01_Project-3 Documentation

Created: *10/11/2017 8:56 AM*
Updated: *10/29/2017 8:21 PM*
Author: *sohan250*
Tags: *PROJECT-3*

01_Project-3 Documentation

Author: sohan250

CSCI8380: Adv. Topics in AIS

Fall 2017

Project 3: Designing your first Amazon Mechanical Turk HIT

Due: October 25th (mid-night)

Design your first HIT using AMT Java SDK. In this project, you will need to use **Twitter API** to collect data and send your data set (**min 100 tweets**) that potentially contain any type of **religious extremism.**

1. Ask workers to verify if a tweet contains extremism,
2. why (justification),
3. scale of extremism
4. and their confidence score.
5. Design you HIT GUI in a simple and user-friendly manner.
6. For final classification use a consensus of at least three workers.

You can send your HITs to students in the class using AMT Sandbox. Optionally you can publish your HITs for actual crowd workers (paid) for undisclosed bonus points.

Programming language/environment: Amazon Mechanical Turk Java SDK (https://requester.mturk.com/developer).

What to submit: Please post

- [ ] source code (zip)
- [ ] brief description
- [ ] post a picture/snapshot of your sample HITs.

☐ include your data set and responses from the crowd workers.

How to Submit:

☐ Submit your ".zip" file using ELC.

Only team leaders need to make a submission.

☐ Also submit a document including several snapshots of user interface.

☐ **Every student needs to submit a peer-evaluation form within 24 hours of the project submission deadline**.

Do not place your solution on a public web site. Submit your own work and follow the course misconduct policy.

02_Using : Amazon Mechanical Turk 11th Oct

Author: sohan250

- **Amazon Mechanical Turk**
- **HITS**

 # 1. Amazon Turk - [Amazon Mechanical Turk]:
[Create project Link - https://requester.mturk.com/create/projects/1085213/edit]
[Mechanical turk hits link - https://www.mturk.com/mturk/findhits?match=false]
[Amazon mechanical Turk sdk - https://requester.mturk.com/developer]
[AMT Java API http://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/welcome.html]

1. Amazon turk -> create project-> new -> categorization
2. Give project Name -> Ex. Extrem Project
3. Set Categories - Extrem, Not Extrem, Not Sure
4. Provide general intructions _ blah blah
5. Selection Criteria- Includes/Excludes for every category
6. Upload .csv file- ideally we are collecting tweets (Twitter API) , store them as csv
7. Open Excel (temporary data):

Account	Date	Location	Content

sohan	11-Oct-17	Athens,GA	Abra Ka Dabra

1. Save as .csv UTF-8 format (if exists)
2. Upload
3. Set type of data - text for all
4. No descriptions added, so u can skip and submit
5. Check whether it looks as required, proceed
6. Number of workers per item - (click change) - Use two workers
7. Reward per submission - 0.02
8. Manage-tab - Review, If you like it then accept else reject
9. Manage-> Qualification types->Friendly Name:Arabic Speaker->Description:Blah Blah->create
10. Developer-tab-> Create AWS Account->Link Mturk account->Register for Mturk Developer Sandbox->Download SDK (no restriction in terms of language java/python/php.. etc) [Override documentation content over this, use any language for this.

03_AWS SDK

Author: sohan250

Getting Started with the AWS SDK for Java

Download the Sample Project

git clone https://github.com/awslabs/aws-java-sample.git

Configure the Access Keys

Create your credentials file at *~/.aws/credentials (C:\Users\USER_NAME\.aws\credentials for Windows users)* and save the following lines after replacing the underlined values with your own.

[default]

aws_access_key_id = YOUR_ACCESS_KEY_ID

aws_secret_access_key = YOUR_SECRET_ACCESS_KEY

Compile and Run the Sample

From the **aws-java-sample** directory, run:

mvn clean compile exec:java

About the Sample

This sample application is designed to show you how to:

- Declare a dependency on the AWS SDK for Java using Maven.
- Read access keys through a configuration file.
- Instantiate an Amazon Simple Storage Service (Amazon S3) client.
- Interact with Amazon S3 in various ways, such as creating a bucket and uploading a file.

The project's README file contains more information about this sample code. If you had trouble getting set up or have other feedback about this sample, let us know on GitHub.

Additional Resources

For in-depth user guides, API documentation, developer forums, and other developer resources, see the AWS SDK for Java page.

04_Amazon Mechanical Turk Sandboxing

Author: sohan250

1. Open https://requestersandbox.mturk.com/ for Requester
2. Open workersandbox,mturk.com for Worker

05_Sources used in program

Author: sohan250

For removing Emojis - https://github.com/vdurmont/emoji-java/releases
Download jar from the download sources

Confidence level confusion

Author: sohan.nipunage@live.com

- Confidence level confusion
- 1 person 20-30 tweets
-