# E-Health Methods & Applications: Project report Part IIA GROUP 4

**Contents**

## PHASE I

## Retrieve Google-Playstore.csv Dataset from Kaggle

We decided to work on a Google-PlayStore database provided by a Kaggle repository. It was the most detailed one. Moreover, the primary key of the database is the appId. This attribute is used also with the google-play-scraper library.

## Filtering the Applications by Relevant Educational Features

We chose those parameters based on our assumptions of what a “real” serious game should be. We considered the feedbacks provided by the users of the app as the most reliable definition of “how good is an application”._Thus, the filter selects an application if:

- It belongs to one of the educational categories ("Education", "Educational", "Family", "Learn", "4-year-old kids" and "4-year-old").
- Its rating is greater or equal than 4.
- Its rating counts are greater than 20 000.

We chose the categories that are semantically related to education and children. The minimum rating of 4.0 represents most of the applications, since the average of good applications is far above this value.

Indeed, if a game has a rating of 5/5 but only has a few ratings, the rating may not be very representative of the quality of the application. We agreed on rating counts greater than 20 000 according to our data source and the number of applications in the final dataset.

We found that other attributes were not as reliable as we thought they could be. For example, the number of installs sometimes does not reflect how famous or not is an app. Some apps could be installed by default or as a dependency.

## Filtering the Games

We noticed that each application has a specific sub-category. It always starts with the string "*GAME_*" in case of a game. We checked the subcategory of each educational app with google-play-scraper to remove all those apps that were not games. Furthermore, since our data source is an old dump of the real Google-PlayStore, some apps could not be found with google-play-scraper because they were deleted or renamed. We chose to select and remove them from our dataset.

## Enriching the Dataset with Descriptions and Reviews

Our dataset presents just the educational games with valuable features. Nevertheless, some important attributes are missing. Thus, we have implemented a first function that enriches our database with the description and the number of reviews of each educational game, using google-play-scraper.

We think that the description can be very useful to understand the purpose of a game and to judge how serious it is. Moreover, the description will be used in the next steps to extract other important parameters from the games. The number of reviews can be used in conjunction with the rating counts to analyze the trustworthiness of the rating of an application.

## Using Natural Language Processing to Identify Learning Category and Age Range

We wanted to classify the different games according to their learning categories and age ranges. Since these parameters could not be found in the initial dataset nor with google-play-scraper, we decided to use Natural Language Processing. For each parameter, we first defined specific categories: ["science", "counting", "language", "creativity", "shape", "food", "music" and "sport"] for the learning categories and ["babies", "children", "adolescents", adults"] for the age ranges.

These categories have been selected to precisely separate the games. Then, we wrote a list of associated keywords by looking at synonyms or semantically related words for each one of these categories. From those lists of keywords, we wrote a function that counts the maximum number of keywords found in the applications' descriptions for each category, to assign the closest category to all of them.

## Export the Final Dataset

The original data source has been shrunk and enriched with new information. The new dataset was exported into a new csv file.

## Benchmark

We chose 120 applications from the Google-PlayStore:

- 40 games judged as serious by the members of the group (human beings).
- 40 random misleading non-serious games that may deceive our filters.
- 40 random applications.

This set of application was used to benchmark the preciseness of our algorithm. The evaluated statistical parameters are:

- *Accuracy = 92.5%*: the ability to distinguish between a serious game and non-serious one or any other application.
- *Sensitivity = 77.5%*: the ability to recognize a serious game.
- *Specificity = 100.0%*: the ability to recognize a non-serious game.

We developed another parameter to test the accuracy, named *Effective Accuracy*. This parameter measures the ability to focus on serious games. The algorithm is penalized anytime that a mistake is made, such as when a serious game is not present, a non-serious game is present, or a random application is present. We found a *relative accuracy* of 85.0%.

## PHASE II.A

## Building Dataset with PubMed Papers

Our purpose is to clinically evaluate the serious games of our dataset according to scientific publications. Thus, we searched papers on PubMed related to all our applications using PyMed library. For each paper, we took into account the title, the abstract, the methods, the conclusions, the results and the associated keywords. We created a new dataset combining the applications' names, their learning categories and their papers.

## Reliability of PubMed papers

In order to measure the reliability of these papers, we used the Natural Language Processing method. We looked for keywords related to specific study types such as observational studies, systematic reviews, meta-analyses, etc. Then, we assigned a score to each paper according to the associated study type. Keywords associated to meta-analyses, systematic reviews and RCTs give a higher score than keywords linked to observational studies. Once the scores are computed, we performed a binary classification. We defined a validation threshold to determine whether a game is considered as *validated*. This threshold coincides with the degree of reliability of an observational study.

## Building a Dataset of Similar Validated Applications per Non-Validated Serious Game

We wanted to provide other validated applications to any user interested into any non-validated game. Therefore, we implemented a function that relates each non-validated application with a list of validated serious games from the same learning category. We chose the learning category as a feature that can subsequently relate clinically validated games with serious ones that do not have scientific validation on PubMed.

## Benchmark for Game Validation

We want to compare the clinical validation of a serious game found by our algorithm with the one found by a person. To begin with, we found 35 mobile games mentioned on PubMed:

- 30 serious validated games
- 5 serious non-validated games

This set of serious games was used to evaluate the performance of our algorithm to analyze the scientific evidence. We based the evaluation on the following statistical parameters, and the results are given below:

- *Accuracy = 44.8%*: the ability to distinguish a validated serious game from a non-validated one.
- *Sensitivity = 86.2%*: the ability to recognize a validated serious game.
- *Specificity = 3.4%:* the ability to recognize a non-validated serious game.

## Benchmark for Study Type Validation

We wanted to compare the study type found by our algorithm to the one found by a person. Therefore, we manually found 10 papers per study type on PubMed, using PubMed filters. Each paper is manually associated to a score, according to its study type. This score will be compared to the score found by our algorithm. For this multi-classification problem, we based our evaluation on the following metrics:

- *Accuracy = 43.2%*: percentage of correctly classified study types.
- *Mean Absolute Error = 26.8%*: mean scores' differences.

## Dashboard

To display all the above-mentioned information, we chose to create a dashboard, using Dash library. The dashboard answers the following questions:

- Given a learning category and an age range, what are the corresponding serious games available on the market?
- Given a specific application: what are the main details about it? How many papers are associated to this application? Even if it has been the subject of clinical studies, what is its level of validation, and can we consider it as "validated"?
- Given a specific paper: what is the main information about it? What is its level of validation, according to the type of study it refers to?
- Given a specific application whose level of validation is insufficient: is there any similar scientifically validated application? If yes, what are their names?

The dashboard is divided into four sections (tabs): field overview, details per application, papers per application and similar applications per non-validated applications.

The first tab allows the user to have a visual summary of the number of applications per learning category and age range, thanks to bar graphs. Then, using the dropdown menus, they can select a specific learning category and age range to display the associated list of serious games. This list contains general information about these applications, such as their rating, the number of ratings, their price, etc.

The second tab enables him to display more detailed information about all the applications: they can read their description, the number of installs, the date of the last update, etc. By selecting one specifically, they can see the list of associated papers and their information. They can visualize its level of validation: if it is green, it means that the validation score is above the defined threshold and the app is considered as “validated”. On the contrary, the level of validation will be displayed in red.

The third tab permits him to visualize the number of papers per application, thanks to a colored pie chart. They can select one specific application and the title of one of its associated papers to display information about it.

Finally, as several applications were not the subjects of any scientific publication in PubMed, we wanted to give to the user the possibility to see the names of similar validated applications. They can search for these applications in PubMed and find papers that can permit him to assess the level of validation of the learning category they are interested in.