X Modal
X Cultural
X Lingual
X Domain
X Site
Global OER Network

Grant Agreement Number: 761758
Project Acronym: X5GON
Project title: X5gon: Cross Modal, Cross Cultural, Cross Lingual, Cross Domain, and Cross Site Global OER Network
Project Date: 2017-09-01 to 2020-08-31
Project Duration: 36 months
Document Title: D4.4 – Final prototype of recommendation engine
Author(s): Jasna Urbančič, Erik Novak (JSI)
Contributing partners: JSI, NA
Date:
Approved by:
Type: P
Status: Final
Contact: Erik Novak

Dissemination Level		
PU	Public	x
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Copyright - This document has been produced under the EC Horizon2020 Grant Agreement H2020-ICT-2014 /H2020-ICT-2016-2-761758. This document and its contents remain the property of the beneficiaries of the X5GON Consortium

Revision

Date	Lead Author(s)	Comments
21/06/2019	Jasna Urbančič	Initial draft
17/07/2019	Victor Connes, Walid Ben Romdhane	Added contributions from Univerité de Nantes
27/08/2019	Colin de la Higuera	Draft review
30/08/2019	Jasna Urbančič, Erik Novak	Final version

TABLE OF CONTENTS

Table of Contents ... 3

List of Figures ... 4

Abbreviations ... 5

Abstract ... 6

1. Introduction ... 7

2. OER material and user activity data ... 8

2.1. Recent updates in the data ... 8

2.2. Enriching the data with the results of other work packages ... 11

3. Recommendation engine ... 13

3.1. Content-based recommendation engine ... 13

3.2. Collaborative filtering ... 13

3.3. User-Item similarity-based recommendations ... 14

4. Evaluation ... 15

5. Conclusion ... 16

References ... 17

LIST OF FIGURES

Figure 1: Distribution of webpages with embedded OER materials per repository..... 8
Figure 2: Distribution of OER materials throughout the repositories. 9
Figure 3: Distribution of materials per type. the most common type group is text. 9
Figure 4: Distribution of materials per language. Most of the materials are in English. 10
Figure 5: The distribution of user views per repository.. 10
Figure 6: Number of users per number of views in log-log scale............................. 11
Figure 7: The PostgresQL query for performing the collaborative filtering method... 14

ABBREVIATIONS

Abbreviation	Definition
OER	Open Educational Resource
WP	Work Package
Y1	Year 1
Y2	Year 2
Y3	Year 3

ABSTRACT

The document presents a report on the final prototype of the recommender engine. The report consists of the newest updates to the platform database and its statistics. In addition, it presents the new methods of the recommendation engine – the bundle recommendations and collaborative filtering – followed by a brief evaluation of the recommender engine.

1. INTRODUCTION

In this document we report on the final prototype of the recommender engine. The recommender engine consists of methods that are both material- and user-based. The material-based methods have already been developed in Y1 and presented in deliverable 4.3 – Early prototype of Recommender Engine. In Y2, we have extended the methods by providing recommendations of webpages (bundles) which are represented as an aggregate of the materials that are located on the webpage.

In addition, user-based models based on the collaborative filtering algorithm were implemented and deployed on the production machine. We have considered various approaches to collaborative filtering, but found that because of the size of the user and material data the platform collected, these matrix factorization/based approaches are not feasible at the moment. We opted for “others also bought/seen that” approach using database queries.

An update on the database statistics is also presented. It describes the amount of data and its distribution based on different dimensions.

The remainder of the documents is as follows. Section 2 describes the current state of the platform database. It provides the data distribution in different dimensions. Next, Section 3 presents the new additions to the recommender engine: the bundle and a description on the collaborative learning-based model. A brief evaluation of the recommender engine is presented in Section 4. Finally, the document is concluded in Section 5.

2. OER MATERIAL AND USER ACTIVITY DATA

We identify two types of data that are useful for the user model development: the OER material metadata and the user activity data. As the data is used and described in other works and deliverables, we direct the user towards the following deliverables for the following:

- *Deliverable 2.2 – Final Server Side Platform* for the data acquisition process,
- *Deliverable 4.1 – Initial Prototype of User Modelling Architecture* for detailed description of user activity data and how we use the data, and
- *Deliverable 4.3 – Early prototype of recommendation engine* for explanation how we combine both types of data.

In this section, we provide the recent updates associated with the acquired data and provide the statistics provided by the X5GON platform, as well as the descriptions of enrichment processes developed in other work packages.

2.1. Recent updates in the data

In the time of the writing of the deliverable we have processed approximately 97k OER webpages from 7 different OER repositories. The target repositories are:

- VideoLectures.NET,
- University of Bologna Digital Library,
- Universitat Politecnica de Valencia,
- MIT OpenCourseWare,
- Univerza v Mariboru,
- University of Osnabruck, and
- Nantes University.

The distribution of webpages per repository is displayed in Figure 1. Most processed webpages are from Videolectures.NET, following by the Universitat Politecnica de Valencia and eUčbeniki.

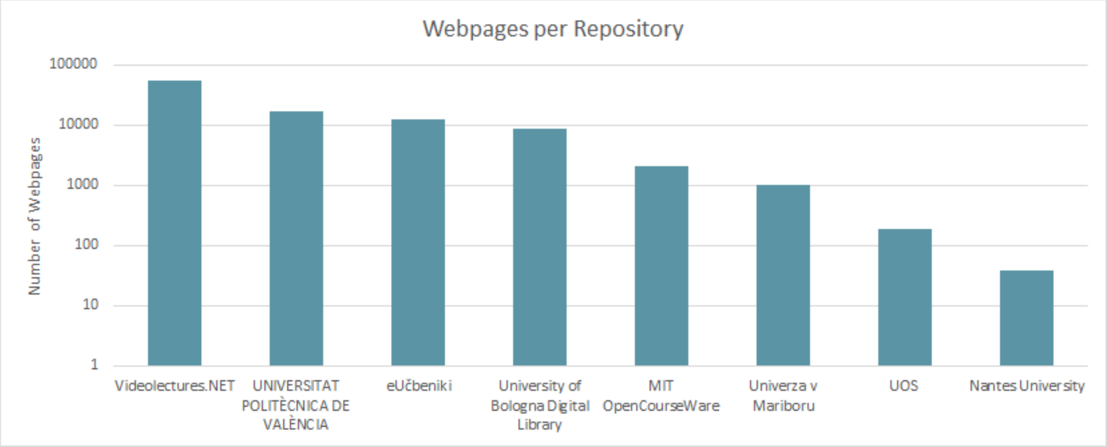

Figure 1: *Distribution of webpages with embedded OER materials per repository.*

These processed webpages contain at least one OER material – accumulated more than 90k OER materials. The number of acquired OER materials per repository is displayed in

These webpages contain more than 90k OER materials. Each of these websites has embedded at least one file containing an OER. The number of files per repository in displayed in Figure 2. Due to its structure, e.g. the webpage containing a whole course, the MIT OpenCourseWare repository offers the largest selection of OER materials – even though it has a small number of webpages. Other OER repositories mostly have fewer OER materials included per webpage.

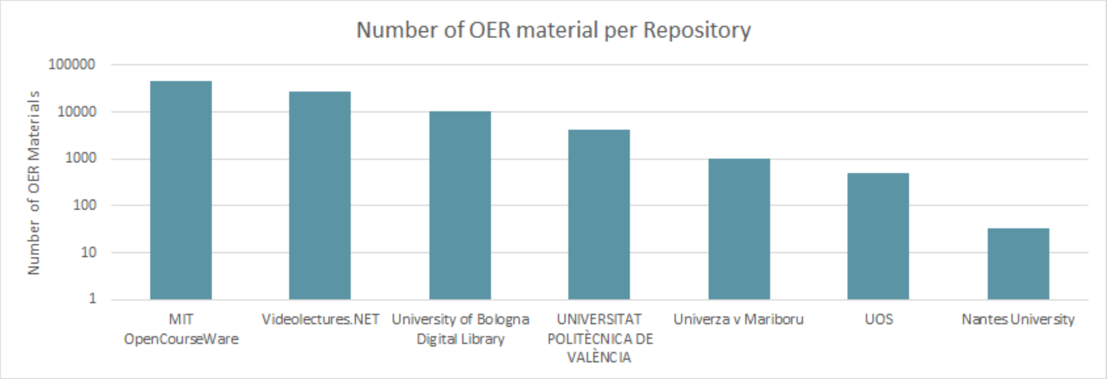

Figure 2: *Distribution of OER materials throughout the repositories.*

We differentiate between OER materials and webpages because the URLs we receive through the user activity acquisition process point to the webpage and not materials. Such differentiation is necessary because of the user modelling and personalized recommendation models, which are based on the user activity data.

The acquired OER materials are found in different modalities. The distribution of OER materials per type is pictured in Figure 3. The most common document type is text with pdfs, followed by video in mp4.

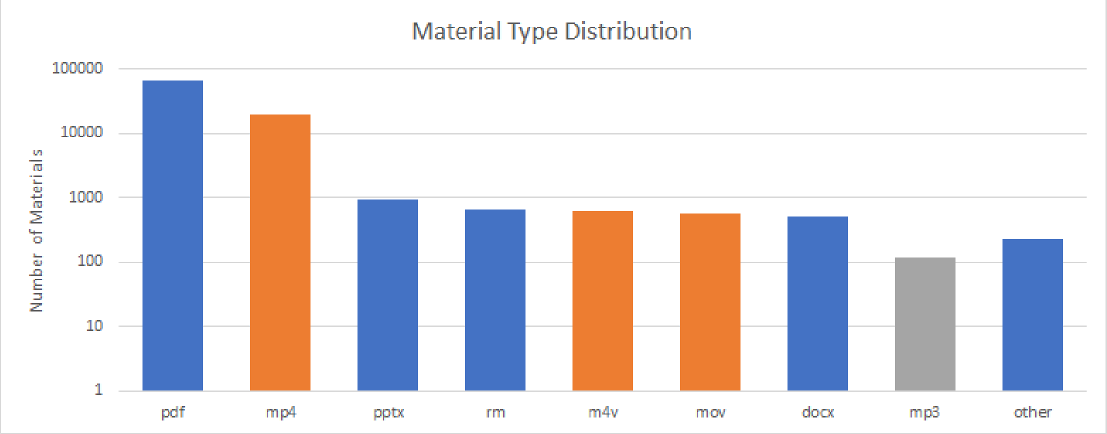

Figure 3: *Distribution of materials per type. the most common type group is text.*

In addition, the materials are in different languages – the distribution of materials per language is shown in Figure 4. The majority language is English as both MIT OpenCourseWare and VideoLectures.NET contain a large number of materials that are in English. The Italian, Spanish, Slovene, and German languages also have a noticeable presence, whereas other languages are underrepresented.

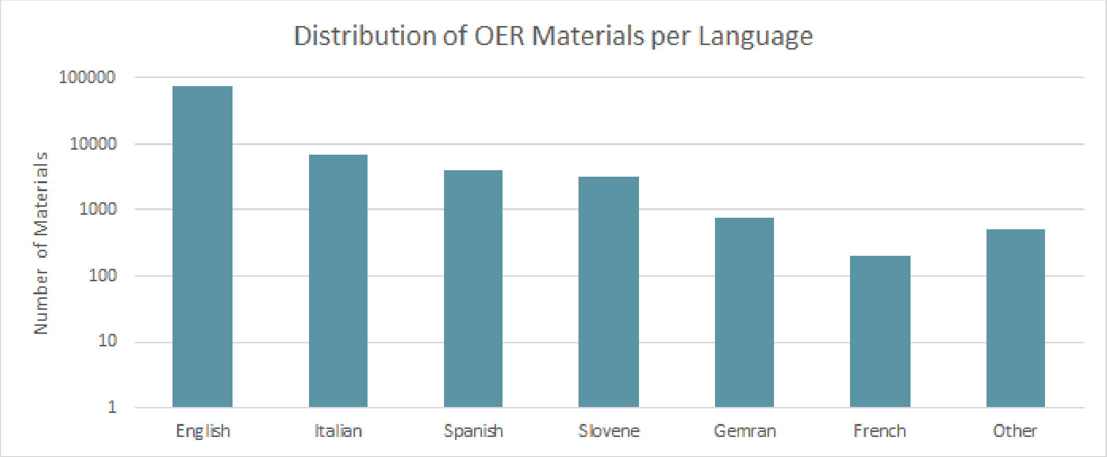

Figure 4: *Distribution of materials per language. Most of the materials are in English.*

We have also made significant progress in the amount of user activity data collected. The technology used to acquire the user activity data – the Connect Service – has been included in the following OER repositories:

- VideoLectures.NET,
- Universitat Politecnica de Valencia,
- eUčbeniki,
- Nantes University, and
- Universitat Osnabrueck.

So far, we recorded more than 1.4M user visit data within the OER network. Among the acquired visit data, almost all activities were produced by more than 375k users that gave an active consent to store a cookie in their browser. We exclude the users that do not allow tracking from further analysis.

The distribution of user views per repository is displayed in Figure 5. Most visits in the X5GON network comes from the Videolectures.NET repository, which is expected as it is one of the most well-known repositories in our network and was also the first repository have integrated the Connect Service.

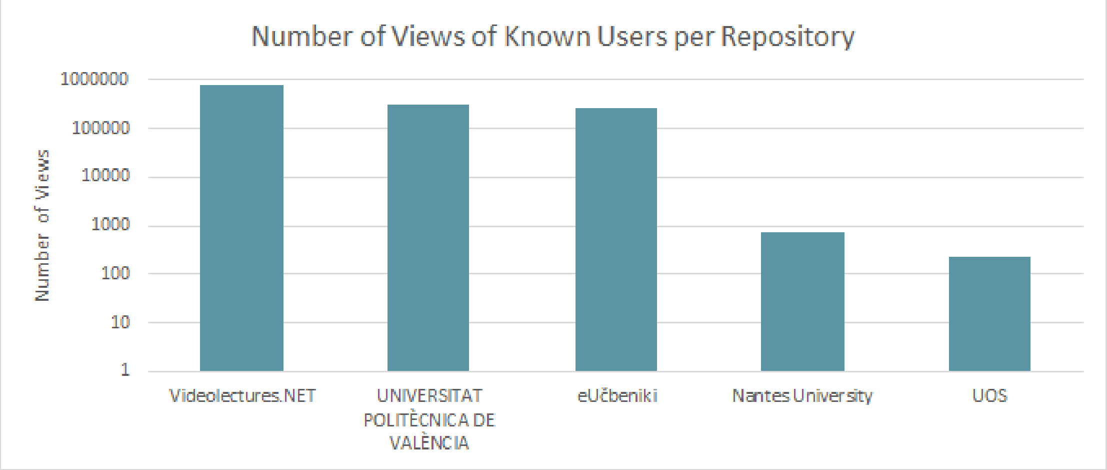

Figure 5: *The distribution of user views per repository.*

The distribution of users per number of views is shown in Figure 6. Most users (more than 220k) access only one material, whereas the most enthusiastic user viewed more than 5000 materials since the beginning of the user activity acquisition process. On average users access just a little over two materials. These two findings pose a great challenge when extracting learning pathways or monitoring users' progress over a period of time.

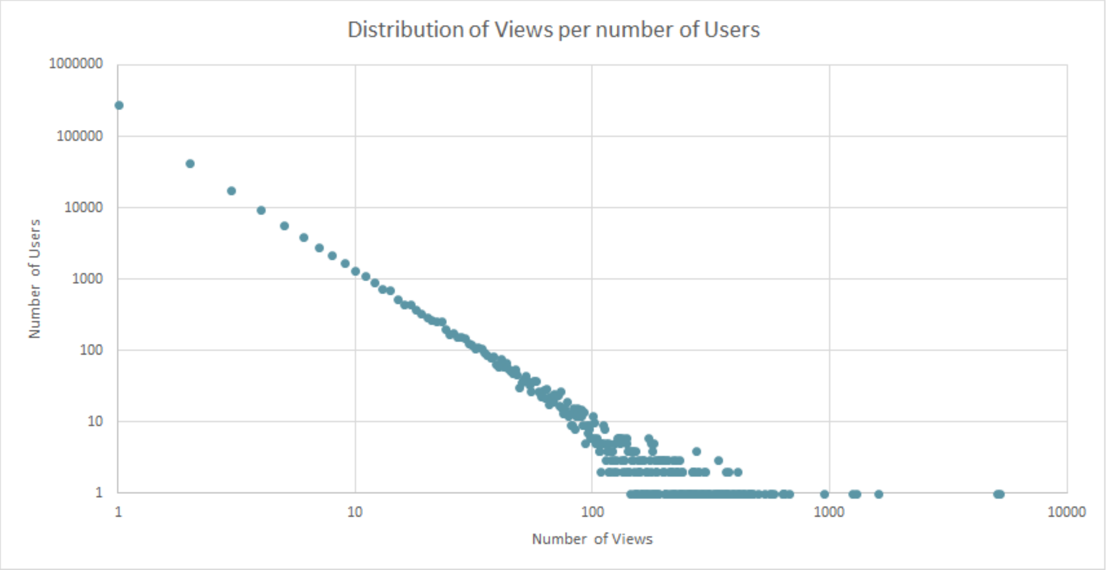

Figure 6: *Number of users per number of views in log-log scale.*

2.2. ENRICHING THE DATA WITH THE RESULTS OF OTHER WORK PACKAGES

In Y2 we were also working towards enriching the data with the results of other work packages, more specifically WP1 and WP3 as we already use the results of WP5 for transcription and translation of OER materials. We will initially use these enrichments as additional information for the users to help their decision-making process.

Difficulty. Recommendation in the context of educational resources raises questions that are quite different from those in a commercial context.

One central question is the hardness of a resources. This question can be treated in two main ways, difficulty and complexity. While difficulty is relating to the relative hardness in a specific field and in comparison, with other resources, complexity can be defined in a more objective way through text properties such as lexicon and grammar, and is an inherent evaluation on the hardness of the resources.

For this year we propose one complexity metric which measures the resource hardness based on lexicon and grammar properties. An implementation for this approach is available in the WP3 API through the service *wikification2conpersec*.

Further details on the metric can be found in the Learning Analytic Engine 2.0 (D3.2) deliverable.

Order. One of the learning approaches is to have a logical order during the learning process when consuming the educative resources. So, this second model try to evaluate a pair of resources and give a relative order to consume these resources basing on 'continuous Wikifier model'. The main idea of this method, is to catch the common concepts between the resources, and to use their distribution over the resources to infer the order.

More precisely, we assume the keys concepts to predict the order are those, which are present in the end a resource and in the beginning of the other. From our observation these patterns correspond to a concept introduction, and our goal is to choose the order which maximises this kind of transition in order to have a fluid transition between the resources and to introduce as many prerequisites as possible.

Implementation for this approach is available in the WP3 API through the service *continuouswikification2order*.

Further details on the metric can be found in the deliverable *3.2 - Learning Analytic Engine 2.0*.

3. RECOMMENDATION ENGINE

This section is dedicated to the description of the current state of the recommender engine. In Y2 we worked towards enriching the recommender engine with additional features, different approaches, and functionalities to support personalization and improve the usability of the content-based recommender. We added two new approaches for personalized recommendations: a) collaborative filtering and b) user-item similarity.

Additionally, we improved the documentation of the recommender engine. The documentation with the examples is available in [1].

3.1. CONTENT-BASED RECOMMENDATION ENGINE

In Y1 we developed a content-based recommender that allows the users to search for relevant materials based on either the text query input or the URL of an existing material. In Y2 we have extended the previously developed recommender to support content-based recommendations on OER webpages – which we defined as *bundles*.

The material bundles are an aggregate of the materials that is located to its corresponding webpage. In other words, if a webpage w contains materials $R_{w1}, R_{w2}, \dots, R_{wn}$, then the material bundle is defined as $c(w) = \frac{1}{n}\sum_{k=1}^{n} c(R_{wk})$, where $c(R_{wk})$ is the material embedding into the Wikipedia concepts semantic space. This approach is similar to user modelling described in *D4.2 – Final prototype of user modelling architecture*.

To recommend bundles we find the nearest neighbours of the query webpage (currently, only the URLs of the existing webpages are supported as queries) in the semantic space of Wikipedia concepts. Afterwards, we provide their metadata back to the user.

3.2. COLLABORATIVE FILTERING

Collaborative filtering [1] is a method which provides recommendations to a user by finding other users that have similar preferences (or taste) and return items that the similar users labelled as useful (or have a high rating). In our case, with collaborative filtering we recommend a list of materials that were previously viewed by users with similar preferences as the user in question. We sort the recommended materials based on the number of users viewed the associated material. More specifically, the recommendations provided by the collaborative filtering are generated in the following steps:

1. User *A* views an OER material
2. We search for other users that have also viewed the same materials as user *A* – the set of found users we call *similarUsers*
3. We filter the OER material that have been viewed by at least one user from the *similarUsers* set and count how many times each material was viewed – the set of (*material, count*) pairs is called *candidateMaterials*
4. From the *candidateMaterials* set we remove materials that were already viewed by user *A*
5. We sort the *candidateMaterials* set based on the *count* value and return the list to user *A*

Often collaborative filtering is associated with matrix factorization of user-item rating matrix. In our case, the user-material matrix at position i, j would have a value 1 if the user i has accessed the material j, and 0 otherwise. Afterwards, the matrix would be split into submatrices with one of the factorization methods – such as the non-negative matrix factorization method. However, since the user-material matrix generated with the data collected by the platform contains round 390M values (and would grow with additional users and materials), the factorization is not a feasible approach at the moment. In the future, we could dedicate effort in researching for an approach which would make it feasible.

Figure 7 shows the query used to perform the collaborative filtering to a particular user.

```
WITH url_count AS (
        SELECT url_id, COUNT(url_id) AS count FROM user_activities WHERE cookie_id IN (
                SELECT cookie_id FROM user_activities WHERE cookie_id<>1 AND cookie_id NOT
                IN (
                        SELECT id FROM cookies WHERE uuid = '${userQuery.uuid}') AND
                url_id IN (
                        SELECT url_id FROM user_activities WHERE cookie_id IN (
                                SELECT id FROM cookies WHERE uuid = '${userQuery.uuid}')
                        )
                )
        GROUP BY url_id ORDER BY count DESC)
SELECT url_count.*, rsmm.* FROM url_count, rec_sys_material_model AS rsmm WHERE
url_count.url_id = rsmm.url_id ORDER BY count DESC LIMIT ${count};
```

Figure 7: *The PostgresQL query for performing the collaborative filtering method.*

3.3. USER-ITEM SIMILARITY-BASED RECOMMENDATIONS

In deliverable *D4.2 – Final user modelling architecture prototype* we reported on various approaches to user modelling – we present the approach of embedding the user in the materials' semantic space. In addition, we present an approach of user modelling which includes the temporal component to increase the presence of the most recent materials the user viewed.

For user-material similarity-based recommendations we use the presented user models to retrieve relevant materials that the user has not yet seen based on their historic interest. We use the k-nearest neighbors method to find the closest materials in the material semantic space and provide these materials as the recommendations.

4. EVALUATION

The current version of the recommendation engine has been evaluated and described in deliverable 5.2 – Second report on piloting. What follows is a brief description of our findings.

The initial findings suggest that the users tend to choose the material that is ranked with an average of 8.89 in the recommended list. Additionally, the preliminary results show that the user prefers to stay on the domain of the source OER material provider when using the recommender plugin (more on the plugin is found in deliverable 4.5 – prototype of cross-site recommendation engine). In addition, the user most often selects the material in the same language and modality as the material which they are viewing at the moment.

Further analysis of the recommendation engine will be performed in the upcoming months and until the end of the project. The analysis results will then be used to improve both the system and the user experience.

5. CONCLUSION

In this document we presented the current state of the material and user activity data, as well as content- and user-based recommendation approaches. The database is slowly and consistently increasing, providing more information about the material consumption – which will then be used in the recommendation engine development. In addition, other enrichment approaches were developed: the hardness level provides a measure for assessing the difficulty level of the material, while the order assesses the order in which two materials should be based on their content.

The recommender engine now contains additional methodology for providing content- and user-based recommendations. We have introduced the concept of the bundle which is an aggregate of OER materials that are found on the same webpage. The user-based approach leverages the collaborative filtering algorithm – recommending materials to the user based on the historic viewings of other users who have similar preferences as the target user. Matrix factorization methods were considered but due to the quantity of the data we omitted the idea.

In the future, we will continue the recommender engine analysis – finding insight into the data and the engine, which will help us understand and develop a better recommender engine.

REFERENCES

[1] “Collaborative filtering - Wikipedia,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Collaborative_filtering. [Accessed 21 03 2018].

[2] Jožef Stefan Institute, “API Documentation | X5GON Platform,” 13 August 2019. [Online]. Available: https://platform.x5gon.org/documentation#recommender-rest-api.