TruePIE: Discovering Reliable Patterns in Pattern-Based Information Extraction

Qi Li[1], Meng Jiang[2], Xikun Zhang[1], Meng Qu[1],

Timothy Hanratty[3], Jing Gao[4], and Jiawei Han[1]

1.University of Illinois at Urbana-Champaign

2. University of Notre Dame

3. US Army Research Laboratory

4. University at Buffalo

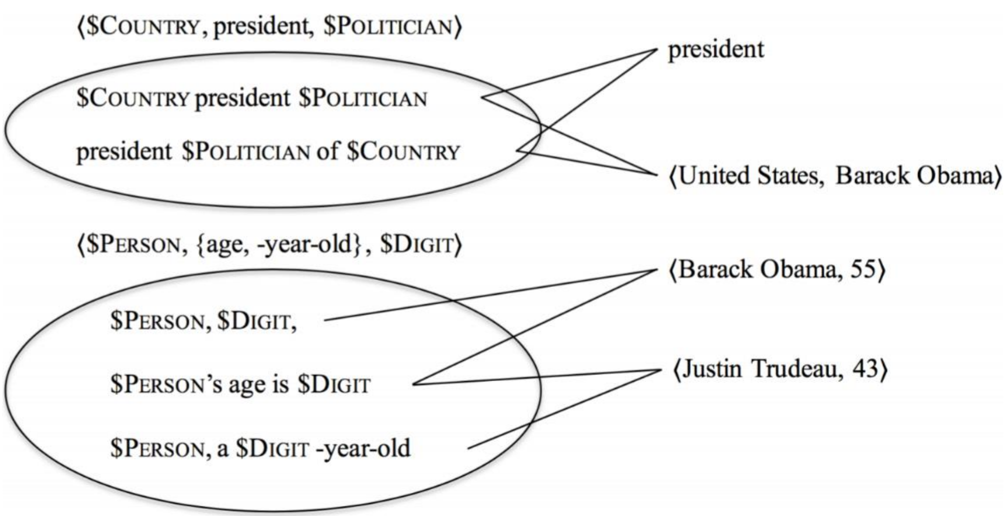
Information Extraction on Text

- **Automatically extracting structured information from unstructured and/or semi-structured documents.**
- **Information extraction from text**
  - **Machine learning methods**
    - Use linguistic features and train machine learning models on a labeled corpus
  - **Textual pattern methods**
    - Based on statistics on a large corpus, such as frequency

Pattern-based Information Extraction

- **Pattern-based IE methods have been applied in finding a huge collection of <Entity, Attribute, Value>-tuples from massive text corpora.**

1. Formation
2. Grouping
3. Extraction



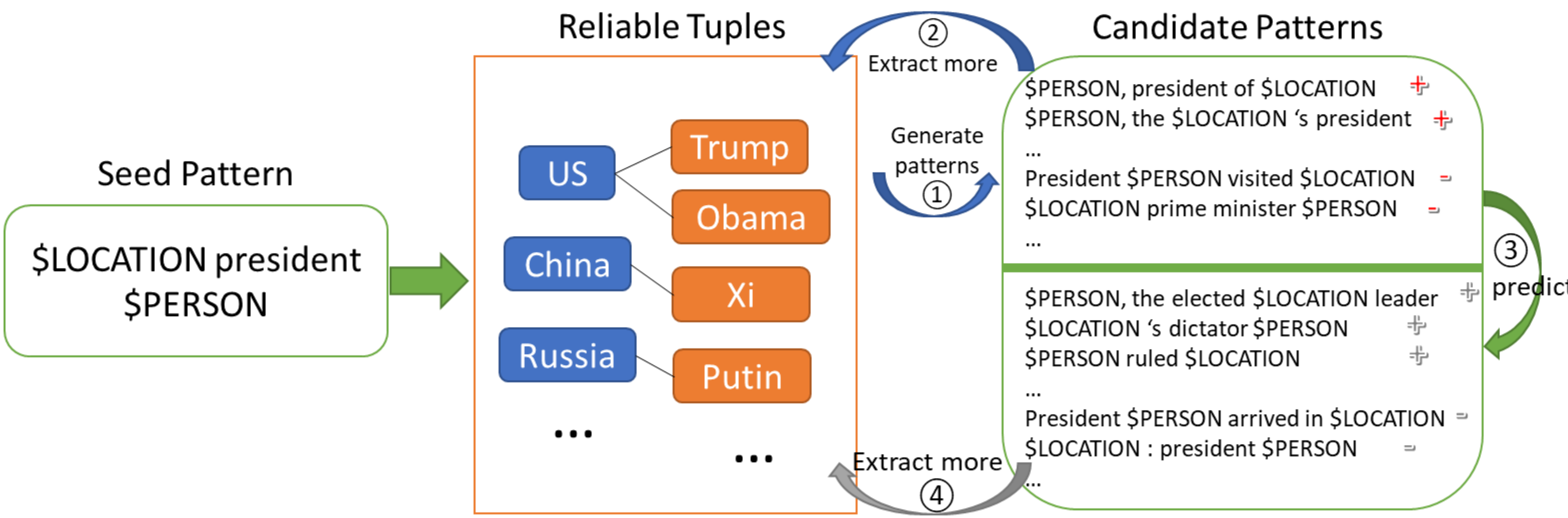
[Meng et al., KDD'17]

Challenge and Solution

- **Issues of existing pattern-based IE methods**
  - group patterns by trigger words (e.g., “married”)
    - Include wrong patterns: [$Person married $Person’s daughter]
    - Miss good patterns: [wedding of $Person and $Person]
  - group patterns by agreement on extractions
    - Miss many good patterns
- **Our solution: pattern reliability estimation**
  - Positive patterns (highly reliable patterns)
  - Negative patterns (highly unreliable patterns)
  - Unrelated patterns: patterns that are unrelated to the task

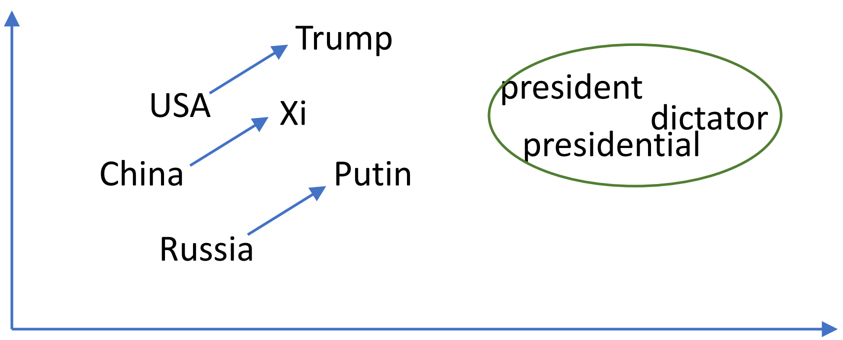
System Overview

- **Given** the text corpus, couple of seed patterns for a specific extraction task *on attribute*
- **Find** as many as possible reliable patterns and correct extractions <entity *e*, attribute *a*, value *v*>



A Intuitive Solution

- **Reliable patterns are semantically similar to the seed patterns**
  - Joint consider pattern constructing words and extractions
  - Eg., $Person , president of $Country
  - Constructing words: president, of
  - Extractions: <Russian, Putin>, <China, Xi>, <USA, Trump>,…
- **Pattern embedding**
  - Adapting word embedding technique



A Intuitive Solution

- **Reliable patterns are semantically similar to the seed patterns**
  - Joint consider pattern constructing words and extractions
  - Eg., $Person , president of $Country
  - Constructing words: president, of
  - Extractions: <Russian, Putin>, <China, Xi>, <USA, Trump>,…
- **Pattern embedding**
  - Adapting word embedding technique
  - $v_p = [\boxed{v_{pw}}, v_{pa}]$

$$\frac{1}{2}(v(president) + v(of))$$

A Intuitive Solution

- **Reliable patterns are semantically similar to the seed patterns**
  - Joint consider pattern constructing words and extractions
  - Eg., $Person , president of $Country
  - Constructing words: president, of
  - Extractions: <Russian, Putin>, <China, Xi>, <USA, Trump>,…
- **Pattern embedding**
  - Adapting word embedding technique
  - $v_p = [v_{pw}, \boxed{v_{pa}}]$

$$\frac{1}{3}\left[\left(v(Russian) - v(Putin)\right) + \left(v(China) - v(Xi)\right) + \left(v(USA) - v(Trump)\right)\right]$$

A Intuitive Solution

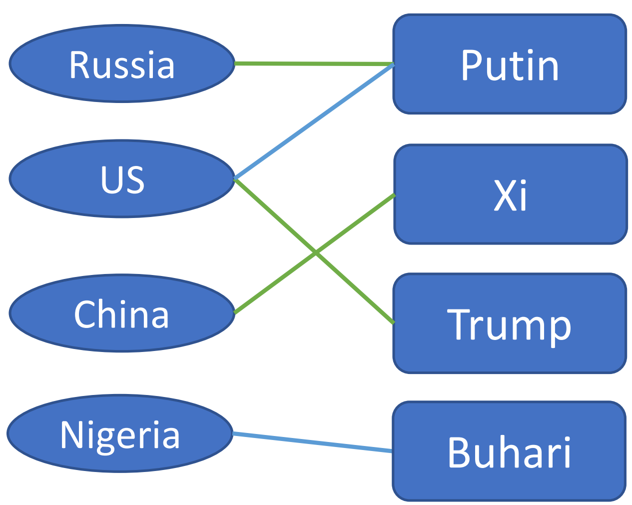
- **Reliable patterns are semantically similar to the seed patterns**
  - Joint consider pattern constructing words and extractions
  - Eg., $Person , president of $Country
  - Constructing words: president, of
  - Extractions: <Russian, Putin>, <China, Xi>, <USA, Trump>,…
- **Pattern embedding**
  - Adapting word embedding technique
  - $v_p = [v_{pw}, v_{pa}]$
  - Reliable patterns are those who are similar to the seed patterns

Issue of the Intuitive Solution

- **Lack of supervision to determine an accurate boundary**
- **Solution**
  - Use the pattern embedding as features
  - Build a training set from the seed patterns
  - Train a classifier

How to Detect Negative Samples

- **Challenge: open world assumption**
  - Eg., the seed pattern does not extract <US, president, Putin> nor <Nigeria, president, Buhari>
- **Arity-constraint**
  - Constraint on degrees of entities and values in an entity-value bipartite graph



Arity-Constraint

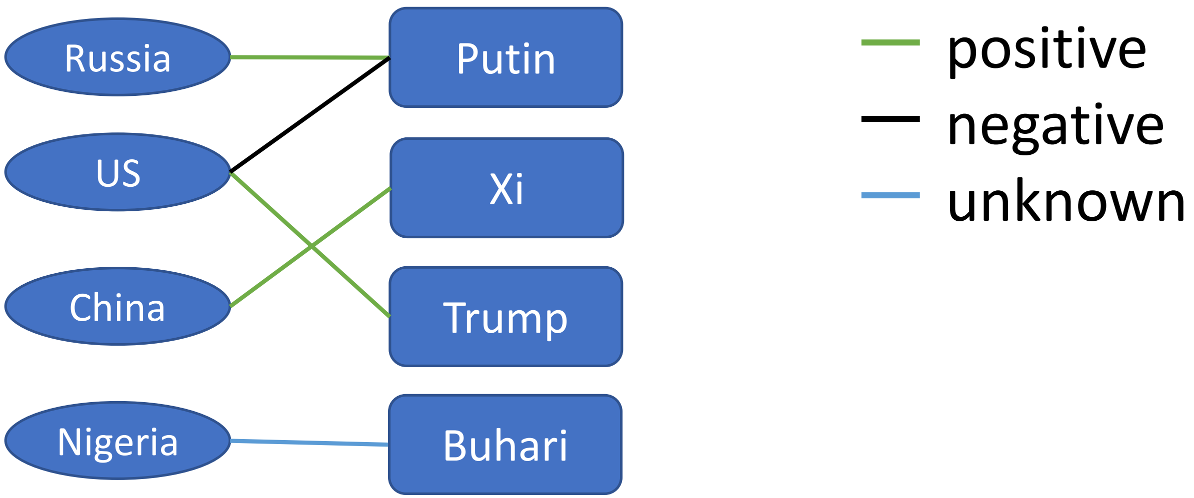
- The arity-constraint is equivalent to setting constraints on the degree of entities C_e and degree of values C_v.

$$C_e^a: \deg(e) \leq median(f_e)$$
$$C_v^a: \deg(v) \leq median(f_v)$$

- **Hard arity-constraint:**
  - If the $median(f) = \beta$-Quantiles(f), we set it as hard arity-constraint
  - For hard arity-constraint, no violation is allowed; e.g., **#country of a president = 1**
- **Soft arity-constraint:**
  - If the $median(f) < \beta$-Quantiles(f), we set it as soft arity-constraint
  - For soft arity-constraint, some violations are allowed; e.g., **#president of a country**
  - If a tuple has a high reliability score, we can add it into the truth tuple set even it may violate the soft arity-constraint.

Arity-constraint-based Conflict Finding

- **Tuple's Polarity**
  - A tuple t is positive, if $t \in T$ (i.e., the true tuple set);
  - t is negative, if $t \notin T$, and adding t to T will cause violation of arity-constraints.
  - t is unknown, if $t \notin T$ and t is not negative



Pattern Reliability

$$\rho_p = \frac{\boxed{|T_p \cap T|} + \frac{1}{2}\boxed{|T_p^u|}}{\boxed{|T_p|}}$$

Pattern reliability score

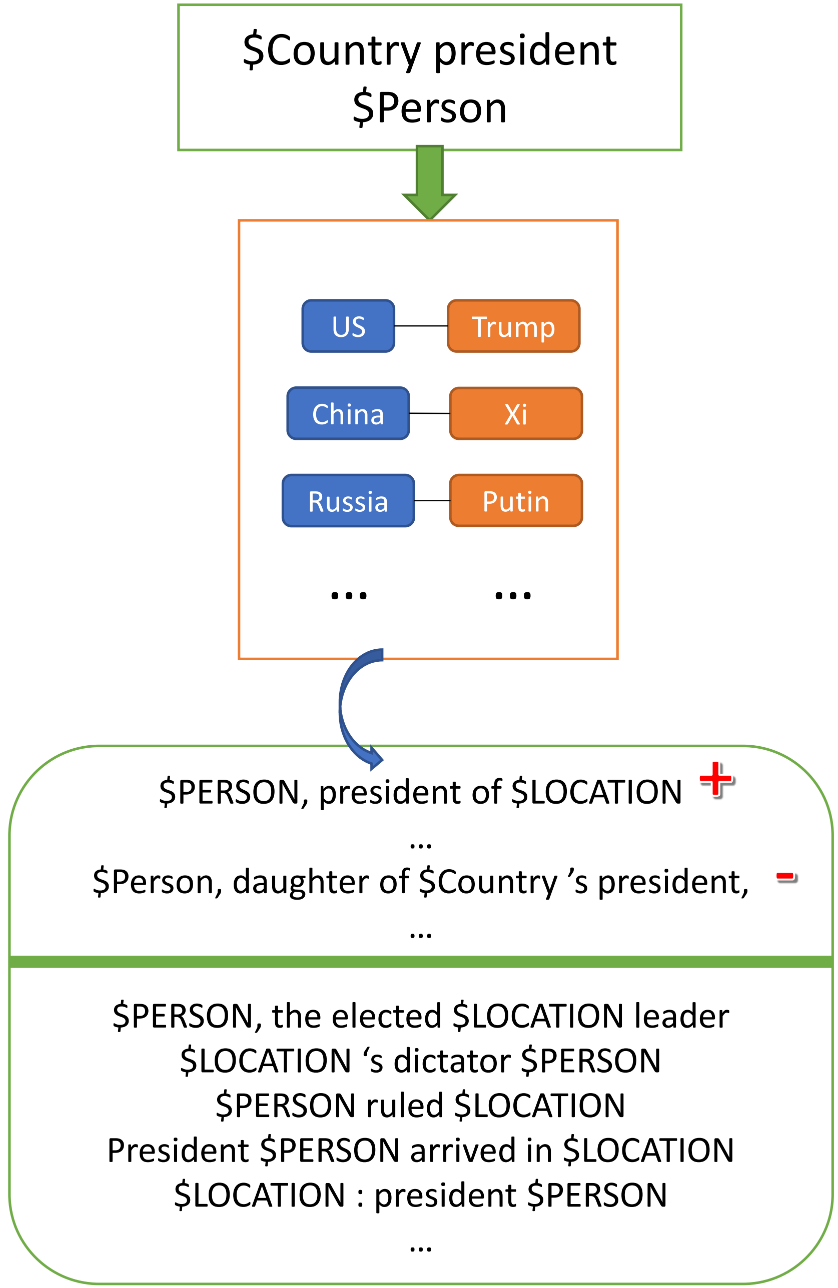
- Extension of precision
  - Number of positive tuples
  - Number of unknown tuples
  - Total number of tuples
- Positive and negative patterns
  - Positive patterns: $\rho_p > \theta$
  - Negative patterns: $\rho_p < 1 - \theta$

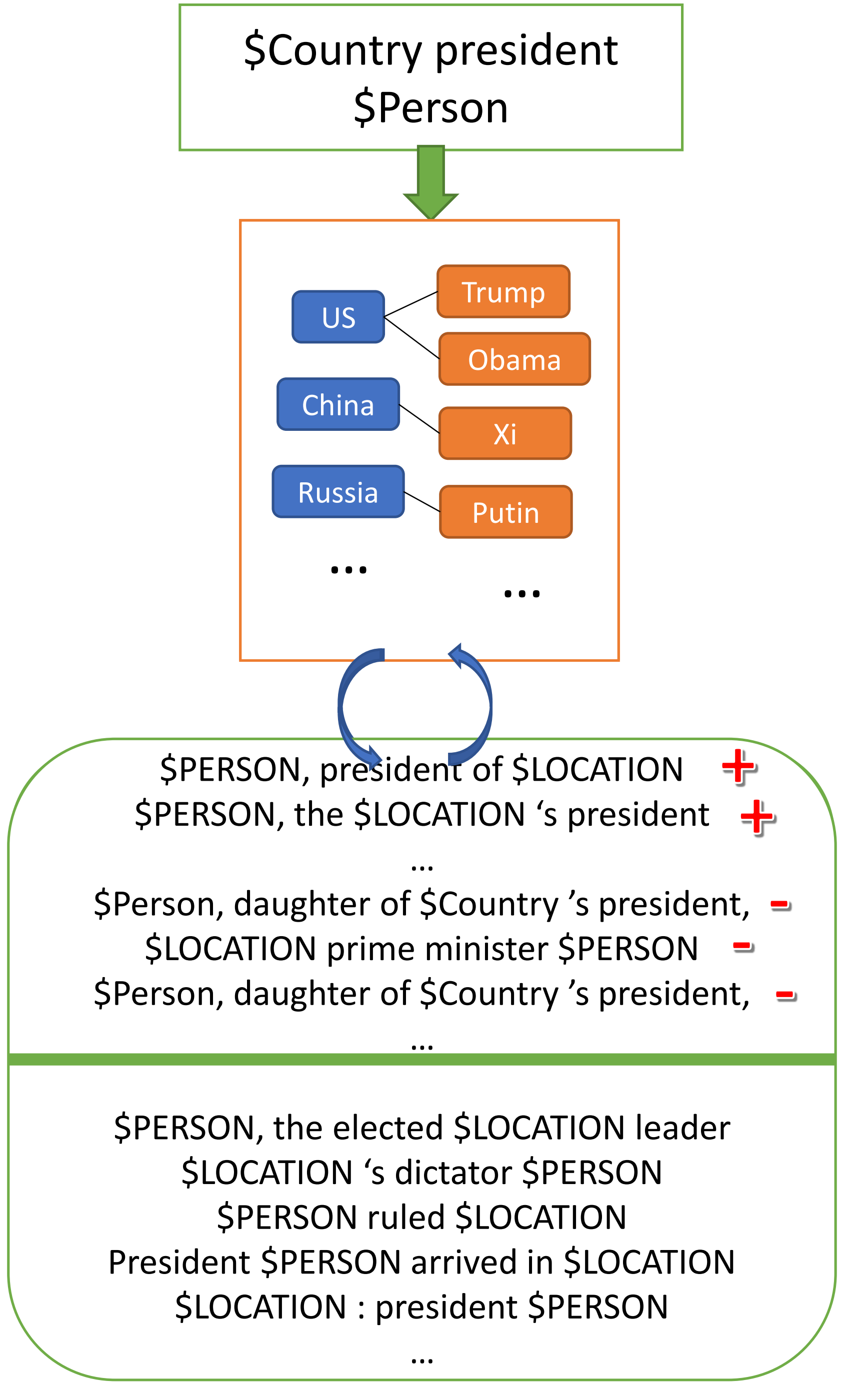
Tuple Reliability

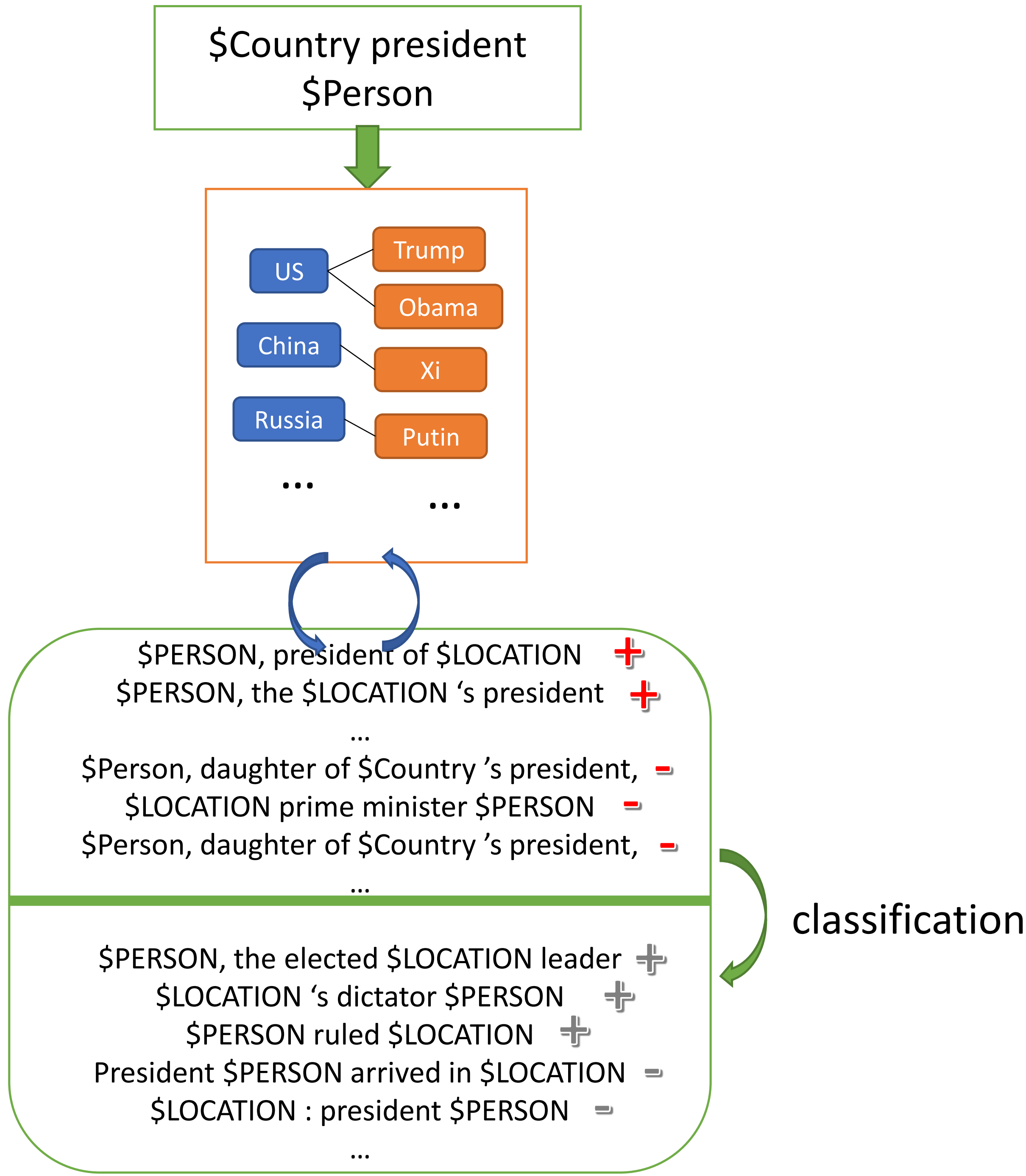
$$\tau_t = \sum_{\{p:p\in P\}} \boxed{\rho_p} \times \boxed{n_t^p}$$

Tuple reliability score

- Edge weight of the entity-value bipartite graph
  - Positive patterns' reliability score
  - Frequency
- Optimization problem: Find the bipartite graph with the maximal sum of edge weights under the arity-constraints
  - Hard arity-constraint: no violation allowed, $+\infty$ penalty
  - Soft arity-constraint: violation allowed with a positive penalty


$Country president
$Person
US
Trump
China
Xi
Russia
Putin
…
…
$PERSON, president of $LOCATION
+
…
$Person, daughter of $Country ’s president,
-
…
$PERSON, the elected $LOCATION leader
$LOCATION ‘s dictator $PERSON
$PERSON ruled $LOCATION
President $PERSON arrived in $LOCATION
$LOCATION : president $PERSON
…


$Country president $Person
US
Trump
Obama
China
Xi
Russia
Putin
…
…
$PERSON, president of $LOCATION +
$PERSON, the $LOCATION ‘s president +
…
$Person, daughter of $Country ’s president, -
$LOCATION prime minister $PERSON -
$Person, daughter of $Country ’s president, -
…
$PERSON, the elected $LOCATION leader
$LOCATION ‘s dictator $PERSON
$PERSON ruled $LOCATION
President $PERSON arrived in $LOCATION
$LOCATION : president $PERSON
…


$Country president $Person
US
Trump
Obama
China
Xi
Russia
Putin
…
…
$PERSON, president of $LOCATION +
$PERSON, the $LOCATION 's president +
…
$Person, daughter of $Country 's president, -
$LOCATION prime minister $PERSON -
$Person, daughter of $Country 's president, -
…
$PERSON, the elected $LOCATION leader +
$LOCATION 's dictator $PERSON +
$PERSON ruled $LOCATION +
President $PERSON arrived in $LOCATION -
$LOCATION : president $PERSON -
…
classification

Experimental Evaluation

- **Corpus**
  - English Gigaword Fourth Edition LDC2009T13
  - 25.7 GB of size including 9.9 million documents and 4.0 billion words
- **State-of-the-art pattern-based IE baselines**
  - PATTY, MetaPAD
- **Performance measure**
  - Precision
    - randomly select 10 sets of 50 extracted tuples and label their correctness
  - Coverage
    - Randomly choose 100 corrected tuples from each method and combine them. Check how many are covered by each method

Performance Comparison

	Task	PATTY	METAPAD	TRUEPIE	Task	PATTY	METAPAD	TRUEPIE
#Extracted Tuples	Leader	2752	4067	2317	President	7801	4917	1490
Average Precision		0.59 ± 0.05	0.43 ± 0.07	**0.87** ± 0.05		0.38 ± 0.08	0.30 ± 0.06	**0.89** ± 0.05
Top 10% Precision		0.89 ± 0.17	0.66 ± 0.30	**0.99** ± 0.03		0.59 ± 0.29	0.42 ± 0.15	**1** ± 0
Top K Precision		0.67 ± 0.12	0.56 ± 0.10	**0.99** ± 0.01		0.56 ± 0.27	0.33 ± 0.07	**0.95** ± 0.04
Coverage Rate		0.56	0.59	**0.61**		**0.87**	0.63	0.71
#Extracted Tuples	Capital	1316	4371	428	Director	10313	14234	5205
Average Precision		0.37 ± 0.07	0.27 ± 0.10	**0.97** ± 0.02		0.54 ± 0.08	0.56 ± 0.07	**0.86** ± 0.05
Top 10% Precision		0.54 ± 0.25	0.47 ± 0.16	**1** ± 0		0.63 ± 0.31	0.65 ± 0.20	**0.93** ± 0.12
Top K Precision		0.51 ± 0.18	0.47 ± 0.16	**0.98** ± 0.02		0.63 ± 0.32	0.67 ± 0.31	**0.89** ± 0.10
Coverage Rate		0.67	**0.92**	0.68		0.52	**0.6**	0.50

Case Study

Task	Positive Patterns	Negative Patterns
Leader	**$Location** president ***$Person*** **$Location** prime minister ***$Person*** **$Location** military ruler ***$Person*** **$Location** 's chancellor , ***$Person*** ,	**$Location** leader told ***$Person*** **$Location** scoring leader ***$Person*** ***$Person*** , son of the **$Location** leader **$Location** 's cricket chief , ***$Person***
Governor	***$Person*** , the **$Location** administrator	**$Location** senator $Person
Capital	**$Location** 's central government in ***$Location*** president sworn in ***$Location*** , **$Location**	**$Location** leader $Person will visit ***$Location*** embassy of **$Location** in ***$Location***
Spouse	***$Person*** 's widower **$Person** $Location president ***$Person*** and first lady **$Person** wedding of prince ***$Person*** and princess **$Person**	***$Person*** 's lover **$Person** , ***$Person*** 's affair with **$Person** ***$Person*** 's girlfriend , **$Person** ,
Parent	**$Person** 's son ***$Person*** **$Person** to his daughter ***$Person***	***$Person*** 's brother , **$Person** , ***$Person*** 's husband **$Person**
Death Year	king **$Person** ($Year - ***$Year***) **$Person** 's ***$Year*** suicide **$Person** 's ***$Year*** funeral killed **$Person** in ***$Year***	**$Person** 's trial in ***$Year*** **$Person** fired him in ***$Year*** **$Person** 's husband died in ***$Year*** **$Person** left in ***$Year***

Error Analysis and Future Work

- Information sparsity
  - Pattern sparsity: extract little information
  - Entity sparsity: appears infrequent in the corpus
- Information ambiguity
  - Fine-grained typing
    - '$Country senator $Person' is semantically different from '$State senator $Person'
  - Entity linking or entity normalization
    - 'John', 'John Kennedy', 'Kennedy', 'J. H. Kennedy'… are they the same person?
    - Are 'John' and 'John' the same person?

Conclusion

- We proposed TruePIE to discover reliable patterns and EAV-tuples from text data
- Only reliable patterns should contribute to the information extraction
- Spotting negative tuples can significantly boost the performance of the information extraction. Arity-constraint is one effective way to do so