# Building & Mining Knowledge Graphs

# (KEN 4256)

# Assignment 2

# Mining relations from KG and KG Quality Assessment

Due date: 20 March 18:00 CET

In this assignment, you will be supplied with an RDF knowledge graph contained in the file "train.nt" and "valid.ttl", which is a subgraph extracted from the Microsoft Academic Graph and Microsoft Academic Knowledge Graph. This subgraph consists of information about conference papers and entities belonging to related classes (e.g. author, domain, paper, conference). For the given types/properties listed below, we have removed **some** triples in which these relations occur. Your task will be to predict these missing links between authors **using only the information in this graph.**

Your general task will be to predict new objects (*o)* for given subject-predicate pairs (*s,p*) such that <s> <p> <o> could be one of the missing triples in the supplied knowledge graph. The given knowledge graph consists of the following semantic types and properties:

Types:

1. Paper (<http://MAGexample.org/type/paper>)
2. Author (<http://MAGexample.org/type/author>)
3. Domain (<http://MAGexample.org/type/domain>)
4. Affiliation (<http://MAGexample.org/type/affiliation>)
5. Conference (<http://MAGexample.org/type/conference>)

Properties

1. memberOf <http://www.w3.org/ns/org#memberOf> (author memberOf affiliation)
2. hascreator <http://purl.org/dc/terms/creator> (paper creator author)
3. appearsInConferenceSeries <https://makg.org/property/appearsInConferenceSeries> (paper appearsInConferenceSeries conference)
4. hasDiscipline <http://purl.org/spar/fabio/hasDiscipline> (paper hasDiscipline domain)
5. hasCoauthor <http://lsdis.cs.uga.edu/projects/semdis/opus#coauthor> (author coauthor author)

6. cites <http://purl.org/spar/cito/cites> (paper cites paper)
7. a <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>

## Tasks:

- Link prediction: (work with `train.nt`)
  - **TASK1:** Define hand-crafted SPARQL queries (based on “horn-like” rules) for each relation *p* above to insert the missing relations <s> <p> <o> into the given graph. (SPARQL query tutorial could be found at colab)
    For example: you define this rule
    `:authorA hasCoauthor :authorB <= :somepaper hascreator :authorA, :somepaper hascreator :authorB`
    , then you can use this SPARQL update-query:

```
INSERT {
        ?authorA
<http://lsdis.cs.uga.edu/projects/semdis/opus#coauthor> ?authorB .
}
WHERE {
        ?somepaper <http://purl.org/dc/terms/creator> ?authorA .
        ?somepaper <http://purl.org/dc/terms/creator> ?authorB .
}
```

    **Note:** You may also supply queries based on **certain** rules (whose confidence score is always 100%) but keep in mind that there might not be any suitable ones for this particular graph. For each **uncertain** hand-crafted rule that you supply, you will give a confidence score for that rule (it will be a number less than 100%). We will consider those rules with confidence scores equal to or higher than 60% as **high quality** rules. Those that have confidence scores of between [30 and 60%) will be considered **medium quality** rules and those that are between [0 and 30%) will be considered **low quality rules**. Choose 5 from predicates 1-7 above and provide at least one **high quality** rule OR **medium quality** rule.

  - **TASK2:** Use automated rule mining techniques that you studied in class to **learn** new rules which predict (s, o) pairs that could be related via the relation *p,* for at least 5 of predicates 1-7 above. Again, one **high quality** rule or **medium-quality** rule per predicate should be specified. Formulate SPARQL queries for inserting the new triples into the graph and specify these queries in the answer sheet provided.

  - **TASK3:** Apply the rules which perform the best (according to confidence scores) for each predicate in Tasks 1 & 2, to the given graph, to predict and insert the missing relations. Provide your queries for insertion. From the

updated KG, also report the new confidence score for each rule and explain any changes.

- Validation: (work with `valid.ttl`)
  - **TASK4:** Your task is to create and apply ShEx or SHACL constraints to verify that the knowledge graph meets a prescribed structure. We inserted errors in the given knowledge graph (`valid.ttl`).
    Some ShEx or SHACL example could be found at http://shex.io/webapps/shex.js/doc/shex-simple.html and https://shacl-playground.zazuko.com (or https://shacl.org/playground/)
    From the given RDF turtle file you will need to consider the domain/range of each predicate:
    - Define a ShEx or SHACL shape file for the types listed to validate those entities
    - Find the errors which can be found in the given RDF knowledge graph (`valid.ttl`)

## Submission instructions:

- For Tasks 1 and 2 above, use the sample submission template called "BMKG_Assignment_2_handcrafted_rules.txt" to record your SPARQL queries (rules) for each predicate along with the confidence scores for each rule.
- For Tasks 3 above, use the submission template called "BMKG_Assignment_2_predictions.xlsx" to record your top 10 (subject,object) pairs with highest confidence scores after applying your rules.
- Your ShEx/SHACL shapes should be in the "BMKG_Assignment_2_valid.txt" file (not PDF). Do not forget to provide the ShapeMap if you use ShEx.
- You will also submit a **written report** (**max 4 pages** excluding the front title page) in PDF format called "BMKG - Assignment 2 - Group *your number here*.pdf" describing:
  - for Task 1, your thought processes and motivation for crafting each rule,
  - for Task 2, your motivations for selecting the methods you did for learning the new rules, your hypotheses about how well they will perform, and your motivations about why you have these hypotheses,
  - for Task 3, explain why applying the rules to predict the relations is necessary and discuss the performance of your predictions, what are the limitations? How could they be improved?
  - For Task 4
    - A brief description of the steps you took to perform the task, and explanations on the constraints you chose to define the schema.
    - Results of the quality assessment using ShEx or SHACL: report on the nodes which passed and did not pass the validation using your ShEx or SHACL code. Which were the errors in this file? Report the erroneous entities, and describe what the cause of the issues was.
- **One person** from your group will submit a **SINGLE .zip archive** to Student Portal, containing the following three files:
  - "BMKG_Assignment_2_handcrafted_rules.txt"
  - "BMKG_Assignment_2_predictions.xlsx"
  - "BMKG - Assignment 2 - Group *your number here*.pdf"

    - “BMKG_Assignment_2_valid.txt”
- **Very important:**
    - Please be very careful to correctly paste your SPARQL queries in the answer sheet for Tasks 1 and 2 (“BMKG_Assignment_2_handcrafted_rules.txt”). We will copy-paste these to test your rules, so if there are any typos or mistakes in the syntax or spelling - we will not be able to test them and you will not receive the marks for that query.
    - Similarly, please carefully paste the correct URIs for the subjects and objects in the answer sheet for Tasks 3 and 4 (“BMKG_Assignment_2_predictions.xlsx”), otherwise we cannot assess your work and you cannot receive the marks.
    - In your report, it is not sufficient to just state the steps you took in the assignment. It is also important to emphasise in your own words **why** each step is necessary or beneficial for the final solution. Please try to state each choice and motivation very clearly throughout the report. Please also state clearly your **group number** on the front page of your report.

# Resources

- ShEx primer: http://shex.io/shex-primer/
- SHACL/SheX web UI:i http://rdfshape.weso.es/validate
- SHACL web UI: https://shacl.org/playground/
- Lab 9 KG Quality - SHACL vs ShEx
- Querying with RDFLib and colab
- Insert/Updating with RDFLib

## Contact:

Dr. Remzi Celebi: remzi.celebi@maastrichtuniversity.nl

Xu Wang: xu.wang@maastrichtuniversity.nl

Shervin Mehryar: shervin.mehryar@maastrichtuniversity.nl