Training Camp on "**Knowledge Graph Completion**"

– Sapienza University, M.Sc. Degree in Data Science –

Fabio Galasso, Laura Laurenti, Alessio Sampieri
Sapienza University of Rome

Ilaria Bordino, Francesco Gullo, Lorenzo Severini
UniCredit Services
"AI, Data & Analytics ICT" Department
"Applied Research & Innovation" unit

https://sapienza-training-camp2021jun.github.io/

June 30th – July 2nd, 2021

Day 3: Combining Rule Mining and Embedding Learning for Knowledge Graph Completion

Schedule

- Introduction to rule mining
  - Mining Rules from Knowledge Graph Data
  - Amie
- Employing rule-learning and embedding learning for Knowledge Graph Reasoning
  - Advantages and Limitations of Rule-Based learning
  - Advantages and Limitations of Embedding-Based learning
- Combining rules and embeddings for Knowledge Graph Reasoning
  - IterE: Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning
- Lab
  - Extracting rules from KG data with AMIE
  - Exploiting rules for knowledge graph completion
  - Combining rule mining and embedding learning

Expected outcome at the end of Day 3

Capability of:

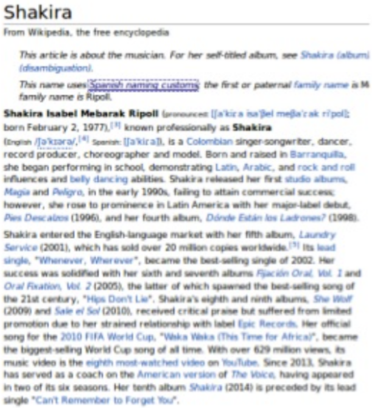
- Extracting rules from knowledge graph data exploiting existing tools (AMIE)
- Instantiate rules and exploit them to produce new triples and/or new features for Knowledge Graph Completion
- Combining embedding learning with rule mining to achieve improved prediction performance

References

- [*amie paper*] AMIE: Association Rule Mining under Incomplete Evidence in Ontological Knowledge Bases
- [*amie repo*] https://github.com/lajus/amie
- [*itere paper*] Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning
- [*itere repo*] https://github.com/wencolani/IterE

Natural Language vs Knowledge Bases (KBs)

Natural Language



Shakira

From Wikipedia, the free encyclopedia

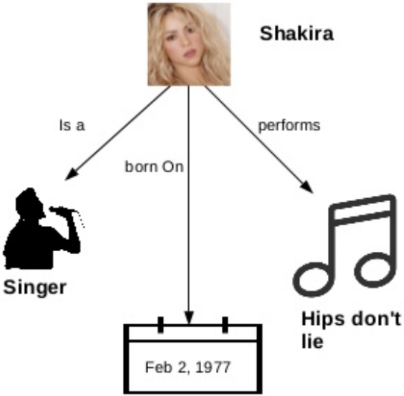
This article is about the musician. For her self-titled album, see Shakira (album) (disambiguation).

This name uses Spanish naming customs: the first or paternal family name is M family name is Ripoll.

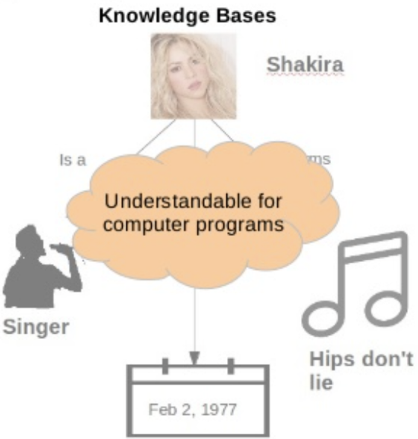
Shakira Isabel Mebarak Ripoll (pronounced: [ʃaˈkiɾa isaˈβel meβaˈɾak riˈpol]; born February 2, 1977),[3] known professionally as **Shakira** (English /ʃəˈkɪərə/,[4] Spanish: [ʃaˈkiɾa]), is a Colombian singer-songwriter, dancer, record producer, choreographer and model. Born and raised in Barranquilla, she began performing in school, demonstrating Latin, Arabic, and rock and roll influences and belly dancing abilities. Shakira released her first studio albums, *Magia* and *Peligro*, in the early 1990s, failing to attain commercial success; however, she rose to prominence in Latin America with her major-label debut, *Pies Descalzos* (1996), and her fourth album, *Dónde Están los Ladrones?* (1998).

Shakira entered the English-language market with her fifth album, *Laundry Service* (2001), which has sold over 20 million copies worldwide.[7] Its lead single, "Whenever, Wherever", became the best-selling single of 2002. Her success was solidified with her sixth and seventh albums *Fijación Oral, Vol. 1* and *Oral Fixation, Vol. 2* (2005), the latter of which spawned the best-selling song of the 21st century, "Hips Don't Lie". Shakira's eighth and ninth albums, *She Wolf* (2009) and *Sale el Sol* (2010), received critical praise but suffered from limited promotion due to her strained relationship with label Epic Records. Her official song for the 2010 FIFA World Cup, "Waka Waka (This Time for Africa)", became the biggest-selling World Cup song of all time. With over 629 million views, its music video is the eighth most-watched video on YouTube. Since 2013, Shakira has served as a coach on the American version of *The Voice*, having appeared in two of its six seasons. Her tenth album *Shakira* (2014) is preceded by its lead single "Can't Remember to Forget You".

Knowledge Bases



Natural Language vs Knowledge Bases (KBs)

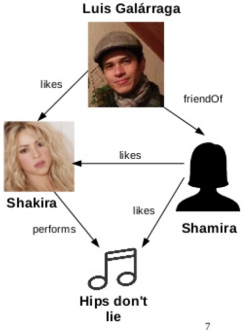


Some popular KBs

Social graphs are KBs

They both share

- A natural **graph-like structure**
- Incompleteness
- Opportunities for data description and prediction
- E.g., If you like Shakira, you are likely to buy her latest song



Rule Mining and KBs

- Data mining is about finding **interesting** and **non-obvious** correlations in the data
- Correlations may be seen as **rules** that hold often
  - You probably live in the same city of your spouse
  - If you like an artist, you probably like her songs
- Correlations can be represented as **logical rules**:
  - $isMarriedTo(x, y) \wedge livesIn(x, city) \Rightarrow livesIn(y, city)$
  - $likes(x, artist) \wedge performs(artist, song) \Rightarrow likes(x, song)$
- Rules allow to exploit social data for real-world applications (e.g., product recommendation)

Applications for social data

- Market basket analysis
  - People who buy laptops also buy laptop cases
- Link and Event Prediction
  - Two people who attended the same high school the same year might know each other
  - If you registered for a conference in Rome, then you are coming to Rome (and you need to book a flight and a hotel)
- Dealing with Incompleteneess
  - If you like German newspapers, fluency in German is probably missing in your profile

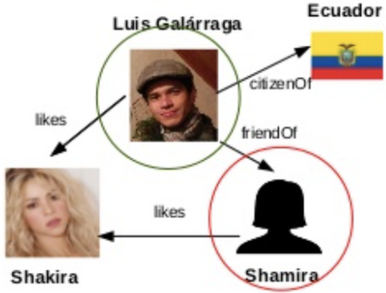
AMIE: Association Rule Mining under Incomplete Evidence

- AMIE is a system that learns **Horn rules** such as:
  - $livesIn(x, city) \land isMarriedTo(x, y) \Rightarrow livesIn(y, city)$
- Starting with all possible head relations $r(x, y)$ and a minimum support threshold:
  - The system explores the search space by means of carefully designed mining operators
  - Search space is restricted to **closed Horn rules**
  - **Head coverage** is used for pruning: we are not interested in rules that cover only very few facts of the head relation
  - E.g., Rules that cover, for example, less than 1% of the facts of the head relation can safely assumed to be marginal
  - Head coverage decreases monotonically as we add more atoms. This allows us to safely discard any rule that trespasses the threshold
  - If a rule $B_1 \land B_2 \land \cdots \land B_n \land B_{n+1} \Rightarrow H$ does not have larger confidence than the rule $B_1 \land B_2 \land \cdots \land B_n \Rightarrow H$, then we do not output the longer rule: both confidence and head coverage of the longer rule are necessarily dominated by the shorter rule.

- **Incompleteness**:
  graph data often contains gaps
- **Open World Assumption** (OWA):
  absence of evidence is not evidence of absence
- Problems to estimate the **confidence** of a rule

likes(x, Shakira) => isCitizenOf(x, Ecuador)

Standard confidence uses a CWA and counts Shamira as counterexample.
Score = 0.5



Challenges of rule mining on KBs

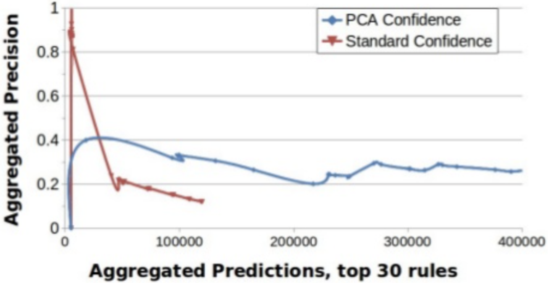
- AMIE uses the **Partial Completeness Assumption** (PCA) to estimate the confidence of rules under OWA
- A KB knows **all or none** of the nationalities of a person

likes(x, Shakira) => isCitizenOf(x, Ecuador)

PCA confidence considers as counterexamples only those people whose nationality is known to be different from Ecuador. Score = 1.0

AMIE: Predictive Behavior

PCA confidence has better predictive behaviour than standard confidence



Examples of rules mined by AMIE on YAGO

```
isMarriedTo(x, y) ∧ livesIn(x, z) => livesIn(y, z)
isCitizenOf(x, y) => livesIn(x, y)
hasAdvisor(x, y) ∧ graduatedFrom(x, z) => worksAt(y, z)
hasWonPrize(x, Gottfried Wilhelm Leibniz Prize) => livesIn(x, Germany)
```

Knowledge Graph Reasoning (KGR)

- **Knowledge graph reasoning** (KGR) can infer new knowledge based on existing ones and check knowledge consistency
- Applications: Knowledge graph **cleaning** and **completion**
- Two main learning methods for KGR:
- **Embedding-based** reasoning
  - learns latent representations of entities and relations in continuous vector spaces, called embeddings, so as to preserve the information and semantics in KGs
  - more efficient when there are a large number of relations or triples to reason over
- **Rule-based** reasoning
  - aims to learn deductive and interpretable inference rules
  - precise and can provide insights for inference results

Limitations of learning methods for KGR

- **Sparsity** Problem for Embedding Learning
  - Poor capability of encoding **sparse entities** (those with only a few triples)
  - Prediction results of entities are highly related to their frequency
- **Efficiency** Problem for Rule Learning
  - Search space exponential to the number of relations

Combining embedding learning and rule learning

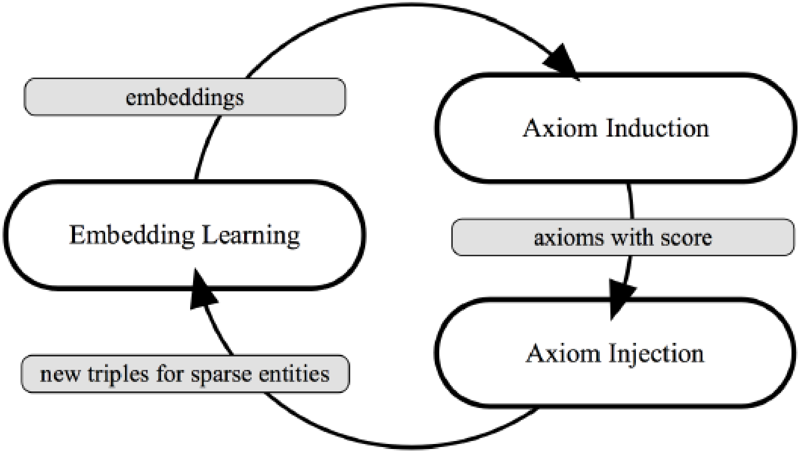
- With different advantages and difficulties, embedding learning and rule learning can benefit and complement each other
- Deductive rules can infer additional triples for sparse entities and help embedding learning methods encode them better.
- Embeddings encoded with rich semantics can turn rule learning from discrete graph search into vector space calculation, so that reduce the search space significantly

IterE: Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning

Zhang et al., 2019

- A framework that iteratively learns embeddings and rules
- Can combine different embedding methods and kinds of rules
- 3 main parts:
  1. Embedding learning: learns embeddings for entities and relations, with input including triples existing in KG and those inferred by rules.
  2. Axiom induction: generates a pool of possible axioms with an effective pruning strategy, then assigns a score to each axiom based on calculation between relation embeddings
  3. Axiom injection: utilizes axioms' deductive capability to infer new triples for sparse entities to be injected into KG
- The three parts are conducted iteratively during training.

IterE: Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning
Zhang et al., 2019



How to exploit rule mining for knowledge graph completion?

- Mine rules on our training set
- Instantiate rules: either all of them, or setting a threshold on PCA confidence
- Derive new KG triples (with a confidence label)
- Use instantiated rules for **direct prediction**: given a test triple, predict it as *true*
  - if I can find it as output of a rule
  - if I can find it as output of a rule with confidence above a threshold
  - if I can find it as output of many rules
- Use instantiated rules to **add new features to the classifier** that predicts whether a triple is *true* or *false*: e.g.,
  - Number of rules which a triple is involved in
  - Min, avg, max confidence of the rules which a triple is involved in
- Use rules to **improve the training of triple embeddings**:
  - Instantiate all rules (with a min confidence) and **add new triples to the training set**
  - Retrain embeddings and check if a better model can be learnt (Once, or iteratively)
  - Try of the existing models that iteratively combine rule mining and embedding learning: [*itere*] Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning