unibz
Freie Universität Bozen
Libera Università di Bolzano
Università Liedia de Bulsan

An Unsupervised Feature Selection algorithm based on Ant Colony Optimization

Aaron Estrada Poggio

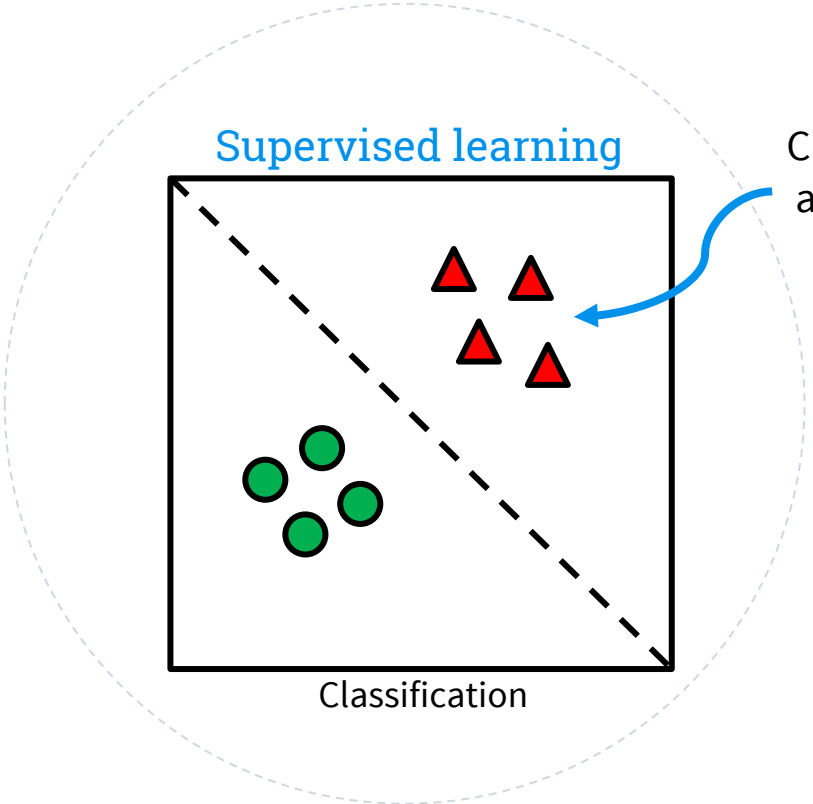
*If you torture the **data** long enough, it will confess.*

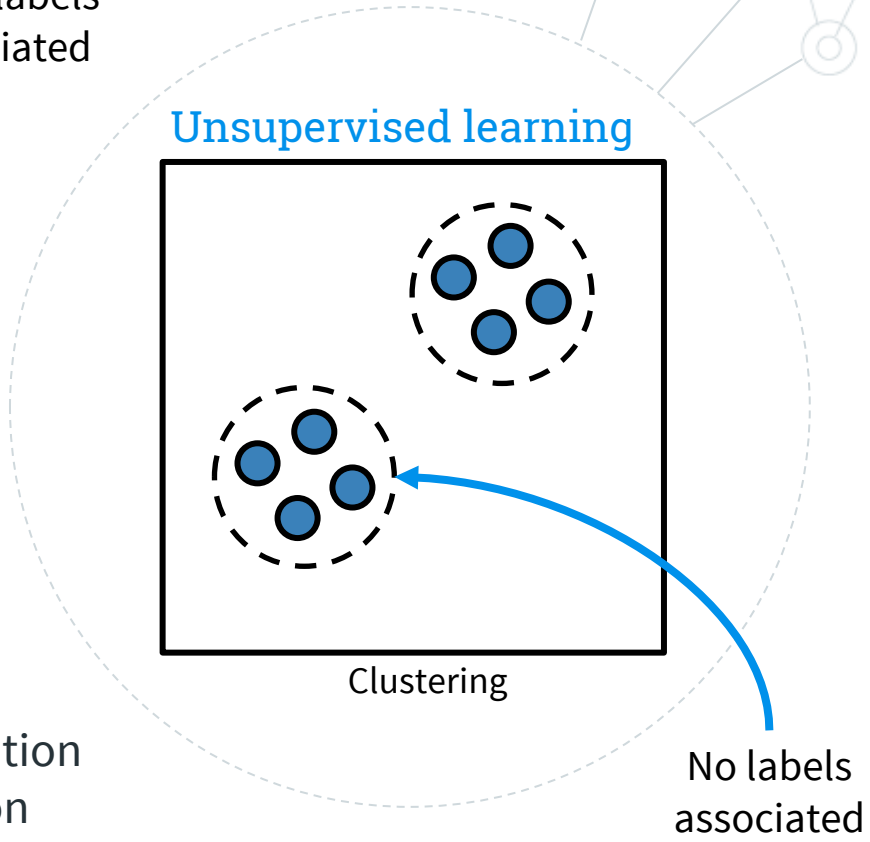
Ronald Coase

Data Mining

The computational process to find knowledge from datasets, which is expressed using a comprehensible structure.

Data Mining – some tasks and applications





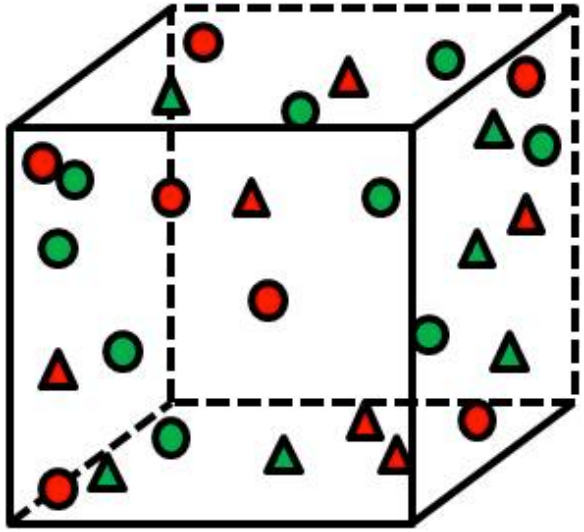
Applications

- Text categorization
- Face recognition
- Cancer classification
- Finance

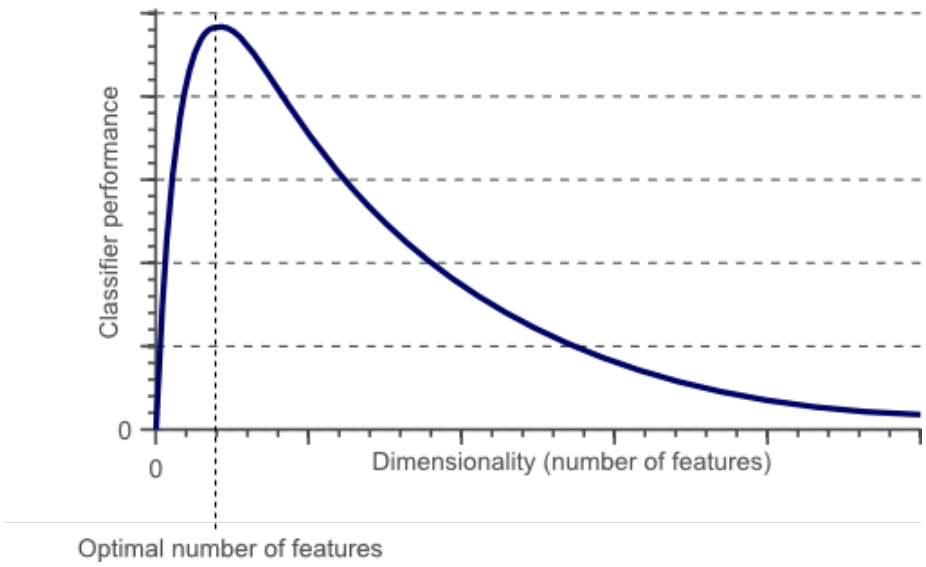
Data Mining – Curse of Dimensionality (1)

Number of features larger than number of patterns

- Performance reduced
- Computational complexity increased



Data Mining – Curse of Dimensionality (2)



Feature Selection

Remove irrelevant and redundant features.

1.000

features

$2^{1.000}$

different feature subsets!

Find a near-optimal feature subset in a reasonable time.

Feature Selection

Remove irrelevant and redundant features.

Data Mining - Feature Selection – Approaches

1. **Filter approach**

- Feature relevance relies on statistic data
- Independent from learning algorithms

Univariate

Feature dependency ignored

- Information gain
- Gain ratio
- Fisher score

Multivariate

Feature dependency considered

- Mutual correlation
- Minimal-redundancy-maximal-relevance (MRMR)
- Random subspace methods (RSM)

2. Wrapper approach

Uses a learning algorithm to select features

Sequential search

Add or remove features sequentially. Tendency to become trapped into a local optimum.

- Sequential backward selection
- Sequential forward selection
- Floating search method

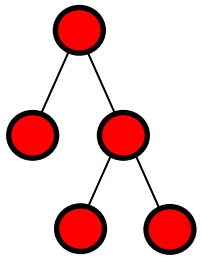
Random search

Embed randomness to escape local optimum solutions

- Random mutation hill-climbing
- Genetic algorithms
- Ant Colony Optimization (ACO)

3. Embedded approach

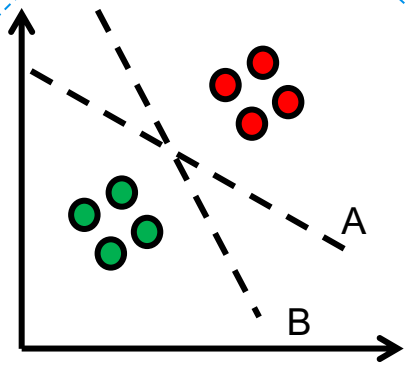
Search of good subset of features performed by learning algorithm



Decision Trees

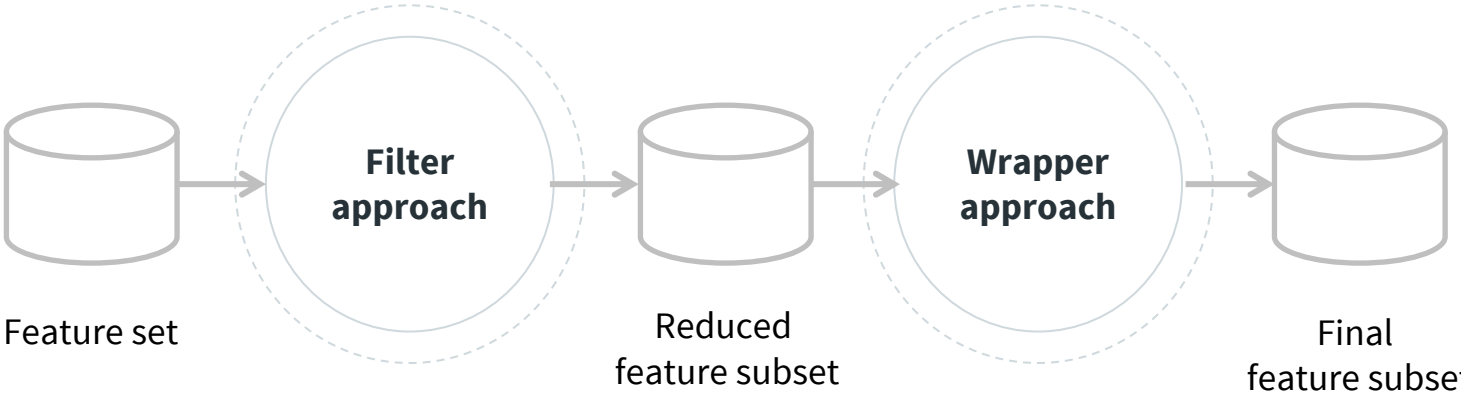
$$P(B|A) = \frac{P(A \text{ and } B)}{P(A)}$$

Naïve Bayes



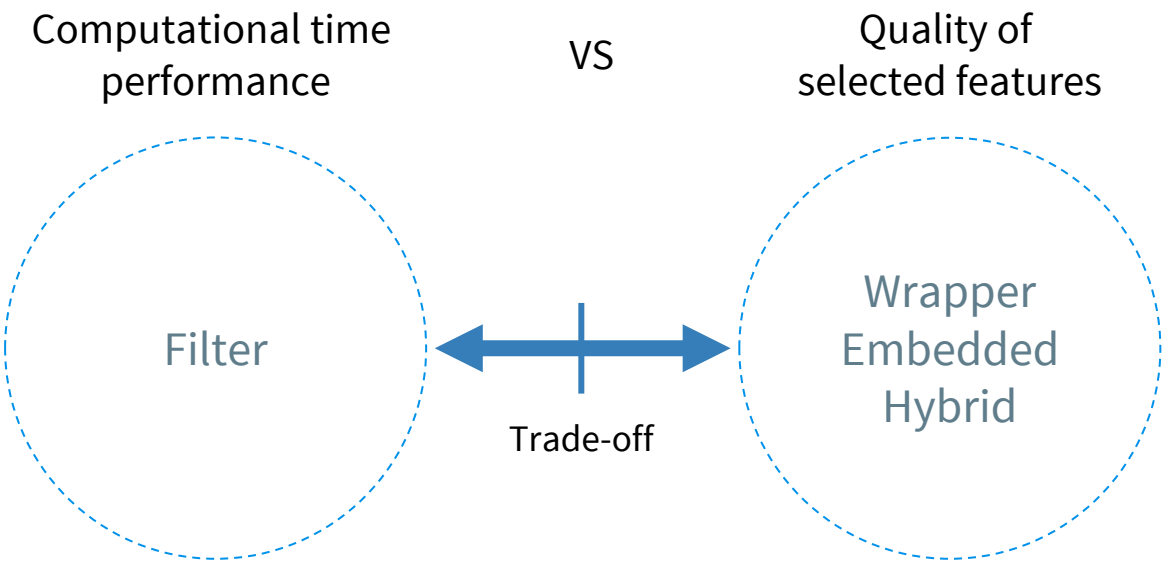
Support Vector Machine (SVM)

4. Hybrid approach



- ACO + mutual information
- ACO + chi-square with SVM
- Mutual information + genetic algorithms
- Mutual information + Swarm optimization with SVM

Feature Selection – Approaches



How to deal with the trade-off?

Unsupervised Feature Selection algorithm based on Ant Colony Optimization (UFSACO)

1. **Unsupervised learning**

Does not need class labels

2. **“Filter approach”**

Acceptable computational time and no learning algorithm needed

3. **Based on ACO**

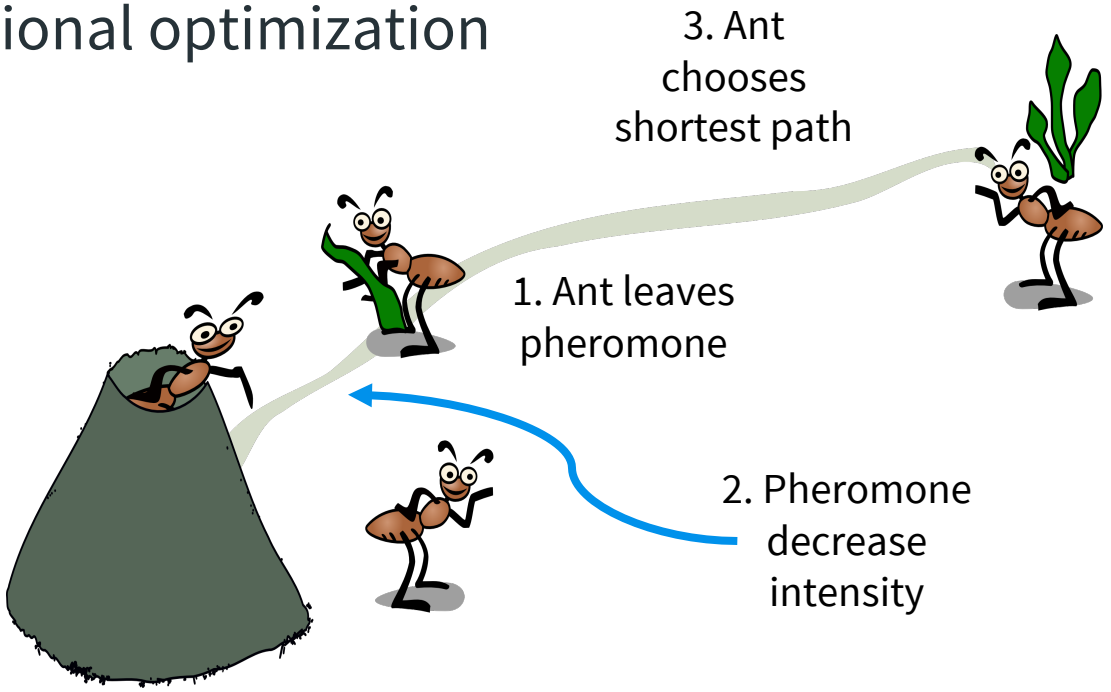
High quality approximation of the solution and efficient for high-dimensional datasets

4. **Minimize redundancy**

Reached through feature similarity heuristics.

Ant Colony Optimization (ACO)

- Based on the social behavior of ants while seeking food
- Cooperative work of ants to solve hard combinational optimization problems



Ant Colony Optimization (ACO) - Strengths

Multi-agent system

Ants cooperate by sharing knowledge through pheromone trail to solve the problem efficiently

Parallel implementation

Distribution allows a decrease in computational time

Distributed long-term memory

Memory used to store knowledge obtained from ants in previous searches

Global and local search capabilities

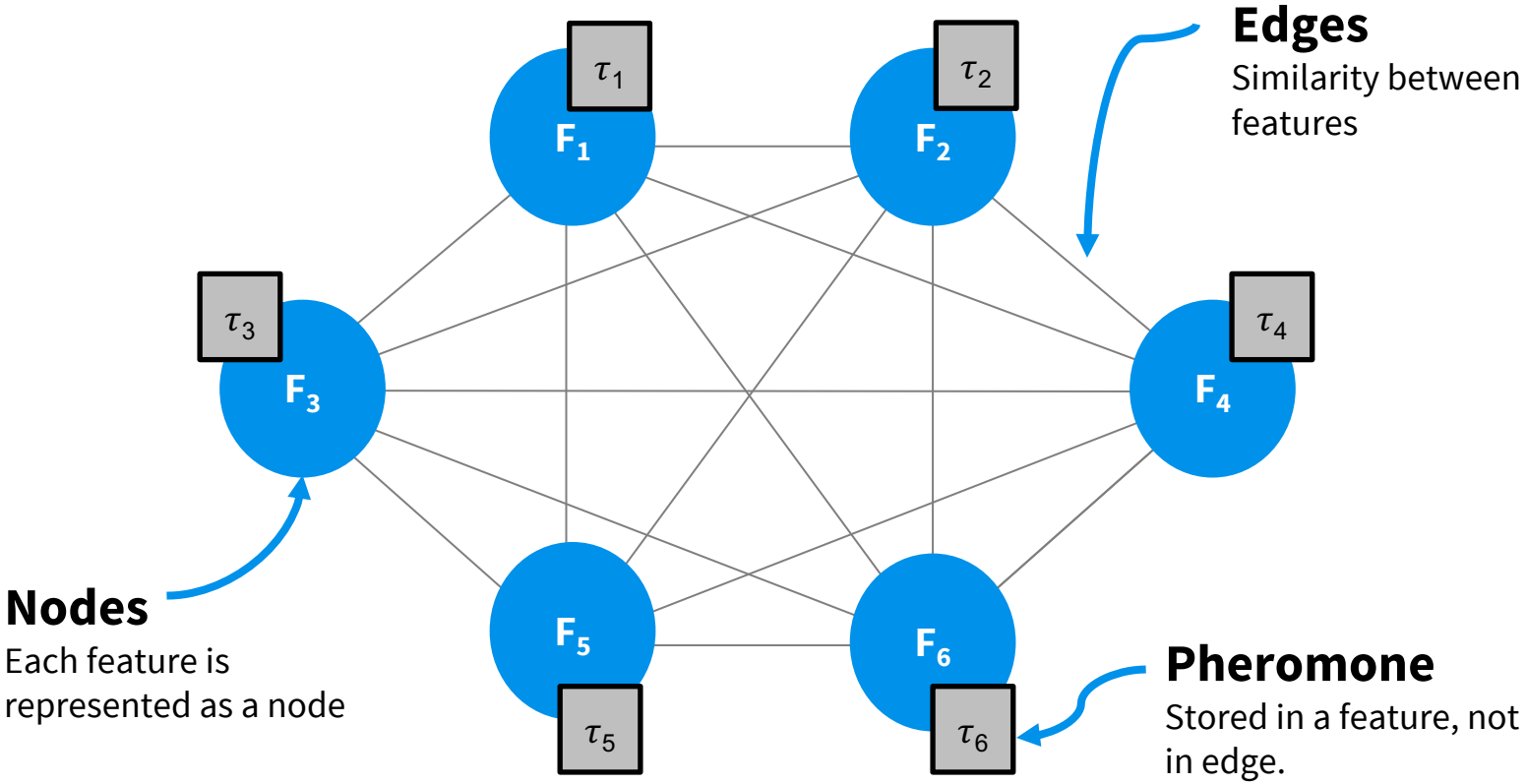
- **Exploitation** - strong local search ability
- **Exploration** - avoid being trapped in local minimum

UFSACO – Components

- **Graph representation for searching**
- Heuristic desirability
- Working cycle
- Pheromone update
- Feature selection

UFSACO – Graph representation for searching

Searching space as undirected graph



Cosine similarity between features

$$sim(F_a, F_b) = sim(F_b, F_a) = \frac{\sum_{i=1}^{p}(a_i * b_i)}{\sqrt{\sum_{i=1}^{p} {a_i}^2} * \sqrt{\sum_{i=1}^{p} {b_i}^2}}$$

F_a, F_b	Feature a/b
p	Total of patterns in dataset
a_i, b_i	Value of feature a/b for pattern$_i$

UFSACO – Components

- Graph representation for searching
- **Heuristic desirability**
- Working cycle
- Pheromone update
- Feature selection

UFSACO – Heuristic desirability

1. Heuristic information

$$\eta(F_a, F_b) = \eta(F_b, F_a) = \begin{cases} \dfrac{1}{sim(F_a, F_b)}, & sim(F_a, F_b) > 0 \\ 0, & otherwise \end{cases}$$

UFSACO – Heuristic desirability

2. Desirability of movement

An ant seeks to select features based on:

- Highest pheromone values
- Lowest similarities to previous selected features (heuristic information)
- Pseudo-random-proportional rule: transition rule based on random value *q*

UFSACO – Heuristic desirability
Pseudo-random-proportional rule

Exploitation – Greedy transition rule

$$j = \arg\max_{u \in J_i^k} \{[\tau_u][\eta(F_i, F_u)^\beta]\}, \qquad if\ q \leq q_0$$

J_i^k	unvisited feature set when ant *k* is in feature F_i
τ_u	Pheromone value of feature u
$\eta(F_i, F_u)$	Heuristic information between features F_i and F_u
β	Importance of pheromone versus heuristic information
q	Random number $q \in [0,1]$
q_0	Exploitation-exploration coefficient $q_0 \in [0,1]$

UFSACO – Heuristic desirability
Pseudo-random-proportional rule

Exploration – Probabilistic transition rule

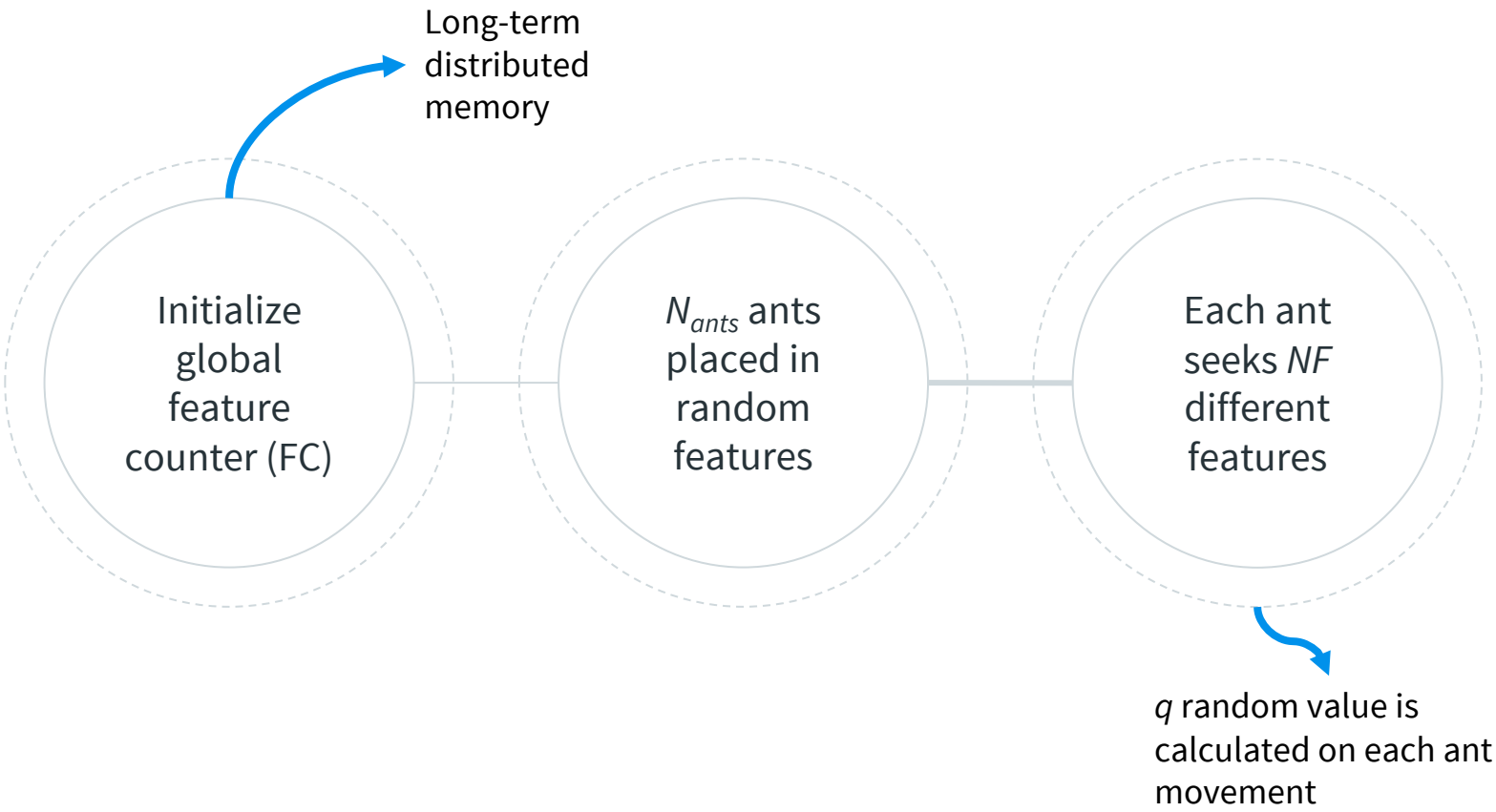
$$P_k(i,j) = \begin{cases} \dfrac{[\tau_j][\eta(F_i,F_j)^{\beta}]}{\sum_{u \in J_i^k}[\tau_u][\eta(F_i,F_u)^{\beta}]}, & if\ j \in J_i^k \\ 0, & otherwise \end{cases}, \quad if\ q > q_0$$

J_i^k	unvisited feature set when ant *k* is in feature F_i
τ_u	Pheromone value of feature u
$\eta(F_i, F_j), \eta(F_i, F_u)$	Heuristic information between features F_i and F_j / F_i and F_u
β	Importance of pheromone versus heuristic information
q	Random number $q \in [0,1]$
q_0	Exploitation-exploration coefficient $q_0 \in [0,1]$

UFSACO – Components

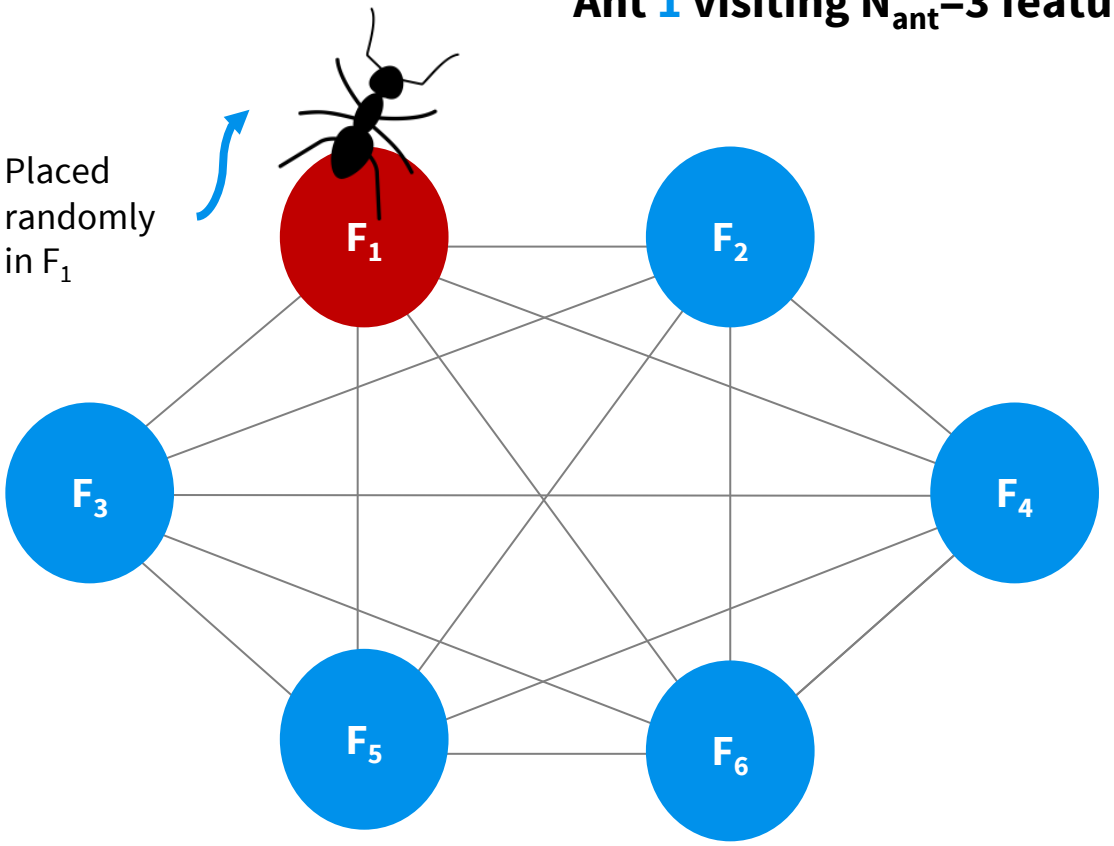
- Graph representation for searching
- Heuristic desirability
- **Working cycle**
- Pheromone update
- Feature selection

UFSACO – Working cycle



UFSACO – Working cycle Example

Ant 1 visiting N_{ant}=3 features

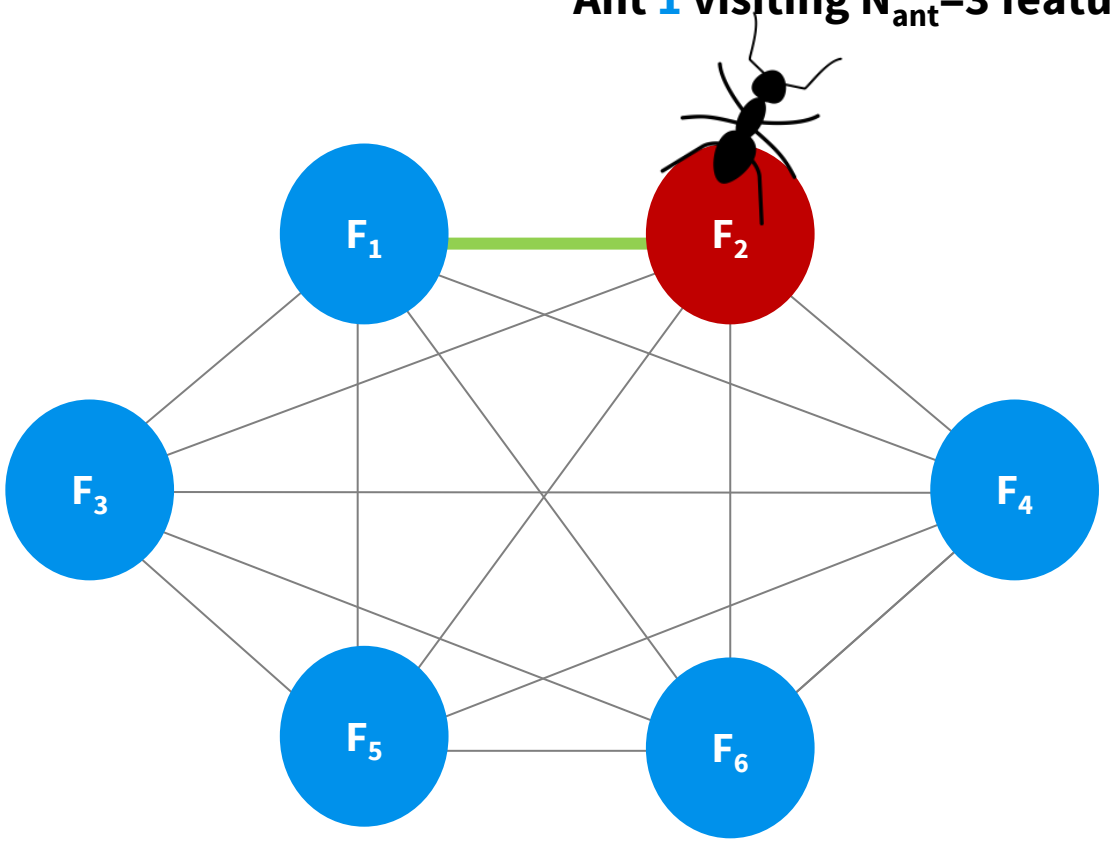


Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**0**
F_3	**0**
F_4	**0**
F_5	**0**
F_6	**0**

UFSACO – Working cycle
Example

Ant 1 visiting N_{ant}=3 features

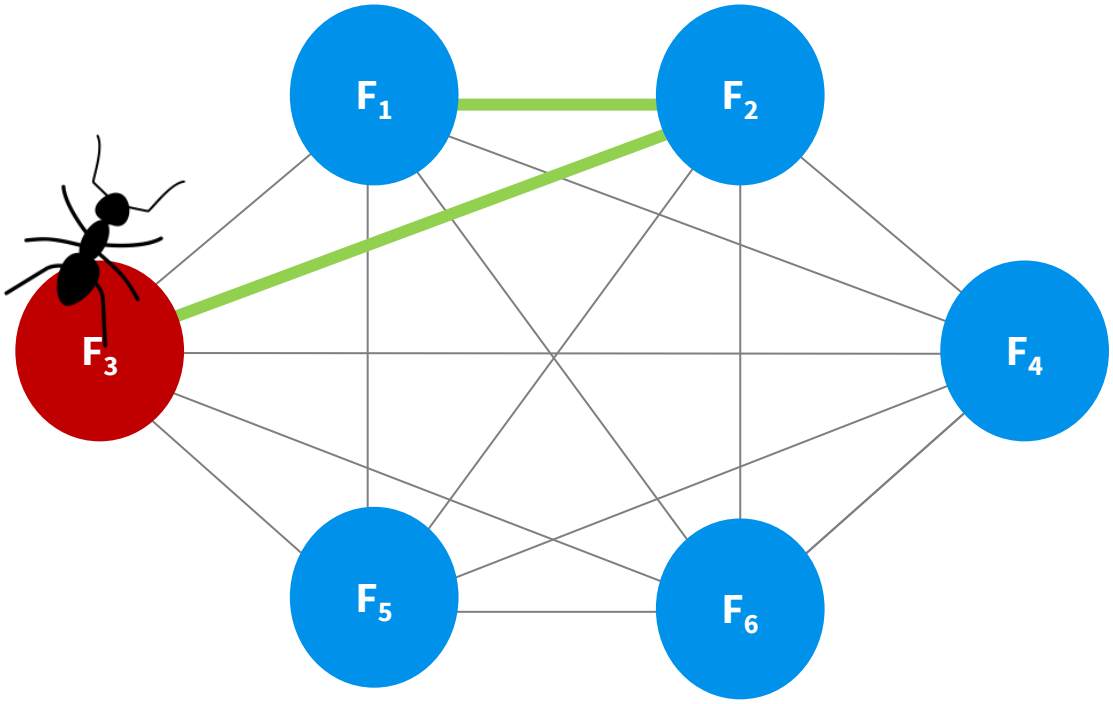


Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**1**
F_3	**0**
F_4	**0**
F_5	**0**
F_6	**0**

UFSACO – Working cycle
Example

Ant 1 visiting N_{ant}=3 features

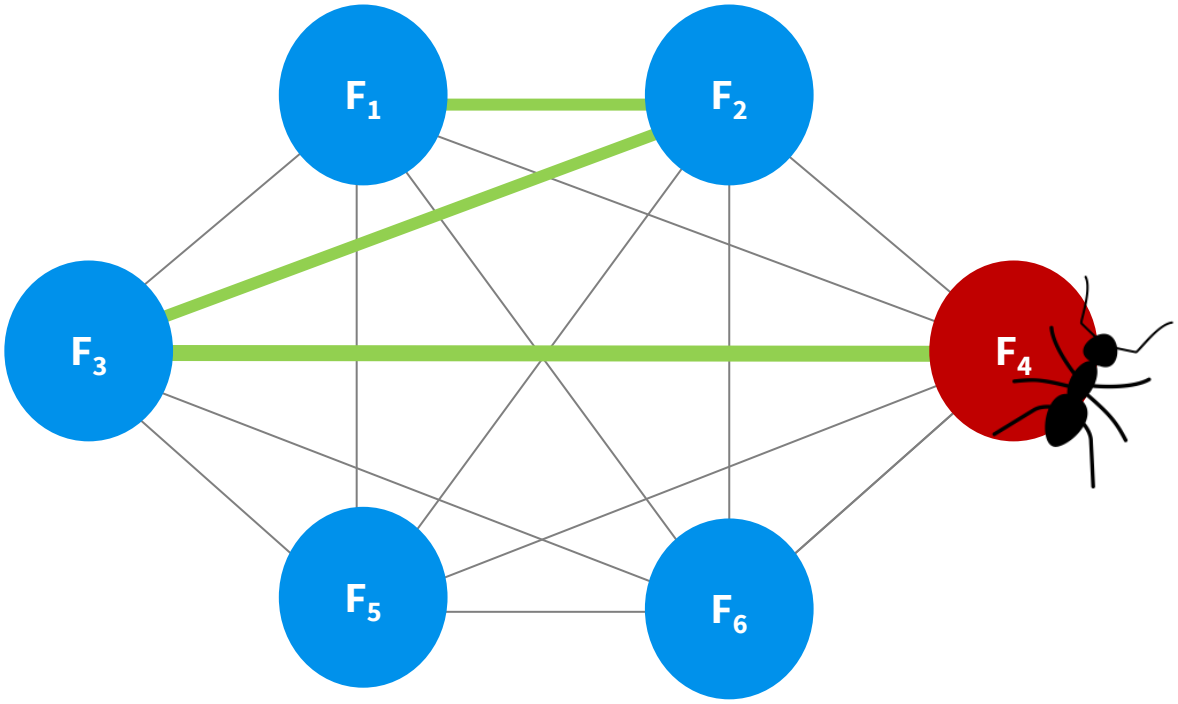


Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**1**
F_3	**1**
F_4	**0**
F_5	**0**
F_6	**0**

UFSACO – Working cycle Example

Ant 1 visiting N_{ant}=3 features

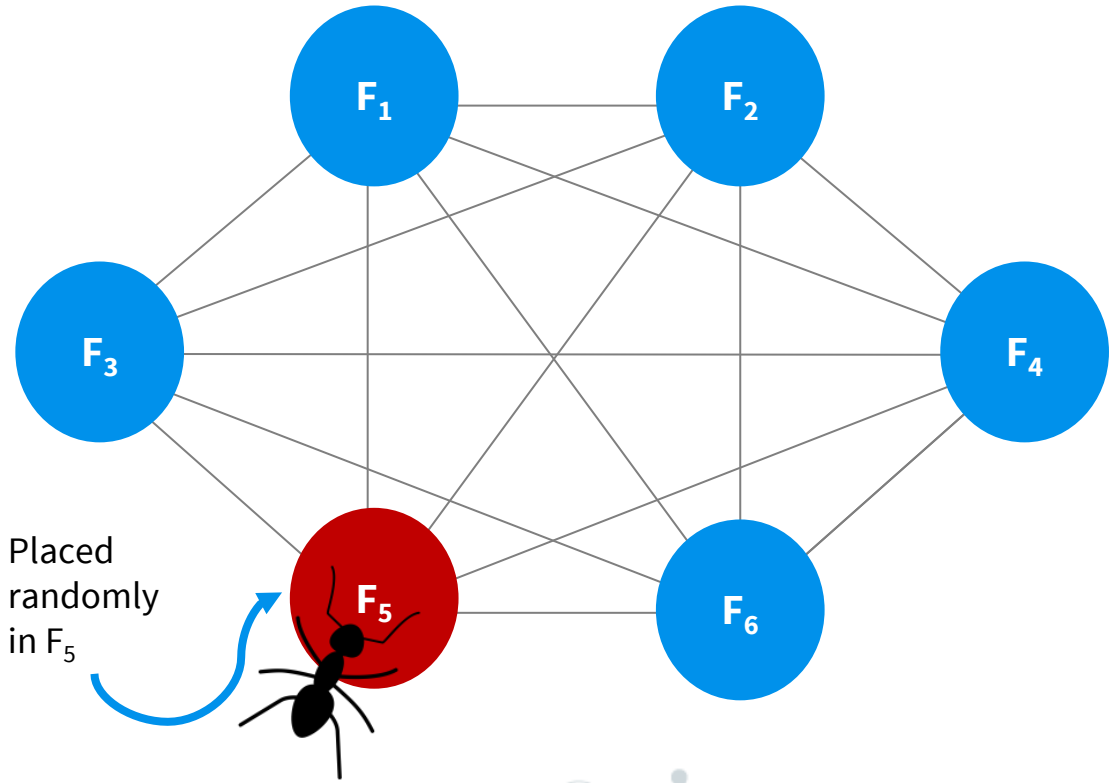


Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**1**
F_3	**1**
F_4	**1**
F_5	**0**
F_6	**0**

UFSACO – Working cycle Example

Ant 2 visiting N_{ant}=3 features

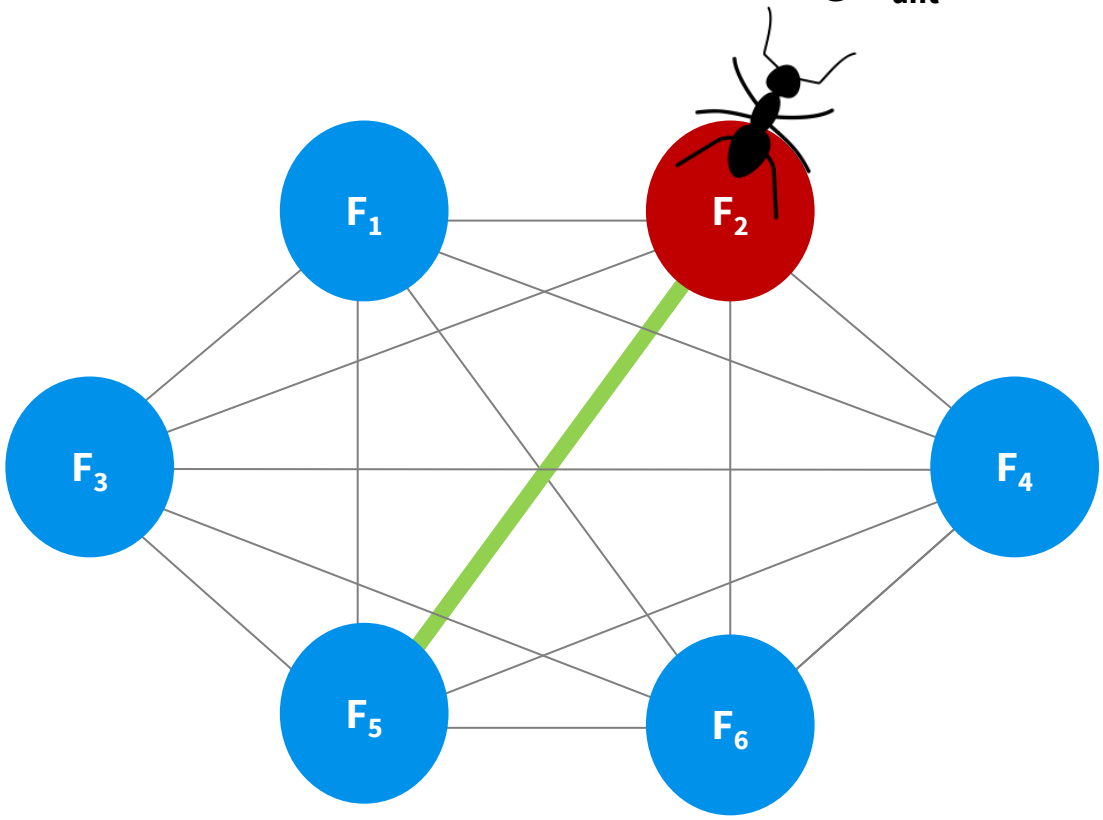


Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**1**
F_3	**1**
F_4	**1**
F_5	**0**
F_6	**0**

UFSACO – Working cycle Example

Ant 2 visiting N_{ant}=3 features

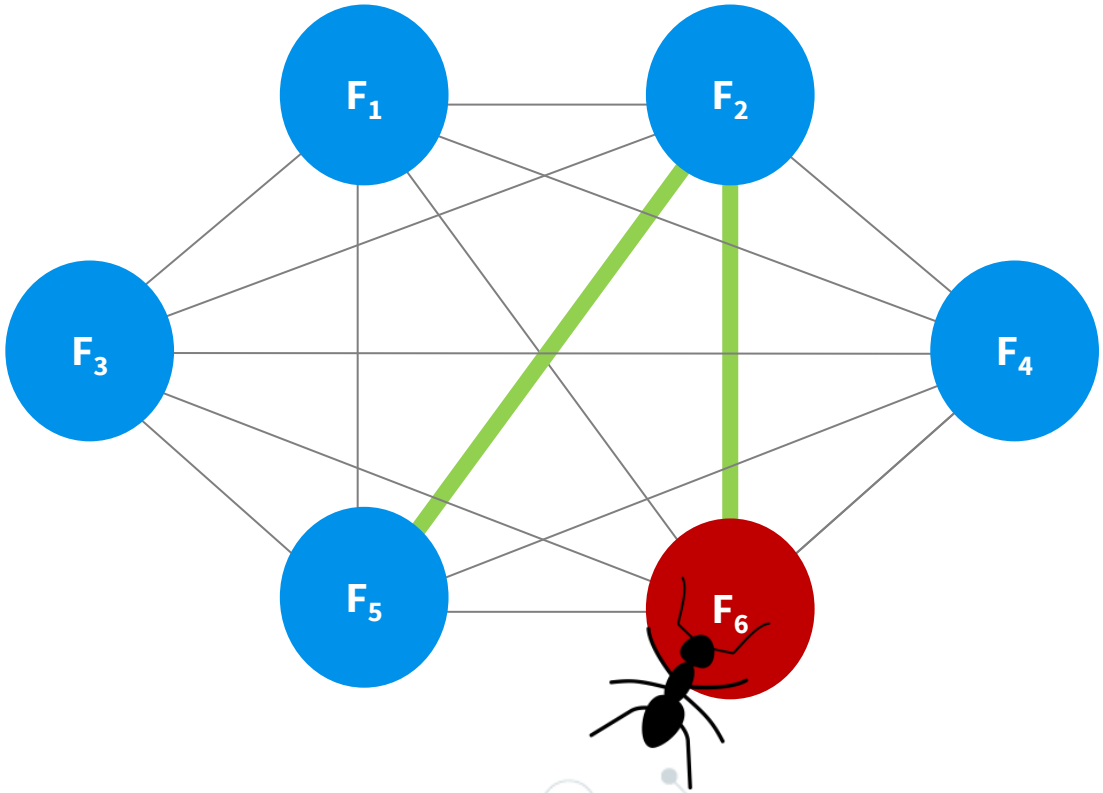


Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**2**
F_3	**1**
F_4	**1**
F_5	**0**
F_6	**0**

UFSACO – Working cycle
Example

Ant 2 visiting N_{ant}=3 features

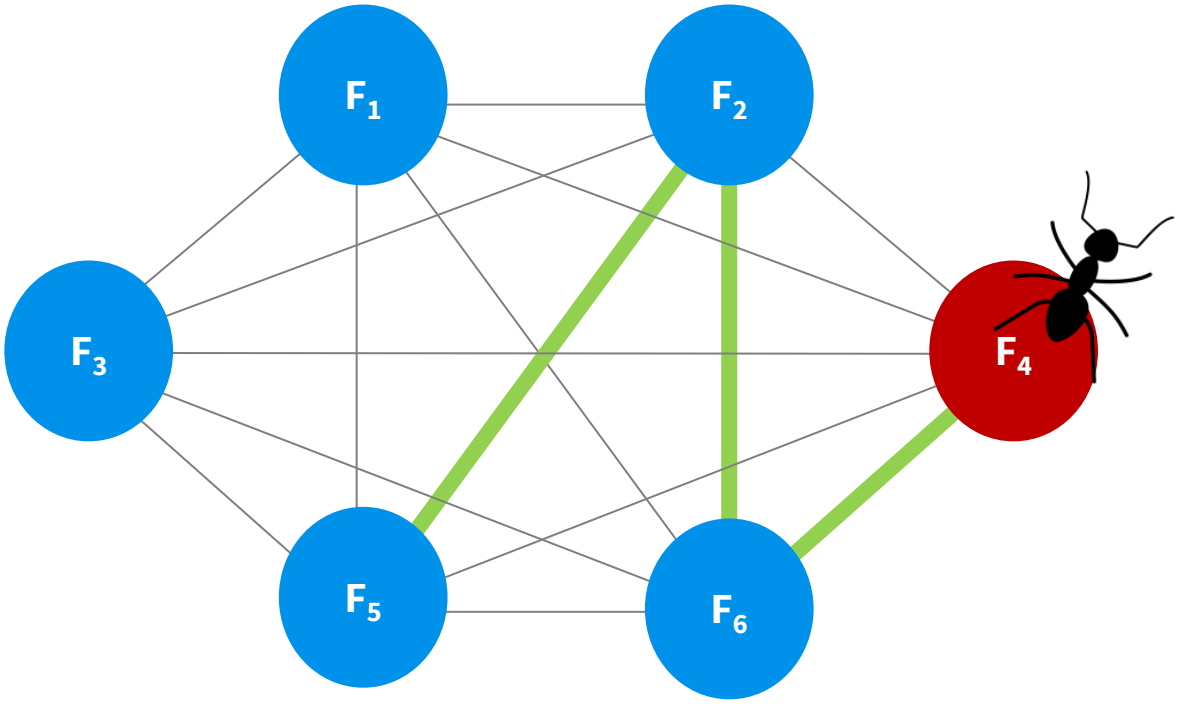


Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**2**
F_3	**1**
F_4	**1**
F_5	**0**
F_6	**1**

UFSACO – Working cycle Example

Ant 2 visiting N_{ant}=3 features



Feature Counter (FC)

Feature	Visits
F_1	**0**
F_2	**2**
F_3	**1**
F_4	**2**
F_5	**0**
F_6	**1**

UFSACO – Components

- Graph representation for searching
- Heuristic desirability
- Working cycle
- **Pheromone update**
- Feature selection

Initial pheromone value

All features start with same pheromone value

$$\tau_i(t = 0) = c$$

$\tau_i(t)$	Pheromone value of feature F_i in time t
c	Pheromone initial value parameter

Global updating rule

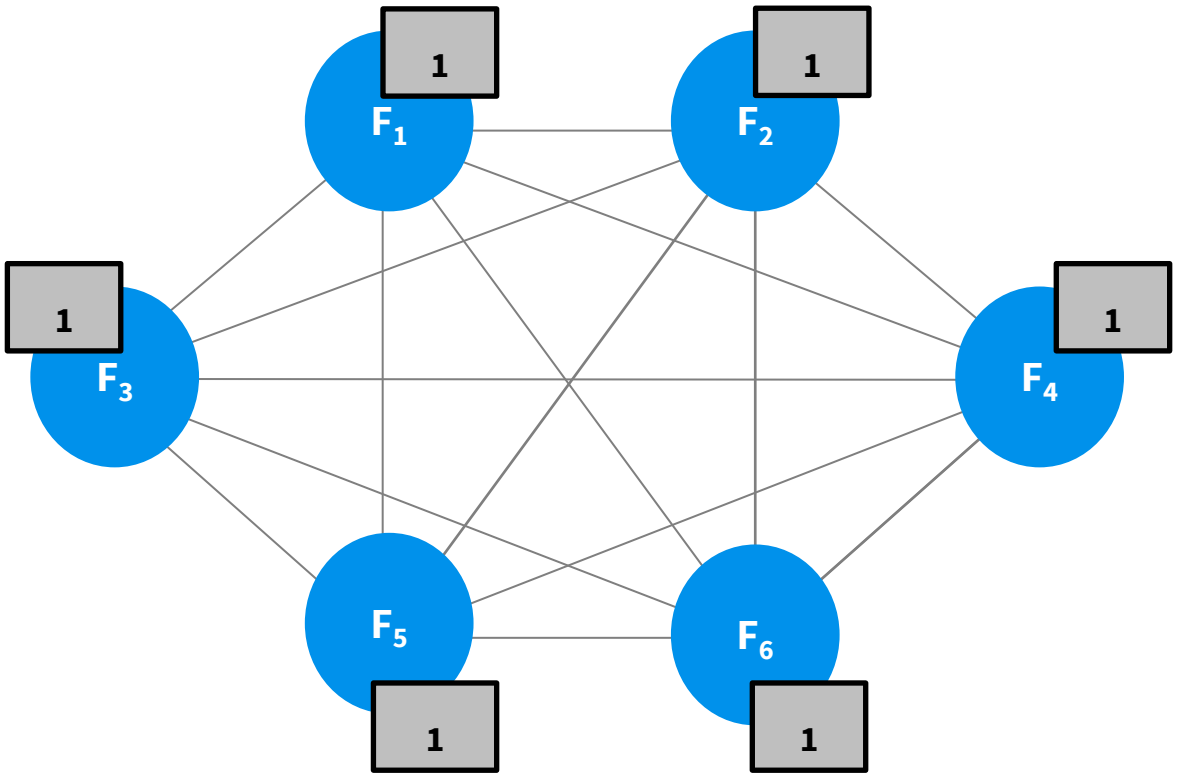
Applied to all nodes after each ants' working cycle.

$$\tau_i(t+1) = (1-\rho) * \tau_i\,(t) + \frac{FC[i]}{\sum_{j=1}^{n} FC[j]}$$

τ_i(t)	Pheromone value of feature F_i in time t
τ_i(t+1)	Pheromone value of feature F_i in time t+1
ρ	Pheromone evaporation parameter
FC[i]	Feature counter value for feature F_i

UFSACO – Pheromone update Example

Initialize pheromone values (c=1)

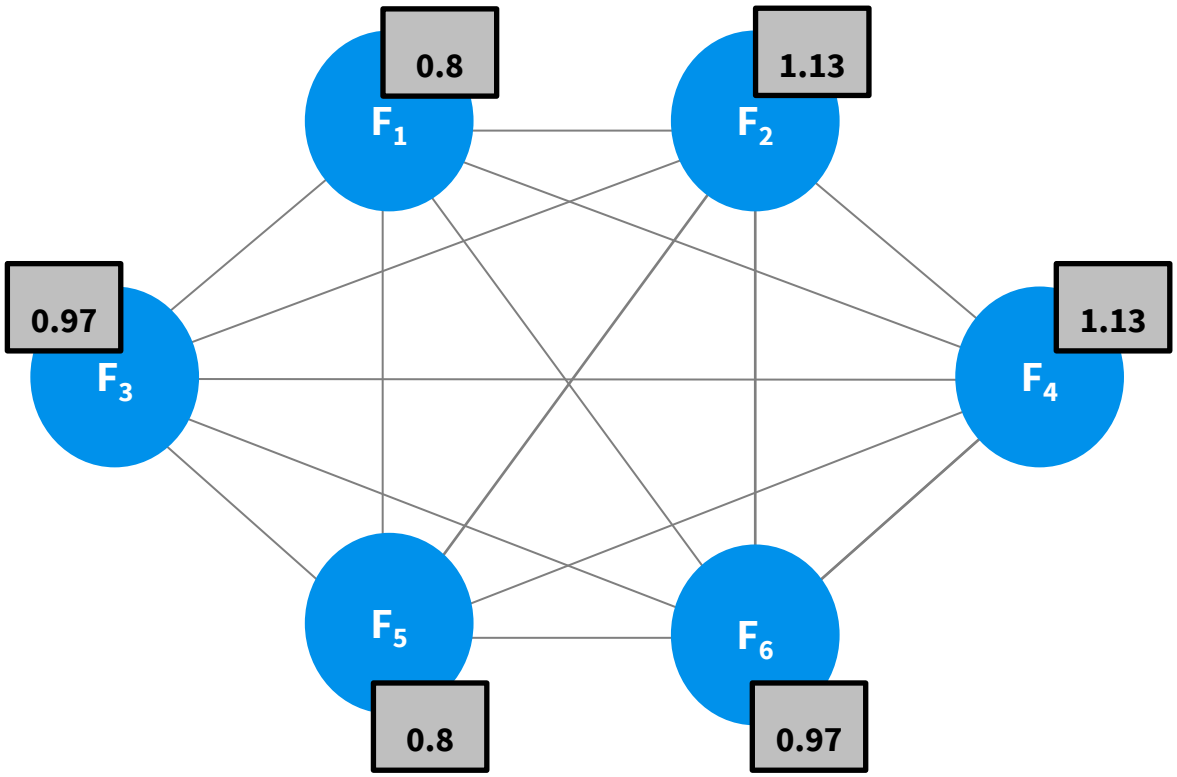


Feature Counter (FC)

Feature	Visits
F_1	0
F_2	0
F_3	0
F_4	0
F_5	0
F_6	0

UFSACO – Pheromone update Example

Update pheromone values after working cycle ($\rho = 0.2$)



Feature Counter (FC)

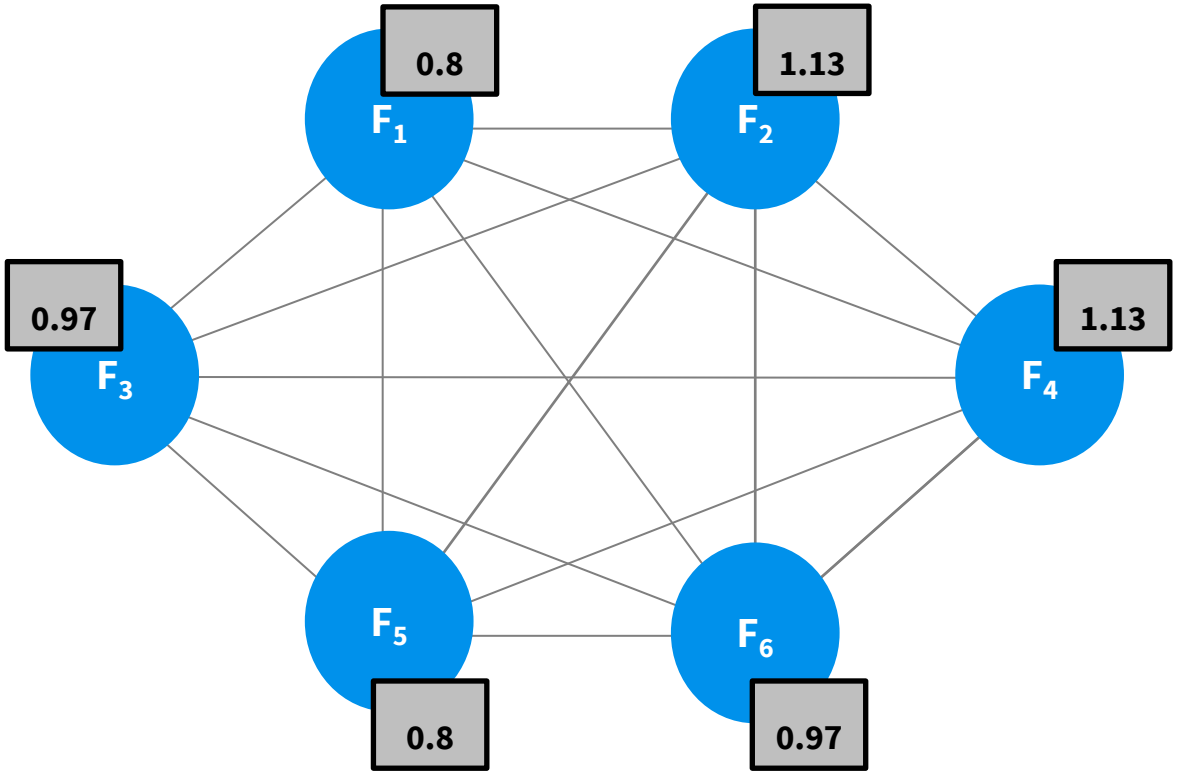
Feature	Visits
F_1	**0**
F_2	**2**
F_3	**1**
F_4	**2**
F_5	**0**
F_6	**1**

UFSACO – Components

- Graph representation for searching
- Heuristic desirability
- Working cycle
- Pheromone update
- **Feature selection**

UFSACO – Feature selection

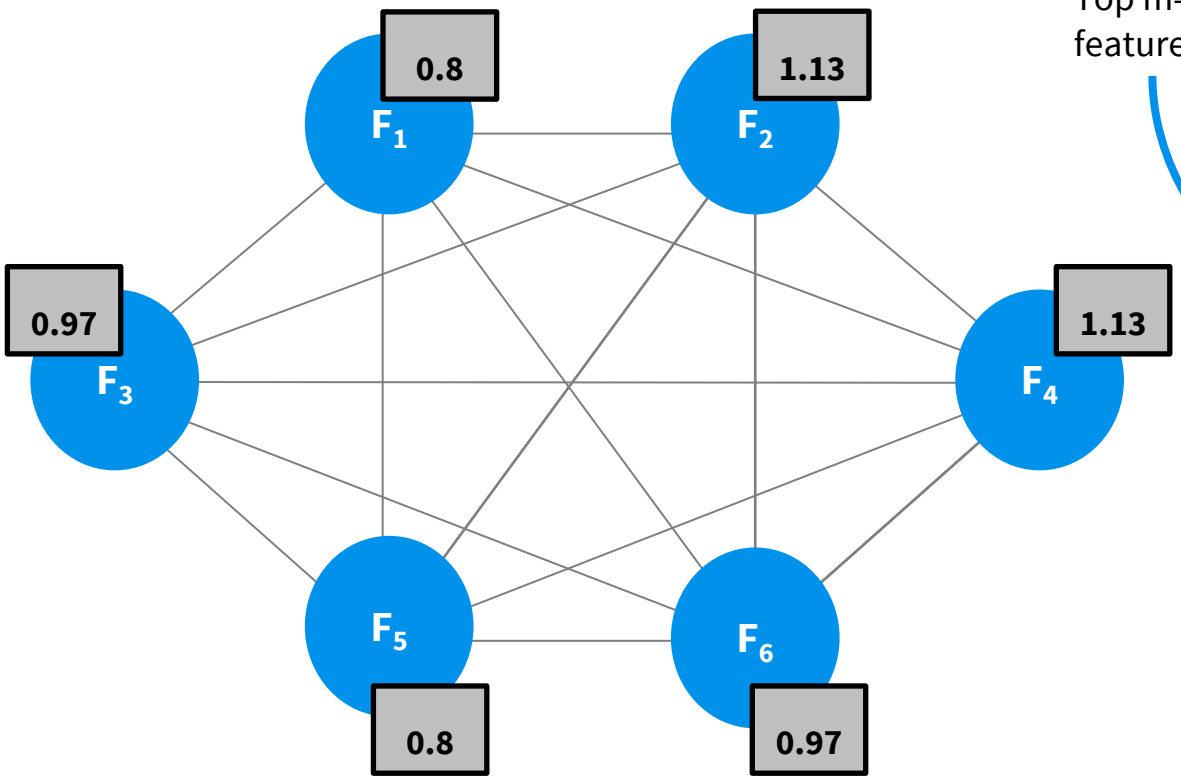
Step 1: sort features by pheromone value (decreasing order)



Feature	Pheromone
F_2	**1.13**
F_4	**1.13**
F_3	**0.97**
F_6	**0.97**
F_1	**0.8**
F_5	**0.8**

UFSACO – Feature selection

Step 2: select top *m* features to use for reduce original data



Top m=3 features

Feature	Pheromone
F_2	**1.13**
F_4	**1.13**
F_3	**0.97**
F_6	**0.97**
F_1	**0.8**
F_5	**0.8**

UFSACO – Algorithm (1)

Algorithm 1. *Unsupervised Feature Selection based on Ant Colony Optimization (UFSACO)*

Input: X: $p \times n$ matrix, n dimensional training set with p patterns.
m ($\leq n$): the number of features to keep for final reduced feature set.
NC_{max}: the maximum number of cycles that algorithm repeated.
NAnt: define the number of agents (number of ants).
NF: the number of features selected by each agent in each cycle.
ρ: define the decay rate of the pheromone on each feature.
@sim: function that computes the similarity between features.

Output: $\tilde{X}$: $p \times m$ matrix, reduced dimensional training set.

UFSACO – Algorithm (2)

Algorithm 1. *Unsupervised Feature Selection based on Ant Colony Optimization (UFSACO)*

begin algorithm
Apply @sim to compute the similarity $S_{i,j}$ between features, $\forall\ i, j = 1...n$.
$\tau_i(1) = c,\ \forall\ i = 1...n.$ /* initial pheromone – c is a constant parameter*/
for $t = 1$ to NC_{max} **do**
    $FC[i] = 0,\ \forall\ i = 1...n.$ /* set the initial features counter to zero */
    Place the agents randomly on the graph nodes.
    for $i = 1$ to NF **do**
        for $k = 1$ to $NAnt$ **do**
            Choose the next unvisited feature f according to (9) and (10). /* pseudo-random-proportional rule */
            Move the k-th agent to the new selected feature f.
            $FC[f] = FC[f] + 1;$ /* update feature counter associated with feature f*/
        end for
    end for
    $\tau_i(t+1) = (1-\rho)\,\tau_i(t) + \frac{FC[i]}{\sum_{j=1}^{n} FC[j]};\ \forall\ i = 1...n.$ /* global updating rule */
end for
Sort the features by decreasing order of their pheromones (τ_i).
Build $\tilde{X}$ from X by keeping the top m features with highest pheromone.
end algorithm

UFSACO – Demo project

https://gitlab.inf.unibz.it/aaronestrada/IntAgACOfeature
https://github.com/aaronestrada/IntAgACOfeature

UFSACO – Demo project
Dataset

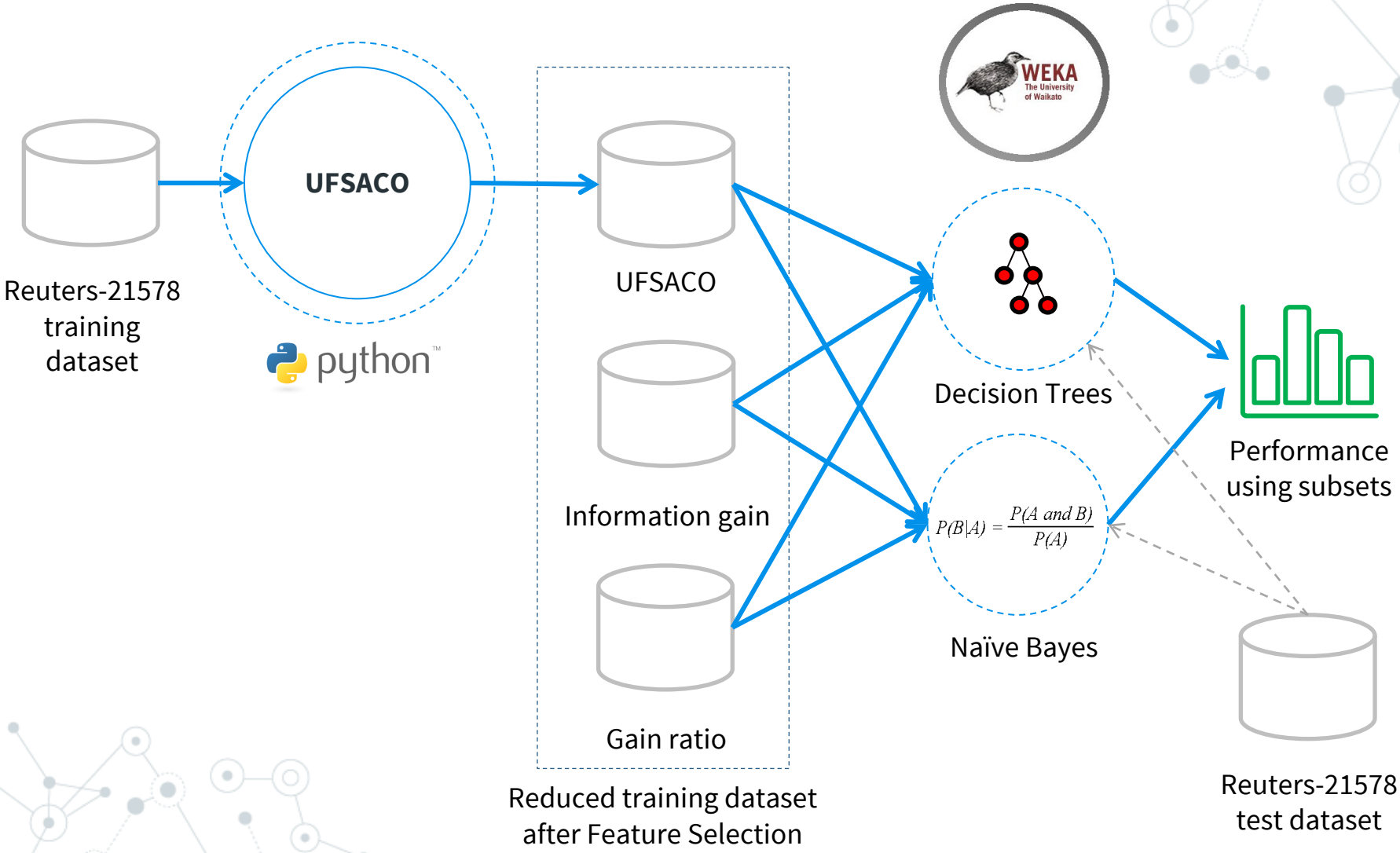
Reuters-21578 test collection for text categorization

	Training	Test
Documents	**6489**	**2545**
Features	**24087**	**15249**

Class labels and number of documents in training and test datasets

Class	Training	Test	Class	Training	Test
earn	**2848**	**1083**	acq	**1636**	**712**
wheat	**208**	**71**	grain	**83**	**38**
money-fx	**369**	**130**	interest	**342**	**131**
corn	**122**	**34**	crude	**389**	**189**
trade	**350**	**108**	ship	**142**	**49**

UFSACO – Demo project



UFSACO – Demo project
Evaluation - settings

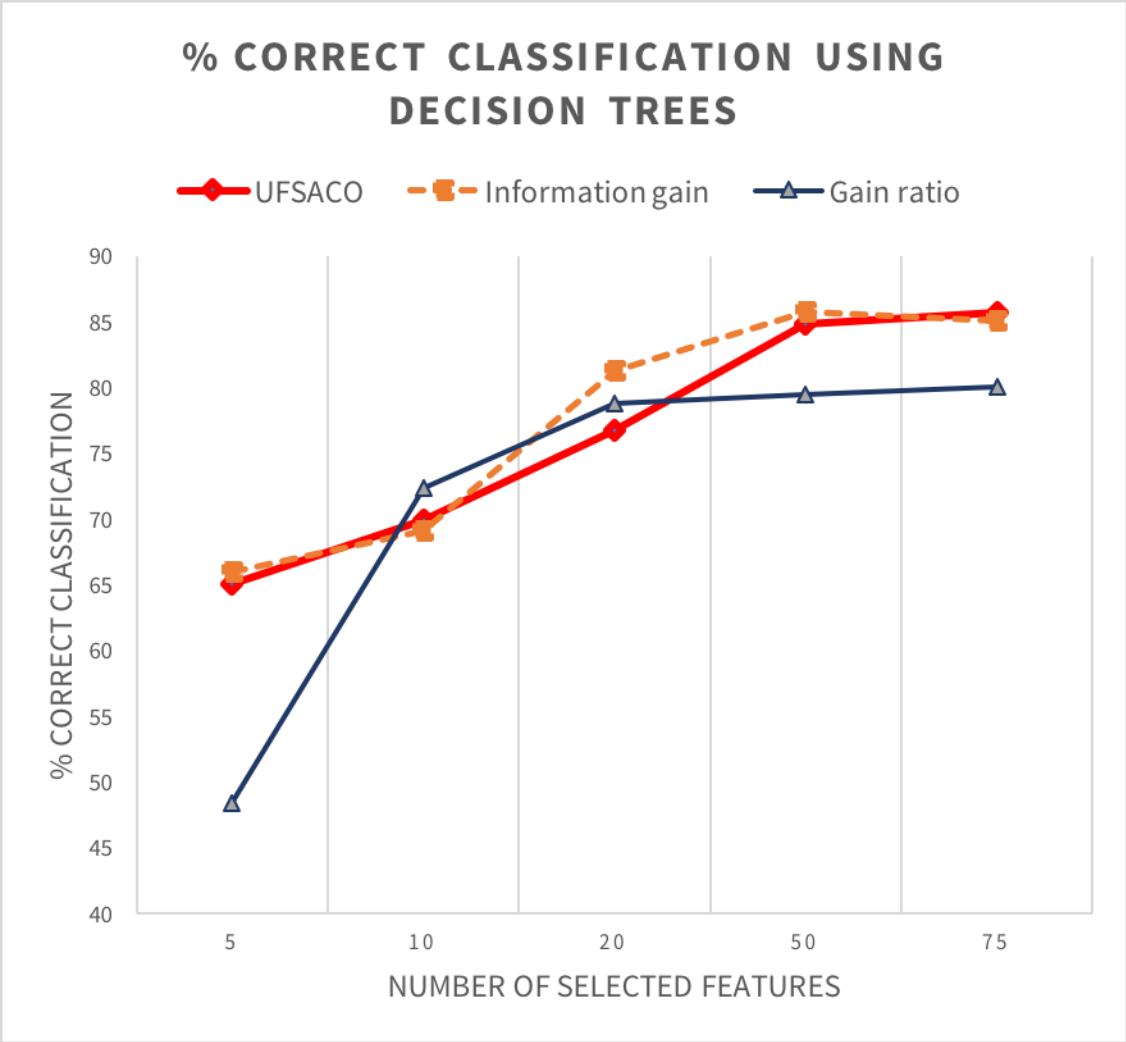
N_{ants}=100	(Number of ants)
NC=50	(Number of cycles)
$\rho = 0.2$	(Decay rate)
$\beta = 1$	
$\tau_i(t = 0) = 0.2$	(Initial pheromone)
q_0=0.7	(Exploration-exploitation coefficient)

UFSACO – Demo project
Evaluation – results

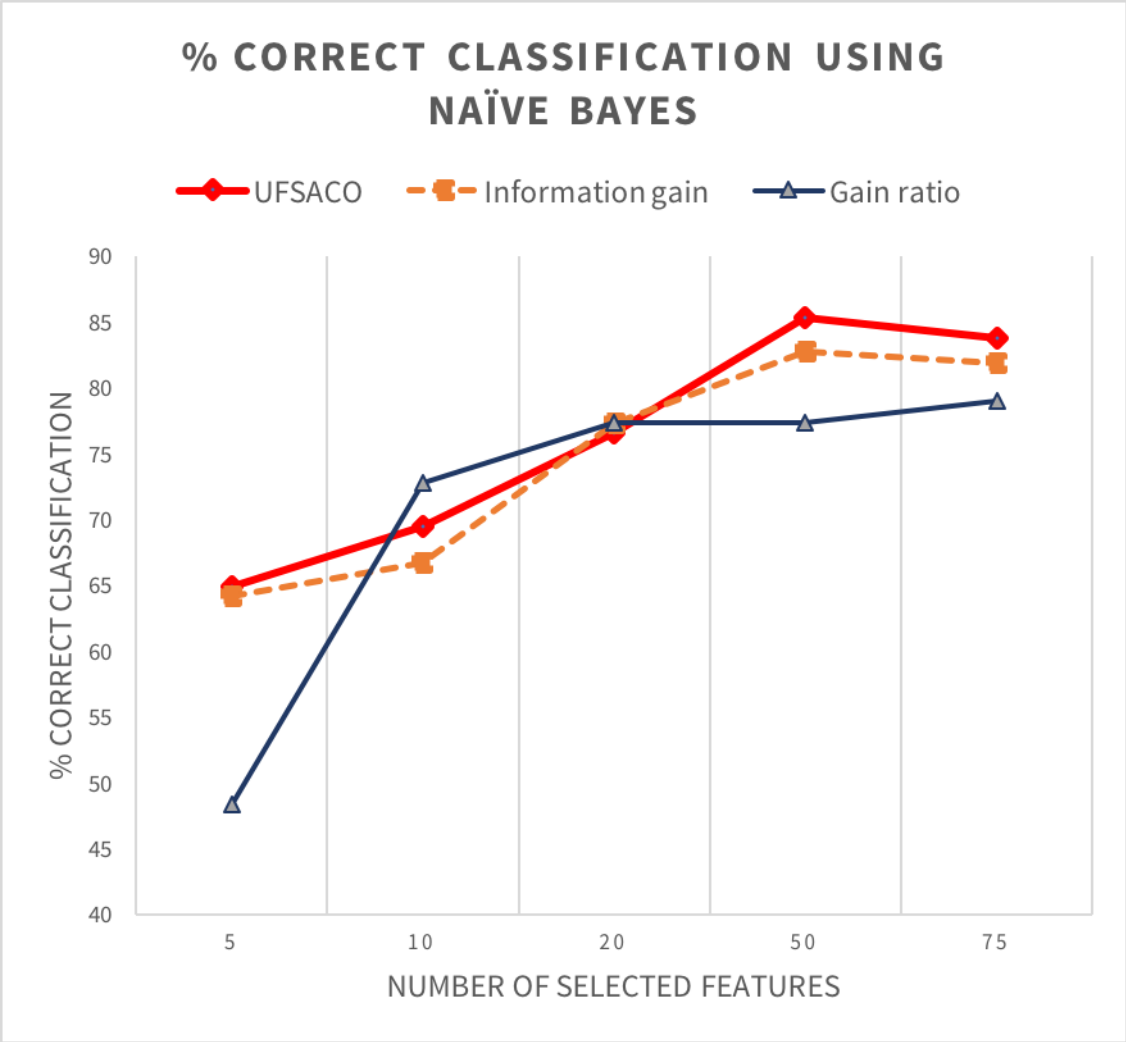
Classification results - % correct classification

Selected features	Classifier	UFSACO	Information gain	Gain ratio
5	DT	64,99	65,93	48,37
	NB	64,83	64,17	48,37
10	DT	69,93	69,12	72,38
	NB	69,43	66,64	72,77
20	DT	76,76	81,22	78,74
	NB	76,52	77,29	77,37
50	DT	84,85	85,74	79,41
	NB	85,35	82,71	77,33
75	DT	85,68	85,03	80,00
	NB	83,77	81,89	78,98

UFSACO – Demo project
Evaluation – results



UFSACO – Demo project
Evaluation – results



Thanks!

Any questions?

ruben.estrada@stud-inf.unibz.it
aaron.estrada.poggio@gmail.com