CS 5542 BIG DATA ANALYTICS AND APPLICATIONS

KANSAS CITY FOUNTAIN MONUMENT IDENTIFIER – REPORT #2

TEAM #2 AJNN: #4 JACQULYN BATSON | #20 CHIA-HUI AMY LIN | #36 NICKSON SIGEI | #46 NANCY YANG

---

## ❖ Project Objectives

### ▪ Significance

- This app is unique in that it will be able to tell the artist and year of work when a picture of building is taken. Apps that exist that are related to this are not able to capture a photo and give such information.

### ▪ Features: Use Case/Scenario

- The app will be able to take a photo and/or when a photo is chosen from the gallery will predict and tell which Fountain monument it is.

## ❖ Approach

Machine learning approach will be used whereby pictures will be put into categories, trained and using the chosen tool will predict which Fountain Monument it belongs to.

### ▪ Data Sources

- Download from google image. Following is the link of the data source: https://www.dropbox.com/home/Public/BigDataAppProjPic
- Amazon Web Services public datasets http://aws.amazon.com/datasets
- Freebase http://www.freebase.com/
- UCI Machine Learning Repository

### ▪ Analytic Tool

- Excel sheets. Results from the error rate from the program.
- Spark RDD.
- Others to be decided and investigate.

### ▪ Analytical Tasks

- Error rate comparison with lab tutorial results.
- Changing the training dataset VS testing dataset ratio, compare the error rate result.
- Changing the number of training dataset number, keeping testing dataset number fixed, compare the result
- Changing the number of testing dataset number, keeping the training dataset number fixed, compare the result.
- Others to be developed.

- Expected Inputs/Outputs

  - Input: 3 categories of individual famous fountains in Kansas City (3 different fountains). The fountain name: J.C. Nichols Memorial Fountain, Children's Fountain and Muse of Missouri. Each fountain will have around 30-40 training dataset and 10-20 picture of testing dataset. Since our tasks is to recognize different fountains, we need to distinguish fountains.
    Example Input →

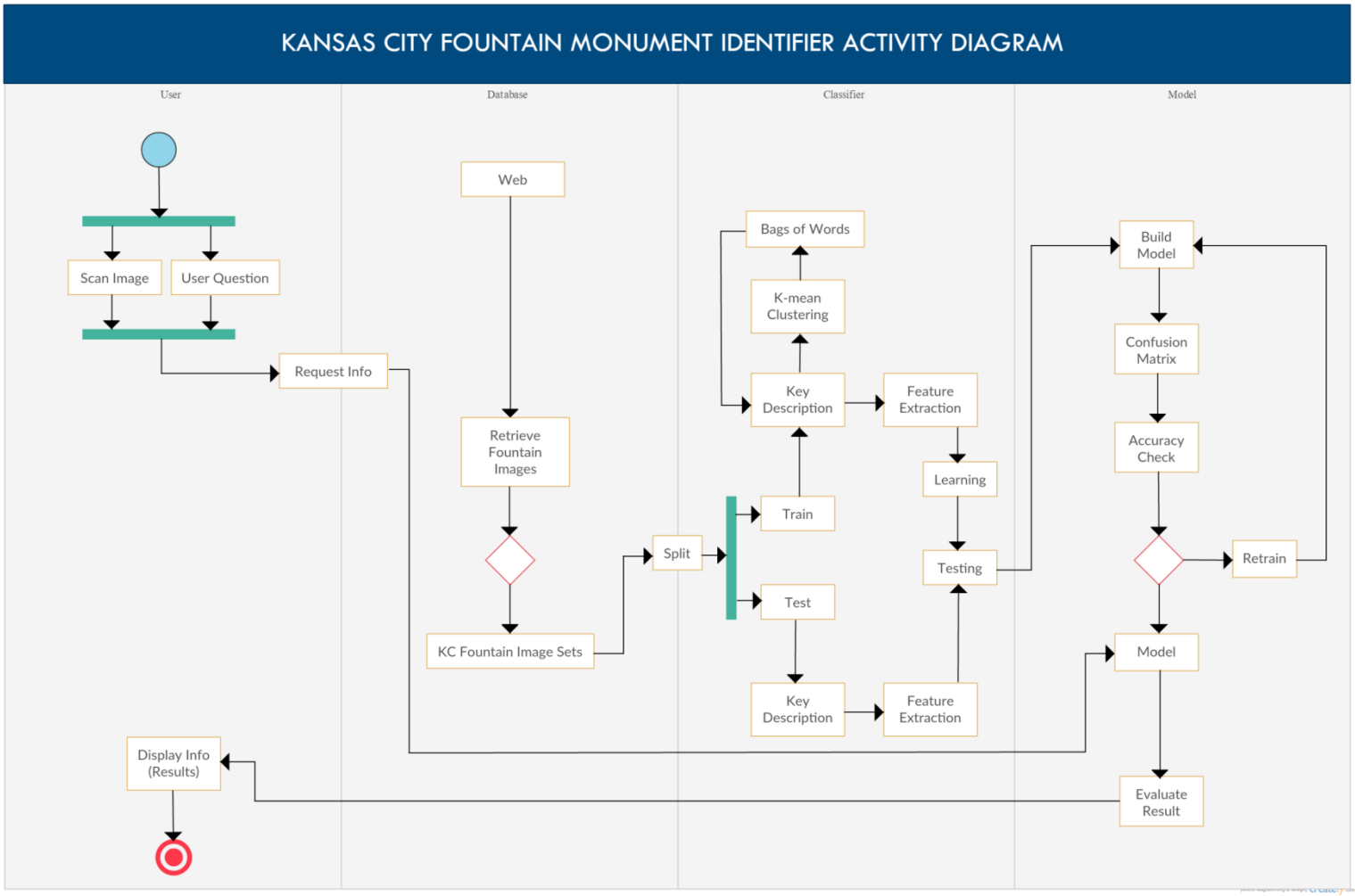
  - Output: The fountain name.
    Example Output → *J.C. Nichols Memorial Fountain*, by Henri-Léon Gréber

+

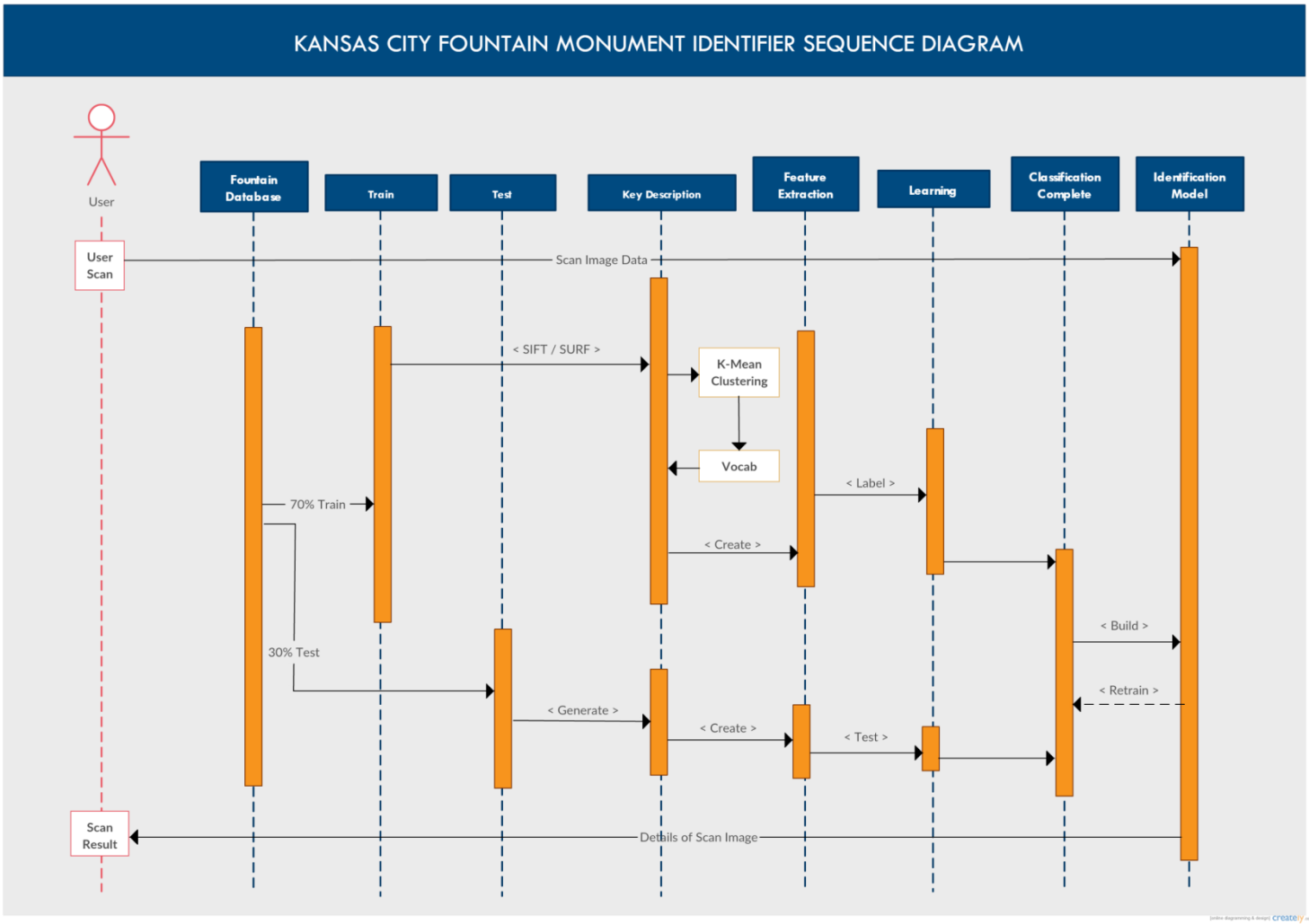
## ❖ Application Specification

- System Specification (Big Data Analytics Server/Client)
  - ✓ Features, workflow, technologies
- Machine Learning Algorithms/Models
  - Shallow Learning (Spark) Model
    - Decision Tree
    - Random Forest
    -
  - Deep Learning Model (Tensorflow)
    - Linear Regression Model (Image Train Data + Train Label / Test Data + Test Label )
    - Softmax + Cross Entropy + Build Model ( Train and Test data )

- Activity Diagram(workflow, data, task)



- Sequence Diagram(interaction/collaboration)




- Feature Specification
  - Display real-time information/details of certain fountain monument in Kansas City.

- Operation Specification(Input/Output, exceptions)
  - <u>Input:</u>
    Scan image of monuments in Kansas City from user's device or allow user to ask questions using the device when it has the image or pointing toward the image.

  - <u>Output:</u>
    Detail information of the monument that user requests. ( Name of the monument, Establish Year, Name of the Artist, Background details … etc.) Show/pop-up details in real-time when the user is pointing toward the image or when the user asks the question of specific monument.

  - <u>Exceptions:</u>
    Monuments outside of Kansas City might not be identified.

- Existing Applications/Services Used

  ✓ Google Arts & Culture
  https://www.google.com/culturalinstitute/beta/

  - Description:
    Partner with different institutions and museums around the world. An app to discover multiple art collections and stories on any device (smartphones, tablets, laptops…). Provide the user with an easy way to explore cultural treasures in details like you're personally visiting/experiencing the piece of art. Have an experiments webpage for developers and programmers to combine technology with artworks. Techniques include machine learning, image classification, serendipity, image similarity, real-time 3D, data visualization, meta data and image similarity.

## ❖ Implementation

- Implementation of your application using Clarifai/Spark/TensorFlow API and Google Conversation API (Revision)
  - For Clarifai, we used the API to recognize sample images of the fountains. It may not work that well for our application because we have very specific data. We may just need to adjust the types of questions that can be asked of Clarifai.
  - For Spark, we are currently using the Decision Tree algorithm, which is achieving only 48.6% accuracy with our own data set. We attempted to use Random Forest but it currently has bugs that need to be worked out in the next iteration. We may also need to increase the size of our dataset.
- Implementation of your Deep Learning Application using your own data
  - Extract picture data from feature extraction program with label on it.
  - Get the label numbers correspond to each images and put them into label data.
  - Aggregate label with actual image pixel data into an asarray.
  - Build Linear Regression Model in tensorflow environment and get the accuracy of the datasets so far.
  - Build Softmax Regression Model and the cost of the model to train and test in tensorflow environment.

## ❖ Documentation

Project page: https://github.com/datarocksAmy/BigDataProject
Wiki page: https://github.com/datarocksAmy/BigDataProject/wiki/Iteration-2

## ❖ Project Management

- Plan & Project Timelines

  - 03/11/2017 – Set up UI + Train/Test Data. Improve accuracy as needed.
  - 03/18/2017 – Build modules.

- 03/25/2017 – Resolve any potential/upcoming issues for each modules.
- 04/04/2017 – Android App/platform development.
- 04/11/2017 – Finish up app + Connect all modules.
- 05/04/2017 – Testing + wrapping up project.

## Implementation Status Report

✓ Work Completed

- Description: Collect fountain data, diagrams(workflow and what model to build), narrow down the whole structure for the project.

- Responsibility (Task – Person):
  Nancy:
  Completed online fountain picture collection.
  Extracted the feature vector from all the pictures using the feature extraction program
  Ran Fuzzy Classification on the picture dataset.

  Amy:
  Convert Training and Testing datasets into arrays from feature extraction to build linear regression model.
  Fix the code to fit our own training sets to the softmax regression, evaluate the cross-entropy and build the model for further training.

  Jackie:
  Added Clarifai functionality to Android app, added Spark functionality to Android app using our dataset. Used Decision Tree for Spark. Tried to use Random Forest and will try to fix errors for next iteration.

  Nickson:
  Worked on Google conversion API

- Time taken ( hours ): each person time add up to 24 hours
- Contributions(members %): Jackie 25% | Nancy 25% | Amy 25% | Nick 25%

✓ Work to be Completed

- Make the images array data into same dimension to wrap up building the Linear Regression and Softmax in tensorflow.
- Add Tensor flow functionality to Android app
- Increase accuracy of Spark model for our data

✓ Issues/Concerns

- Building the Tensor flow dataset with language model is a huge obstacle.
- Capsulate the pixel data from one image category to match the label sets and do a multiple data type transformation when reading in the raw data.
- Improve accuracy of Spark model (currently only 48% accuracy of testing data)