With this information we can:

1. Describe your input data
   - OR
   - Directly submit your dataset

2. Describe your objectives
   - "I want to detect potential forest fires on satellite images, with the best accuracy possible."

3. Provide Algorithm properties
   - What is the algorithm used for?
   - Does the algorithm work with numerical and/or nominal data?
   - Can the algorithm work with missing values?
   - What is the implementation platform?
   - Description, examples of use, properties for the algorithm
   - Can you provide executable code?

4. Internal process when adding an algorithm
   - If code was provided, run experiments on the new algorithm
   - Ensure experiments did not reveal inconsistencies with the provided information
   - Compare algorithms results with the other algorithms
   - Add the new algorithm to the system:
     - Add it in the Feature Model
     - Generate appropriate constraints
     - Create display metadata based on provided information

5. Comparing algorithms
   - First similar datasets, following a dataset pattern
   - For each pattern, rank the workflows (pre-processing + algorithm) that were applied on it.
   - Ranking is based on statistical computations: for each pair of algorithms, we test if the results are significantly different.

6. Knowledge obtained through experiments
   - Which pre-processing techniques can be applied on a given dataset pattern.
   - What are estimated results for a given tuple (dataset pattern, pre-processing, algorithm).
   - "Best accuracy", "average memory usage", "slower execution"

7. Perspectives & Future Work
   - Allow for a more precise trade-off between non-functional requirements through the use of goal models
   - Adapt the tool for specific domains such as images or medical data analysis
   - Look for more relevant data patterns
   - Capitalize on experiments made by users in the tool
   - Focus on privacy aspects: local execution, data anonymisation

---

Request Your Own Knowledge Flows (ROCKFlows)

Cécile Camilleri, Mireille Blay-Fornarino, Frédéric Précioso, Michel Riveill

As an End User
You request a Machine Learning Workflow to answer your needs

Description of input data
- With which workflows can be applied?
- Estimate the results of the possible workflows, by comparing with results on similar datasets we already know.

Your data

Provide Algorithm properties
- What is the algorithm used for?
- Does the algorithm work with numerical and/or nominal data?
- Can the algorithm work with missing values?
- What is the implementation platform?
- Description, examples of use, properties for the algorithm
- Can you provide executable code?

Your algorithm

Comparing algorithms
- First similar datasets, following a dataset pattern
- For each pattern, rank the workflows (pre-processing + algorithm) that were applied on it.
- Ranking is based on statistical computations: for each pair of algorithms, we test if the results are significantly different.

ROCKFlows

Configuration interface
- Web-based interface
- Oriented towards non-expert users
- With each element, we associate metadata:
  - Question
  - Description
  - Examples
  - References
  - Images
  => You make your choices by answering yes/no questions

Model-driven Workflow Generation
- User configuration
- Workflow Generation
- Platform-independent Workflow

A Feature Model to express variability

Constraints between features:
- Algo A excludes Numerical Data
- Algo B requires Numerical Data

Similar Works
Some of the similar works:
- IBM Watson automatically learns from different data sources at the same time (pdfs, database, etc.)
- Microsoft Azure Machine Learning provides with an advances Graphical User Interface to build workflows
- NLBase aims to provide users with a suitable workflow through active research

Compared to those, ROCKflows:
- Is non-expert oriented;
- Knowledge is based on experiments;
- Aims to bring together a community;
- At the same time, we try to understand better the impact of different parameters (input data, ...) on the results of the workflows

Perspectives & Future Work
- Allow for a more precise trade-off between non-functional requirements through the use of goal models
- Adapt the tool for specific domains such as images or medical data analysis
- Look for more relevant data patterns
- Capitalize on experiments made by users in the tool
- Focus on privacy aspects: local execution, data anonymisation

http://rockflows3s.unice.fr

---

Some of the similar works:
- IBM Watson automatically learns from different data sources at the same time (pdfs, database, etc.)
- Microsoft Azure Machine Learning provides with an advances Graphical User Interface to build workflows
- NLBase aims to provide users with a suitable workflow through active research

Compared to those, ROCKflows:
- Is non-expert oriented;
- Knowledge is based on experiments;
- Aims to bring together a community;
- At the same time, we try to understand better the impact of different parameters (input data, ...) on the results of the workflows

Perspectives & Future Work
- Allow for a more precise trade-off between non-functional requirements through the use of goal models
- Adapt the tool for specific domains such as images or medical data analysis
- Look for more relevant data patterns
- Capitalize on experiments made by users in the tool
- Focus on privacy aspects: local execution, data anonymisation

http://rockflows3s.unice.fr