mantistable.disco.unimib.it

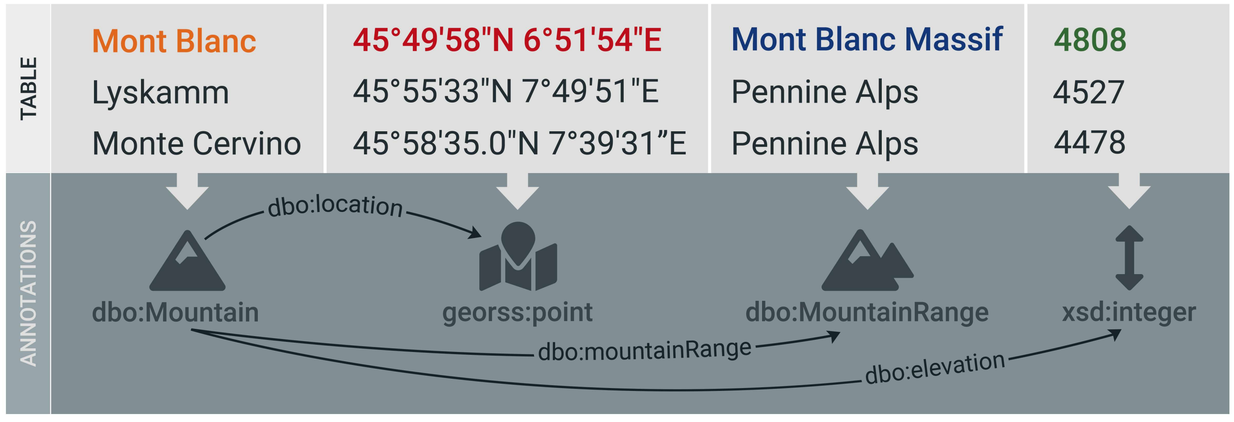
MANTISTABLE

a Tool for Creating **Semantic Annotations** on **Tabular Data**

Marco Cremaschi, Anisa Rula, Alessandra Siano and Flavio De Paoli | **Contact**: marco.cremaschi@unimib.it

University of Milano - Bicocca
Milano, Italy

MantisTable is an open source **Semantic Table Interpretation** tool, which automatically annotates **tables** using a **Knowledge Graph**

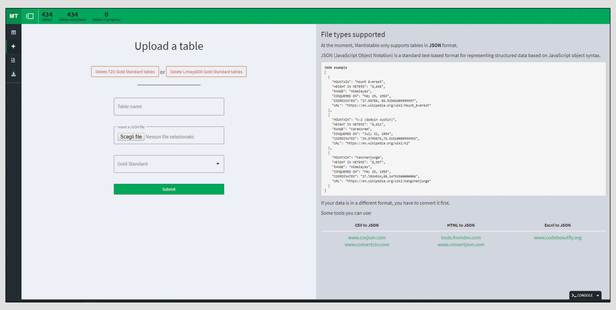


APPLICATION INTERFACE

LOADING AND STORING

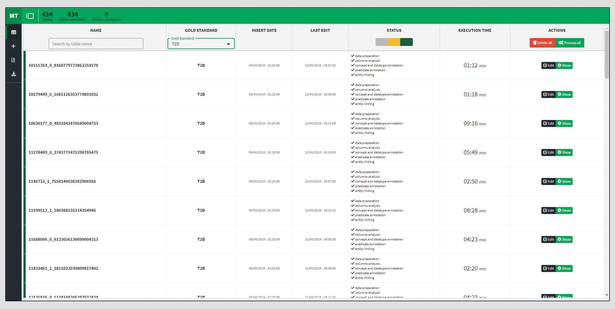
Users can **add tables** in JSON format

Tables are imported and stored in a **MongoDB** database



A **list of loaded tables** is displayed on the main page

Users can **download** the annotated tables at the end of the annotation process



EXECUTION

5-step annotation process

Users can run all steps together or step by step

EXPLORATION

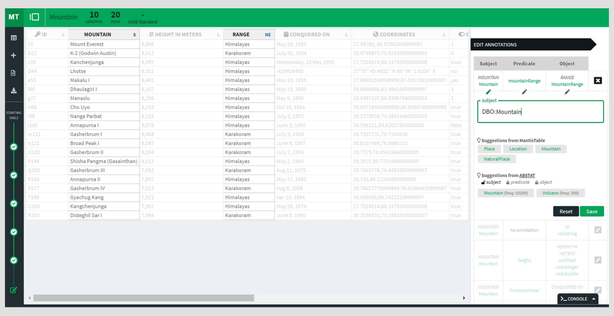
Users can navigate back and forth through the steps performed

Each step shows **additional information** on the right sidebar

EDITING

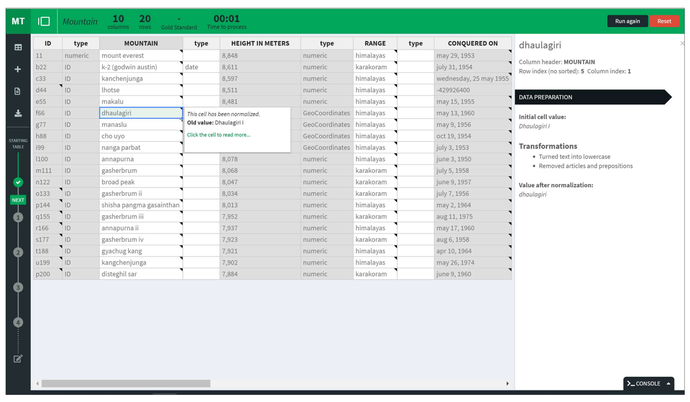
Users can **edit** the annotations

Editing annotations is supported by **autocompletion** and **suggestions**



MAIN PROCESS

0 DATA PREPARATION

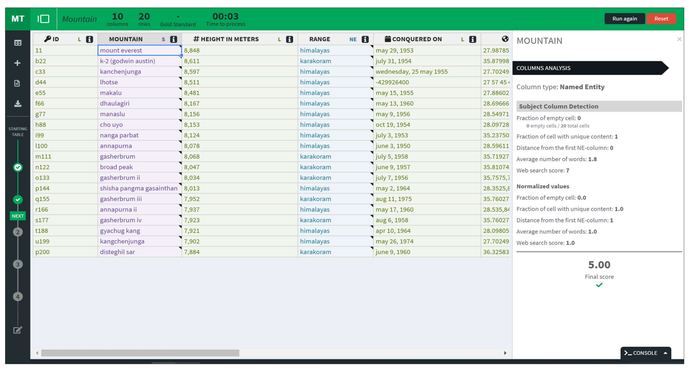


The **data** in the table are **cleaned** and **uniformed**

The applied **transformations** are:

- removal of HTML tags and stop words
- transformation of the text into lowercase
- resolution of acronyms and abbreviation
- normalization of units of measurement by applying regular expressions

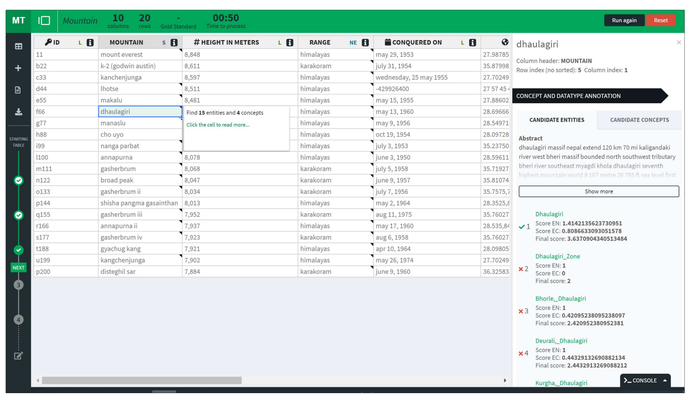
1 COLUMN ANALYSIS



Columns are classified as named-entity column (**NE-column**) or literal column (**L-column**) and a subject column (**S-column**) is detected

- Detection of **L-columns** by 16 regular expressions to identify regextype (e.g., geo coordinate, address, hex color code, URL)
- Detection of **S-column** considers different statistic features

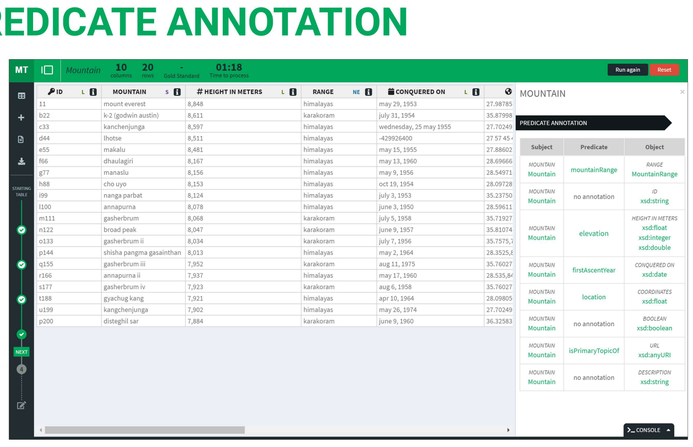
2 CONCEPT AND DATATYPE ANNOTATION



Column headers are mapped to **semantic elements** (*concepts* or *datatypes*) of a **Knowledge Graph**

- Retrieval of a set of **candidate entities** performing the **entity-linking** by searching the Knowledge Graph with the content of a cell
- Retrieval of **abstract** and **concepts** for each item in the set of retrieved entities
- Application of heuristics for the identification of the most frequent **concept** of the column

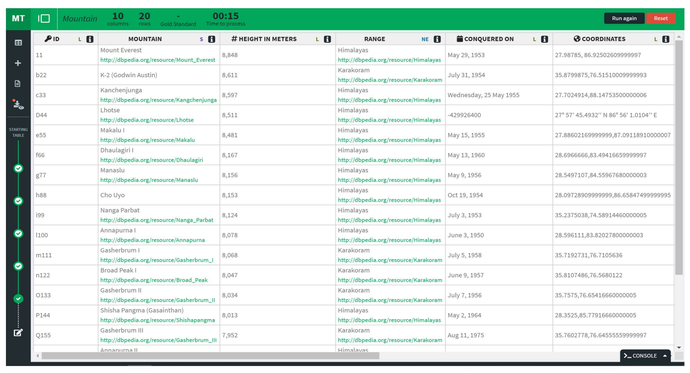
3 PREDICATE ANNOTATION



Relations (predicates) between the subject column and the other columns are identified

- The winning concept of the **S-column** are considered as the **subject** of the relationship and annotations of the other columns as **objects**
- The Knowledge Graph is searched for the subject and the object to collect possible predicates

4 ENTITY LINKING



The **content of cells** is mapped to entities in the **Knowledge Graph**

- Already discovered annotations are used to create a query for the **disambiguation of the cell content**
- If more than one entity is returned for a cell, the one with a smaller edit distance is taken