

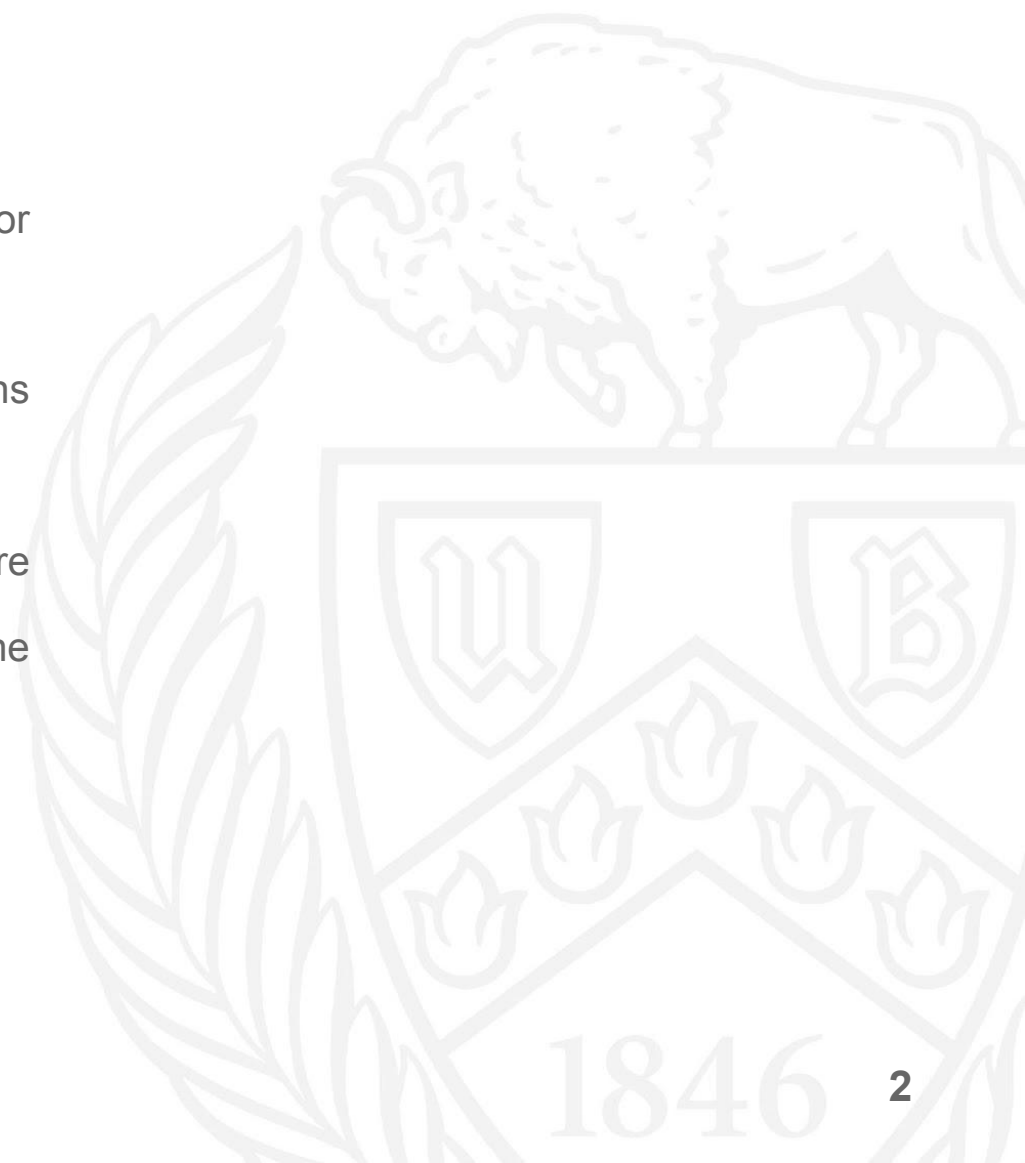
DIFFICULTY ANALYSIS OF EXAM QUESTIONS

Developed and Presented by: Jayant Solanki - 50246821



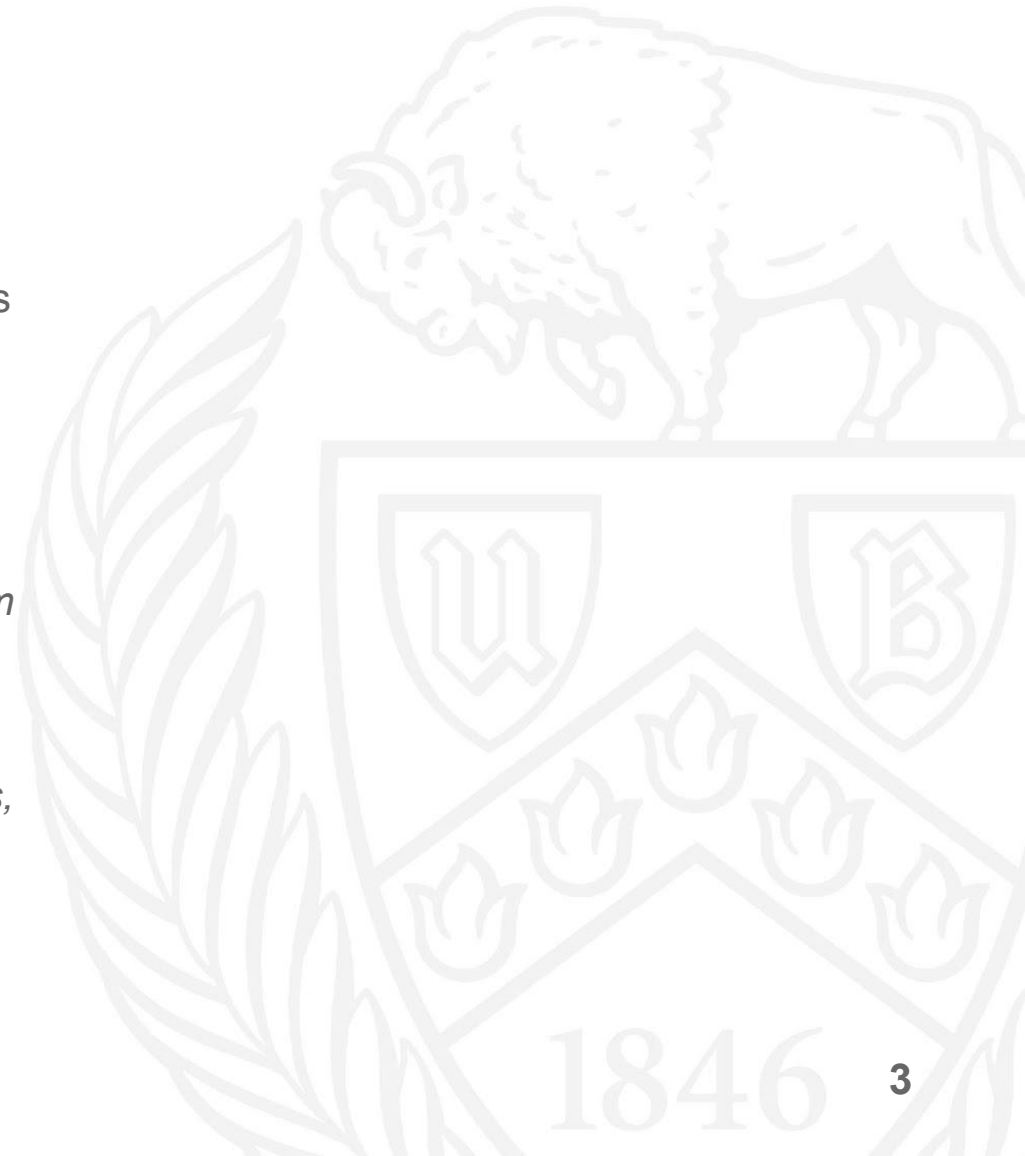
Motivation

- To address the challenges faced in ensuring authenticity and fairness for Test-Takers to who attempt Online Exams
- The challenge is to ensure each question set for test-taker contains questions of similar difficulty levels
- Come-up with a web-application which will ensure that the questions are tagged with correct difficulty level for next iteration of Exam, using machine learning analysis on the performance of test-takers for previous exam.



Objectives Achieved

- Identification of the features in the Exam Question Data
- Development of 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*



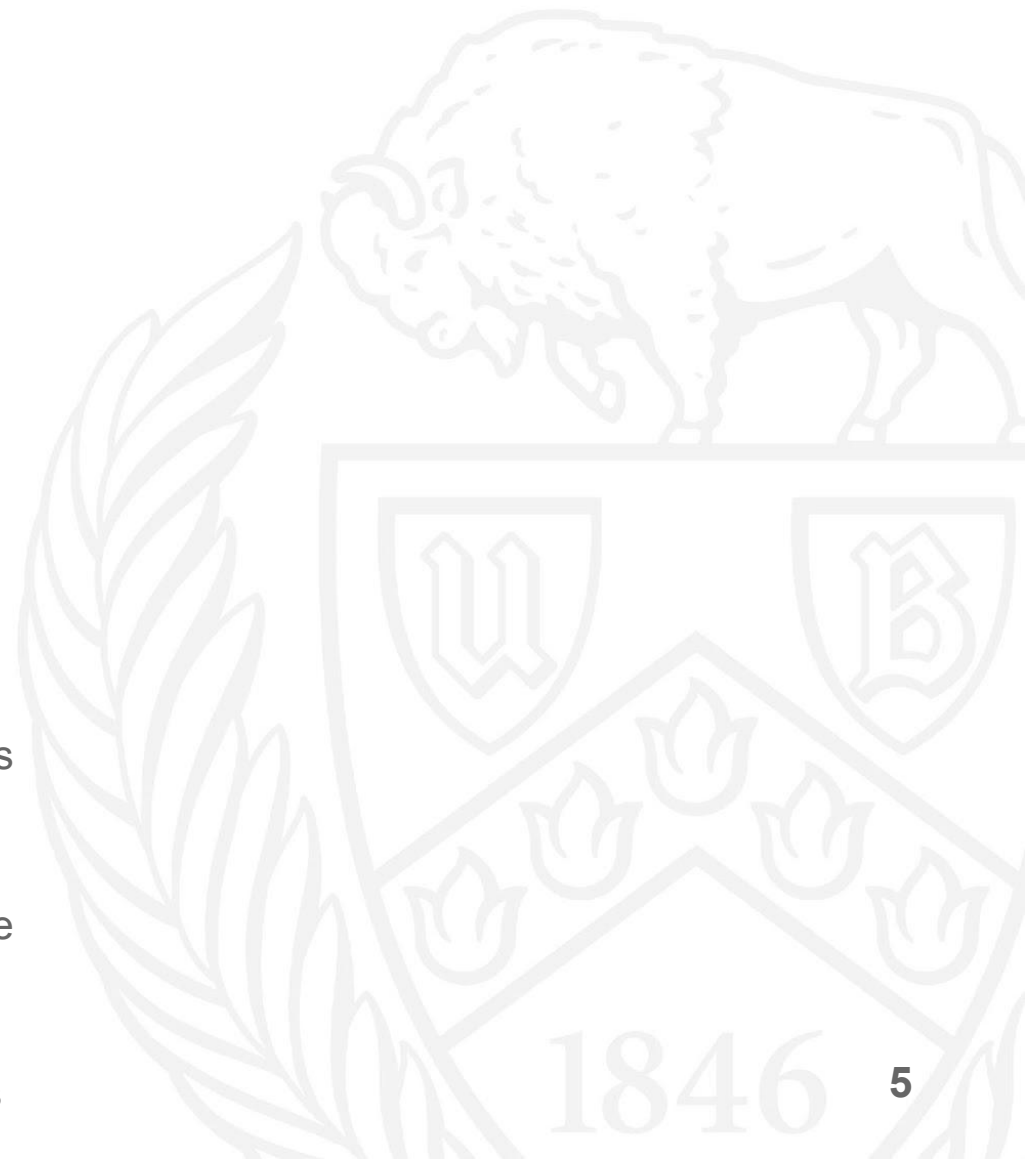
Methodology

- 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

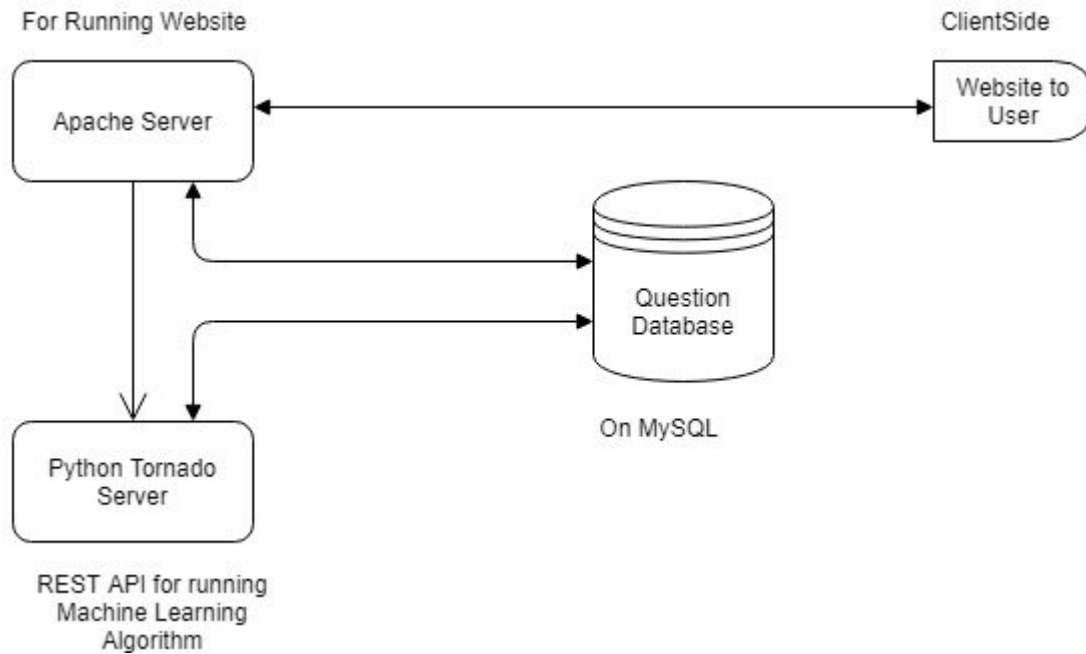


Methodology: continued

- 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 Mysql Database for the respective questions



Block Diagram



Tech-Stack Used

- Machine Learning
 - *Python 3.6*
 - *Tornado for Rest Api*
 - *SQLAlchemy*
 - *Matplotlib for plotting the clusters*
- Web Development
 - *Laravel 5.4 with Symphony and Eloquent ORM*
 - *PHP 7*
 - *MySQL*
 - *Apache Server for deploying the Website*
 - *Bootstrap 3.6*
 - *Google Charts*



Demo

[Visit the Website](#)



Room for Improvements

- Creation of Question Sets after the new difficulty level tags are obtained
- Adding Question Revision History
- Adding support for more Machine Learning Algorithm



Challenges faced

- Identification of proper features for the Machine Learning Analysis, acknowledging the fact that some of the question were incorrect
- Synchronising the Front-end and the Back-end Machine Learning Model
- Going through the awful iteration of User input validation and security testings
- Creating the Website with certain degree of action flow.



Acknowledgement

- This project was developed under the guidance of Prof. Alan Hunt, University at Buffalo
- Question Bank and Test-takers' data was provided by Indian Institute of Technology - Bombay
- Special thanks to Prof. Kavi Arya, Dr. Saraswathi Krithivasan, Dr. Krishna Lala



