COMA 3.0

Community Edition (CE)

User's Guid

Introduction

COMA is a schema and ontology matcher that receives two database schemas or ontologies as input and creates a mapping between the two resources. The program uses different measures to calculate the similarity between two schema elements or ontology concepts (on a scale between 0 and 1). These similarity measures are organized in so-called workflows. To create a mapping, several of such workflows can be used. Mappings are visualized in COMA and can be optionally exported. COMA uses a program-specific export format which is similar to CSV.

Using COMA usually comprises the following four steps:

- Load two resource files (e.g., two ontologies)
- Add the two resources to the mapping area
- Select a workflow and carry out the mapping (match phase)
- Possibly, edit the mapping (add/remove correspondences)
- Export the mapping to DB or to file

Resources imported to COMA are automatically stored in the database. Mappings (frequently referred to as *match results*) are created in the workspace. They can be saved to database by the user; otherwise they will not be persistently stored. The database containing the user's data (schemas, instances and mappings) is called *Repository*.

COMA is a research prototype provided for research issues and teaching. It contains already known bugs and is not designed for professional usage. If you have problems with COMA, you may carefully read this user's guid or squarely check the Troubleshooting part, yet any further technical support cannot be provided.

Installation and Setup

Once COMA 3.0 CE has been downloaded and the ZIP file has been extracted, it can be launched by executing coma.bat in the COMA program folder. Please make sure that...

(1) Java is installed on your machine
(2) MySQL is installed and running
(3) COMA has access rights to create a database in MySQL

Before you use COMA, you have to create a new database schema "coma-project" in MySQL. This will be the schema where STROMA saves all schemas, instances and mappings.

IMPORTANT NOTE: When COMA is launched for the first time, please click *Match* > *Reset Workflow Variables* in order to initialize the workflows.

More details for the COMA setup can be also found on our website.

GUI Overview

The COMA GUI consists of three areas: The *Repository tab*, the *Workspace tab* and the *Mapping panel*. The *Repository tab* allows access to the repository where imported resources and mappings are stored. The *Workspace tab* allows access to a currently opened or created mapping. Workspace data is deleted as soon as COMA is shut down. For this reason, generated mappings must be saved to the repository if they are to be persistently available.

The *mapping panel* is the main part of the GUI. In this area, two schemas or ontologies are added by the user and a mapping will be created and displayed in this panel. The links between the elements or concepts are the correspondences, their color indicating the confidence (similarity value).

Menu structure:

- **Repository:** *Load/delete schemas, instances and mappings to/from the repository (database) or delete the entire repository.*
- **Match:** *Execute and configure the workflows for schema matching.*
- **Matchresult:** *Mapping-related operations (e.g. save mapping to repository).*
- **View:** *Operations related to the view (e.g. show textual representation of the match result).*

Loading Resources

Loading Schemas/Ontologies

As a first step, resources that are to be matched must be loaded (imported). Please click *Repository* > *Schemas* > *Import file*. Basically, you can import XML schemas (.xsd), Ontologies, RDF Files, SQL files or CSV files.

If you choose CSV, your input schema is a flat list of concepts. Your CSV file should consist of just one line, with each concept resp. schema element being separated by a comma.

NOTE: While loading CSV and XML files usually works smoothly and without any difficulties, loading ontologies or SQL files can encounter errors. Since there are many different standards for ontologies and SQL, some schemas that do not use the default format may cause an error and COMA may be unable to parse and load them. In this case, the error message "Schema was aborted" may appear, or the resource file may not be loaded in the first place.

CAUTION: The program stores the schemas in MySQL tables and identifies them by a name object. For some formats (e.g. owl) the whole file path is used as identifier. COMA is restricted to maximum path lengths of 100. In case that your schema or ontology mapping cannot be loaded, the overall file path of your file may be too long and you may copy it into another (more superior) directory.

Loading Instance Data

After a resource has been loaded, you can optionally load instance data. Please click *Repository* > *Instances*. If your instances are included in your schema file, as it is with OWL files, click *Parse Schema File*. COMA shows you the available (i.e. the already imported) resource files. Click a file of your choice and COMA will, if possible, extract the instance data from this file and automatically attach it to the already imported resource.

If your instance data resides in a separate file, i.e., is not part of the schema file (as it is with XML), please click *Repository* > *Instances* > *Parse Instance File.* Then select the resource to which the instance data refers (it must be an already imported resource). You can then commence the import of instance files. If your instance file does not match the specified schema, COMA may encounter an error and nothing may be loaded at all. Otherwise, the instance data is automatically attached to your resource.

Creating a Mapping

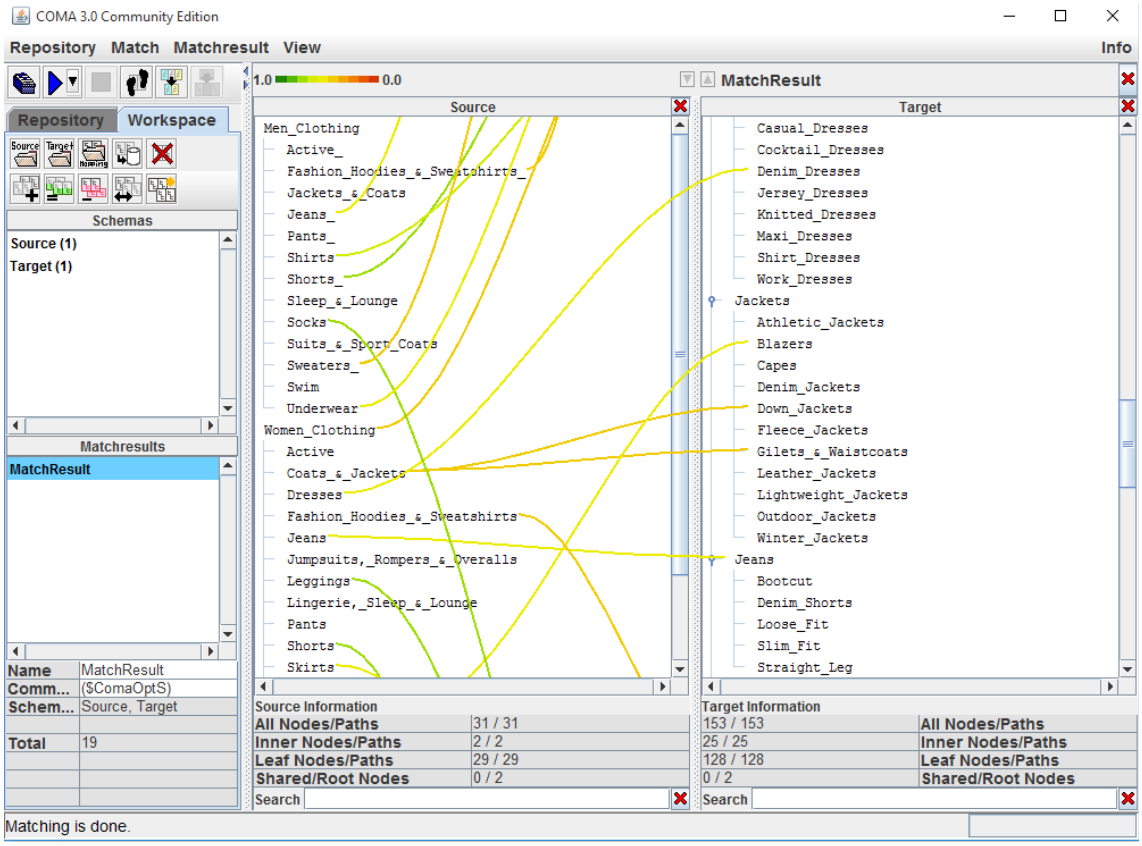
The loaded schemas are listed in the *Repository* tab. These schemas will also remain after COMA has been exited, i.e., they do not need to be re-imported at a later time. Double-clicking on one schema automatically puts it in the mapping area on the right hand. The first resource will be the source schema, the second resource will be the target schema.

Once two resources have been added to the mapping area, click *Match > Execute Workflow* and select a workflow of your choice. COMA comes with several workflows, though only two are recommended for standard scenarios:

- **$AllContextW:** Used for all standard scenarios if no instance data is available (or if instance data should not be regarded).
- **$AllContextInstW:** Used for all standard scenarios if instance data is available and should be regarded.

Warning: Please don't run *$AllContentInstW* if no instance data has been loaded!

Depending on the size of your resources, COMA will then create the mapping between the two resources. For two resources containing 500 elements, execution times should range between a few seconds and few minutes.



Post Processing

After the mapping has been calculated, you can right-click on a specific correspondence and carry out some advanced tasks (more precisely, you click on a node participating in a correspondence). For example, you can set the highest possible confidence value for a correspondence (1.0), which means a manual confirmation. You can also delete or create new correspondences. To create a new correspondence, simply select two nodes, right-click on a node and click *Create Correspondence*.

NOTE: You cannot edit a mapping without running a workflow, that is, you always have to carry out the matching before you can manually add correspondences between the two resources.

Mapping Export

To obtain a textual representation of the mapping, please click *View > Matchresult Correspondences*. The correspondences are shown in a COMA-specific format. You may use it for further processing, like merging or data transformation. However, COMA is a sole match tool, i.e., it does not provide these feats!

To save the mapping in the database, please click the icon "Save current Schema or MatchResult to Database" icon in the *Workspace tab* (or click *Matchresult > Save*). The mapping will then be saved to DB and can be loaded again at a later time or exported as a text file (you will find it in the *Repository tab*, in the *MatchResults* list). To export it, click *Repository > Matchresult > Export File.* COMA will save the mapping as text-file in a COMA-specific format.

Workflow Management

COMA consists of several atomic similarity measures that determine the likelihood that two concepts or elements are a match. Each similarity measure returns a value between 0 (no match) and 1 (certain match). The similarity measures are combined in complex workflows. Because of the workflow's complexity, there is a whole hierarchy of combined similarity measures.

Workflows are organized as follows:

- A workflow (W) consists of one or several strategies.
- A strategy (S) consists of one or several complex matchers.
- A complex matcher (CM) consists of one or several matchers.
- A matcher (M) consists of one or several similarity strategies.

Matchers and complex matchers are triples of the form *(scope, measure, result calculations)*.

- **Scope of a Matcher:** Describes what is used as input for the similarity calculation (e.g., name/label, datatype, instances etc.).
- **Scope of a Complex Matcher:** Describes what part of the schema is considered (e.g., the leave node, the entire node path, the parents of a node etc.). Note that COMA always calculates the similarity between two schema elements, but may regard more information than just the elements.
- **Measure of a Complex Matcher:** A matcher.
- **Measure of a Matcher:** A similarity measure (a match algorithm).
- **Result Calculation:** Describes how results of different measures are combined (Average, Maximum, Minimum).

You can get an overview of all matchers, strategies and workflows used in COMA by clicking *Match > Workflow Variables*. The program uses a specific grammar to create and represent workflows. By double-clicking on a variable (e.g., a strategy or matcher), you can edit the variable and thus change the behavior of the workflow. Getting more acquainted with the workflow management, this feature allows you to build new workflows consisting of individual match strategies.

By clicking *Reset > Workflow Variables* the default workflows will be reloaded and all changes will be reset. You can also click *Match > Workflow Hierarchy* to get an overview about the hierarchical arrangement of the COMA workflows.

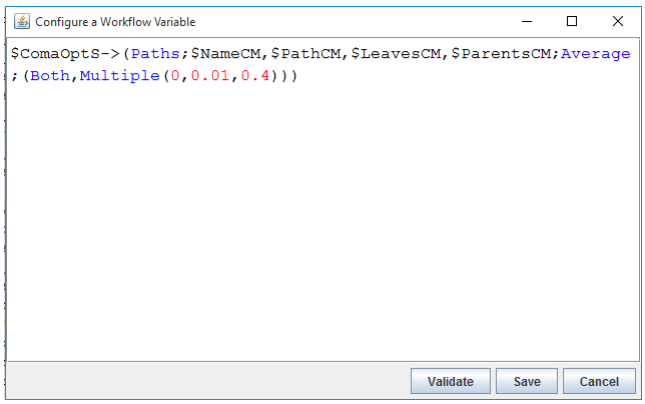
Workflow Example: $AllContext

$AllContextW consist of one strategy: $ComaOptS. This strategy is defined as follows:

Paths;NameCM;PathCM;LeavesCM;ParentCM;Average;(Both,Multiple(0,0.01,0.4))

It basically means that this strategy uses the four complex matchers NameCM, PathCM, LeavesCM, ParentCM, based on their element paths. The confidence values are averaged.

The specification Multiple(0,0.01,0.4) at the end of the line is quite important. It specifies the basic behavior of the mapping creation.



- **MaxN:** The first value specifies the maximum number of correspondences a node can be involved in. A value of 0 or 1 means that nodes can only be in 1:1-relations to each other. A value above 1 means that COMA will also consider 1:n and n:1 correspondences. For example, a value of 5 means that a source element can be related to up to 5 target elements.
- **MaxDelta:** The second value specifies the scope in which the correspondence scores must be in order to consider them as a 1:n or n:1 correspondence. A higher value leads to more 1:n and n:1 correspondences.
- **Threshold:** Most importantly, this value specifies the minimal confidence a correspondence between two elements must reach in order to be accepted to the final match result (mapping). Increasing this value leads to less

correspondences in the final mapping (but possibly a better precision). Decreasing this value leads to more correspondences in the mapping (but possibly to a worse precision).

Strategies consist of complex matchers. As an example, the name complex matcher $NameCM is organized as follows:

SelfNode;$NameM;Set_Average

This complex matcher uses only one matcher (Name Matcher) and considers the node name as input (in contrast to the node path, parent nodes etc.). If there was more than one matcher, the final score would be the average score of all matchers.

Finally, the name matcher $NameM is represented as follows:

Name;Trigram;Set_Average

This matcher uses the Trigram similarity measure and works directly on the node name (in contrast to the data type, instances etc.). If there was more than one matcher, the final score would be the average score of all measures.

Many details about the workflow management and different strategies used in COMA have been presented in the following publication:

http://dbs.uni-leipzig.de/file/COMA.pdf

Please confer to this publication (or possibly to further publications found on our website) to get a more profound insight in the anatomy of COMA.

Troubeshooting / FAQ

COMA cannot be launched.

- ➔ Java may not be installed.
- ➔ MySQL may not be installed (or the service may not be running).
- ➔ COMA may not have access rights to create a database in MySQL.
- ➔ The schema "coma-project" may not have been created by the user.

When trying to load a schema or ontology, COMA freezes/nothing happens.

- ➔ Your resource file may use a format that is not supported or which is invalid. You may try to convert it into another (possibly simpler) format.
- ➔ The overall file path of your resource may be too long (longer than 100 characters). You may copy your files in a superior directory and try again.
- ➔ The resource file may be too big. You may grant the JVM more memory and/or run COMA on a server.

I cannot select any workflow.

- ➔ Please click *Match > Reset Workflow Variables*.

COMA freezes when creating the mapping.

- ➔ The schema files may be too large. You may grant the JVM more memory and/or run COMA on a server.
- ➔ You may try another (simpler) workflow.
- ➔ Some larger mappings may take some time... just wait & see.