

Project Name: Difficulty Analysis of Exam Questions

Requirements Document (version 2.0)

Project: **Difficulty Analysis of Exam Questions**

Date(s): **05/10/2018**

Prepared by: **Jayant Solanki**

Document status: ☐ Draft ☐ Proposed ☒ Validated ☐ Approved

1. Introduction

Main task of any Organisation conducting the Online/Offline Exams is to address the challenges faced in ensuring authenticity and fairness for Test-Takers to who attempt Online Exams. They have to ensure that each question set for test-taker contains questions of similar difficulty levels and those difficulty levels should be almost free from human biases. In this project I have came-up with an Web-application which will ensure that the questions are automatically tagged with correct difficulty level for next iteration of Exam, using Machine Learning analysis on the performance of test-takers for previous exam.

1.1 How to Use This Document

We expect that this document will be used by people with different skill sets. This section explains which parts of this document should be reviewed by various types of readers.

Types of Reader

This document is aimed for readers which are:

- **Project-Manager:** This application will provide manager with intuitive information about the Question creation and tagging life-cycle and help in conducting the Exams.
- **Question-Master:** For people assigned with the task for creating and moderating the questions, this application will be an useful tools for them to achieve their goals.
- **Web-Developer:** This project will be an important learning experience for the people assigned with the task of designing a better Question Creation and analysis website.

Technical Background Required

- The reader must be aware of the Examinations and types which are prevalent.
- The reader must know the basic usage of a computer Web Browser.
- The reader must know about Full-Stack Web Development

1.3 Scope of the Product

The product will briefly encompassed following categories of features mentioned below:

- **Core Features**
 - Identifications of those questions whose content remained same for all iterations of Selection Tests held in year 2014, 2015, 2016 and 2017.
 - Identification of features in the Selection Test data. This step will be done in sync with the next step and will be altered frequently.
 - Development of Machine Learning Algorithm for predicting the tags of the questions served to the Test-Takers in year 2014, 2015, 2016 and 2017.
- **Intermediate Features:**
 - Development of Front-end Website for with User login features.
 - Feature for adding new questions.
 - Feature for showing manual tags and predicted tags for a particular question.

- Feature for showing overall comparison between different taggings of Selection tests data.
- Showing statistics for change in the tags, that is, deficits and excess, overall.

2. General Description

This project proposes to develop a Machine Learning based software/web-app product for analysis of the data generated during the Online Selection Test for e-Yantra Robotics Competition (eYRC) which is conducted annually by the e-Yantra, Indian Institute of Technology - Bombay.

Data generated during the Online Selection Test contains raw information regarding the performance of the Test-Takers. This project aims to critically evaluate the raw information using Machine Learning and present the person tasked with administering the Question data, with tools and analytics for creating new questions or modifying old ones. This project will revolve around the difficulty taggings of the questions which are used for the Online Selection Test, and one of the core objective of this Project will be to predict the accurate difficulty tagging, which are easy, medium and hard, for each questions, and compare it with the manual tagging provided by the question-masters who initially designed those questions.

2.1 Product Perspective

During a Selection test, it is important to provide questions to the Test-Takers, which are at par in terms of overall difficulty and at the same time the question-master should be well versed to tell the difficulty level of the questions, so that questions with wrongly tagged difficulty levels do not end up to the test-takers.

This project encompasses extensive research in the field of machine learning so as to come up with certain features set which will be used to design a machine learning algorithm to accurately predict the consistent difficulty levels of questions across successive Online Selection Tests. Furthermore the project will employ certain data analytic methods to build a recommender system Website which will train the question-master to tag new questions with difficulty with more accuracy. This way personal biases of the question-master is reduced during the manual tagging of the new questions or old questions.

This project will be a full fledge product, which will combine the Machine Learning tool with the Web development Front-end. The product will primarily help any organisation to conduct free and fair Exams. This project will hand them the tools to easily analyse the questions of previously held Selection Tests or Exams.

2.2 Product Functions

- Identification of the features in the Exam Question Data
- Machine Learning Algorithm for predicting the difficulty levels or tags of the Question Data
- Provide User Interface for Question Creators with following features:
 - Provision for performing Machine Learning analysis on Selected Exam Data
 - Show analysis report on difficulty tagging for each year's Exam data
 - Show statistics for change in the tags, that is, deficits and excess, overall
 - Adding new Questions
 - Editing Previous Questions
 - User Management: providing admin or normal user roles, blocking, deactivating activating different user accounts.
 - Profile management: Editing own user account details.
 - Dashboard for showing overall summary of all the analytics

2.3 User Characteristics

This product will be mainly used by the user who has the role of question-master. This product can be used as a tool to train the question-master to tag the new question efficiently. The user of this product doesn't need to be highly technical oriented to use it.

One of the main motivation of this product is to make usability and data analysis a mouse clicks play, provide simple interface with easier to understand the directions. The user will need to know the working of a Online Selection Test. It will be more better if the user knows about the question creation methodology. The user will be required to understand about basic terminology related to the Selection Tests. How the selection test works, difficulty level terms, etc.

There will be no public release of this product. The broader target of the product can be those organisations who conduct the online test or Exams, and will like to have Feedback on the quality of questions which they employed, instead of using the surveys to gauge the views of test-takers on the difficulty of the questions, they can get detailed analysis of the those questions using the responses generated by the test-takers on those questions.

2.4 General Constraints

- This product will use the questions from the databases of the Selection Test held in year 2014, 2015, 2016 and 2017.
- This product will utilize the responses from the Test-Takers on different questions used during the Selection Test.

2.5 Assumptions and Dependencies

- There will be certain assumption on the questions which will be later on added in this document, such as where will the new questions go, some database challenges for incorporating previous tags.
- The project assumes that the future iteration of Selection Test or any other Exam will follow the database structure retrospectively.
- The project will need an administrator who will maintain the website, add new users, etc.
- The administrator will be need to setup the Website and migrate question bank data from respective database of Selection Test held since year 2014.
- The administrator will need to understand about the working of the website and the database and should be technically qualified for handling the database setup.

3. Specific Requirements

This section of the document lists specific requirements for ***Difficulty Analysis of Exam Questions***. Requirements are divided into the following sections:

1. User requirements. These are requirements written from the point of view of end users, usually expressed in narrative form.
2. System requirements. These are detailed specifications describing the functions the system must be capable of doing.

3.1 User Requirements

Front-end Website

- There will be a Web application where the users can choose which Selection Test data will be processed by the machine learning algorithm for the data analytics generation.
- The interface will show the predicted difficulty taggings for the questions which were used in the Selection Tests.

- The Website will show the changes in the difficulty level of questions, show overall deficit and excess in the question categories caused by the change in the difficulty levels.
- The Website will provide the tool for adding new question with different interfaces to help the users to efficiently tag new questions.
- The Website will display real-time information of any ongoing Machine Learning Analysis initiated by the user.

3.2 System Requirements

Back-end

Development will be done on a workstation running Ubuntu Operating System.

- The application will provide the comparative analysis between the manual taggings and the predicted taggings.
- The application will provide the comparative analysis between the predicted taggings of different Selection tests.
- It will provide questions specific information, such as flagged questions which shows unusually high incorrect attempts in successive Selection Tests, question category-wise accuracy of predicted taggings.
- The application will enforce user authentication, access restrictions, user input validation and security for the database from external intrusion

4. Appendices

- Technical Requirements
- Machine Learning report

5. Glossary

- **Difficulty-tags:** 0 for easy, 1 for medium and 2 for hard difficulty. These tags are allotted to every questions.
- **e-Yantra:** it is an initiative by IIT Bombay that aims to create the next generation of embedded systems engineers with a practical outlook to help provide practical solutions to some of the real world problems.
- **eYRC:** Also called e-Yantra Robotics Competition, conduction by e-Yantra.
- **Questions:** In the Selection Test, there are three categories of questions, C programming, Electricals and General Aptitude. Each have 10 questions for each Questions set. each 10 questions are in ratios of 3:4:3 (easy:medium:hard).
- **Question-Master:** user who creates new question for the Selection Test.
- **Question-Set:** Each test-taker gets a set of 30 Questions. Each set will have mutually exclusive questions.
- **Selection Tests:** The Online Exam conducted by the e-Yantra for their Robotics Competition

6. References

Background

- Each Exam year has a **Question Bank** of **N** Questions equally divided into 3 Categories: **Aptitude, Electronics and Programming**
- Each question is tagged with a Difficulty Level: **Easy, Medium, or Hard**
- Following metrics are used for assessing the Difficulty levels:
 - Questions that require direct application of concepts and formulae are categorized as Easy
 - When solving a question involves extra processing such as application of assumptions and prior knowledge are categorized as Medium
 - Questions that require derivations and application of logic that involve multiple steps to arrive at a solution are categorized as Hard

- Each Test-Taker gets a Unique Question Set of 30 questions each having equal division in three categories and difficulty level in some fixed ratio
- Analysis on responses of Test-takers on previous Exams
- Identification of features such as:
 - the proportion of Test-takers who attempted a question
 - correctly answered a question,
 - incorrectly answered the questions,
 - response of top performers
- Application of several Machine Learning Algorithms such as K-Means Clustering, Neural Network and Autoencoder on the identified features
- Choosing the Algorithm with most accurate result, which was in this case **K-Means Clustering**
- Saving the predicted tags in the Database for the respective questions
- Displaying the Front-end Website which has access to same Database, to the user

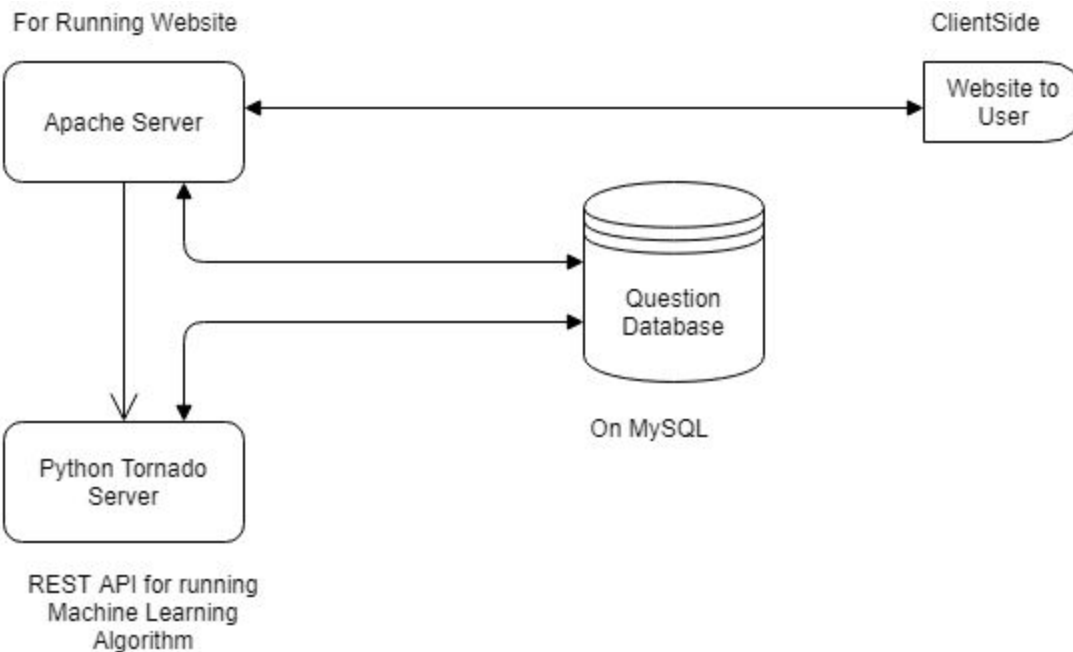


Figure 6

END