Building Structured Databases of Factual Knowledge from Massive Text Corpora

Xiang Ren  Meng Jiang  Jingbo Shang  Jiawei Han

Department of Computer Science

University of Illinois at Urbana-Champaign
Turning Unstructured Text Data into Structures

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

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

![Diagram](attachment:image.png)
Why Text to Structures?

Structured Search & Exploration

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

Pattern / Association Rule Mining

Graph Mining & Network Analysis

Structured Feature Generation

Input Space

Feature Space

Cluster
Periphery
Core
Hub
Link
Node

φ
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

A Life Science Use Case: Identifying “Distinctively Related Entities”

Collaborate with UCLA Heart BD2K Center & Mayo Clinic

What proteins are distinctively associated with Cardiomyopathy?

Prior Art: Extracting Structures with Repeated Human Effort

Extraction Rules
Machine-Learning Models

Structured Facts

Broadways shows
NYC
Times square
hotel
Hilton property

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 ...

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

Labeled data
Text Corpus

Stanford CoreNLP
CMU NELL
UW KnowItAll
USC AMR
IBM Alchemy APIs
Google Knowledge Graph
Microsoft Satori
...

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?

**Human labeling effort**

**Hand-crafted Systems**
UCB Hearst Pattern, 1992
NYU Proteus, 1997

**Supervised structure discovery**
Stanford CoreNLP, 2005 - present
UT Austin Dependency Kernel, 2005
IBM Watson Language APIs

**Weakly-supervised structure discovery**
CMU NELL, 2009 - present
UW KnowItAll, Open IE, 2005 - present
Max-Planck YAGO, 2008 - present

**Distantly-supervised structure discovery**
Stanford DeepDive, MIML-RE 2012 - present
UW FIGER, MultiR, 2012

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

A Review of Previous Efforts

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

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

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

Text corpus

Knowledge Bases

“Un-matchable”

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?*

---

... next to restaurants like *Junior’s Cheesecake*

(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

---

NYT: 82.2
Yelp: 91.7
Tweet: 95.5

MATCHABLE: 17.8
UN-MATCHABLE: 8.3
UN-MATCHABLE: 4.5

---

It is my favorite city in the United States.

The United States needs a new strategy to meet this challenge.

---

Location

Government

---
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**

- **Text corpus**
- **Wikipedia**, **WIKIDATA**, **Freebase**, **DBpedia**, **yago**

- **Location**
- **Government**

- **United States**

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

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

Text corpus

- Data-driven text segmentation (SIGMOD’15, WWW’16)
  - Entity names & context units

Corpus-specific Structure Discovery (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

Knowledge bases

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

- Text corpus
- StructMine
- Data-driven text segmentation (SIGMOD’15, WWW’16)
- Entity names & context units
- Structures from the remaining unlabeled data
- Partially-labeled corpus
- Knowledge bases
- Corpus-specific Structure Discovery (KDD’15, KDD’16, EMNLP’16, WWW’17)
- Pre-defined Type Structures vs Open Type Structures

Pre-defined Type Structures vs Open Type Structures
Effort-Light StructMine: Methodology

- **Text corpus**
- **Data-driven text segmentation** (SIGMOD’15, WWW’16)
- **Entity names & context units**
- **Corpus-specific Structure Discovery** (KDD’15, KDD’16, EMNLP’16, WWW’17)
- **Partially-labeled corpus**

- **Structures** from the remaining *unlabeled* data

- Typing of Entities and Relations vs Meta Pattern-Based Attribute Discovery

- Knowledge bases

15
Tutorial Outline

• Introduction
• Part I: Quality Phrase Mining
• Part II: Joint Typing of Entities and Relations
• Part III: Meta Pattern-Based Attribute Discovery
• Summary & Future Directions
Building Structured Databases of Factual Knowledge from Massive Text Corpora

Part I: Quality Phrase Mining
Effort-Light StructMine: Methodology

Text corpus

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

Entity names & context units

Corpus-specific Structure Discovery (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

Knowledge bases

Structures from the remaining unlabeled data
Quality Phrase Mining

• Quality phrase mining seeks to extract a ranked list of phrases with decreasing quality from a large collection of documents

• Examples:

<table>
<thead>
<tr>
<th>Scientific Papers</th>
<th>Expected Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>data mining</td>
</tr>
<tr>
<td></td>
<td>machine learning</td>
</tr>
<tr>
<td></td>
<td>information retrieval</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>support vector machine</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>the paper</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>News Articles</th>
<th>Expected Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>US President</td>
</tr>
<tr>
<td></td>
<td>Anderson Cooper</td>
</tr>
<tr>
<td></td>
<td>Barack Obama</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>Obama administration</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>a town</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

...
Why Phrase Mining?

w/o phrase mining

- What is “united”?  
- Which Dao?

w/ phrase mining

- United Airline!  
- David Dao!

Applications in NLP, IR, Text Mining

- Document analysis  
- Indexing in search engine

- Keyphrases for topic modeling  
- Summarization
What Kind of Phrases Are of “High Quality”?  

- **Popularity**  
  - “information retrieval” > “cross-language information retrieval”

- **Concordance**  
  - “strong tea” > “powerful tea”  
  - “active learning” > “learning classification”

- **Informativeness**  
  - “this paper” (frequent but not discriminative, not informative)

- **Completeness**  
  - “support vector machine” > “vector machine”
Three Families of Methods

- Supervised (linguistic analyzers)
- Unsupervised (statistical signals)
- Weakly / Distantly Supervised
Supervised Phrase Mining

• Phrase mining was originated from the NLP community

• How to use linguistic analyzers to extract phrases?
  • Parsing (e.g., stanford NLP parsers)
  • Noun Phrase (NP) Chunking

• How to rank extracted phrases?
  • C-value [Frantzi et al.’00]
  • TextRank [Mihalcea et al.’04]
  • TF-IDF
Linguistic Analyzer – Parsing

• Minimal Grammatical Segments ⇔ Phrases

Raw text sentence (string) → Full-text Parsing → Full parse tree (grammatical analysis)

The chef cooks the soup.

• Phrases: “the chef”, “the soup”
Linguistic Analyzer – Chunking

• Noun phrase chunking is a light version of parsing
  1. Apply tokenization and part-of-speech (POS) tagging to each sentence
  2. Search for noun phrase chunks
Inefficiencies of Linguistic Analyzer

• Difficult to directly apply pre-trained to new domains (e.g. twitter, biomedical, yelp)
  • Unless sophisticated, manually curated, domain-specific training data are provided

• Computationally slow.
  • Cannot be applied on web-scale data to support emerging applications

• Lack of the usage of corpora-level information
  • NP sometimes can’t meet the requirements of quality phrases

• We need “shallow” phrase mining techniques
Ranking

• C-Value
  • Prefers “maximal” phrases
  • Popularity & Completeness

• TextRank
  • Similar to PageRank
  • Popularity & Informativeness

• TF-IDF
  • Term Frequency
  • Inverse Document Frequency
  • Popularity & Informativeness
Three Families of Methods

- Supervised (linguistic analyzers)
- Unsupervised (statistical signals)
- Weakly / Distantly Supervised
Unsupervised Phrase Mining

- Statistics based on massive text corpora
- Popularity
  - Raw frequency
  - Frequency distribution based on Zipfian ranks [Deane’05]
- Concordance
  - Significance score [Church et al.’91][El-Kishky et al.’14]
- Completeness
  - Comparison to super/sub-sequences [Parameswaran et al.’10]
Raw Frequency

• Raw frequency could NOT reflect the quality of phrases, because

• Combine with topic modeling
  • Merge adjacent unigrams of the same topic [Blei & Lafferty’09]
  • Frequent pattern mining within the same topic [Danilevsky et al.’14]

• Limitations
  • Tokens in the same phrase may be assigned to different topics
  • E.g. knowledge discovery using least squares support vector machine classifiers...
Frequency Distribution

• Idea: ranks in a Zipfian frequency distribution is more reliable than raw frequency

• Heuristic: Actual Rank / Expected Rank

• Example:
  • Given a phrase like “east end”
  • **Actual Rank**: rank “east end” among all occurrences of “east” (e.g., “east end”, “east side”, “the east”, “towards the east”, etc.)
  • **Expected Rank**: rank “__ end” among all contexts of “east” (e.g., “__ end”, “__ side”, “the __”, “towards the __”, etc.)
Significance score

- Significance score [Church et al.’91]
  - A.k.a. Z score

- ToPMine [El-Kishky et al.’15]
  - If a phrase can be decomposed into two parts
    - \( P = P_1 \cdot P_2 \)
  - \( \alpha(P_1, P_2) \approx (f(P_1 \cdot P_2) - \mu_0(P_1, P_2))/\sqrt{f(P_1 \cdot P_2)} \)
Significance score (cont’d)

• Merge adjacent unigrams greedily if their significance score is above the threshold.
Comparison to super/sub-sequences

• Frequency ratio between an n-gram phrase and its two (n-1)-gram phrases

• Example

<table>
<thead>
<tr>
<th>Phrase</th>
<th>Raw frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>San</td>
<td>14585</td>
</tr>
<tr>
<td>Antonio</td>
<td>2855</td>
</tr>
<tr>
<td>San Antonio</td>
<td>2385</td>
</tr>
</tbody>
</table>

• Pre-confidence of San Antonio: 2385 / 14585
• Post-confidence of San Antonio: 2385 / 2855

• Expand / Terminate based on thresholds
Comparison to super/sub-sequences (cont’d)

• Assumption
  
  An n-gram quality phrase

  Two (n-1)-gram sub-phrases

  At least one of them is not a quality phrase.

• Anti-example
  
  • “relational database system” is a quality phrase.
  
  • Both “relational database” and “database system” can be quality phrases.
Limitations of Statistical Signals

• The thresholds should be carefully chosen.
• Only consider a subset of quality phrase requirements.
• Combining different signals in an unsupervised manner is difficult.
  • Introduce some supervision may help!
Three Families of Methods

- Supervised (linguistic analyzers)
- Unsupervised (statistical signals)
- Weakly / Distantly Supervised
Weakly / Distantly Supervised Phrase Mining Methods

• SegPhrase [Liu et al.’15]
  • Weakly supervised

• AutoPhrase [Shang et al.’17]
  • Distantly supervised
**SegPhrase**

- Outperform all above methods on *domain-specific* corpus (e.g., Yelp reviews)

---

**Raw Corpus**

**Quality Phrases**

**Segmented Corpus**

- **Document 1**
  Citation recommendation is an interesting but challenging research problem in *data mining* area.

- **Document 2**
  In this study, we investigate the problem in the context of *heterogeneous information networks* using *data mining* technique.

- **Document 3**
  Principal Component Analysis is a linear dimensionality reduction technique commonly used in *machine learning* applications.

---

**Input Raw Corpus**

**Phrase Mining**

**Quality Phrases**

**Phrasal Segmentation**

**Segmented Corpus**
Quality Estimation

• Weakly Supervised
  • Labels: Whether a phrase is a quality one or not
    • “support vector machine”: 1
    • “the experiment shows”: 0
  • For ~1GB corpus, only 300 labels

• Pros
  • Binary annotations are easy

• Cons
  • The selection of hundreds of varying-quality phrases from millions of candidates should be careful.
Phrasal Segmentation

- Phrasal segmentation can tell which phrase is more appropriate
  - Ex: A standard [feature vector] [machine learning] setup is used to describe...

Not counted towards the rectified frequency

- Effects on quality re-estimation (real data)
  - np hard in the strong sense
  - np hard in the strong sense
  - data base management system
Interesting Phrases Mined (From Titles & Abstracts of SIGMOD/SIGKDD Proceedings)

<table>
<thead>
<tr>
<th>SIGMOD</th>
<th>SIGKDD</th>
</tr>
</thead>
<tbody>
<tr>
<td>SegPhrase+</td>
<td>SegPhrase+</td>
</tr>
<tr>
<td>Chunking</td>
<td>Chunking</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
</tr>
<tr>
<td>1 data base</td>
<td>data mining</td>
</tr>
<tr>
<td></td>
<td>data base</td>
</tr>
<tr>
<td>2 database system</td>
<td>database system</td>
</tr>
<tr>
<td></td>
<td>data set</td>
</tr>
<tr>
<td>3 relational database</td>
<td>association rule</td>
</tr>
<tr>
<td></td>
<td>knowledge discovery</td>
</tr>
<tr>
<td>4 query optimization</td>
<td>query optimization</td>
</tr>
<tr>
<td></td>
<td>knowledge discovery</td>
</tr>
<tr>
<td>5 query processing</td>
<td>relational database</td>
</tr>
<tr>
<td></td>
<td>time series</td>
</tr>
<tr>
<td></td>
<td>decision tree</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>51 sql server</td>
<td>assoc. rule mining</td>
</tr>
<tr>
<td></td>
<td>search space</td>
</tr>
<tr>
<td>52 relational data</td>
<td>database server</td>
</tr>
<tr>
<td></td>
<td>rule set</td>
</tr>
<tr>
<td>53 data structure</td>
<td>large volume</td>
</tr>
<tr>
<td></td>
<td>concept drift</td>
</tr>
<tr>
<td>54 join query</td>
<td>knowledge acquisition</td>
</tr>
<tr>
<td></td>
<td>concurrency control</td>
</tr>
<tr>
<td>55 web service</td>
<td>gene expression data</td>
</tr>
<tr>
<td></td>
<td>conceptual graph</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>201 high dimensio. data</td>
<td>efficient impl.</td>
</tr>
<tr>
<td></td>
<td>web content</td>
</tr>
<tr>
<td>202 location based serv.</td>
<td>sensor network</td>
</tr>
<tr>
<td></td>
<td>frequent subgraph</td>
</tr>
<tr>
<td>203 xml schema</td>
<td>large collection</td>
</tr>
<tr>
<td></td>
<td>intrusion detection</td>
</tr>
<tr>
<td>204 two phase locking</td>
<td>important issue</td>
</tr>
<tr>
<td></td>
<td>categorical attribute</td>
</tr>
<tr>
<td>205 deep web</td>
<td>frequent itemset</td>
</tr>
<tr>
<td></td>
<td>user preference</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

42
AutoPhrase

- No label selection and annotation effort
- Smoothly support multiple languages
How to get rid of human effort?

• Basic Idea:
  • Knowledge bases can give us a clean positive pool
  • The remaining frequent n-grams form a noisy negative pool. However, the ratio of false negative is low.
  • Ensemble: average the predictions from base classifiers
    • Independence helps to denoise

Ideal error $= \frac{\delta}{2K} \approx$ noise in the negative pool $\approx 10\%$. Empirical error $p$ should be similar.
### AutoPhrase’s Example Results

<table>
<thead>
<tr>
<th>Rank</th>
<th>EN</th>
<th>CN</th>
<th>Translation (Explanation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Elf Aquitaine</td>
<td>江苏 舜 天</td>
<td>(the name of a soccer team)</td>
</tr>
<tr>
<td>2</td>
<td>Arnold Sommerfeld</td>
<td>苦 艾 酒</td>
<td>Absinthe</td>
</tr>
<tr>
<td>3</td>
<td>Eugene Wigner</td>
<td>白发 魔 女</td>
<td>(the name of a novel/TV-series)</td>
</tr>
<tr>
<td>4</td>
<td>Tarpon Springs</td>
<td>笔记 型 电脑</td>
<td>notebook computer, laptop</td>
</tr>
<tr>
<td>5</td>
<td>Sean Astin</td>
<td>党委 书记</td>
<td>Secretary of Party Committee</td>
</tr>
<tr>
<td>20,001</td>
<td>ECAC Hockey</td>
<td>非洲 国家</td>
<td>African countries</td>
</tr>
<tr>
<td>20,002</td>
<td>Sacramento Bee</td>
<td>左翼 党</td>
<td>The Left (German: Die Linke)</td>
</tr>
<tr>
<td>20,003</td>
<td>Bering Strait</td>
<td>菲 沙 河谷</td>
<td>Fraser Valley</td>
</tr>
<tr>
<td>20,004</td>
<td>Jacknife Lee</td>
<td>海马 体</td>
<td>Hippocampus</td>
</tr>
<tr>
<td>20,005</td>
<td>WXYZ-TV</td>
<td>斋 贺 光 希</td>
<td>Mitsuki Saiga (a voice actress)</td>
</tr>
<tr>
<td>99,994</td>
<td>John Gregson</td>
<td>计算机 科学技术</td>
<td>Computer Science and Technology</td>
</tr>
<tr>
<td>99,995</td>
<td>white-tailed eagle</td>
<td>恒 天然</td>
<td>Fonterra (a company)</td>
</tr>
<tr>
<td>99,996</td>
<td>rhombic dodecahedron</td>
<td>中国 作家 协会 副 主席</td>
<td>The Vice President of Writers Association of China</td>
</tr>
<tr>
<td>99,997</td>
<td>great spotted woodpecker</td>
<td>维他命 B</td>
<td>Vitamin B</td>
</tr>
<tr>
<td>99,998</td>
<td>David Manners</td>
<td>奥论 导向</td>
<td>controlled guidance of the media</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>


References


Building Structured Databases of Factual Knowledge from Massive Text Corpora

Part II: Joint Extraction of Typed Entity and Relation
Effort-Light StructMine: Methodology

Text corpus

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

Entity names & context units

Corpus-specific Structure Discovery (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

Structures from the remaining unlabeled data

Knowledge bases

Typing of Entities and Relations vs Meta Pattern-Based Attribute Discovery

Typing of Entities and Relations
Effort-Light StructMine: Typing

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

Structures from the remaining unlabeled data → Corpus-specific Structure Discovery (KDD’15, KDD’16, EMNLP’16, WWW’17) → Partially-labeled corpus

Knowledge bases

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

Text corpus

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

Corpus-specific Structure Discovery (KDD’15, KDD’16, EMNLP’16, WWW’17)

Partially-labeled corpus

entity names & context units

Knowledge bases

Corpus-specific Structure Discovery

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)
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. ...
Traditional Named Entity Recognition (NER) Systems

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

A manual annotation interface

The best BBQ I’ve tasted in Phoenix

NER Systems:
Stanford NER
Illinois Name Tagger
IBM Alchemy APIs

...
Weak-Supervision Systems: Pattern-Based Bootstrapping

- Requires manual seed selection & mid-point checking

Seed entities and corpus

Annotate corpus using entities

Generate candidate patterns

Score candidate patterns

Select Top patterns

Apply patterns to find new entities

Patterns for Food

the best <X> I’ve tried in their <X> tastes amazing

... Systems:

CMU NELL
UW KnowItAll
Stanford DeepDive
Max-Planck PROSPERA

Seeds for Food

Pizza
French Fries
Hot Dog
Pancake
...

e.g., (Etzioni et al., 2005), (Talukdar et al., 2010), (Gupta et al., 2014), (Mitchell et al., 2015), ...
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><strong>Phoenix</strong> is my all-time favorite dive bar in <strong>New York City</strong>.</td>
</tr>
<tr>
<td>S2</td>
<td>The best <strong>BBQ</strong> I’ve tasted in <strong>Phoenix</strong>.</td>
</tr>
<tr>
<td>S3</td>
<td><strong>Phoenix</strong> has become one of my favorite bars in <strong>NY</strong>.</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>Phoenix is my all-time favorite dive bar in New York City.</td>
</tr>
<tr>
<td>S2</td>
<td>The best BBQ I’ve tasted in Phoenix.</td>
</tr>
<tr>
<td>S3</td>
<td>Phoenix has become one of my favorite bars in NY.</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><em>Phoenix</em> is my all-time favorite dive bar in <em>New York City</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>
**ClusType**: Data-Driven Entity Mention Detection

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

\[
\rho_x(S_1, S_2) = \frac{\nu(S_1 \oplus S_2) - N \frac{\nu(S_1)}{N} \frac{\nu(S_2)}{N}}{\sqrt{\nu(S_1 \oplus S_2)}} \cdot \text{Syntactic quality} \cdot I_X(S_1 \oplus S_2)
\]

<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>

**Quality of merging**

Markov Blanket Feature Selection for Support Vector Machines.

Good Concordance
**My Solution: ****ClusType** **(KDD’15)**

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

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

Two subtasks mutually enhance each other

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

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

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

<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 The Apprentice.</td>
</tr>
</tbody>
</table>

A few common types:
- Location
- Person
- Organization

A type hierarchy with 100+ types (from knowledge base):

(Ling et al., 2012), (Nakashole et al., 2013), (Yogatama et al., 2015)
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><em>Donald Trump</em> spent 14 television seasons presiding over a game show, NBC’s The Apprentice</td>
</tr>
</tbody>
</table>

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

**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 The Apprentice</td>
</tr>
</tbody>
</table>

**Text features:** TOKEN_Donald, CONTEXT: television, CONTEXT: season, TOKEN_trump, SHAPE: AA

**S1: Donald Trump**

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

More effective classifiers

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

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

<table>
<thead>
<tr>
<th>ID</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_i$</td>
<td>President <em>Trump</em> gave an all-hands address to troops at the U.S. Central Command headquarters</td>
</tr>
</tbody>
</table>

Vectors for text features

Test mention: $S_i \_Trump$

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

Accuracy on different type levels

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

- **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 WASBIE 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)
- Learning Corpus-specific Model (KDD’15, KDD’16, EMNLP’16, WWW’17)
- Partially-labeled corpus
- 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: An “Incremental” System Pipeline

Error propagation cascading down the 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:

(protest, *January 21, 2017*)

(protest, January 21, 2017)

Relation type errors:

(is a)

(protest, *January 21, 2017*)

Entity type errors:

 erroneously

(protest, *January 21, 2017*)

Relation type errors:

(is a)

(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

- Entity mention detection
- Context-aware entity typing
- Relation mention detection
- Context-aware relation typing

(Ren et al. WWW’17)
**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>Entity 1: <em>Donald J. Trump</em></th>
<th>Entity 2: <em>United States</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>Types: person, politician, businessman, author, actor</td>
<td>Types: location, organization</td>
</tr>
</tbody>
</table>

**Relations between E1, E2 in KB:**

*president of*, *live in*, *born in*
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.

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

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 \( m_2 \):

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

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

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

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

\[ O_{ZM} = \sum_{z_i \in Z_L} \sum_{\nu=1}^V \max \{ 0, 1 + \tau(z_i) - \tau(z_\nu) \} \]

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

\[ O_M = \mathcal{L}_{MF} + \sum_{i=1}^{N_L'} \ell_i' + \frac{\lambda}{2} \sum_{i=1}^{N_L'} \| m_i \|^2 + \frac{\lambda}{2} \sum_{k=1}^{K_r} \| r_k \|^2 \]

\[ O_Z = \mathcal{L}_{ZF} + \sum_{i=1}^{N_L} \ell_i + \frac{\lambda}{2} \sum_{i=1}^{N_L} \| z_i \|^2 + \frac{\lambda}{2} \sum_{i=1}^{K_y} \| y_k \|^2 \]

(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

**LifeNet:**
A Knowledge Exploration and Analytics System for Life Sciences

- **BioInfer Network by human labeling**
  - Human-created
  - 1,100 sentences
  - 94 protein-protein interactions
  - 2,500 man-hours
  - 2,662 facts
  (Pyysalo et al., BMC Bioinformatics’07)

- **LifeNet by Effort-Light StructMine**
  - Machine-created
  - 4 Million+ PubMed papers
  - 1,000+ entity types
  - 400+ relation types
  - <1 hour, single machine
  - 10,000x more facts

Performance evaluation on BioInfer:
Relation Classification Accuracy = 61.7%
(11%↑ over the best-performing baseline)

(Pyysalo et al., BMC Bioinformatics’07)
(Ren et al., ACL’17 demo, under review)
Corpus to Structured Network: The Roadmap

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

Corpus-specific Structure Discovery (KDD’15, KDD’16, EMNLP’16, WWW’17) → Partially-labeled corpus

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)

Knowledge bases
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.
Building Structured Databases of Factual Knowledge from Massive Text Data

Part III: Meta Pattern-Based Attribute Discovery
Attribute Discovery

Given a sentence “President Blaise Compaoré’s government of Burkina Faso was founded…”, …

Given text corpora (news, tweets, paper text, etc.), find

1. \( \langle \text{entity, attribute name, attribute value} \rangle \)
   
   Ex. \( \langle \text{Burkina Faso, president, Blaise Compaoré} \rangle \)
   
   \( \langle \text{Burkina Faso, population, 17 million} \rangle \)
   
   \( \langle \text{Blaise Compaoré, age, 65} \rangle \)

2. \( \langle \text{entity type, attribute name} \rangle \)

   Ex. \( \langle $\text{COUNTRY, president} \rangle \)
   
   \( \langle $\text{LOCATION, population} \rangle \)
   
   \( \langle $\text{PERSON, age} \rangle \)
Approaches

Relation/attribute

Learning

Textual pattern mining and bootstrapping

- Hearst patterns
- Patterns in open-domain information extraction
- Textual patterns with semantic types
Text Mining and Textual Pattern Mining

*Text mining* is the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends …

*Frequent patterns* are itemsets, subsequences, or substructures that appear in a data set with frequency no less than a user-specified threshold. [Wikipedia]

**Definition (Textual pattern mining).** Mining frequent substructures from text data.

```
 instances
Sentence #15  
Sentence #71  
Sentence #366  
```

```
“United States”, “Japan”  → NP, $ENTITY, $LOCATION, $COUNTRY
```

```
textual pattern
+3
```
Hearst’s Lexico-Syntactic Patterns (1992)

NP such as {NP,}* { (or l and) } NP
such NP as {NP,}* { (or l and) } NP
NP {, NP}* {,} (or l and) other NP
NP {,} (including l especially) {NP,}* (or l and) NP …

**PRO:** Designed for very high precision.

**CONs:** But low recall. Only cover “is-a”, later, extended to “part-of” relation – more like “typing”. Unclear if such patterns can be signed for all relations/attributes.

**How to improve recall?**
Bootstrapping

Initialization:

- few seed examples, e.g., for “is-a”,
- for “is-a”, cat-animal, banana-fruit …
- for “organization-location_of_headquarters”,
  - microsoft-redmond, boeing-seattle …

Expansion:

- new patterns
- new instances

Several iterations

CON: Semantic drift. “pattern-based method”, “turn-based strategy”
Tackling Semantic Drift using **Semantic Types**

The **Snowball** System [Agichtein & Gravano, 2000]

- `$STRING1`’s headquarters in `$STRING2`
- `$STRING2`-based `$STRING1`
- `$STRING1`, `$STRING2`

- `$ORGANIZATION`’s headquarters in `$LOCATION`
- `$LOCATION`-based `$ORGANIZATION`
- `$ORGANIZATION`, `$LOCATION`

**Never-Ending Language Learner (NELL)** [Mohamed et al. 2011]

- **Specify** relations/attributes, e.g.,
  - country:president → `$COUNTRY × $POLITICIAN`
  - organization:headquarters → `$ORGANIZATION × $LOCATION`

- **Start with seed examples**
- **Learn**: new entities, new instances, **novel relations**

Approach: bootstrapping + coupled learning called **continuous open-domain information extraction**
Machine Reading at University of Washington

- **KnowItAll** [Etzioni et al., 2005] – bootstrapping using Hearst patterns
- **TextRunner** [Banko et al., 2007] – self-supervised, specific relation models from a small corpus, applied to a large corpus
- **Kylin** [Wu & Weld, 2007] and **WPE** [Hoffmann et al., 2010] – bootstrapping starting with Wikipedia infoboxes and associated articles
- **WOE** [Wu & Weld, 2010] – extends Kylin to open information extraction, using part-of-speech or dependency patterns
- **ReVerb** [Fader et al., 2011] – lexical and syntactic constraints on potential relation expressions
- **OLLIE** [Mausam et al., 2012] – extends WOE with better patterns and dependencies (e.g., some relations are true for some period of time, or are contingent upon external conditions)
Entity-based Textual Patterns: Google’s Advantages

**Biperpedia** [Gupta et al. 2014]: Pipeline and **E-A patterns**

**PRO:** 36 billion anonymized unique queries

Ontology with 1.6M (CLASS, ATTRIBUTE) pairs and 67K attribute names

**CONS:** (query log) Highly constrained and unavailable in academia
Entity-based Textual Patterns: Google’s Advantages

**ARI** [Halevy et al. 2016]: Discover structure in attribute names

Learning the attribute grammar

*Sportscars* tyre price in Singapore

$\text{Attribute\_price} ::= \text{price}$
$\text{Attribute\_price} ::= \text{Nation} \ $\text{Attribute\_price}$
$\text{Attribute\_price} ::= \text{Attribute\_price in Nation}$
$\text{Attribute\_price} ::= \text{Product} \ $\text{Attribute\_price}$
$\text{Nation} ::= \text{Singapore} | \text{USA} | \text{UAE} | \text{UK} | \ldots$
$\text{Product} ::= \text{battery} | \text{insurance} | \text{kit} | \text{door} | \ldots$
Entity-based Textual Patterns: Google’s Advantages

ReNoun [Yahya et al. 2014]: Pipeline and S-A-O patterns

PROs: 8 manually crafted high-precision patterns to find seed triples in corpus

680K unique facts
400M news docs

CONs: (annotated corpus) domain-limited and expensive

1. the A of S, O – the CEO of Google, Larry Page
2. the A of S is O – the CEO of Google is Larry Page
3. O, S A – Larry Page, Google CEO
4. O, S’s A – Larry Page, Google’s CEO
6. SAO – Google CEO Larry Page
7. S A, O – Google CEO, Larry Page
8. S’s A, O – Google’s CEO, Larry Page
**Syntactic-Ontological-Lexical Patterns with Semantic Types**

**PATTY** [Nakashole et al. 2012]

*Definition.* An SOL pattern is the shortest path between two entities in the dependency parse tree.

poss (“government”, “Barack Obama”)
nmod:of (“government”, “United States”)

Output: $POLICITIAN$ government [of] $COUNTRY$

**PRO:** Harnessing typing information (O) from a typing system

**CONS:** Relying on Stanford’s dependency parsers (S & L).
Losing pattern contexts. Lacking pattern organization.
Reminder: Attribute Discovery

Given text corpora (news, tweets, paper text, etc.), find

1. \( \langle \text{entity, attribute name, attribute value} \rangle \)
   
   Ex. \( \langle \text{Burkina Faso, president, Blaise Compaoré} \rangle \)
   
   \( \langle \text{Burkina Faso, population, 17 million} \rangle \)
   
   \( \langle \text{Blaise Compaoré, age, 65} \rangle \)

2. \( \langle \text{entity type, attribute name} \rangle \)
   
   Ex. \( \langle $\text{COUNTRY}, \text{president} \rangle \)
   
   \( \langle $\text{LOCATION}, \text{population} \rangle \)
   
   \( \langle $\text{PERSON}, \text{age} \rangle \)
Organizing Context-Aware Textual Patterns into Synonymous Groups

“President Blaise Compaoré’s government of Burkina Faso …”

→ \( \langle $COUNTRY, \text{president} \rangle, \langle \text{Burkina Faso, president, Blaise Compaoré} \rangle \)

\$COUNTRY President $POLITICIAN
\$COUNTRY’s president $POLITICIAN
President $POLITICIAN of $COUNTRY

…
President $POLITICIAN’s government of $COUNTRY

A new textual pattern: Meta Pattern and a synonymous meta pattern group

Meng Jiang, Jingbo Shang, Taylor Cassidy, Xiang Ren, Lance Kaplan, Timothy Hanratty and Jiawei Han, "MetaPAD: Meta Patten Discovery from Massive Text Corpora", in ACM SIGKDD, 2017.
The MetaPAD Framework: Meta PAHten Discovery from Massive Text Corpora

(#1) “President Blaise Compaoré’s government of Burkina Faso was founded …”
The MetaPAD Framework: Meta PAten Disovery from Massive Text Corpora

(#1) “President Blaise Compaoré’s government of Burkina Faso was founded …”
(#2) “President Barack Obama’s government of U.S. claimed that…”

Meta patterns:

[president $PERSON.POLITICIAN 's government of $LOCATION.COUNTRY] was founded…

Generate patterns with massive instances in the data

($)COUNTRY, {president}, $POLITICIAN)

(U.S., {president}, Barack Obama)
The MetaPAD Framework: Meta PAStten Discovery from Massive Text Corpora

(#1) “President Blaise Compaoré’s government of Burkina Faso was founded …
(#2) “President Barack Obama’s government of U.S. claimed that…”

Meta patterns:

[president $PERSON.POILITICIAN ’s government of $LOCATION.COUNTRY] was founded…

⟨$COUNTRY, {president}, $POLITICIAN⟩

Generate massive triples by matching the meta patterns

⟨Burkina Faso, {president}, Blaise Compaoré⟩
⟨U.S., {president}, Barack Obama⟩
The **MetaPAD** Framework: 
**Meta PAtte n Discovery from Massive Text Corpora**

(#1) “President Blaise Compaoré’s government of Burkina Faso was founded …”
(#2) “President Barack Obama’s government of U.S. claimed that…”
(#3) “U.S. President Barack Obama visited …”

**Meta patterns:**

\[
\text{[president $PERSON.POLITICIAN$’s government of $LOCATION.COUNTRY$] was founded…}
\]

\[
\text{[$LOCATION.COUNTRY$ president $PERSON.POLITICIAN$] …}
\]

\[
\langle $LOCATION.COUNTRY$, \{president\}, $PERSON.POLITICIAN$\rangle
\]

**Group synonymous patterns by massive triples**

\[
\langle \text{Burkina Faso, \{president\}, Blaise Compaoré} \rangle
\]

\[
\langle \text{U.S., \{president\}, Barack Obama} \rangle
\]

**frequency ↑↑↑**
The MetaPAD Framework: Meta PAllen Discovery from Massive Text Corpora

(#1) “President Blaise Compaporé’s government of Burkina Faso was founded …”
(#2) “President Barack Obama’s government of U.S. claimed that…”
(#3) “U.S. President Barack Obama visited …”

Meta patterns:

```
'[president $PERSON.POLITICIAN 's government of $LOCATION.COUNTRY] was founded…
[$LOCATION.COUNTRY president $PERSON.POLITICIAN] …
```

Adjust entity types in meta patterns for appropriate granularity with triples

```
⟨$COUNTRY, {president}, $POLITICIAN⟩

⟨Burkina Faso, {president}, Blaise Compaporé⟩
⟨U.S., {president}, Barack Obama⟩
```
The MetaPAD Framework: Meta PAtt en Dis covery from Massive Text Corpora

(#1) “President Blaise Compaoré’s government of Burkina Faso was founded …”
(#2) “President Barack Obama’s government of U.S. claimed that…”
(#3) “U.S. President Barack Obama visited …”

Meta patterns:

Meta pattern segmentation

 prezident $PERSON.POLITICIAN ’s government of $LOCATION.COUNTRY was founded…
 $LOCATION.COUNTRY president $PERSON.POLITICIAN …

⟨Burkina Faso, {president}, Blaise Compaoré⟩
 ⟨U.S., {president}, Barack Obama⟩

Adjust types for appropriate granularity

Joint extraction

Group synonymous meta patterns
The MetaPAD Framework: Meta PAillon Discovery from Massive Text Corpora

(#1) “President Blaise Compaoré’s government of Burkina Faso was founded …”
(#2) “President Barack Obama’s government of U.S. claimed that…”
(#3) “U.S. President Barack Obama visited …”

Meta patterns:

- No heavy annotation required
- No domain knowledge required
- No query log required

if we can recognize and type the entities in the same manner…

Adjust types for appropriate granularity

synonymous meta patterns
Effort-Light Text Mining

“President Blaise Compaoré’s government of Burkina Faso was founded …”

Phrase mining (SegPhrase Liu et al. 2015)

“president blaise_compaoré ’s government of burkina_faso was founded …”

Entity recognition and typing with Distant Supervision (ClusType Ren et al. 2015)

“president $PERSON ’s government of $LOCATION was founded …”

Fine-grained typing (PLE Ren et al. 2016)

“president $PERSON.POLITICIAN ’s government of $LOCATION.COUNTRY was founded …”
Meta-Pattern Quality Assessment and Segmentation

A rich set of features:

✓ Frequency
✓ Concordance: “$PERSON ’s wife”
✓ Completeness: “$COUNTRY president” vs “$COUNTRY president $POLITICIAN”
✓ Informativeness: “$PERSON and $PERSON ” vs “$PERSON ’s wife, $PERSON”

Regression Q(.): random forest with only 300 labels
Grouping Synonymous Patterns

\(<\text{COUNTRY}, \text{president}, \text{POLITICIAN}\)
Adjusting Types in Meta Patterns for Appropriate Granularity

$\text{PERSON}, \text{DIGIT},$

$\text{PERSON}'s \text{ age is } \text{DIGIT}$

$\text{PERSON}, \text{a DIGIT-year-old}$

$\text{COUNTRY} \text{ president } \text{POLITICIAN}$

$\text{POLITICIAN of } \text{COUNTRY}$
## Results in General Domain

<table>
<thead>
<tr>
<th>Meta patterns</th>
<th>Entity</th>
<th>Attribute value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{COUNTRY} \space \text{President} \space \text{POLITICIAN} $</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\text{COUNTRY}'s \space \text{president} \space \text{POLITICIAN} $</td>
<td></td>
<td></td>
</tr>
<tr>
<td>President $\text{POLITICIAN}'s \space \text{government} \space \text{of} $\text{COUNTRY}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>United States</td>
<td>Barack Obama</td>
</tr>
<tr>
<td>...</td>
<td>Russia</td>
<td>Vladimir Putin</td>
</tr>
<tr>
<td>...</td>
<td>France</td>
<td>Francois Hollande</td>
</tr>
<tr>
<td>...</td>
<td>Burkina Faso</td>
<td>Blaise Compaoré</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\text{COMPANY} \space \text{CEO} \space \text{PERSON} $</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\text{COMPANY} \space \text{chief} \space \text{executive} \space \text{PERSON} $</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\text{PERSON}, \text{the} \space \text{COMPANY} \space \text{CEO}, \space \text{...}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\text{COMPANY} \space \text{former} \space \text{CEO} \space \text{PERSON} $</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\text{PERSON}, \text{the} \space \text{COMPANY} \space \text{former} \space \text{CEO}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>Apple</td>
<td>Tim Cook</td>
</tr>
<tr>
<td>...</td>
<td>Facebook</td>
<td>Mark Zuckerberg</td>
</tr>
<tr>
<td>...</td>
<td>Hewlett-Packard</td>
<td>Carly Fiorina</td>
</tr>
<tr>
<td>...</td>
<td>Infor</td>
<td>Charles Phillips</td>
</tr>
<tr>
<td>...</td>
<td>Afghan Citadel</td>
<td>Roya Mahboob</td>
</tr>
</tbody>
</table>
## Results in Biomedical Domain

<table>
<thead>
<tr>
<th>Meta patterns</th>
<th>Entity</th>
<th>Attribute value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{TREATMENT was used to treat}$ $\text{DISEASE}$</td>
<td>zoledronic acid therapy</td>
<td>Paget's disease of bone</td>
</tr>
<tr>
<td>$\text{DISEASE using the}$ $\text{TREATMENT}$</td>
<td>bisphosphonates</td>
<td>osteoporosis</td>
</tr>
<tr>
<td>$\text{TREATMENT has been used to treat}$ $\text{DISEASE}$</td>
<td>calcitonin</td>
<td>Paget's disease of bone</td>
</tr>
<tr>
<td>$\text{TREATMENT of patients with}$ $\text{DISEASE}$</td>
<td>calcitonin</td>
<td>osteoporosis</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Meta patterns</th>
<th>Entity</th>
<th>Attribute value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{BACTERIA was resistant to}$ $\text{ANTIBIOTICS}$</td>
<td>corynebacterium striatum BM4687</td>
<td>gentamicin</td>
</tr>
<tr>
<td>$\text{BACTERIA are resistant to}$ $\text{ANTIBIOTICS}$</td>
<td>corynebacterium striatum BM4687</td>
<td>tobramycin</td>
</tr>
<tr>
<td>$\text{BACTERIA is the most resistant to}$ $\text{ANTIBIOTICS}$</td>
<td>methicillin-susceptible S aureus</td>
<td>vancomycin</td>
</tr>
<tr>
<td>$\text{BACTERIA, particularly those resistant to}$ $\text{ANTIBIOTICS}$</td>
<td>multidrug-resistant enterobacteriaceae</td>
<td>gentamicin</td>
</tr>
</tbody>
</table>
References


References


Li, J. and Jurafsky, D., 2015. Do multi-sense embeddings improve natural language understanding? In ACL.

Li, J., Luong, M.T., Jurafsky, D. and Hovy, E., 2015. When are tree structures necessary for deep learning of representations? In ACL.


References


References


Building Structured Databases of Factual Knowledge from Massive Text Corpora

Summary
Overall Contributions

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

- Apply the framework to solve three **text structuring tasks**
  → **quick** construction of structured networks for different corpora

- Technology Transfer:
  ![technology_transfer_logos](image)

- A principled approach to manage, explore and analyze “Big Text Data”

![Diagram](image)

- Massive corpus
- Corpus-to-network
- Structured Network
- Network-to-knowledge
- Knowledge
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: Meta Pattern Discovery

• Combining network mining and attribute mining

Text Corpus

Attribute mining

Entities, attributes, relations

Information Network

Network mining

Ranking, clustering, prediction …

Human evaluation

Human in the loop!
Looking Forward: What’s Next?

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

- Knowledge bases
  - Wikipedia
  - DBpedia
  - Yago

- 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 → Personalized Intelligent Systems

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

Collaborate with doctors, social scientists, economists, ...

User content to structured network
Structured information from eHealth records

Smart Health, Business intelligence, Conversational agent
Looking Forward: Integrating with Our Physical World

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

- News streams
- Social media post
- Organization report
- Geo-sensors
- Audio/video sensors
- Bio-sensors

Data Analytics

Smart-City Operating Systems

Traffic management, Sustainable urban system, Cyber-physical system

Collaborate with network & system researchers, environmental scientists, ...
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)

• 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)

• Funding

![ARL](logo.png)  ![NSF](logo.png)  ![NIH](logo.png)  ![Google](logo.png)  ![Microsoft](logo.png)
Thank you! Q&A

- **Effort-Light StructMine**: “accurate” expansion of sparse “matchable” → Corpus-specific labeling free, domain/language-independent
- Quick construction of structured networks for different corpora
- Technology Transfer:
- A principled approach to manage, explore and analyze “Big Text Data”

[Diagram showing the process from massive corpus to knowledge]