# Describing and Organizing Semantic Web and Machine Learning Systems in the SWeMLS-KG

Fajar J. Ekaputra[1,2], Majlinda Llugiqi[1], Marta Sabou[1], Andreas Ekelhart[3,4], Heiko Paulheim[5], Anna Breit[6], Artem Revenko[6], Laura Waltersdorfer[2], Kheir Eddine Farfar[7], and Sören Auer[7,8]

[1] WU (Vienna University of Economics and Business) first.last@wu.ac.at
[2] TU Wien first.last@tuwien.ac.at
[3] University of Vienna first.last@univie.ac.at
[4] SBA Research first.last@sba-research.org
[5] University of Mannheim first.last@uni-mannheim.de
[6] Semantic Web Company first.last@semantic-web.com
[7] TIB Leibniz Information Centre for Science and Society first.last@tib.eu
[8] L3S Research Center, Leibniz University of Hannover auer@l3s.de

**Abstract.** The overall AI trend of creating neuro-symbolic systems is reflected in the Semantic Web community with an increased interest in the development of systems that rely on both *Semantic Web resources* and *Machine Learning components* (SWeMLS, for short). However, understanding trends and best practices in this rapidly growing field is hampered by a lack of standardized descriptions of these systems and an annotated corpus of such systems. To address these gaps, we leverage the results of a large-scale systematic mapping study collecting information about 470 SWeMLS papers and formalize these into one resource containing: (i) the *SWeMLS ontology*, (ii) the *SWeMLS pattern library* containing machine-actionable descriptions of 45 frequently occurring SWeMLS workflows, and (iii) *SWEMLS-KG*, a knowledge graph including machine-actionable metadata of the papers in terms of the SWeMLS ontology. This resource provides the first framework for semantically describing and organizing SWeMLS thus making a key impact in (1) understanding the status quo of the field based on the published paper corpus and (2) enticing the uptake of machine-processable system documentation in the SWeMLS area.

**Keywords:** Neuro-symbolic System, Semantic Web, Machine Learning, Knowledge Graphs

**Resource type:** Knowledge Graph

**DOI:** https://doi.org/10.5281/zenodo.7445917
**URL:** https://w3id.org/semsys/sites/swemls-kg/

## 1 Introduction

The field of Artificial Intelligence (AI) is currently witnessing a great interest in (more closely) integrating and bridging between symbolic and sub-symbolic

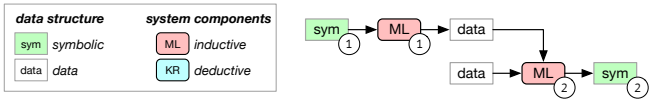


Fig. 1: Schematic representation of a SWeMLS workflow for art classification [13].

(AI) [7] techniques. This substantial trend led to the establishment of the new sub-research field of *neuro-symbolic systems*[9] [6,12], which focuses on the theoretical and practical aspects of creating such complex systems. Against this backdrop, it is not surprising that this AI trend is also reflected in the Semantic Web (SW) research community which has popularized AI-based knowledge representation techniques and resources in the last two decades [17]. There is increased interest in neuro-symbolic integration in the context of the Semantic Web [18], such as the development of systems that rely on both Semantic Web resources and Machine Learning components. We coined the term *Semantic Web and Machine Learning System* (SWeMLS) to refer to such systems [8].

For example, in [13] authors propose a system for automatic art analysis that can be classified as an SWeMLS. To that end, they augment a deep learning based system that classifies artistic images purely based on visual features with contextual art information in a form of a knowledge graph about painters, paintings, artistic schools, etc. Schematically, the system's workflow is depicted in Fig. 1 with the boxology notation introduced by [29]: starting with the `sym1` knowledge graph, graph embeddings are created through `ML1` which is a CNN deep learning model (`sym1-ML1-data`); subsequently, these embeddings together with visual data (i.e., images) are input to a CNN model (`ML2`) to create image classifications (`sym2`). Authors experimentally show that the inclusion of the SW component leads to performance increases by 7.3% in art classification and 37.24% in image retrieval tasks, thus demonstrating the potential of such systems.

Given the potential of SWeMLS and the increased interest in this field, the key *motivation* for our work was to gain a systematic understanding of the SWeMLS area by identifying trends among such systems and clustering them to better characterize the landscape of published systems. The main *challenges* in achieving a large-scale, data-driven, representative and systematic analysis of the SWeMLS field were:

- (i) a lack of understanding of important *system characteristics* that should be considered when analyzing SWeMLS. Approaches to characterise neuro-symbolic systems either focus on broader families of systems than SWeMLS [4], [21], or on a specific aspect of the systems (e.g., their internal processing flow [29,5]). Additionally, none are formalized for the purpose of using them as a basis for machine-actionable descriptions of the systems.
- (ii) the lack of a corpus of systematically collected (and therefore representative) papers annotated in terms of such characteristics, to allow for a data-

[9] The Neurosymbolic Artificial Intelligence journal will be launched in 2023: https://www.iospress.com/catalog/journals/neurosymbolic-artificial-intelligence

driven research trend analysis. While a number of papers about systems that learn and reason were collected as a basis for the analysis described in [29], these were not offered as a corpus of annotated papers to the community.

We addressed both challenges by conducting a large-scale Systematic Mapping Study (SMS [23]) on SWeMLS [8], through which we (i) proposed a set of characteristics for describing SWeMLS and (ii) systematically collected, selected and extracted data from nearly 500 papers describing such systems. This led to the following artifacts which together are offered as one *resource*:

- *the SWeMLS ontology* that describes the main aspects of SWeMLS including their internal workflow in terms of boxology patterns as shown in Fig. 1. The ontology schema (i.e., capturing important SWeMLS characteristics, e.g., $StatisticalModel$) and relevant instances (e.g., $DeepLearningModel$) were derived systematically during the scoping and analysis phases of the SMS,
- *the SWeMLS-KG*: a knowledge graph containing the machine-actionable description of almost 500 systems in terms of the SWeMLS ontology, and
- *the SWeMLS Pattern Library* containing the machine-actionable description of 45 SWeMLS patterns and their associated SHACL-based validation constraints. This pattern library extends the initial pattern catalog of 10-15 patterns originally identified by [29] both *quantitatively* with additional patterns observed during the SMS and *qualitatively*, by offering the patterns in a machine-actionable rather than graphical representation.

This resource is *timely* considering the recent trend in the SW community (and beyond) to create systems that leverage both SW and ML components. To the best of our knowledge, it is also *novel* by (i) providing the first ontology (and associated pattern library) for describing SWeMLS in a machine-actionable way and (ii) a methodologically collected corpus of SWeMLS and their semantic description. The resource is of immediate benefit for (SW) researchers that aim to explore trends in the SWeMLS field by analysing the data in the SWeMLS-KG and as such promises to have an impact on the understanding of the status-quo in this emerging field. Furthermore, the resource provides a semantic framework for describing SWeMLS and their internal details, thus potentially strongly influencing this field in terms of being well-documented, data-driven, and transparent.

We continue by discussing the impact of this resource (Sect. 2) and then detail its main components and the methodology used to produce them (Sect. 3), availability (Sect. 4) and usage in two use cases (Sect. 5). We summarize related work in Sect. 6 and conclude with an outlook on future work in Sect. 7.

## 2 Impact of the Resource

This resource is interesting to the Semantic Web community both in terms of its immediate and potential future impact on the field. An *immediate impact* is *enabling the understanding of general trends in the emerging area of SWeMLS*. The SWeMLS-KG allows for the first time to perform data-driven analysis in

order to better understand this family of systems. This can be achieved as part of two scenarios as described next and in more detail in Sections 5.1 and 5.2.

- *Asking concrete research questions*, e.g., *What kind of processing patterns are the most frequent? Which ML methods are used most often in combination with which SW resources?* Such targeted analysis was performed as part of the SMS [8] from which the SWeMLS ontology and KG were derived. While we investigated a limited number of questions that were feasible within the scope of the SMS, by making this resource available openly we enable the research community at large to perform additional analysis.
- *Identifying new insights* (e.g., through graph embedding) allows uncovering a new understanding of the field by exploring latent semantics encoded in the data.

Furthermore, the presented resources could have an important future impact by enabling the following use cases:

- *Search for SWeMLS-related work.* Researchers that create a SWeMLS could more easily find related systems, as part of related work search, during the design, evaluation, and publishing of their own systems. The current resource supports answering questions such as: *Which system patterns/pattern types are most frequent for graph completion tasks in the medical domain?*
- *Machine readable documentation and validation of SWeMLS.* Researchers that want to document a SWeMLS, can now (1) describe the system in a machine-processable way in terms of the SWeMLS ontology and (2) verify the correctness of their description through the SHACL validation. While the core technical artifacts are in place to enable other researchers to document their systems, future work will focus on more user-friendly annotation tools to entice large-scale adoption of research documentation for SWeMLS.
- *Improved scientific reviewing and publication processes.* AI-related conferences are struggling with high numbers of submissions which leads to challenges (i) for conference organisers to meaningfully assign papers to reviewers; as well as (ii) for reviewers who are overloaded with receiving very diverse papers and challenged to compare new systems to related work. We envision that SWeMLS-related events could use the SWeMLS ontology as a basis for annotating the submitted system papers. Such in-depth annotation of the systems could support (i) assigning relevant/similar papers to reviewers by clustering papers in terms of (the intersection of) several dimensions (task solved, domain addressed, system pattern used); (ii) allow reviewers to more easily comprehend the design of the system by referring to a structured or even visual notation of the system besides its textual description in the paper. Naturally, reviewers could leverage other collections of annotated systems (e.g., the SWeMLS KG), to identify papers similar to the one reviewed to make an informed assessment of novelty.

To conclude, the proposed resource could have a major impact on the way the research-documentation-publication cycles of SWeMLS happen, leading to a data-driven field and supporting faster growth and shorter innovation cycles.

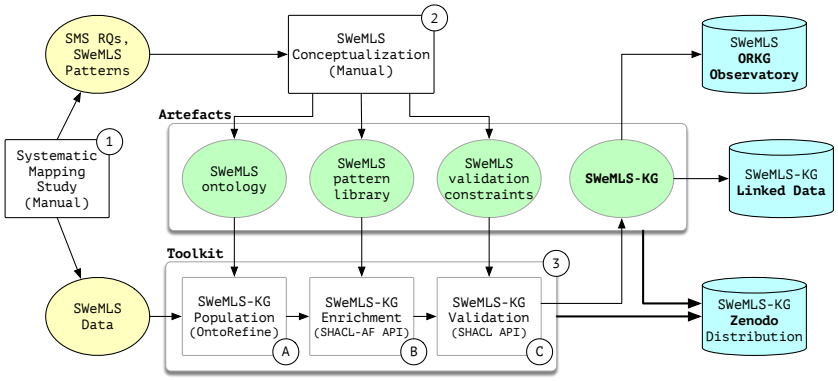


Fig. 2: Overview of the SWeMLS-KG construction process

## 3 Knowledge Graph Construction

We hereby describe the overall methodology for the construction of SWeMLS-KG (Sect. 3.1) and focus on key elements in this process such as the SWeMLS ontology (Sect. 3.2), the SWeMLS pattern library (Sect. 3.3) and the SWeMLS-KG and its population process (Sect. 3.4).

### 3.1 SWeMLS-KG Construction Process and Methodology

Fig. 2 depicts the process and methodology followed to construct the SWeMLS-KG. The starting point for the process was our prior large-scale SMS on the topic of SWeMLS [8] (cf. **Step 1** in Fig. 2) during which we collected information from 476 papers on SWeMLS in spreadsheet format. Starting from this SMS, we converted its results into a machine-processable format, through the next steps.

In **Step 2** we *conceptualised the SWeMLS related information* from two inputs provided in the SMS results: (i) the SMS research questions were a basis for the competency questions of the SWeMLS ontology, and (ii) the 45 SWeMLS patterns identified from the papers, which have been described and depicted as drawings but were not yet formalized in a machine-processable manner. From this step, we produce three types of outputs: (a) the SWeMLS ontology (Sect. 3.2), (b) the SWeMLS pattern library, which consists of pattern templates represented as RDF instances and SHACL-Advanced Features (SHACL-AF) rules, and (c) SWeMLS constraint definitions, which provide users with a mean to validate SWeMLS instances based on the existing patterns.

In **Step 3** we perform the *population of SWeMLS-KG* using the SWeMLS data and the artifacts produced in Step 2. This step (detailed in Sect. 3.4) consists of three sub-steps: (3A) populating the KG with spreadsheet data extracted from the SMS, (3B) constructing workflows between components and variables linked to each SWeMLS, and (3C) validating the KG using pattern-specific validations. The integrated and validated SWeMLS-KG is published through three distribution channels: (i) Linked Data interface, (ii) ORKG observatory, and (iii) a Zenodo repository as further explained in Sect. 4.

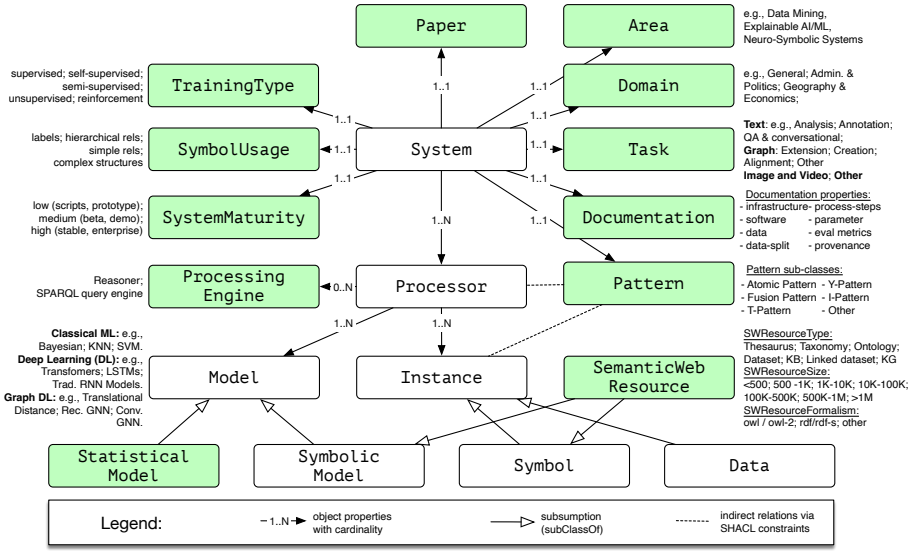


Fig. 3: SWeMLS Ontology Overview (adapted from [8])

### 3.2 The SWeMLS Ontology

**Ontology Creation** To create the ontology, we followed the Ontology Development 101 guideline [25]. We started by determining the domain and the scope of the ontology, using SMS research questions as competency questions:

- **Bibliographic characteristics** - How are the publications temporally and geographically distributed? How are the systems positioned, and which keywords are used to describe them?
- **System Architecture** - What processing patterns are used in terms of inputs/outputs and what is the order of processing units?
- **Application Areas** - What kind of tasks are solved (e.g., text analysis)? In which domains are SWeMLS applied (e.g., life sciences)?
- **Characteristics of the ML Module** - What ML models are incorporated (e.g., SVM)? Which ML components can be identified (e.g., attention)? What training type(s) is used during the system training phase?
- **Characteristics of the SW Module** - What type of Semantic Web structure is used (e.g., taxonomy)? What is the degree of semantic exploitation? What are the size and the formalism of the resources? Does the system integrate semantic processing modules (i.e., KR)?
- **Maturity, Transparency and Auditability** - What is the level of maturity of the systems? How transparent are the systems in terms of sharing source code, details of infrastructure and evaluation setup? Does the system have a provenance-capturing mechanism?

We considered reusing existing ontologies, especially to represent the patterns' workflows such as Wings Workflow [15], Taverna [19], and the Common

Workflow Language [1]. However, our patterns are very specific and none of them could be used. Thus, we decided to develop our own SWeMLS ontology by adapting and extending the P-PLAN ontology [14] to describe SWeMLS workflows and the OPMW ontology [24] to describe system patterns.

As the next step, we enumerated important terms from the SMS that should be represented in the ontology such as *System*, *Paper*, *Processor*, *Model*, and *Instance*. Then we defined the classes and the class hierarchy based on these terms, using a top-down approach, e.g., class *Model* has *Semantic Model* and *Statistical Model* as sub-classes. After establishing the classes and their hierarchy, we defined the class properties based on the data gathered from the SMS, e.g., system application area, task, and system maturity. Finally, we created individuals from the SMS data, i.e., *Data Mining* is an instance of *Area*.

**Ontology Description** The resulting SWeMLS ontology is intended to represent the systems described in the publications reported in [8]. A high-level overview of its main classes, properties, and an excerpt of named individuals is shown in Fig. 3. Overall, the SWeMLS ontology includes: (i) paper details, (ii) system properties reported in such papers, and (iii) workflow-style representations of patterns:

- **Paper** details such as title, year of *publication*, *publication type*, *venue*, *authors' countries*, *keywords*, a short *summary*, and the *link* to the paper.
- **SWeMLS** properties such as the *targeted tasks*, *level of maturity*, *application domain*, *semantic web resources* being used, *machine learning model*, type of *semantic processor*, *the pattern* being used, as well as *documentation* properties which include: e.g., *infrastructure*, *provenance*, and *evaluation*.
- **SWeMLS patterns** representing the structure of each system workflow pattern with each pattern's component including their inputs/outputs. We detail the representation of the SWeMLS patterns in Sect. 3.3.

An example of how the terms defined by the ontology are used to describe a paper reporting a SWeMLS is presented in Sect. 3.1.

### 3.3 SWeMLS Pattern Library

*Pattern representation.* We use the P-PLAN [14] and OPMW [24] as the basis for SWeMLS pattern representation. More specifically, we follow the separation of three major types of workflow structures outlined by Garijo et al. [14]:

- (i) Workflow Template (`opmw:WorkflowTemplate`), a generic pattern that indicates the type of steps in the workflow and their dataflow dependencies,
- (ii) Workflow Instance (`swemls:System` as a sub-class of `p-plan:Plan`), a workflow that specifies the application algorithms to be executed and data to be used, and
- (iii) Workflow Execution (`p-plan:Bundle`), a workflow execution trace containing details of what happened during an execution.

We focus on the first two types (i.e., Workflow Template and Instance) and plan for Workflow Execution as part of our future work.

*The SWeMLS Pattern Library* consists of the representation of 45 system processing patterns identified during the SMS (each pattern is captured in a .ttl file in the zenodo distribution of the resource). An example representation of the Workflow Template for pattern T-3 is shown in Listing 3.1. The template contains the definition of the patterns (e.g., `res:Pattern.T3`), its components/steps (i.e., `res:Pattern.T3.ML1` and `res:Pattern.T3.ML2`) and how they use or generate variables (e.g., `res:Pattern.T3.ML1` use `res:Pattern.T3.SW1` and generate `res:Pattern.T3.Data2`).

```
@prefix swemls: <https://w3id.org/semsys/ns/swemls#> .
@prefix res: <http://semantic-systems.net/swemls/> .
@prefix p-plan: <http://purl.org/net/p-plan#> .
@prefix opmw: <http://www.opmw.org/ontology/> .
/* ... rdf, rdfs are ommitted */

/* Pattern T3 as an instance of opmw:WorkflowTemplate */
res:Pattern.T3 a opmw:WorkflowTemplate ; rdfs:label "T3";
  rdfs:comment "[{sym -> ML -> data / data} -> ML -> sym]" .

/* Component T3.ML1 with T3.SW1 (SW resource) as input */
res:Pattern.T3.ML1 a swemls:WorkflowTemplateProcessML ;
  opmw:isStepOfTemplate res:Pattern.T3 ; opmw:uses res:Pattern.T3.SW1 .
/* Component T3.ML2 with T3.Data1 and T3.Data2 (data) as inputs */
res:Pattern.T3.ML2 a swemls:WorkflowTemplateProcessML ;
  p-plan:isPreceededBy res:Pattern.T3.ML1 ;
  opmw:isStepOfTemplate res:Pattern.T3; opmw:uses res:Pattern.T3.Data1,
       res:Pattern.T3.Data2 .

/* Variable T3.SW1 */
res:Pattern.T3.SW1 a swemls:TemplateArtifactSW ; opmw:isVariableOfTemplate
    res:Pattern.T3 .
/* Variable T3.Data1; used as input for component T3.ML2 */
res:Pattern.T3.Data1 a swemls:TemplateArtifactData;
    opmw:isVariableOfTemplate res:Pattern.T3 .
/* Variable T3.Data2; generated by T3.ML1; input for T3.ML2 */
res:Pattern.T3.Data2 a swemls:TemplateArtifactData ;
  opmw:isVariableOfTemplate res:Pattern.T3 ; opmw:isGeneratedBy
       res:Pattern.T3.ML1 .

/* Variable T3.SW2; generated by T3.ML2 as the final result of the System */
res:Pattern.T3.SW2 a swemls:TemplateArtifactSW ;
  opmw:isVariableOfTemplate res:Pattern.T3 ; opmw:isGeneratedBy
       res:Pattern.T3.ML2 .
```

Listing 3.1: T-3 pattern in turtle format

### 3.4 SWeML-KG Population and Update Mechanisms

After defining the underlying ontology, we populated the SWeMLS KG with the details of SWeMLS collected in the course of the SMS.

*An example SWeMLS semantic description* is depicted in Fig. 4 which shows the SWeMLS-KG instance of the SWeMLS discussed in Sect. 1 [13]. For this system, we display the paper in which it is reported, together with other paper details, such as title, keywords, and year of publication. In addition, the target task to be solved falls into the category of '*Image and Video*', the system maturity is reported as '*Low*', the application domain is listed as '*Human Culture and Education*', the training type as '*Supervised*' and the symbol usage as '*Complex Structure*'. The depicted system also contains documentation information, including transparency and auditability components of the system.

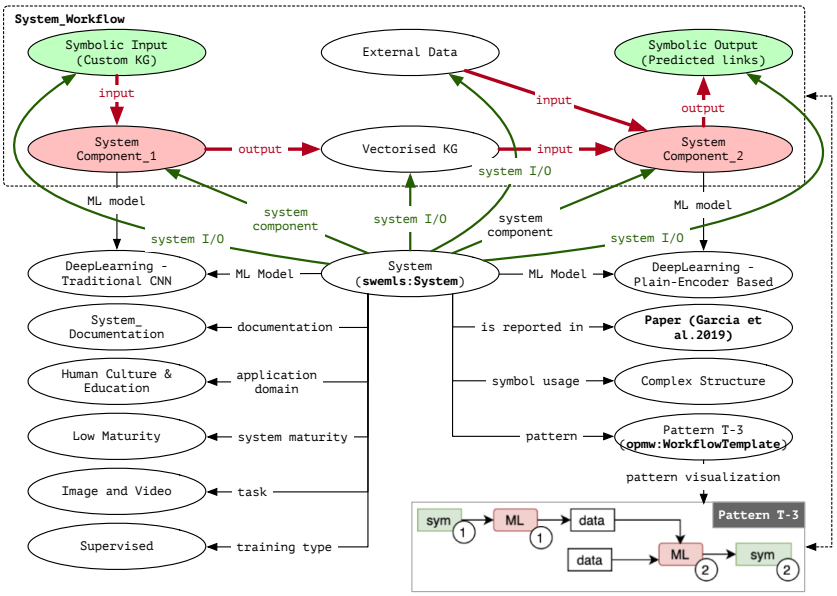


Fig. 4: Example of the semantic representation of paper [13] and the art classification SWeMLS described by it (adapted from [8]). *Green arrows* represent relations between a system and its components and variables, *Red arrows* represent workflow information generated with SHACL-AF rules.

The chosen system instantiates pattern "T-3" (depicted visually in the bottom right of Fig. 4) which involves two symbolic data components, two machine learning components, and two data components in its workflow. The upper part of Fig. 4 shows the semantic representation of the system workflow: starting from custom KG as symbolic input to a traditional CNN machine-learning component, the model produces a vectorized KG, which, along with some external data, serves as input for a deep learning plain encoder-based model. Finally, the system produces predicted links as symbolic output.

*The SWeMLS population process* consisted of the following steps. We created a mapping to the SWeMLS Ontology[10] and transformed the SMS data into RDF format (**Step 3A** of Fig. 2). The generated RDF graphs from Step 3A already connect each system with its respective system components and I/O variables (cf. green arrows on Fig. 4). However, the connection between I/O variables and components is not yet available (cf. red arrows on Fig. 4). To build these connections, we ran an enrichment process (**Step 3B**) using SHACL-AF rules[11].

Lastly, we validated the resulting data against SHACL constraints (**Step 3C**). We defined a set of constraints for general SWeMLS as well as instances of specific SWEMLS patterns to ensure completeness, validity and conformance of the KG to the pattern definitions.

[10] We use Ontotext Refine https://www.ontotext.com/products/ontotext-refine/

[11] Example SHACL-AF rules and SHACL validation constraints can be accessed in our GitHub repo, e.g., https://bit.ly/sweml-t3-pattern for pattern T-3.

*SWeML-KG update mechanisms.* To promote community-based contributions, we published all the source code necessary for the KG creation (i.e., Ontotext refine projects and mapping files, SHACL-AF rules, and SHACL constraints). Furthermore, we plan to make updating the SWeMLS-KG with new system descriptions easier, for example, by relying on features provided by the Open Research Knowledge Graph to enable community-wide contributions.

## 4 Availability of the SWeMLS-KG

The SWeMLS-KG landing page[12] provides pointers to the various resources covered in this paper, i.e., the Linked Data resources[13], the SPARQL query interface[14], a Zenodo link for the complete RDF snapshots[15], the source code for SWeMLS toolkit[16]. This allows users to choose the most appropriate resources and access mechanisms most suitable for their context.

*Publication as ORKG Observatory.* SWeMLS-KG was also made available as part of the Open Research Knowledge Graph [3,20][17]. ORKG is a scholarly knowledge organization facility, where contributions conveyed in scientific articles are represented semantically in machine- and human-readable ways. The FAIR semantic description of research contributions facilitates a number of applications, such as overviews of the state-of-the-art for certain research questions (comparisons), visualizations, or leaderboards. The ORKG organizes the semantic contribution descriptions along research fields but also in thematic *observatories*, where a team of curators from one or several organizations curates the contributions related to a specific topic.

Together with the ORKG development team, we imported the SWeMLS-KG, so that its data is browsable, accessible, citable and reusable. The ORKG observatory for SWeMLS-KG allows browsing patterns, instantiations of the patterns as well as searching and filtering contributions from articles by certain characteristics. An initial version of the SWeMLS-specific ORKG observatory is available online on the ORKG server[18], providing an overview of the collected papers as well as detailed metadata for each paper[19].

## 5 Use Cases

We hereby report on use cases that explore SWeMLS-KG via (i) SPARQL queries (Sect. 5.1, and (ii) Knowledge Graph Embedding methods (Sect. 5.2).

[12] https://w3id.org/semsys/sites/swemls-kg/
[13] e.g., Garcia et al. [13] https://semantic-systems.net/swemls/System_4QP5XAGX
[14] https://semantic-systems.net/sparql/
[15] https://doi.org/10.5281/zenodo.7445917
[16] https://github.com/semanticsystems/swemls-toolkit
[17] https://orkg.org
[18] https://orkg.org/observatory/Neurosymbolic_artificial_intelligence
[19] e.g., Garcia et al. [13] in ORKG: https://orkg.org/paper/R574440

### 5.1 Use Case 1: Understanding SWeMLS trends through Querying the SWeMLS-KG

The SWeMLS-KG can support researchers and reviewers in exploring and understanding trends in the SWeMLS field through queries executed on the SPARQL enpoint of the KG. We first motivate exemplary knowledge questions in natural language, show their SPARQL representations, and discuss their results.

```
PREFIX swemls: <https://w3id.org/semsys/ns/swemls#>
PREFIX res: <http://semantic-systems.net/swemls/>
/* ... rdf, rdfs, and dc-terms are ommitted*/
select ?swModel ?statisticalModel ?trainingType ?title ?year
where {
  ?system a swemls:System ;
    swemls:hasApplicationDomain res:Domain.Medicine_Health ;
    swemls:hasTask res:Task.Patient_Diagnosis_Prediction ;
    swemls:hasTrainingType / rdfs:label ?trainingType .
  ?system swemls:hasSymbolIO / rdfs:label ?swModel .
  {
    select ?system (group_concat(?statisticalModelName;separator=",") as
        ?statisticalModel)
    where { ?system swemls:hasStatisticalModel / rdfs:label
        ?statisticalModelName}
    group by ?system
  }
  ?paper swemls:reports ?system ; terms:title ?title ; swemls:year ?year .
}
```

Listing 5.1: Query for components in the medical domain for diagnosis prediction

*Task/domain driven queries for components of SWeMLS.* We want to support researchers and reviewers in identifying and exploring existing SWeMLS and their components: *What SWeMLS components (SW resources and ML models) have been used to solve a specific task x in the domain y?* A researcher might ask this question e.g., in the course of designing a new system as part of state-of-the-art research or when looking for additional datasets that have been used in a target domain. A reviewer on the other hand, could quickly identify publications with similar components and use these to highlight the innovation and advantages of the submission under review. A SPARQL representation of this question in the domain `Medicine_Health` and for the task `Patient_Diagnosis_Prediction` is given in Listing 5.1. Table 1 shows an excerpt of the query results, which could be further explored.

| swModel | statisticalModel | trainingType | title | year |
|---|---|---|---|---|
| CCS | Attention,GloVe,MLP,RNN | Self-supervised | GRAM: Graph-Based Attention Model for Healthcare Representation Learning | 2017 |
| UMLS | ARM | Self-supervised | Guiding supervised learning by bio-ontologies in medical data analysis | 2018 |
| ICD | Graph-based Attention Model,Knowledge Attention,Gated Recurrent Unit (GRU) | Supervised | KAME: Knowledge-based attention model for diagnosis prediction in healthcare | 2018 |
| DBpedia | SVM | Supervised | Improving rare disease classification using imperfect knowledge graph | 2019 |

Table 1: Query results for components in the medical domain for diagnosis prediction (excerpt)

*Pattern-driven queries for SWeMLS* are queries that explore the system workflow patterns' structure and its components. This allows researchers and reviewers to identify *structurally* identical or similar SWeMLS and their relevant aspects, i.e., the integration of ML models and SW resources: *What SWeMLS exist that use a specific SW resource x as input for a ML Model y that produces symbolic output?*

```
PREFIX swemls: <https://w3id.org/semsys/ns/swemls#>
PREFIX res: <http://semantic-systems.net/swemls/>
/* ... rdf, rdfs, skos, and dc-terms are ommitted */
select ?domain ?task ?pattern ?sw ?groupSw ?title ?year
where {
  ?system a swemls:System ; swemls:hasApplicationDomain ?domain ;
    swemls:hasTask ?task ; swemls:hasCorrespondingPattern ?pattern ;
        swemls:hasStepML ?ml .
  ?paper swemls:reports ?system ; terms:title ?title ; swemls:year ?year .
  ?sw a swemls:SemanticWebResource .
  { ?sw skos:broader res:Resource.Facebook }
  UNION {
    select ?sw (group_concat(?compoundSwInput;separator=",") as ?groupSw)
    where {
      ?sw swemls:hasCompoundElement ?compoundSwInput .
      ?compoundSwInput skos:broader res:Resource.Facebook
    } group by ?sw
  }
  ?ml swemls:componentInput ?sw .
  ?ml swemls:componentOutput/rdf:type swemls:SemanticWebResource .
  { ?ml swemls:componentModel res:StatisticalModel.TransX }
  UNION {
    ?ml swemls:componentModel ?compoundML .
    ?compoundML swemls:hasCompoundElement res:StatisticalModel.TransX
  }
}
```

Listing 5.2: SPARQL query for systems using a translation model on Facebook benchmark data, producing symbolic output as part of their architecture

A SPARQL representation for this question is given in Listing 5.2. We search for systems that use a translation model (TransX) as ML module which operates on Facebook benchmark semantic web resources (e.g., FB15k, FB13, FB500k). Furthermore, the module has to generate symbolic output. The ML module can be placed anywhere in the system architecture, hence, we do not look for specific patterns or architectures. Table 2 shows an excerpt of the found systems,

| domain | task | pattern | sw | groupSw | title | year |
|---|---|---|---|---|---|---|
| General | KG_Completion | F4 | ...SW_cc6bef6e | FB122 | Jointly embedding knowledge graphs ... | 2016 |
| General | KG_Completion | F2 | ...SW_d5ee1a61 | FB_500K | Probabilistic Belief Embedding ... | 2016 |
| General | KG_Completion | A1 | ...SW_f27afb1c | FB13,FB15k | Learning Knowledge Embeddings by ... | 2017 |
| General | KG_Completion | A1 | FB15k | | Knowledge Graph Embedding via ... | 2018 |
| General | Question_Answering | F3 | ...SW_1fb71cdc | FB15k | Representation Learning of ... | 2019 |

Table 2: Query results for systems processing Facebook resources with a translation model, producing symbolic output (excerpt)

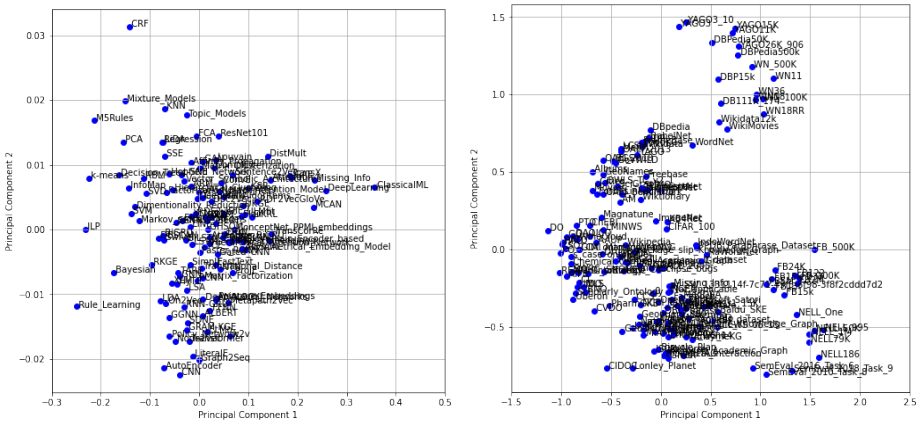


Fig. 5: Scatterplots of statistical model methods (left) and resources (right). For the plots, the embedding spaces have been reduced to 2D using principal component analysis.

including different patterns and specific SW resources. As an example, the system presented in the paper "Learning Knowledge Embeddings by Combining Limit-Based Scoring Loss", uses pattern A1 and two Facebook benchmark datasets, namely FB13 and FB15k, for KG completion.

### 5.2 Use Case 2: Embedding-based Exploration of the SWeMLS KG

In order to allow further exploration of the SWeMLS-KG, we have computed RDF2vec embeddings [27] on the graph. With the help of those embeddings, visualizations can be generated, and additional queries, based on entity similarity in the embedding space, can be performed. Unlike the previous use case where the exploration of the data is guided by explicitly stated information needs, this use case explores the latent semantics encoded in the SWeMLS-KG.

Fig. 5 shows scatter plots of embeddings for the statistical modeling methods and semantic web resources in the knowledge graph. Especially for the resources, one can observe that the grouping is actually sensible, forming, e.g., a cluster of DBpedia and YAGO-related resources in the top area.

Typical scenarios for searching in the embedding space would be triggered using one entity and then searching for further entities in the neighborhood. One typical example use case would be searching for alternative resources and methods. For example, a neighborhood query for the FB15k link prediction benchmark provides a list of other link prediction benchmark datasets, whereas a neighborhood query for DBpedia provides a list of other general-purpose knowledge graphs, such as YAGO or Wikidata. Likewise, neighborhood queries for methods can be conducted. For example, a neighborhood search for graph neural networks gives rule learning and On2vec as nearest neighbors[20].

[20] Source code: https://w3id.org/semsys/sites/swemls-kg/rdf2vec

Neighborhood queries can also be useful for finding related papers. We probed the embedding space with a randomly selected paper, describing a knowledge-graph-based recommender system using embeddings, attention networks, and freebase as a resource. The neighborhood contains mostly other papers describing knowledge-based recommenders and papers using the same pattern and/or resources for other purposes, such as question answering.

## 6 Related Work

*Ontologies for describing neuro-symbolic/ML systems.* For the proposed SWeMLS ontology, related work is represented by earlier efforts to characterize neuro-symbolic systems. For example, Bader and Hitzler [4] made an early attempt at such characterization and proposed eight dimensions for classification purposes. More recently, Van Harmelen and ten Teije [29] introduced a set of 13 design patterns, similar to design patterns in software engineering. This taxonomy has been extended with processes and models in [5]. Another taxonomy comprising six different types of systems but without focus on the internal architectures of the investigated systems has been presented by Kautz [21]. While all these efforts focus on the broader family of neuro-symbolic systems, in our recent work [8] we proposed a classification system tailored for SWeMLS. With the ontology presented here we provide the first machine-actionable (i.e., formally represented) system classification.

To ensure that ML research outcomes are properly comparable, understandable, reusable, and reproducible several ontologies have been proposed. OntoDM [26] for instance provides generic representations of entities in data mining, DMOP [22] supports meta-learning from ML processes, Exposé [30] can be used to describe and reason about ML experiments, and the MEX Vocabulary [11] aims to support managing ML outcomes and sharing of provenance data. In order to offer a flexible approach for mapping existing ML ontologies and to support extensions, a W3C Community Group[21] developed ML-Schema (MLS)[22]. Compared to our approach, existing ontologies focus on ML experiment *executions* and not on system descriptions. They also do not focus on representing SW elements of the systems. However, main ML concepts in our ontology can be mapped to ML-Schema, such as `mls:Experiment`, which is comparable to `swemls:System`, and `mls:Data` is similar to `swemls:Instance`. With a focus on reproducibility, ML-Schema and other approaches also cover detailed ML settings, such as hyperparameters and evaluation results. While this is not in the focus of our current work, we plan to extend our knowledge graph in this direction.

*Machine-processable publication of domain-specific scientific knowledge* is reported in several disciplines. For example, in social sciences, in the domain of

[21] https://www.w3.org/community/ml-schema/
[22] http://ml-schema.github.io/documentation/MLSchema.html

human cooperation, experts annotated nearly 3,000 studies in terms of 60 features as part of the COoperation DAtabank initiative[23]. This systematically collected scientific knowledge has been published using semantic technologies as a knowledge graph [28] and as nanopublications [2] in order to support the automation of scientific tasks such as (comparative) meta-analysis and the detection of contradictory claims respectively.

Going beyond domain-specific efforts for publishing scientific knowledge, the Open Research Knowledge Graph (ORKG) aims to provide a platform for the publication of open research knowledge. ORKG describes scientific articles semantically in a machine- and human-readable way. It offers concepts and properties to classify articles, and extract various metadata such as authors and publication date, but also to describe research contributions and results. To contribute to this initiative, we mapped our ontology to the ORKG schema and published our SWeMLS results within the ORKG (see Sect. 4), thus being one of the first communities to leverage the capabilities of this system and to benefit by the sustainability of the data publication on a long term.

## 7 Conclusion and Future Work

In this work, we used a semantic technology approach to provide a machine-processable way to represent a large number of SWeMLS reported in scientific publications. We introduced the SWeMLS ontology and SWeMLS knowledge graph to support researchers and reviewers using a more automated approach to search, conduct analysis and test existing SWeMLS. The SWeMLS-KG was also imported into ORKG, making the data browseable, accessible, citable, and reusable. The use cases we discussed have shown that the SWeMLS-KG is useful for researchers and reviewers on a variety of levels, including identifying and analyzing existing SWeMLS, drawing conclusions about the components being used, or identifying similar components using SPARQL queries and embedding-based exploration of the SWeMLS-KG.

Regarding future work, we plan to include audit support for SWeMLS by capturing Workflow Execution traces to complement the workflow templates (i.e., SWeMLS patterns) and workflow instances (i.e., SWeMLS instance), building on our prior work on auditability [10]. Furthermore, we strive to enable semi-automatized description extraction from SWeMLS papers and generation of SWeMLS pipeline code from patterns by building on existing works [9,16]. Finally, we want to support a two-way transformation of data from the SWeMLS-KG and the ORKG-observatory. Beyond the scope of our research, we hope to inspire broader research communities to provide their research results in a structured representation, which in turn will allow others to build their research *by standing on the shoulder of giants.*

[23] COoperation DAtabank: https://amsterdamcooperationlab.com/databank/

## Acknowledgments

This work has been supported by the Austrian Science Fund (FWF) under grant V0745 (HOnEst) and FFG Project OBARIS (Grant Agreement No 877389). SBA Research (SBA-K1) is a COMET Center within the COMET – Competence Centers for Excellent Technologies Programme and funded by BMK, BMAW, and the federal state of Vienna. The COMET Programme is managed by FFG. Moreover, financial support by the Christian Doppler Research Association, the Austrian Federal Ministry for Digital and Economic Affairs, the National Foundation for Research, Technology and Development, DFG NFDI4DataScience (No. 460234259) and ERC ScienceGRAPH (GA ID: 819536) is gratefully acknowledged.

## References

1. Amstutz, P., Crusoe, M.R., Tijanić, N., Chapman, B., Chilton, J., Heuer, M., Kartashov, A., Leehr, D., Ménager, H., Nedeljkovich, M., et al.: Common workflow language, v1. 0 (2016). https://doi.org/10.6084/m9.figshare.3115156
2. Asif, I., Tiddi, I., Gray, A.J.G.: Using nanopublications to detect and explain contradictory research claims. In: 2021 IEEE 17th International Conference on eScience (eScience). pp. 1–10 (2021). https://doi.org/10.1109/eScience51609.2021.00010
3. Auer, S., Oelen, A., Haris, M., Stocker, M., D'Souza, J., Farfar, K.E., Vogt, L., Prinz, M., Wiens, V., Jaradeh, M.Y.: Improving access to scientific literature with knowledge graphs. Bibliothek Forschung und Praxis **44**(3), 516–529 (2020). https://doi.org/doi:10.1515/bfp-2020-2042
4. Bader, S., Hitzler, P.: Dimensions of neural-symbolic integration - A structured survey. CoRR **abs/cs/0511042** (2005). https://doi.org/10.48550/arXiv.cs/0511042
5. van Bekkum, M., de Boer, M., van Harmelen, F., Meyer-Vitali, A., ten Teije, A.: Modular Design Patterns for Hybrid Learning and Reasoning Systems. Applied Intelligence **51**(9), 6528–6546 (2021). https://doi.org/10.1007/s10489-021-02394-3
6. Besold, T.R., d'Avila Garcez, A., Bader, S., Bowman, H., Domingos, P., Hitzler, P., Kühnberger, K.U., Lamb, L.C., Lima, P.M.V., de Penning, L., et al.: Neural-Symbolic Learning and Reasoning: A Survey and Interpretation. In: Neuro-Symbolic Artificial Intelligence: The State of the Art, pp. 1–51. IOS Press (2021). https://doi.org/10.48550/arXiv.1711.03902
7. Booch, G., Fabiano, F., Horesh, L., Kate, K., Lenchner, J., Linck, N., Loreggia, A., Murugesan, K., Mattei, N., Rossi, F., Srivastava, B.: Thinking fast and slow in ai. In: AAAI. AAAI Press (2021). https://doi.org/10.48550/arXiv.2010.06002
8. Breit, A., Waltersdorfer, L., Ekaputra, F.J., Sabou, M., Ekelhart, A., Iana, A., Paulheim, H., Portisch, J., Revenko, A., Teije, A.t., van Harmelen, F.: Combining machine learning and semantic web: A systematic mapping study. ACM Comput. Surv. (Mar 2023), https://doi.org/10.1145/3586163, Just Accepted.
9. Daga, E., Groth, P.: Data journeys: explaining AI workflows through abstraction. Semantic Web pp. Early–Access (2023), http://oro.open.ac.uk/88012/
10. Ekaputra, F.J., Ekelhart, A., Mayer, R., Miksa, T., Šarčević, T., Tsepelakis, S., Waltersdorfer, L.: Semantic-enabled architecture for auditable privacy-

preserving data analysis. Semantic Web **pre-press**(Preprint), 1–34 (2021). https://doi.org/10.3233/SW-212883
11. Esteves, D., Moussallem, D., Neto, C.B., Soru, T., Usbeck, R., Ackermann, M., Lehmann, J.: Mex vocabulary: a lightweight interchange format for machine learning experiments. In: Proceedings of the 11th International Conference on Semantic Systems. pp. 169–176 (2015). https://doi.org/10.1145/2814864.2814883
12. Garcez, A., Broda, K., Gabbay, D., et al.: Neural-symbolic learning systems: foundations and applications. Springer (2002). https://doi.org/10.1007/978-1-4471-0211-3
13. Garcia, N., Renoust, B., Nakashima, Y.: Context-aware embeddings for automatic art analysis. In: Proceedings of the 2019 on International Conference on Multimedia Retrieval. pp. 25–33 (2019). https://doi.org/10.1145/3323873.3325028
14. Garijo, D., Gil, Y., Corcho, Ó.: Towards workflow ecosystems through semantic and standard representations. In: Montagnat, J., Taylor, I.J. (eds.) Proceedings of the 9th Workshop on Workflows in Support of Large-Scale Science, WORKS '14, New Orleans, Louisiana, USA, November 16-21, 2014. pp. 94–104. IEEE (2014). https://doi.org/10.1109/WORKS.2014.13
15. Gil, Y., Ratnakar, V., Kim, J., González-Calero, P.A., Groth, P., Moody, J., Deelman, E.: Wings: Intelligent workflow-based design of computational experiments. IEEE Intell. Syst. **26**(1), 62–72 (2011). https://doi.org/10.1109/MIS.2010.9
16. Grafberger, S., Groth, P., Stoyanovich, J., Schelter, S.: Data distribution debugging in machine learning pipelines. The VLDB Journal **31**(5), 1103–1126 (2022). https://doi.org/10.1007/s00778-021-00726-w
17. Hitzler, P.: A review of the semantic web field. Communications of the ACM **64**(2) (2021). https://doi.org/10.1145/3397512
18. Hitzler, P., Bianchi, F., Ebrahimi, M., Sarker, M.K.: Neural-Symbolic Integration and the Semantic Web. Semant. Web **11**(1), 3–11 (jan 2020). https://doi.org/10.3233/SW-190368
19. Hull, D., Wolstencroft, K., Stevens, R., Goble, C.A., Pocock, M.R., Li, P., Oinn, T.: Taverna: a tool for building and running workflows of services. Nucleic Acids Res. **34**(Web-Server-Issue), 729–732 (2006). https://doi.org/10.1093/nar/gkl320
20. Jaradeh, M.Y., Oelen, A., Farfar, K.E., Prinz, M., D'Souza, J., Kismihók, G., Stocker, M., Auer, S.: Open research knowledge graph: Next generation infrastructure for semantic scholarly knowledge. In: Proceedings of the 10th International Conference on Knowledge Capture. p. 243–246. K-CAP '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3360901.3364435
21. Kautz, H.: The Third AI Summer, AAAI Robert S. Engelmore Memorial Lecture, 34th AAAI, 2020. https://doi.org/10.1002/aaai.12036
22. Keet, C.M., Ławrynowicz, A., d'Amato, C., Kalousis, A., Nguyen, P., Palma, R., Stevens, R., Hilario, M.: The data mining optimization ontology. Journal of Web Semantics **32**, 43–53 (2015). https://doi.org/https://doi.org/10.1016/j.websem.2015.01.001, https://www.sciencedirect.com/science/article/pii/S1570826815000025
23. Kitchenham, B., Charters, S., et al.: Guidelines for performing systematic literature reviews in software engineering. Tech. rep., Keele University and Durham University Joint Report (2007), https://www.researchgate.net/publication/302924724
24. Moreau, L., Clifford, B., Freire, J., Futrelle, J., Gil, Y., Groth, P., Kwasnikowska, N., Miles, S., Missier, P., Myers, J., Plale, B., Simmhan, Y., Stephan, E.G., den Bussche, J.V.: The open provenance model core

specification (v1.1). Future Gener. Comput. Syst. **27**(6), 743–756 (2011). https://doi.org/10.1016/j.future.2010.07.005
25. Noy, N.F., McGuinness, D.L., et al.: Ontology development 101: A guide to creating your first ontology (2001), https://www.researchgate.net/publication/243772462
26. Panov, P., Džeroski, S., Soldatova, L.: Ontodm: An ontology of data mining. In: 2008 IEEE International Conference on Data Mining Workshops. pp. 752–760 (2008). https://doi.org/10.1109/ICDMW.2008.62
27. Ristoski, P., Paulheim, H.: Rdf2vec: Rdf graph embeddings for data mining. In: International Semantic Web Conference. pp. 498–514. Springer (2016). https://doi.org/10.1007/978-3-319-46523-4_30
28. Tiddi, I., Balliet, D., ten Teije, A.: Fostering scientific meta-analyses with knowledge graphs: A case-study. In: The Semantic Web. ESWC 2020. pp. 287–303. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer (2020). https://doi.org/10.1007/978-3-030-49461-2_17
29. van Harmelen, F., ten Teije, A.: A boxology of design patterns for hybrid learning and reasoning systems. J. of Web Engineering **18**(1-3), 97–124 (2019). https://doi.org/10.13052/jwe1540-9589.18133
30. Vanschoren, J., Soldatova, L.: Exposé: An ontology for data mining experiments. In: International workshop on third generation data mining: Towards service-oriented knowledge discovery (SoKD-2010). pp. 31–46 (2010), https://www.researchgate.net/publication/228525536