KompaRe: A Knowledge Graph Comparative Reasoning System

Lihui Liu (UIUC)  Boxin Du (UIUC)  Yi Ren Fung (UIUC)  Heng Ji (UIUC)  Jiejun Xu (HRL)  Hanghang Tong (UIUC)
Knowledge Graph

• KG = collection of interlinked entities
  • Objects, events or concepts
  • Multiple types of entities and relations exist

• Facts are represented as triples (h, r, t)
  • <‘Paris’, ‘is_a’, ‘city’>
  • <‘Alice’, ‘is_friend_of’, ‘Bob’>
  • ...

![Knowledge Graph Diagram]
Knowledge Graph Applications

Computer Vision [Y. Fang IJCAI-17]

Recommendation [F. Zhang KDD-16]

Question Answering [S. Hu TKDE-18]

Fact Checking [P. Shiralkar ICDM-17]

Traditional Methods for Fact Checking

Claim: <Berkshire_Hathaway, keyPerson, Warren_Buffett>

Knowledge Stream [Prashant et al. ICDM’ 17]

• Limitations
  • Only perform fact checking w.r.t. a single claim
  • None of them supports fact checking w.r.t. multiply claims at the same time

• G. Ciampaglia, P. Shiralkar, and Rocha. 2015. Computational Fact Checking from Knowledge Networks.(PLOS ’15).
Comparative Reasoning

• **Goal**: Find commonality and inconsistency

• An Example

• Advantages: a more complete picture w.r.t. the input clues

Outline

✅ Motivations

→ Problem Definitions
  ▪ Key Ideas and Solutions
  ▪ Experiments and Prototype
  ▪ Conclusion
Problem Definition #1: Single claim fact checking

- **Goal**: Answering whether a claim is true or false

- **Input**:
  - A background knowledge graph $G$
  - A claim as a triple $<s, p, o>$

- **Output**:
  - True or false
Problem Definition #2: Pair-wise fact checking

- **Goal:** Answering whether two separate claims are consistent
  - **Consistent:** Both claims are true at the same time

- **Input:**
  1. A knowledge graph $\mathcal{G}$
  2. A pair of claims which are denoted as:
     \[ <s_1, p_1, o_1> \text{ and } <s_2, p_2, o_2> \]

- **Output:**
  - The two facts are consistent or not

$s_1, o_1, s_2, o_2$ are nodes, $p_1, p_2$ are relationships.

- **Example:**
  - $<$Barack_Obama, majorIn, Political Science$>$
  - $<$Barack_Obama, graduatedFrom, Harvard$>$
Problem Definition #3: Collective fact checking

- **Goal:** Answering whether a query graph is consistent. A query graph consists of a set of inter-connected edges/triples
  - **Consistent:** All claims are true at the same time

- **Input:**
  - (1) A knowledge graph $G$
  - (2) A query graph $Q$

Barack Obama finished both his bachelor’s degree in political science and a master’s degree in law at Harvard University

- **Output:**
  - True or false
Outline

- Motivations
- Problem Definitions
- Key Ideas and Solutions
  - Experiments and Prototype
  - Conclusion
Challenges and Research Questions

- **Goal**: Detecting inconsistency inside a pair of claims or a query graph

- **Challenge #1**: How to express the claim?
  - Claim might not exist in the knowledge graph
  - Q1: How to utilize other related information in the knowledge graph

- **Challenge #2**: How to quantify inconsistency?
  - Too much irrelevant or noise information in the knowledge graph
  - Q2: How to decide the relevant/important information

---

**C1**: Obama

- Study
- Law

- Not exist

**C2**: Obama

- Study
- StudyAt
- Law
- Columbia

Conflict

---

Knowledge Graph

- Obama
- Columbia
Key Idea #1: Knowledge segment extraction

• Challenge #1: How to express the claim?

• Our solution:
  • Transform the knowledge graph into a weighted graph
  • Use K-simple shortest paths between two nodes to find knowledge segment

• Knowledge segment: (KS for short)
  • A connection subgraph of the knowledge graph
  • Describes the semantic context of a piece of given clue
    • i.e., a node, a triple or a query graph

• Advantages:
  • Useful when query claim does not directly exist in KG
  • Utilizing the ‘background’ or related entities
Key Idea #1: Knowledge segment extraction

- **Details**: Knowledge segment extraction
  - Converting the knowledge graph into a weighted graph according to predicate-predicate similarity
    - Co-occurrence matrix
      \[
      \begin{array}{cccc}
      & p_i & 0 & 1 & 0 \\
      p_i & 3 & 4 & 0 & 1 \\
      p_j & 0 & 3 & 1 & 0 & 2 \\
      p_j & 4 & 1 & 5 & 0 & 1 \\
      \end{array}
      \]
    - Similarity function:
      \[
      \text{Sim}(p_i, p_j) = \text{cosine}(\text{row}_i, \text{row}_j)
      \]
  - Finding K-simple shortest paths between two nodes

- Query:
  - Alan Turing
    - wasBornIn
    - United Kingdom

- Knowledge Segment:
  - Alan Turing
    - workAt
    - University of Manchester
    - isCitizenOf
    - United Kingdom
    - isLocatedIn
    - Maida Vale
    - wasBornIn

---

Key Idea #2: Inconsistency quantification

- **Challenge #2:** How to quantify inconsistency?

- Our solution for pair-wise fact checking
  - Influence function
  - Similarity between two knowledge segments
  - Find the nodes, edges, node attributes with the highest influence

\[
\text{Sim}(KS_1, KS_2) = q'(I - cNxAx)^{-1}N_{x}P_{x}
\]

\[
\frac{\partial \text{Sim}(KS_1, KS_2)}{\partial \text{Node}}
\]

\[
\frac{\partial \text{Sim}(KS_1, KS_2)}{\partial \text{Edge}}
\]

\[
\frac{\partial \text{Sim}(KS_1, KS_2)}{\partial \text{Node\_Attribute}}
\]

Key Idea #2: Inconsistency quantification

- Our solution for collective fact checking
  - Transforming the query graph and knowledge segment graph into two line graphs
    - Given a graph $G$, its line graph $L(G)$ is a graph such that
      - Each vertex of $L(G)$ represents an edge of $G$
      - Two vertices of $L(G)$ are adjacent if and only if their corresponding edges share a common endpoint ("are incident") in $G$

- Finding the importance of nodes/edges/node attributes with influence function
  - Node/edge/node attribute influence

\[
\frac{\partial \text{Loss}}{\partial \text{Node}} \quad \frac{\partial \text{Loss}}{\partial \text{Edge}} \quad \frac{\partial \text{Loss}}{\partial \text{Node Attribute}}
\]
Outline

✔ Motivations
✔ Problem Definitions
✔ Key Ideas and Solutions

→ Experiments and Prototype
  ▪ Conclusion
System Architecture

User Interface
- Function selection
- Query input
- Visualization

Online Reasoning
- Single claim fact checking
- Pair-wise claim fact checking
- Collective fact checking

Offline Mining
- Predicate-predicate similarity calculation
- Predicate entropy calculation
Experiments

Datasets Summary:
- **YAGO**
  - 4,295,825 entities
  - 39 predicates
  - 12,430,705 triples
- **Covid-19**
  - 55,434 entities
  - 5,527,628 triples

**Baseline Methods:**
- TransE [Bordes et al. NeurIPS’ 13]
- Jaccard Similarity [Liben-Nowell et al. JASIST’ 07]
- Knowledge Linker [Ciampaglia et al. PLOS’ 15]
- KGMiner [Shi et al. KBS’ 16]
Effectiveness Results

• Pair-wise comparative reasoning
  • 10 query sets. For each query set, each of them contains 300 query pairs
  • Accuracy = $\frac{N}{M}$, $N$ is the number of queries correctly classified. $M$ is the total number of queries

<table>
<thead>
<tr>
<th>Dataset</th>
<th># of queries</th>
<th>TransE</th>
<th>Jaccard</th>
<th>KL</th>
<th>KGMiner</th>
<th>Kompare</th>
</tr>
</thead>
<tbody>
<tr>
<td>Family members positive</td>
<td>300</td>
<td>0.682</td>
<td>0.831</td>
<td>0.618</td>
<td>0.983</td>
<td>0.944</td>
</tr>
<tr>
<td>Family members negative</td>
<td>300</td>
<td>0.335</td>
<td>0.169</td>
<td>1.000</td>
<td>1.000</td>
<td>0.941</td>
</tr>
<tr>
<td>Graduated college positive</td>
<td>300</td>
<td>0.686</td>
<td>0.335</td>
<td>0.502</td>
<td>0.769</td>
<td>0.794</td>
</tr>
<tr>
<td>Graduated college negative</td>
<td>300</td>
<td>0.626</td>
<td>0.993</td>
<td>0.947</td>
<td>0.901</td>
<td>0.994</td>
</tr>
<tr>
<td>Live place positive</td>
<td>300</td>
<td>0.567</td>
<td>0.415</td>
<td>0.489</td>
<td>0.834</td>
<td>0.762</td>
</tr>
<tr>
<td>Live place negative</td>
<td>300</td>
<td>0.802</td>
<td>0.585</td>
<td>0.907</td>
<td>0.900</td>
<td>0.888</td>
</tr>
<tr>
<td>Birth place positive</td>
<td>300</td>
<td>0.590</td>
<td>0.435</td>
<td>0.537</td>
<td>0.698</td>
<td>0.800</td>
</tr>
<tr>
<td>Birth place negative</td>
<td>300</td>
<td>0.845</td>
<td>1.000</td>
<td>0.973</td>
<td>0.927</td>
<td>0.927</td>
</tr>
<tr>
<td>Work place positive</td>
<td>300</td>
<td>0.751</td>
<td>0.319</td>
<td>0.445</td>
<td>0.698</td>
<td>0.720</td>
</tr>
<tr>
<td>Work place negative</td>
<td>300</td>
<td>0.624</td>
<td>0.994</td>
<td>0.942</td>
<td>0.927</td>
<td>0.995</td>
</tr>
<tr>
<td>mean ± std</td>
<td>-</td>
<td>0.651 ± 0.424</td>
<td>0.608 ± 0.302</td>
<td>0.736 ± 0.221</td>
<td>0.864 ± 0.105</td>
<td>0.877 ± 0.095</td>
</tr>
</tbody>
</table>

Accuracy of pair-wise comparative reasoning.
Effectiveness Results

- Collective comparative reasoning
  - YAGO: 6 query sets. Each query of collective comparative reasoning contains 3 edges

<table>
<thead>
<tr>
<th>Dataset</th>
<th># of queries</th>
<th>TransE</th>
<th>Jaccard</th>
<th>KL</th>
<th>KGMiner</th>
<th>Kompare</th>
</tr>
</thead>
<tbody>
<tr>
<td>Birth place positive</td>
<td>300</td>
<td>0.542</td>
<td>0.418</td>
<td>0.389</td>
<td>0.678</td>
<td>0.795</td>
</tr>
<tr>
<td>Birth place negative</td>
<td>300</td>
<td>0.465</td>
<td>0.996</td>
<td>0.968</td>
<td>0.970</td>
<td>0.829</td>
</tr>
<tr>
<td>Live place positive</td>
<td>300</td>
<td>0.448</td>
<td>0.451</td>
<td>0.465</td>
<td>0.635</td>
<td>0.989</td>
</tr>
<tr>
<td>Live place negative</td>
<td>300</td>
<td>0.558</td>
<td>1.000</td>
<td>0.860</td>
<td>0.924</td>
<td>0.743</td>
</tr>
<tr>
<td>Graduated college positive</td>
<td>300</td>
<td>0.488</td>
<td>0.269</td>
<td>0.335</td>
<td>0.585</td>
<td>0.963</td>
</tr>
<tr>
<td>Graduated college negative</td>
<td>300</td>
<td>0.545</td>
<td>0.996</td>
<td>0.928</td>
<td>0.907</td>
<td>0.829</td>
</tr>
<tr>
<td>mean ± std</td>
<td>-</td>
<td>0.508 ± 0.045</td>
<td>0.688 ± 0.313</td>
<td>0.658 ± 0.265</td>
<td>0.783 ± 0.152</td>
<td>0.858 ± 0.089</td>
</tr>
</tbody>
</table>

 Accuracy of collective comparative reasoning.

- Covid-19: Each query contains less than 8 nodes

<table>
<thead>
<tr>
<th>Dataset</th>
<th># of queries</th>
<th>TransE</th>
<th>Jaccard</th>
<th>KL</th>
<th>KGMiner</th>
<th>Kompare</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>36</td>
<td>0.667</td>
<td>0.611</td>
<td>1.000</td>
<td>0.694</td>
<td>1.000</td>
</tr>
<tr>
<td>Negative</td>
<td>36</td>
<td>0.528</td>
<td>0.361</td>
<td>0.722</td>
<td>0.553</td>
<td>0.863</td>
</tr>
<tr>
<td>Average accuracy</td>
<td>-</td>
<td>0.598 ± 0.071</td>
<td>0.486 ± 0.126</td>
<td>0.861 ± 0.138</td>
<td>0.623 ± 0.071</td>
<td>0.932 ± 0.063</td>
</tr>
</tbody>
</table>

 Accuracy of Covid-19 dataset.
**Effectiveness vs. Running Time**

- The runtime for semantic subgraph extraction scales sub-linearly w.r.t. the number of nodes in the knowledge graph.
- Average runtime of comparative reasoning is less than 8 seconds.
System Demonstration

- System demonstration: https://github.com/lihuiliullh/KompaRe
Outline

- Motivations
- Problem Definitions
- Key Ideas and Solutions
- Experiments and Prototype

Conclusion
Conclusion

• Contribution:
  • *We build a knowledge graph comparative reasoning system*

• Support functions:
  • Key function (1): single claim fact inconsistency checking
  • Key function (2): pair-wise fact inconsistency checking
  • Key function (3): collective fact inconsistency checking

• Results:
  • High accuracy of fact inconsistency checking
  • Fast running time on large knowledge graphs
  • Sublinear scalability
Thank you!

- KompaRe: A Knowledge Graph Comparative Reasoning System
- System demonstration: https://github.com/lihuiliullh/KompaRe
- Lihui Liu: lihuil2@illinois.edu