
My Solution: **Partial Label Embedding** (KDD’16)

**Extract Text Features**

**“Label Noise Reduction” with PLE**

**Train Classifiers on De-noised Data**

**Prediction on New Data**

<table>
<thead>
<tr>
<th>ID</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td><em>Donald Trump</em> spent 14 television seasons presiding over a game show, NBC’s <em>The Apprentice</em></td>
</tr>
</tbody>
</table>

**Text features:** TOKEN:

- Donald
- Trump

CONTEXT:

- television
- season

SHAPE:

- AA

**Entity Types:** person, artist, actor, author, businessman, politician

More effective classifiers

“De-noised” labeled data

(Ren et al., KDD’16)
**PLE: Modeling Clean and Noisy Mentions Separately**

For a **clean mention**, its “**positive types**” should be **ranked higher** than all its “**negative types**”

<table>
<thead>
<tr>
<th>ID</th>
<th>Noisy Entity Mention</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td><em>Donald Trump</em> spent 14 television seasons presiding over a game show, NBC’s The Apprentice</td>
</tr>
</tbody>
</table>

**Types in KB:** person, artist, actor, author, businessman, politician

**Types ranked**

1. (+) actor 0.88
2. (+) artist 0.74
3. (+) person 0.55
4. (+) author 0.41
5. (+) politician 0.33
6. (+) business 0.31

For a **noisy mention**, its “**best candidate type**” should be **ranked higher** than all its “**non-candidate types**”

*S1: Donald Trump*

(Ren et al., KDD’16)
Type Inference in PLE

- Top-down nearest neighbor search in the given type hierarchy

Vector for text features

Test mention: $S_i^{Trump}$

(Ren et al., KDD’16)
**PLE: Performance of Fine-Grained Entity Typing**

Accuracy = \(rac{\text{# mentions with all types correctly predicted}}{\text{# mentions in the test set}}\)

Accuracy on different type levels:

- **Raw**: candidate types from distant supervision
- **WSABIE** (Google, ACL’15): joint feature and type embedding
- **Predictive Text Embedding** (MSR, WWW’15): joint mention, feature and type embedding
  - Both WSABIE and PTE suffer from “noisy” training labels
- **PLE** (KDD’16): partial-label loss for context-aware labeling

OntoNotes public dataset (Weischedel et al. 2011, Gillick et al., 2014):
13,109 news articles, 77 annotated documents, 89 entity types
Corpus to Structured Network: The Roadmap

Text corpus

Data-driven text segmentation (SIGMOD’15, WWW’16)

entity names & context units

Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

Structures from the remaining unlabeled data

Knowledge bases

Entity Recognition and Coarse-grained Typing (KDD’15)

Fine-grained Entity Typing (KDD’16)

Joint Entity and Relation Extraction (WWW’17)
The Women’s March was a worldwide protest on January 21, 2017. The protest was aimed at Donald Trump, the recently inaugurated president of the United States. The first protest was planned in Washington, D.C., and was known as the Women’s March on Washington.
Prior Work: Relation Extraction (RE)

**Supervised RE systems**
- Hard to be ported to deal with different kinds of corpora

**Pattern-based bootstrapping RE systems**
- Focus on “explicit” relation mentions
- “Semantic drift”

**Distantly-supervised RE systems (cont.)**
- Error propagation
- Noisy candidate type labels

---


Prior Work: An “Incremental” System Pipeline

Entity mention detection

Context-aware entity typing

Relation mention detection

Context-aware relation typing

Entity boundary errors:
The *Women* ’s March was a worldwide *protest* on *January 21, 2017*.

Entity type errors:
The *Women* ’s March was a worldwide *protest* on *January 21, 2017*. → *person*

Relation mention errors:

(women, protest) X
(protest, January 21, 2017)

Relation type errors:

(women, protest) → *is a* X
(protest, January 21, 2017)

Entity type errors:

(women, protest) → *is a* X
(protest, January 21, 2017)

(Mintz et al., 2009), (Riedel et al., 2010), (Hoffmann et al., 2011), (Surdeanu et al., 2012), (Nagesh et al., 2014), ...
My Solution: **CoType** (WWW’17)

1. Data-driven detection of entity and relation mentions
   - Data-driven text segmentation
   - Syntactic pattern learning from KBs

2. Joint typing of entity and relation mentions
   - **Context-aware** type modeling
   - Model **entity-relation interactions**

(Ren et al. WWW’17)
Data-Driven Entity and Relation Detection

**S2**: The protest was aimed at Donald Trump, the recently inaugurated president of the United States.

**Frequent Pattern Mining**

**S2**: The *protest* was *aimed at* Donald Trump, the recently inaugurated *president of* the United States.

**Segment Quality Estimation**

Phrases quality: *United States*: 0.9, *was aimed at*: 0.4, ....

Part-of-speech (POS) patterns quality: *ADJ NN*: 0.85, *V PROP*: 0.4, ...

**POS-guided Segmentation**

**S2**: The *protest was aimed at Donald Trump*, the recently inaugurated *president of* the United States.

**Quality Re-estimation & Re-segmentation**

**(S2: protest, Donald Trump), (S2: Donald Trump, United States)**
## Entity Mention Detection: Results

<table>
<thead>
<tr>
<th>POS Tag Pattern</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Good</strong> (high score)</td>
<td><img src="http://example.com" alt="POS Tag Pattern" /></td>
</tr>
<tr>
<td>NNP NNP</td>
<td>San Francisco/Barack Obama/United States comedy drama</td>
</tr>
<tr>
<td>NN NN</td>
<td>car accident/club captain seven network/seven dwarfs/2001</td>
</tr>
<tr>
<td>CD NN</td>
<td>census crude oil/nucletic acid/baptist church</td>
</tr>
<tr>
<td>JJ NN</td>
<td></td>
</tr>
<tr>
<td><strong>Bad</strong> (low score)</td>
<td><img src="http://example.com" alt="POS Tag Pattern" /></td>
</tr>
<tr>
<td>DT JJ NND</td>
<td>a few miles/the early stages/the late 1980s 2 : 0 victory</td>
</tr>
<tr>
<td>CD CD NN IN</td>
<td>over/1 : 0 win over rating on rotten tomatoes worked</td>
</tr>
<tr>
<td>NN IN NNP NNP</td>
<td>together on/spent much of</td>
</tr>
<tr>
<td>VVD RB IN</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>NYT</th>
<th>Wiki-KBP</th>
<th>BioInfer</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIGER segmenter [UW, 2012]</td>
<td>0.751</td>
<td>0.814</td>
<td>0.652</td>
</tr>
<tr>
<td>Our Approach</td>
<td><strong>0.837</strong></td>
<td>0.833</td>
<td>0.785</td>
</tr>
</tbody>
</table>
CoType: Co-Embedding for Typing Entities and Relations

Object interactions in a heterogeneous graph

Low-dimensional vector spaces

Model entity-relation interactions

(Ren et al. WWW’17)
Modeling Mention-Feature Co-occurrences

- **Second-order Proximity**
  Mentions with similar distributions over text features should have similar types

Vertex $m_i$ and $m_j$ have a large second-order proximity

(Tang et al., WWW’15), (Ren et al. WWW’17)
**Challenge:** Context-Agnostic Labeling

<table>
<thead>
<tr>
<th>ID</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2</td>
<td>The protest was aimed at <em>Donald Trump</em>, the recently inaugurated president of the <em>United States</em>.</td>
</tr>
</tbody>
</table>

**Type labels for relation mention:**

<table>
<thead>
<tr>
<th>E1: Donald J. Trump</th>
<th>E2: United States</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1 Types: person, politician, businessman, author, actor</td>
<td>E2 Types: location, organization</td>
</tr>
</tbody>
</table>

**Relations between E1, E2 in KB:** president of, live in, born in

**Relation types from knowledge base:**

- president of
- born in
- live in

Entity 1: Donald Trump

Entity 2: United States
Context-Aware Type Modeling

Partial-label Loss Function

- Vector representation of the relation mention should be more similar to its "best" candidate type, than to any other non-candidate type.

\[ \ell_i = \max \left\{ 0, 1 - \max_{y \in \mathcal{V}_i} s(m_i, y) - \max_{y' \in \mathcal{V}_i} s(m_i, y') \right\} \]

Maximal score for non-candidate types

(Ren et al. WWW’17)
Modeling Entity-Relation Interactions

Object “Translating” Assumption

For a relation mention \( z \) between entity arguments \( m_1 \) and \( m_2 \):

\[
\text{vec}(m_1) \approx \text{vec}(m_2) + \text{vec}(z)
\]

Error on a relation triple \((z, m_1, m_2)\):

\[
\tau(z) = \|m_1 + z - m_2\|_2^2
\]

Low-dimensional vector space

(Bordes, NIPS’13), (Ren et al., WWW’17)
Reducing Error Propagation: A Joint Optimization Framework

Modeling entity-relation interactions

\[ O_{ZM} = \sum_{z_i \in Z_L} \sum_{v=1}^{V} \max \left\{ 0, 1 + \tau(z_i) - \tau(z_v) \right\} \]

\[ \min \mathcal{O} = \mathcal{O}_M + \mathcal{O}_Z + \mathcal{O}_{ZM} \]

Modeling types of relation mentions

\[ O_Z = \mathcal{L}_{ZF} + \sum_{i=1}^{N_L} \ell_i + \frac{\lambda}{2} \sum_{i=1}^{N_L} \|z_i\|^2_2 + \frac{\lambda}{2} \sum_{k=1}^{K_r} \|r_k\|^2_2 \]

Modeling types of entity mentions

\[ O_M = \mathcal{L}_{MF} + \sum_{i=1}^{N'_L} \ell'_i + \frac{\lambda}{2} \sum_{i=1}^{N'_L} \|m_i\|^2_2 + \frac{\lambda}{2} \sum_{k=1}^{K_y} \|y_k\|^2_2 \]

(Ren et al., WWW’17)
CoType: Comparing with State-of-the-Arts RE Systems

- Given candidate relation mentions, predict its relation type if it expresses a relation of interest; otherwise, output “None”

- DS+Logistic (Stanford, ACL’09): logistic classifier on DS
- MultiR (UW, ACL’11): handles inappropriate labels in DS
- DeepWalk (StonyBrook, KDD’14): homogeneous graph embedding
- LINE (MSR, WWW’15): joint feature & type embedding
- CoType-RM (WWW’17): only models relation mentions
- CoType (WWW’17): models entity-relation interactions

NYT public dataset (Riedel et al. 2010, Hoffmann et al., 2011): 1.18M sentences in the corpus, 395 manually annotated sentences for evaluation, 24 relation types
## An Ongoing Application to Life Sciences

<table>
<thead>
<tr>
<th>LifeNet by Effort-Light StructMine</th>
<th>BioInfer Network by human labeling (Pyysalo et al., 2007)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine-created</td>
<td>Human-created</td>
</tr>
<tr>
<td>4 Million+ PubMed papers</td>
<td>1,100 sentences</td>
</tr>
<tr>
<td>1,000+ entity types</td>
<td>94 protein-protein interactions</td>
</tr>
<tr>
<td>400+ relation types</td>
<td>2,500 man-hours</td>
</tr>
<tr>
<td>&lt;1 hour, single machine</td>
<td>2,662 facts</td>
</tr>
<tr>
<td>10,000x more facts</td>
<td></td>
</tr>
</tbody>
</table>

(Pyysalo et al., BMC Bioinformatics‘07)
(Ren et al., ACL’17 demo, under review)

Performance evaluation on BioInfer:
Relation Classification Accuracy = 61.7%
(11%↑ over the best-performing baseline)
Biomedical Named Entity Recognition by Multi-tasking different datasets

**Single-task/dataset learning**

- CRF
- word BiLSTM
- word emb
- concat
- char BiLSTM
- char emb
- LM

**Multi-task/dataset learning**

- CRF1
- CRF2
- word BiLSTM
- Word emb
- concat
- char BiLSTM
- Char emb

(Liu et al., AAAI’18)
## Performance of NER on Biomed Benchmark Datasets

<table>
<thead>
<tr>
<th>Dataset Benchmark</th>
<th>Liu et al. 2017 (single-task)</th>
<th>Multi-task</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BC2GM</strong> (gene/protein)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prec</td>
<td>88.48</td>
<td>83.82</td>
</tr>
<tr>
<td>Rec</td>
<td>85.97</td>
<td>82.12</td>
</tr>
<tr>
<td>F1</td>
<td><strong>87.21</strong></td>
<td>82.96</td>
</tr>
<tr>
<td><strong>BC4CHEMD</strong> (Chemical)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prec</td>
<td>89.09</td>
<td>90.21</td>
</tr>
<tr>
<td>Rec</td>
<td><strong>85.75</strong></td>
<td>84.82</td>
</tr>
<tr>
<td>F1</td>
<td><strong>87.39</strong></td>
<td>87.44</td>
</tr>
<tr>
<td><strong>BC5CDR</strong> (Chemical, Diseases)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prec</td>
<td><strong>89.21</strong></td>
<td>85.71</td>
</tr>
<tr>
<td>Rec</td>
<td>84.45</td>
<td>84.71</td>
</tr>
<tr>
<td>F1</td>
<td><strong>86.76</strong></td>
<td>85.21</td>
</tr>
<tr>
<td><strong>NCBI</strong> (Diseases)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prec</td>
<td>85.10</td>
<td>84.06</td>
</tr>
<tr>
<td>Rec</td>
<td>80.80</td>
<td>84.57</td>
</tr>
<tr>
<td>F1</td>
<td><strong>82.90</strong></td>
<td>84.32</td>
</tr>
<tr>
<td><strong>JNLPBA</strong> (Gene, DNA, Cell Line, etc.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prec</td>
<td>69.42</td>
<td>72.10</td>
</tr>
<tr>
<td>Rec</td>
<td><strong>75.99</strong></td>
<td><strong>77.52</strong></td>
</tr>
<tr>
<td>F1</td>
<td><strong>72.55</strong></td>
<td>74.72</td>
</tr>
</tbody>
</table>
“Heterogeneous Supervision” for Relation Extraction

- A principled I framework to **unify** KB-supervision, manual rules, crowd-sourced labels, etc.
- Multiple “**labeling functions**” annotate one instance → resolve conflicts & redundancy → “**expertise**” of each labeling function

---

(Liu et al., EMNLP’17)
Indirect Supervision for Relation Extraction -- using QA Pairs

- Questions $\rightarrow$ positive / negative answers
- pos pairs $\rightarrow$ similar relation; neg pairs $\rightarrow$ distinct relations

(Wu et al., WSDM’18)
Pattern-enhanced Distributional Representation Learning

**Pattern Module**
- [ENT] [ENT] ’s capital
- [ENT] capital [ENT]
- capital [ENT] [ENT]

**Distributional Module**
- Germany
- France
- Paris
- Berlin
- China
- Beijing

**Existing Integration Frameworks**

- Pattern Module
- Distributional Module

**Our Co-training Framework**

- Pattern Module
- Distributional Module
- Seeds
Corpus to Structured Network: The Roadmap

Text corpus → Data-driven text segmentation (SIGMOD’15, WWW’16) → Partially-labeled corpus

Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17) → entity names & context units

Structures from the remaining unlabeled data

Knowledge bases

Entity Recognition and Coarse-grained Typing (KDD’15) → Fine-grained Entity Typing (KDD’16) → Joint Entity and Relation Extraction (WWW’17)
**References I**

- **Xiang Ren**, Zeqiu Wu, Wenqi He, Meng Qu, Clare R. Voss, Heng Ji, Tarek F. Abdelzaher, Jiawei Han. CoType: Joint Extraction of Typed Entities and Relations with Knowledge Bases. WWW, 2017.

- **Xiang Ren**, Ahmed El-Kishky, Heng Ji, and Jiawei Han. Automatic Entity Recognition and Typing in Massive Text Data (Conference Tutorial). SIGMOD, 2016.

- **Xiang Ren**, Wenqi He*, Meng Qu, Lifu Huang, Heng Ji, Jiawei Han. AFET: Automatic Fine-Grained Entity Typing by Hierarchical Partial-Label Embedding. EMNLP, 2016.

- **Xiang Ren**, Wenqi He*, Meng Qu, Heng Ji, Clare R. Voss, Jiawei Han. Label Noise Reduction in Entity Typing by Heterogeneous Partial-Label Embedding. KDD, 2016.

- **Xiang Ren**, Wenqi He, Ahmed El-Kishky, Clare R. Voss, Heng Ji, Meng Qu, Jiawei Han. Entity Typing: A Critical Step for Mining Structures from Massive Unstructured Text (Invited Paper). MLG, 2016.


- Tarique A. Siddiqui*, **Xiang Ren***, Aditya Parameswaran, Jiawei Han. FacetGist: Collective Extraction of Document Facets in Large Technical Corpora. CIKM, 2016.

- Jialu Liu, Jingbo Shang, Chi Wang, **Xiang Ren**, Jiawei Han. Mining Quality Phrases from Massive Text Corpora. SIGMOD, 2015.
References II

• Marina Danilevsky, Chi Wang, Nihit Desai, **Xiang Ren**, Jingyi Guo, and Jiawei Han. Automatic Construction and Ranking of Topical Keyphrases on Collections of Short Documents. SDM, 2014.

• **Xiang Ren**, Yuanhua Lv, Kuansan Wang, Jiawei Han. Comparative Document Analysis for Large Text Corpora. WSDM, 2017.

• Jialu Liu, **Xiang Ren**, Jingbo Shang, Taylor Cassidy, Clare R. Voss, Jiawei Han. Representing Documents via Latent Keyphrase Inference. WWW, 2016.

• Hyungsul Kim, **Xiang Ren**, Yizhou Sun, Chi Wang, and Jiawei Han. Semantic Frame-Based Document Representation for Comparable Corpora. ICDM, 2013.

• **Xiang Ren**, J. Liu, X. Yu, U. Khandelwal, Q. Gu, L. Wang, and J. Han. ClusCite: Effective Citation Recommendation by Information Network-Based Clustering. KDD, 2014.


• **Xiang Ren**, Yujing Wang, Xiao Yu, Jun Yan, Zheng Chen, Jiawei Han. Heterogeneous Graph-Based Intent Learning from Queries, Web Pages and Wikipedia Concepts. WSDM 2014b.


• Xiao Yu, Xiang Ren, Quanquan Gu, Yizhou Sun and Jiawei Han. Collaborative Filtering with Entity Similarity Regularization in Heterogeneous Information Networks. IJCAI-HINA, 2013.
Construction and Querying of Large-scale Knowledge Bases

Part II: Schema-agnostic Knowledge Base Querying
Transformation in Information Search

Desktop search

Mobile search

“Which hotel has a roller coaster in Las Vegas?”

Lengthy Documents? Direct Answers!

Answer: New York-New York hotel

Surge of mobile Internet use in China

Source: China Internet Network Information Center
Application: Facebook Entity Graph

People, Places, and Things

Facebook’s knowledge graph (entity graph) stores as entities the users, places, pages and other objects within the Facebook.

Connecting

The connections between the entities indicate the type of relationship between them, such as friend, following, photo, check-in, etc.
QA Engine instead of Search Engine

• Behind the scene: A **knowledge graph** with millions of entities and billions of facts
Structured Query: RDF + SPARQL

Triples in an RDF graph

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barack_Obama</td>
<td>parentOf</td>
<td>Malia_Obama</td>
</tr>
<tr>
<td>Barack_Obama</td>
<td>parentOf</td>
<td>Natasha_Obama</td>
</tr>
<tr>
<td>Barack_Obama</td>
<td>spouse</td>
<td>Michelle_Obama</td>
</tr>
<tr>
<td>Barack_Obama_Sr.</td>
<td>parentOf</td>
<td>Barack_Obama</td>
</tr>
</tbody>
</table>

RDF graph

SPARQL query

```sparql
SELECT ?x WHERE {
  Barack_Obama_Sr. parentOf ?y.
  ?y parentOf ?x.
}
```

Answer

```
<Malia_Obama>
<Natasha_Obama>
```
Why Structured Query Falls Short?

<table>
<thead>
<tr>
<th>Knowledge Base</th>
<th># Entities</th>
<th># Triples</th>
<th># Classes</th>
<th># Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freebase</td>
<td>45M</td>
<td>3B</td>
<td>53K</td>
<td>35K</td>
</tr>
<tr>
<td>DBpedia</td>
<td>6.6M</td>
<td>13B</td>
<td>760</td>
<td>2.8K</td>
</tr>
<tr>
<td>Google Knowledge Graph*</td>
<td>570M</td>
<td>18B</td>
<td>1.5K</td>
<td>35K</td>
</tr>
<tr>
<td>YAGO</td>
<td>10M</td>
<td>120M</td>
<td>350K</td>
<td>100</td>
</tr>
<tr>
<td>Knowledge Vault</td>
<td>45M</td>
<td>1.6B</td>
<td>1.1K</td>
<td>4.5K</td>
</tr>
</tbody>
</table>

* as of 2014

- It’s more than large: High heterogeneity of KBs
- If it’s hard to write SQL on simple relational tables, it’s only harder to write SPARQL on large knowledge bases
  - Even harder on automatically constructed KBs with a massive, loosely-defined schema
Certainly, You Do Not Want to Write This!

“find all patients diagnosed with eye tumor”

```
WITH Traversed (cls,syn) AS ( 
(SELECT R.cls, R.syn 
FROM XMLTABLE ('Document("Thesaurus.xml")' 
/terminology/conceptDef/properties 
[property/name/text()="Synonym" and 
property/value/text()="Eye Tumor"] 
/property[name/text()="Synonym"]/value’ 
COLUMNS 
cls CHAR(64) PATH './parent::*/parent::* 
  /parent::*/name’, 
tgt CHAR(64) PATH’.’) AS R) 
UNION ALL 
(SELECT CH.cls,CH.syn 
FROM Traversed PR, 
XMLTABLE ('Document("Thesaurus.xml")' 
/terminology/conceptDef/definingConcepts/ 
concept.[/text()=$parent]/parent::*/$parent::*/ 
properties/property[name/text()="Synonym"]/value’ 
PASSING PR.cls AS ”parent” 
COLUMNS 
cls CHAR(64) PATH ‘./parent::*/ 
  /parent::*/parent::*/name’, 
syn CHAR(64) PATH’.’) AS CH)) 
SELECT DISTINCT V.* 
FROM Visit V 
WHERE V.diagnosis IN 
(SELECT DISTINCT syn FROM Traversed)
```

“Semantic queries by example”, Lipyeow Lim et al., EDBT 2014
Schema-agnostic KB Querying

"Barack Obama Sr. grandchildren"

Keyword query: query like search engine

"Who are Barack Obama Sr.'s grandchildren?"

Graph query: add a little structure

Natural language query: like asking a friend

Query by example: Just show me examples

<Barack Obama Sr., Malia Obama>
“Find a professor, ~70 yrs., who works in Toronto and joined Google recently.”

Search intent

Graph query

A match (result)
# Mismatch between Knowledge Base and Query

<table>
<thead>
<tr>
<th>Knowledge Base</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>“University of Washington”</td>
<td>“UW”</td>
</tr>
<tr>
<td>“neoplasm”</td>
<td>“tumor”</td>
</tr>
<tr>
<td>“Doctor”</td>
<td>“Dr.”</td>
</tr>
<tr>
<td>“Barack Obama”</td>
<td>“Obama”</td>
</tr>
<tr>
<td>“Jeffrey Jacob Abrams”</td>
<td>“J. J. Abrams”</td>
</tr>
<tr>
<td>“teacher”</td>
<td>“educator”</td>
</tr>
<tr>
<td>“1980”</td>
<td>“~30”</td>
</tr>
<tr>
<td>“3 mi”</td>
<td>“4.8 km”</td>
</tr>
<tr>
<td>“Hinton” - “DNNresearch” - “Google”</td>
<td>“Hinton” - “Google”</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Schema-less Graph Querying (SLQ)

Query

A Match

Prof., 70 yrs.

Geoffrey Hinton
(1947-)

Univ. of Toronto

DNNResearch

Toronto  Google

✓ Acronym transformation: ‘UT’ → ‘University of Toronto’
✓ Abbreviation transformation: ‘Prof.’ → ‘Professor’
✓ Numeric transformation: ‘≈70’ → ‘1947’
✓ Structural transformation: an edge → a path
<table>
<thead>
<tr>
<th>Transformation</th>
<th>Category</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>First/Last token</td>
<td>String</td>
<td>“Barack Obama” &gt; “Obama”</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>String</td>
<td>“Jeffrey Jacob Abrams” &gt; “J. J. Abrams”</td>
</tr>
<tr>
<td>Prefix</td>
<td>String</td>
<td>“Doctor” &gt; “Dr”</td>
</tr>
<tr>
<td>Acronym</td>
<td>String</td>
<td>&quot;International Business Machines&quot; &gt; &quot;IBM&quot;</td>
</tr>
<tr>
<td>Synonym</td>
<td>Semantic</td>
<td>“tumor” &gt; “neoplasm”</td>
</tr>
<tr>
<td>Ontology</td>
<td>Semantic</td>
<td>&quot;teacher&quot; &gt; &quot;educator&quot;</td>
</tr>
<tr>
<td>Range</td>
<td>Numeric</td>
<td>“~30” &gt; “1980”</td>
</tr>
<tr>
<td>Unit Conversion</td>
<td>Numeric</td>
<td>&quot;3 mi&quot; &gt; &quot;4.8 km&quot;</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Candidate Match Ranking

Query: $Q$

- Prof., 70 yrs.
- Toronto
- Google

Candidate Match: $\varphi(Q)$

- Geoffrey Hinton (1947-)
- Univ. of Toronto
- Google
- DNNResearch

**Features**

- **Node matching features:**
  \[ F_V(v, \varphi(v)) = \sum_i \alpha_i f_i(v, \varphi(v)) \]

- **Edge matching features:**
  \[ F_E(e, \varphi(e)) = \sum_j \beta_j g_j(e, \varphi(e)) \]

**Overall Matching Score**

\[
P(\varphi(Q) \mid Q) \propto \exp\left( \sum_{v \in V_Q} F_V(v, \varphi(v)) + \sum_{e \in E_Q} F_E(e, \varphi(e)) \right)
\]

[Yang et al. VLDB’14]
Query-specific Ranking via Relevance Feedback

• Generic ranking: sub-optimal for specific queries
  • By “Washington”, user A means Washington D.C., while user B might mean University of Washington

• Query-specific ranking: tailored for each query
  • But need additional query-specific information for further disambiguation

Relevance Feedback:
Users indicate the (ir)relevance of a handful of answers
Problem Definition

- \( Q \): A graph query
- \( G \): A knowledge graph
- \( \phi(Q) \): A candidate match to \( Q \)
- \( F(\phi(Q) | Q, \theta) \): A generic ranking function
- \( M^+ \): A set of positive/relevant matches of \( Q \)
- \( M^- \): A set of negative/non-relevant matches of \( Q \)

**Graph Relevance Feedback (GRF):**
Generate a query-specific ranking function \( \tilde{F} \) for \( Q \) based on \( M^+ \) and \( M^- \)
A General GRF Framework

<table>
<thead>
<tr>
<th>Input</th>
<th>Component</th>
<th>Stage Outcome</th>
<th>Final Ranking Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F(\phi(Q)</td>
<td>Q, \theta) ) ( \mathcal{M}^+, \mathcal{M}^- )</td>
<td>Query-specific Tuning</td>
<td>( F(\phi(Q)</td>
</tr>
<tr>
<td></td>
<td>Type Inference</td>
<td>( Q^* )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Context Inference</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Query-specific Tuning

• The $\theta$ represents (query-independent) feature weights. However, each query carries its own view of feature importance

• Find query-specific $\theta^*$ that better aligned with the query using user feedback

\[
g(\theta^*) = (1 - \lambda) \left( \frac{\sum_{\phi(Q) \in M^+} F(\phi(Q) | Q, \theta^*)}{|M^+|} - \frac{\sum_{\phi(Q) \in M^-} F(\phi(Q) | Q, \theta^*)}{|M^-|} \right) + \lambda R(\theta, \theta^*)
\]

User Feedback

Regularization

[Su et al. KDD’15]
Type Inference

• Infer the implicit type of each query node
• The types of the positive entities constitute a composite type for each query node
Context Inference

- **Entity context**: neighborhood of the entity
- The contexts of the positive entities constitute a composite context for each query node

[Su et al. KDD’15]
Experiment Setup

- Knowledge graph: DBpedia (4.6M nodes, 100M edges)
- Graph query sets: WIKI and YAGO

YAGO Class

Naval Battles of World War II Involving the United States

Instances

Battle of Midway
Battle of the Caribbean

Graph Query

Structured Information need

Naval Battle

Links between YAGO and DBpedia

Answer

Battle of Midway

United States

World War II

Naval Battles of World War II Involving the United States

[Su et al. KDD’15]
• Explicit feedback: User gives relevance feedback on top-10 results
• GRF improves SLQ for over 100%
• Three GRF components complement each other

Metric: mean average precision (MAP)

[Su et al. KDD’15]
• Pseudo feedback: Blindly assume top-10 results are correct
• Erroneous feedback information but no additional user effort

<table>
<thead>
<tr>
<th>MAP@K</th>
<th>1</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLQ_WIKI</td>
<td>0.23</td>
<td>0.21</td>
<td>0.24</td>
<td>0.25</td>
<td>0.27</td>
<td>0.28</td>
</tr>
<tr>
<td>GRF_WIKI</td>
<td>0.73</td>
<td>0.58</td>
<td>0.52</td>
<td>0.50</td>
<td>0.49</td>
<td>0.49</td>
</tr>
<tr>
<td>SLQ_YAGO</td>
<td>0.40</td>
<td>0.35</td>
<td>0.33</td>
<td>0.32</td>
<td>0.36</td>
<td>0.39</td>
</tr>
<tr>
<td>GRF_YAGO</td>
<td>0.82</td>
<td>0.66</td>
<td>0.60</td>
<td>0.57</td>
<td>0.58</td>
<td>0.61</td>
</tr>
</tbody>
</table>
Who is Justin Bieber’s sister?

Jazmyn Bieber

Knowledge Base

λx. sibling_of(Justin Bieber, x) ∧ gender(x, female)

Figure credit to Scott Yih
Challenges

• Language mismatch
  • Lots of ways to ask the same question
    • *Find terrorist organizations involved in September 11 attacks*
    • *Who did September 11 attacks?*
    • *The nine eleven were carried out with the involvement of what terrorist organizations?*
  • All need to be mapped to the KB relation: **terrorist_attack**
Challenges

• Language mismatch
• Large search space
  • United_States has over 1 million neighbors in Freebase
Challenges

• Language mismatch

• Large search space
  • United_States has over 1 million neighbors in Freebase

• Scalability
  • How to scale up to more advanced inputs, and scale out to more domains?
  • KBQA data is highly domain-specific
Challenges

• Language mismatch

• Large search space
  • United_States has over 1 million neighbors in Freebase

• Scalability
  • How to scale up to more advanced inputs, and scale out to more domains?
  • KBQA data is highly domain-specific

• Compositionality
  • If a model understands relation A and B, can it answer A+B?
What will be covered

• Model
  • General pipeline
  • Semantic matching: CNN and Seq2Seq

• Data
  • Low-cost data collection via crowdsourcing
  • Cross-domain semantic parsing via neural transfer learning
General pipeline

1. Topic Entity Linking
2. Candidate Logical Form Generation
3. Semantic Matching
4. Execution

**Seq2Seq:**
- [Jia and Liang, ACL’16]
- [Liang et al. ACL’17]

**CNN:**
- [Yih et al. ACL’15]

**Seq2Seq:**
- [Su and Yan, EMNLP’17]
Query Graph

Who first voiced Meg on Family Guy?

\[ \lambda x. \exists y. \text{cast}(\text{FamilyGuy}, y) \land \text{actor}(y, x) \land \text{character}(y, \text{MegGriffin}) \]
Topic Entity Linking

• An advanced entity linker for short text

• Prepare surface form lexicon for KB entities

• Entity mention candidates: all consecutive word sequences in lexicon

• Score entity mention candidates with the statistical model, keep top-10 entities

[Yih et al. ACL’15]
Candidate Logical Form Generation

• (Roughly) *enumerate* all admissive logical forms up to a certain complexity (2-hop)

[Yih et al. ACL’15]
Discriminative model: $p(R|P) = \frac{\exp(\cos(y_R,y_P))}{\sum_{R'} \exp(\cos(y_{R'},y_P))}$

who voiced meg on ⟨e⟩  cast–actor

[Yih et al. ACL’15]
Generative model: \( p(R|P) = \prod_i p(R_i|P, R_{<i}) \)
What will be covered

• Model
  • General pipeline
  • Semantic matching: CNN and Seq2Seq

• Data
  • Low-cost data collection via crowdsourcing
  • Cross-domain semantic parsing via neural transfer learning
Scalability

• Vertical scalability
  • Scale up to more complex inputs and logical constructs

Who was the head coach when Michael Jordan started playing for the Chicago Bulls?

In which season did Michael Jordan get the most points?

What team did Michael Jordan play for?
Scalability

• **Vertical scalability**
  • Scale up to more complex inputs and logical constructs

• **Horizontal scalability**
  • Scale out to more domains
  • Weather, calendar, hotel, flight, restaurant, ...
  • Knowledge base, relational database, API, robots, ...
  • Graph, table, text, image, audio, ...

• **More data + Better (more data-efficient) model**

On Generating Characteristic-rich Question Sets for QA Evaluation (EMNLP’16)
Cross-domain Semantic Parsing via Paraphrasing (EMNLP’17)
Building Natural Language Interfaces to Web APIs (CIKM’17)
Low-cost Data Collection via Crowdsourcing

“How many children of Eddard Stark were born in Winterfell?”

“How many children of Eddard Stark were born in Winterfell, and who is child of Eddard Stark?”

$\text{count}(\lambda x. \text{children(}\text{Eddard}_\text{Stark}, x) \land \text{place}_\text{of}_\text{birth}(x, \text{Winterfell}))$

1: Logical form generation

2: Canonical utterance generation

3: Paraphrasing via crowdsourcing

[Wang+ ACL’15, Su+ EMNLP’16, Su+ CIKM’17]
Existing KBQA datasets mainly contain *simple questions*

“*Where was Obama born?*”

“*What party did Clay establish?*”

“*What kind of money to take to bahamas?*”

… …
GraphQuestions: A New KBQA Dataset with Rich Characteristics

• Structural complexity
  • “people who are on a gluten-free diet can’t eat what cereal grain that is used to make challah?”

• Quantitative analysis (functions)
  • “In which month does the average rainfall of New York City exceed 86 mm?”

• Commonness
  • “Where was Obama born?” vs.
  • “What is the tilt of axis of Polestar?”

• Paraphrase
  • “What is the nutritional composition of coca-cola?”
  • “What is the supplement information for coca-cola?”
  • “What kind of nutrient does coke have?”

• …

https://github.com/ysu1989/GraphQuestions
<table>
<thead>
<tr>
<th>Model</th>
<th>Average F1 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sempre (Berant+ EMNLP’13)</td>
<td>10.8</td>
</tr>
<tr>
<td>Jacana (Yao+ ACL’14)</td>
<td>5.1</td>
</tr>
<tr>
<td>ParaSempre (Berant+ ACL’14)</td>
<td>12.8</td>
</tr>
<tr>
<td>UDepLambda (Reddy+ EMNLP’17)</td>
<td>17.6</td>
</tr>
<tr>
<td>Para4QA (Li+ EMNLP’17)</td>
<td>20.4</td>
</tr>
</tbody>
</table>
Crowdsourcing is great, but…

- There is an unlimited number of application domains; prohibitive cost to collect (sufficient) training data for every one.

- **Transfer learning**: Use existing data of some source domains to help target domain

- **Problem**: KBQA data is highly domain-specific
What is **transferrable** in semantic parsing?

*In which season did Kobe Bryant play for the Lakers?*

\[ \text{R[season].(player.KobeBryant} \ \land \ \text{team.Lakers)} \]

\[ p(\text{team|"play for")} \]

When did Alice start working for Mckinsey?

\[ \text{R[start].(employee.Alice} \ \land \ \text{employer.Mckinsey)} \]
• First convert logical forms to canonical utterances
• Train a neural paraphrase model on the source domains; adapt the model to the target domain
• Source domain: “play for” ⇒ “whose team is”
• Word embedding: “play” ⇒ “work”, “team” ⇒ “employer”
• Target domain: “work for” ⇒ “whose employer is”
Neural Transfer Learning for Semantic Parsing

Source Domain

Target Domain

Pre-trained Word Embedding
• Overnight dataset: 8 domains (basketball, calendar, etc.), each with a knowledge base
• For each target domain, use other 7 domains as source
Construction and Querying of Large-scale Knowledge Bases

Summary
Overall Contributions

• **Effort-Light StructMine**: “accurate” expansion of “matchable”
  → Corpus-specific labeling free, domain/language-independent

• **Schema-agnostic Query**: query without programming

• Technology Transfer:
  - A principled approach to manage, explore, analyze, and search “Big Text Data”
Future Work: Phrase Mining

• Refine quality phrases to entity mentions

• Further use the refined entity mention results to improve phrase mining

• Use high-quality phrases in different languages to improve the entity tagging
Future Work: Phrase Mining

• For popular languages with sufficient NLP tools
  • Incorporate more NLP features and structures

• For low-/zero-resource languages
  • Better unsupervised method
Future Work: Attribute Discovery

• Combining complementary methods towards attribute discovery from massive text corpora
  • Learning Approaches
  • Linguistic patterns using POS tagging, NP chunking, clause analysis, dependency parsing …
  • Meta pattern-driven approaches
    • Harnessing entity recognition and (fine-grained) typing systems
    • Quality assessment and meta-pattern segmentation based on contexts
    • Grouping synonymous patterns
    • Adjusting type levels for appropriate granularity
Future Work: Attribute Discovery

- Combining network mining and attribute mining

Text Corpus

**Attribute mining**

Entities, attributes, relations

Information Network

**Network mining**

Ranking, clustering, prediction ...

Human in the loop!

Human evaluation
Looking Forward: What’s Next?

Text corpus

Data-driven text segmentation (SIGMOD’15, WWW’16, ...)

entity names & context units

Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17, ...)

Partially-labeled corpus

Structures from the remaining unlabeled data

Knowledge bases

Network-to-knowledge

Structured Network

Knowledge & Insights
Looking Forward: Analyzing Literature to Facilitate Scientific Research

- Literature $\rightarrow$ Structured Network $\rightarrow$ Scientific Discovery
- More disciplines & More structure analysis functions

Scientific Hypothesis Generation by predicting missing relationships

Gaining insights for various research tasks in different disciplines

Collaborate with life scientists, chemists, physicists, computer scientists, ...
Looking Forward: Engaging with Human Behaviors

User-generated Content (Structured Network) + Structured Behavior Data

Social media post, Customer review, Chats & messages
Social network, Electronic health record, Transaction record

Personalized Intelligent Systems
Smart Health, Business intelligence, Conversational agent

Collaborate with doctors, social scientists, economists, ...
Looking Forward: Integrating with Our Physical World

Textual Signals (Structured Network) + Physical Sensor (Network) Signals

- Geo-sensors
- Audio/video sensors
- Bio-sensors

News streams, Social media post, Organization report

Smart-City Operating Systems
- Traffic management,
  Sustainable urban system,
  Cyber-physical system

Data Analytics

Collaborate with network & system researchers, environmental scientists, ...
Application to Vertical Domains

“Which cement stocks go up the most when a Category 3 hurricane hits Florida?”
One Interface for All

- All domains in a unified knowledge base
- Incrementally learn new domains without forgetting (or instead boosting) existing ones
Acknowledgement

• Academic Collaborators
  Jiawei Han (UIUC), ChengXiang Zhai (UIUC), Tarek Abdelzaher (UIUC), Aditya Parameswaran (UIUC), Saurabh Sinha (UIUC), Heng Ji (RPI), Yizhou Sun (UCLA), Peipei Ping (UCLA), David Liem (UCLA), Shih-Fu Chang (Columbia), Morteza Dehghani (USC), Richard Weinshilboum (Mayo Clinic), Clare V. Ross (ARL), Lance Kaplan (ARL), James Hendler (RPI), Xifeng Yan (UCSB), Brian Sadler (ARL), Michelle Vanni (ARL), Sue Kase (ARL), Huan Sun (OSU)

• Industry Collaborators
  Surajit Chaudhuri (MSR), Kuansan Wang (MSR), Kaushik Chakrabarti (MSR), Chi Wang (MSR), Hao Ma (MSR), Bin Bi (MSR), Yuanhua Lv (Microsoft Bing), Cong Yu (Google Research), Jialu Liu (Google Research), Tao Cheng (Pinterest), Mike Tung (DiffBot), Craig Schmidt (TripAdvisor), Mudhakar Srivatsa (IBM), Ahmed Awadallah (MSR), Patrick Pantel (MSR), Michael Gamon (MSR), Scott Yih (AI2)

• Funding

[Images of logos for ARL, NSF, NIH, Google, Microsoft, and IBM]
Thank you! Q&A

- **Effort-Light StructMine**: “accurate” expansion of “matchable”
  - Corpus-specific labeling free, domain/language-independent
- **Schema-agnostic Query**: query without programming
- Technology Transfer:
  - A principled approach to manage, explore, analyze, and search “Big Text Data”

![Diagram](image)

- Massive corpus → Structured Network → Knowledge → User-friendly Search