

Docker Edu2Com API User Guide

What is the Edu2Com

Edu2com is an algorithm for allocating teams to tasks, based on competencies and preferences. Building teams and match each one with some task is a very complicating job, especially as the number of individuals, and the number of task rise. Consider, for example, only 6 individuals and only 3 tasks, where for each task we need a team of 2, and no overlaps are allowed—that is, each individual can participate in at most one team, each team can work in at most one task, while each task can be assigned to at most one team. Even in this over simplified example we need to evaluate matching quality of 15 different teams against each task, and then choose a combination of 3 pairs $\langle team, task \rangle$ with no overlaps (in total there are 90 such combinations).

Competencies. So Edu2Com analyses the competencies required by the tasks, along with the ones acquired by the individuals, and builds teams of certain size (determined by the task) that are capable of undertaking the tasks. Within a team, each member is responsible for some competencies based on the task assigned to the team. Collectively the competence-responsibilities of all the members of the team must cover all the task's competence-requirements. that is for each required competence there is at least one member that is responsible for this competence. Similarly, all the members of the team must be actively participating, i.e., each member is responsible for at least one competence.

Preferences. Moreover, individuals may (or not) express their preferences over the tasks. Depending on the application domain, the individuals might have a saying on which task they prefer to work on. For example, in the domain of education we have the scenario where students need to form team in order to carry out some semester project, and each team needs to select a different project. In this case, the students express their interest over the different projects, and these preferences are taken into consideration when they are put into teams and assigned to a task. A similar case is when students are to be placed into internship programs.

Thus Edu2Com weighs both the competence-based matching quality of the team to the assigned task, and the individuals' preferences over the several tasks. Edu2Com provides a team allocation to tasks, which is of the following form:

```

{
  TaskID_1 : {IndividualID_1 , IndividualID_2 , IndividualID_3}
  TaskID_2 : {IndividualID_4 , IndividualID_5}
  TaskID_3 : {IndividualID_6 , IndividualID_7 , IndividualID_8 , IndividualID_9}
  TaskID_4 : {IndividualID_10}
}

```

The API

The screenshot shows the Swagger API interface for 'Team Formation'. The top navigation bar includes the Swagger logo, the URL '/static/swagger_api.json', and an 'Explore' button. The main content area displays the API title 'Team Formation' with a version indicator '1.0.0'. Below the title, there is a 'Schemes' dropdown menu set to 'HTTP'. The API endpoints are listed in a table-like format:

Method	Endpoint	Description
GET	/ontology/alreadyLoaded	List of already loaded ontologies
POST	/ontology/{OntologyType}	Load ontology
POST	/ontology/{OntologyType}/{OntologyName}	Load ontology
GET	/dataset/alreadyLoaded	List of already loaded dataset
POST	/dataset/{DataSetType}	Load dataset
POST	/team_formation	Team formation in one call. Post all data (students, projects, preferences), and get the allocation
POST	/team_formation/{OntologyID}/{StudentsFile}/{ProjectsFile}	Team formation in one call. Post all data (students, projects, preferences), and get the allocation
POST	/team_formation/{OntologyID}/{StudentsFile}/{ProjectsFile}/{PreferencesFile}	Team formation in one call. Post all data (students, projects, preferences), and get the allocation

The provided API allows the user to run the Edu2Com algorithm using their own data.

1. The user can choose the competence ontology used in Edu2Com; that is, our algorithm needs an ontology that relates the competencies with each other, or a similarity matrix that specifies the similarity between every two competencies (for more information see our accompanying paper [1]).
2. The user can run Edu2Com in two ways:
 - (a) “in one call”, provide all the necessary data when calling Edu2com; or
 - (b) provide all the necessary data before calling Edu2Com.

In the 2nd way the data are stored, and can be used in multiple calls of the algorithm; while in the 1st way nothing is stored. The individuals' preferences are 'optional', in both ways.

How to use the API

We use an example of education in the following data.

Upload Ontology

```
curl -X POST "http://0.0.0.0:5000/ontology/Graph" -H "accept: *application/json*" -H "Content-Type: application/json" -d "{ \"Graph\": { \"Root\": [ \"c1\", \"c2\" ], \"c1\": [ \"c3\", \"c4\", \"c5\" ], \"c2\": [ \"c4\", \"c6\" ] } }"
```

```
curl -X POST "http://0.0.0.0:5000/ontology/Graph" -H "accept: *application/json*" -H "Content-Type: application/json" -d "{ \"Precalc\": { \"c1-c1\": 1, \"c2-c2\": 1, \"c3-c3\": 1, \"c4-c4\": 1, \"c5-c5\": 1, \"c6-c6\": 1, \"c1-c2\": 0.7, \"c2-c1\": 0.7, \"c1-c3\": 0.3, \"c3-c1\": 0.3, \"c1-c4\": 0.8, \"c4-c1\": 0.8, \"c1-c5\": 0, \"c5-c1\": 0, \"c1-c6\": 0.1, \"c6-c1\": 0.1, \"c2-c3\": 0.4, \"c3-c2\": 0.4, \"c2-c4\": 0.6, \"c4-c2\": 0.6, \"c2-c5\": 0.2, \"c5-c2\": 0.2, \"c2-c6\": 0.4, \"c6-c2\": 0.4, \"c3-c4\": 0.7, \"c4-c3\": 0.7, \"c3-c5\": 0.9, \"c5-c3\": 0.9, \"c3-c6\": 0.65, \"c6-c3\": 0.65, \"c4-c5\": 0.8, \"c5-c4\": 0.8, \"c4-c6\": 0.4, \"c6-c4\": 0.4, \"c5-c6\": 0.75, \"c6-c5\": 0.75 } }"
```

Upload Data (a) uploading 2 students

```
curl -X POST "http://0.0.0.0:5000/dataset/Students" -H "accept: *application/json*" -H "Content-Type: application/json" -d "{
  \"Students\": {
    \"Ludovica\": { \"competences\": {
      \"http://data.europa.eu/esco/skill/33051fa7-bc25-4a75-80ae-c3b7cf5c744d\":[1,1],
      \"http://data.europa.eu/esco/skill/f441f55b-7ba0-4c5b-9ac2-1335a4d68813\":[1,1],
      \"http://data.europa.eu/esco/skill/473a8b64-a05c-4d49-962d-b9d9219950bc\":[1,1],
      \"http://data.europa.eu/esco/skill/098498a8-d57f-4ddd-b5bf-16e46475e768\":[1,1] },
      \"max_work_load\": null, \"name\": \"Ludovica\", \"school\":
        \"ISTITUTO TECNICO ECONOMICO E TECNOLOGICO 'F.e G.Fontana' - ROVERETO\"
    },
    \"Marco\": { \"competences\": {
      \"http://data.europa.eu/esco/skill/33051fa7-bc25-4a75-80ae-c3b7cf5c744d\":[1,1],
      \"http://data.europa.eu/esco/skill/f441f55b-7ba0-4c5b-9ac2-1335a4d68813\":[1,1],
      \"http://data.europa.eu/esco/skill/db77825e-0f3e-47d0-abdb-356794484272\":[1,1],
      \"http://data.europa.eu/esco/skill/da629999-d226-4ff3-9e1a-01788132664f\":[1,1],
      \"http://data.europa.eu/esco/skill/4211283b-617e-4ded-8636-4eed896c33b0\":[1,1],
      \"http://data.europa.eu/esco/skill/ee6630cd-f44d-4968-aa48-c179703fa012\":[1,1],
    }
  }
}
```

```

    \“http://data.europa.eu/esco/skill/94bc9278-dc2b-41e5-b337-5bda2c020818\”: [1,1] },
    \“max_work_load\”: null, \“name\”: \“Marco\”, \“school\”:
      \“ISTITUTO TECNICO ECONOMICO E TECNOLOGICO 'F.e G.Fontana' - ROVERETO\”
    }
  }
}”

```

(b) uploading 1 project

```

curl -X POST "http://0.0.0.0:5000/dataset/Projects" -H "accept: *application/json*" -H "Content-Type:
application/json" -d "{
  \“Projects\”: {
    \“GeometraRF\”: { \“competences\”: {
      \“http://data.europa.eu/esco/skill/473a8b64-a05c-4d49-962d-b9d9219950bc\”: [1,1],
      \“http://data.europa.eu/esco/skill/d1a2437d-7eee-4c13-9104-0bb3dd7b45d6\”: [1,1],
      \“http://data.europa.eu/esco/skill/d94d9eaa-073f-40bc-b111-5f6278e7d742\”: [1,1],
      \“http://data.europa.eu/esco/skill/748b5c82-bf2f-4a76-af21-9ef24667e6c8\”: [1,1],
      \“http://data.europa.eu/esco/skill/ee6630cd-f44d-4968-aa48-c179703fa012\”: [1,1],
      \“http://data.europa.eu/esco/skill/4211283b-617e-4ded-8636-4eed896c33b0\”: [1,1] },
      \“weeks\”: null, \“teamsize\”: 2, \“interview\”: false, \“description\”: null,
      \“institute\”: \“GEOMETRA ROBERTO FRANCHINI\”, \“researcher\”: null
    },
  }
}”

```

(c) uploading preferences

```

curl -X POST "http://0.0.0.0:5000/dataset/Preferences" -H "accept: *application/json*" -H "Content-
Type: application/json" -d "{ \“Preferences\”: { \“1\”: { \“sid\”: \“Ludovica\”, \“pid\”: \“GeometraRF\”,
\“value\”: 4 }, \“2\”: { \“sid\”: \“Marco\”, \“pid\”: \“GeometraRF\”, \“value\”: 3 } } }”

```

Running edu2com (a) using uploaded data (Ontology/Students/Projects/Preferences—optionally)

```

curl -X POST "http://0.0.0.0:5000/team_formation/Precalc-ESCO-demo/Students-ESCO-demo/Projects-
ESCO-demo" -H "accept: *application/json*"

```

```

curl -X POST "http://0.0.0.0:5000/team_formation/Precalc-ESCO-demo/Students-ESCO-demo/Projects-
ESCO-demo/Preferences-ESCO-demo" -H "accept: *application/json*"

```

Note that the preferences file *must be compliant* with the projects' and students' files; that is, the preferences must use project ids and student ids existing in the respective files. Moreover, the preferences *must be complete*, i.e., for each combination project_id, student_id must exist an entry.

(b) one-shot call

```

curl -X POST "http://0.0.0.0:5000/team_formation" -H "accept: *application/json*" -H "Content-Type:
application/json" -d "{
  \"Students\": {
    \"Ludovica\": { \"competences\": {
      \"http://data.europa.eu/esco/skill/33051fa7-bc25-4a75-80ae-c3b7cf5c744d\":[1,1],
      \"http://data.europa.eu/esco/skill/f441f55b-7ba0-4c5b-9ac2-1335a4d68813\":[1,1],
      \"http://data.europa.eu/esco/skill/473a8b64-a05c-4d49-962d-b9d9219950bc\":[1,1],
      \"http://data.europa.eu/esco/skill/098498a8-d57f-4ddd-b5bf-16e46475e768\":[1,1] },
      \"max_work_load\": null, \"name\": \"Ludovica\", \"school\":
        \"ISTITUTO TECNICO ECONOMICO E TECNOLOGICO 'F.e G.Fontana' - ROVERETO\"
    },
    \"Marco\": { \"competences\": {
      \"http://data.europa.eu/esco/skill/33051fa7-bc25-4a75-80ae-c3b7cf5c744d\":[1,1],
      \"http://data.europa.eu/esco/skill/f441f55b-7ba0-4c5b-9ac2-1335a4d68813\":[1,1],
      \"http://data.europa.eu/esco/skill/db77825e-0f3e-47d0-abdb-356794484272\":[1,1],
      \"http://data.europa.eu/esco/skill/da629999-d226-4ff3-9e1a-01788132664f\":[1,1],
      \"http://data.europa.eu/esco/skill/4211283b-617e-4ded-8636-4eed896c33b0\":[1,1],
      \"http://data.europa.eu/esco/skill/ee6630cd-f44d-4968-aa48-c179703fa012\":[1,1],
      \"http://data.europa.eu/esco/skill/94bc9278-dc2b-41e5-b337-5bda2c020818\":[1,1] },
      \"max_work_load\": null, \"name\": \"Marco\", \"school\":
        \"ISTITUTO TECNICO ECONOMICO E TECNOLOGICO 'F.e G.Fontana' - ROVERETO\"
    }
  },
  \"Projects\": {
    \"GeometraRF\": { \"competences\": {
      \"http://data.europa.eu/esco/skill/473a8b64-a05c-4d49-962d-b9d9219950bc\":[1,1],
      \"http://data.europa.eu/esco/skill/d1a2437d-7eee-4c13-9104-0bb3dd7b45d6\":[1,1],
      \"http://data.europa.eu/esco/skill/d94d9eaa-073f-40bc-b111-5f6278e7d742\":[1,1],
      \"http://data.europa.eu/esco/skill/748b5c82-bf2f-4a76-af21-9ef24667e6c8\":[1,1],
      \"http://data.europa.eu/esco/skill/ee6630cd-f44d-4968-aa48-c179703fa012\":[1,1],
      \"http://data.europa.eu/esco/skill/4211283b-617e-4ded-8636-4eed896c33b0\":[1,1] },
      \"weeks\": null, \"teamsize\": 2, \"interview\": false, \"description\": null,
      \"institute\": \"GEOMETRA ROBERTO FRANCHINI\", \"researcher\": null
    },
    },
  \"Preferences\": {
    \"1\": { \"sid\": \"Ludovica\", \"pid\": \"GeometraRF\", \"value\": 4 },
    \"2\": { \"sid\": \"Marco\", \"pid\": \"GeometraRF\", \"value\": 3 }
  },
  \"OntologyID\": \"Precalc-ESCO-demo\"
}"

```

The user can replace the preferences in the one-shot call with `\“Preferences\”: {}`, in case they don't want to consider the preference profiles during the team formation.

Running the docker

Download the docker from the AI4EU Catalog. Then navigate in the folder you downloaded the docker, and run:

```
$ docker load -i teamformation.tar.gz
$ docker run -p 5000:5000 --name teamformation teamformation:latest
```

Access the API at the page localhost:5000/AI4EU/teamformationui

The user is kindly requested to check the sample data files existing in the docker through the command:

`curl -X GET "http://0.0.0.0:5000/dataset/alreadyLoaded" -H "accept: *application/json*" before using the API.`

References

- [1] Athina Georgara, Carles Sierra, and Juan A. Rodríguez-Aguilar. Taip: an anytime algorithm for allocating student teams to internship programs, 2020.