Data Labeling System

Requirements Analysis Document (RAD) / ITERATION 2

Purpose of the system

Data labeling, in the context of machine learning, is the process of detecting and tagging data samples. The process can be manual but is usually performed or assisted by software. Data labeling is an important part of data preprocessing for ML, particularly for supervised learning, in which both input and output data are labeled for classification to provide a learning basis for future data processing. This system can be used to label news from on online newspaper, articles as sports economy, world, politics other also can be used to label customer comments in e-commerce web sites. This system also calculates the percentages of labelling.

Stakeholders

- Murat Can Ganiz
- Lokman Altın

Scope of the system

Data labeling service comprises many different tasks. This project we use data labelling system for users' comments. Data labelling system can be used by customer services, complaint/request sites and in extracting data from these databases.

Functional Requirements

- System will be a multi-user system.
- There must be many to many relationship users and instances.
- Program will be getting user information as a json file.
- Program must support easily pluggable labeling mechanisms.
- The labeling mechanism will be random labeling mechanism.
- Program will write results in a json file
- Program must show one of the instances to this user for labeling again with a certain probability in a dataset
- Previously labeled instance will be selected randomly from the instances that are already labeled by this user in this dataset
- Config Json file have at least 3 different users configured.
- Program can add datasets to your config json by providing an id, name, and file path.
- Program can assign any number of existing users in our config json to a particular dataset for labeling.
- Program can set an existing dataset to start labeling simulation from our config json.
- In our program if an instance is labeled more than then assign the most frequent class label as its final label.
- Program can stop the simulation at any time and access the reports.
- User performance metrics will be created and update in every labelling.
- Instance performance metrics will be created and update in every labelling.
- Dataset performance metrics will be created and update in every labelling.
- In each run, only one dataset will be labeled.
- Previous reports will be persistent.

Non-Functional Requirements

- A user can label many instances
- An instance can be labeled by one or more users (possibly with different class labels).
- The LabelingMechanism needs to assign one (or more if multilabeled) of these labels to a given instance.

USE CASES

- Users are randomly chosen according to the given number of users for each dataset.
- If a dataset is read for the first time, then the users make their first assignments.
- If a user is labeled before, the program checks its ConsistencyCheckProbability. A random number between 0 and 100 will be assigned. If this number is less than ConsistencyCheckProbability, the user labels a new instance.
- During these labeling, user metrics, instance metrics and dataset metrics are updated.
- In each labeling, total number of label assignment, unique label assignment, unique label, unique user, most frequent class label and their percentages, class labels and their percentages and entropy for each instance are calculated for instance metrics.
- Number of datasets assigned, list of all datasets with their completeness percentage, total number of instances labeled, total number of unique instances labeled, consistency percentage, average time spent in labeling an instance in seconds, standard. deviation of time spent in labeling an instance in seconds are calculated for user metrics.
- Completeness percentage, class distribution based on final instance labels, list number of unique instances for each class label, number of users assigned to this dataset, list of users assigned and their completeness percentage, list of users assigned, and their consistency percentage are calculated for dataset metrics.
- These data are kept for the next simulations.
- In the next simulation, If the current dataset is not used before, the data from the previous simulations are transferred and updated.
- In the next simulation, If the current dataset is not used before, New metrics files are created for this dataset.
- Program continues to work like that.

Glossary of Terms

***User* –** People who label the given data

***Label* –** A classifying phrase or name applied to a person or thing

***Instance* –** Sentences that need to be labeled

***Dataset* –** A collection of related sets of information

***Labeling mechanism* -** Mechanism that we used in our project to label data.

Check consistency probability- It is a parameter which shows the users selection probability.

Instance completeness percentage- It shows the percentage of the instances are labeled.

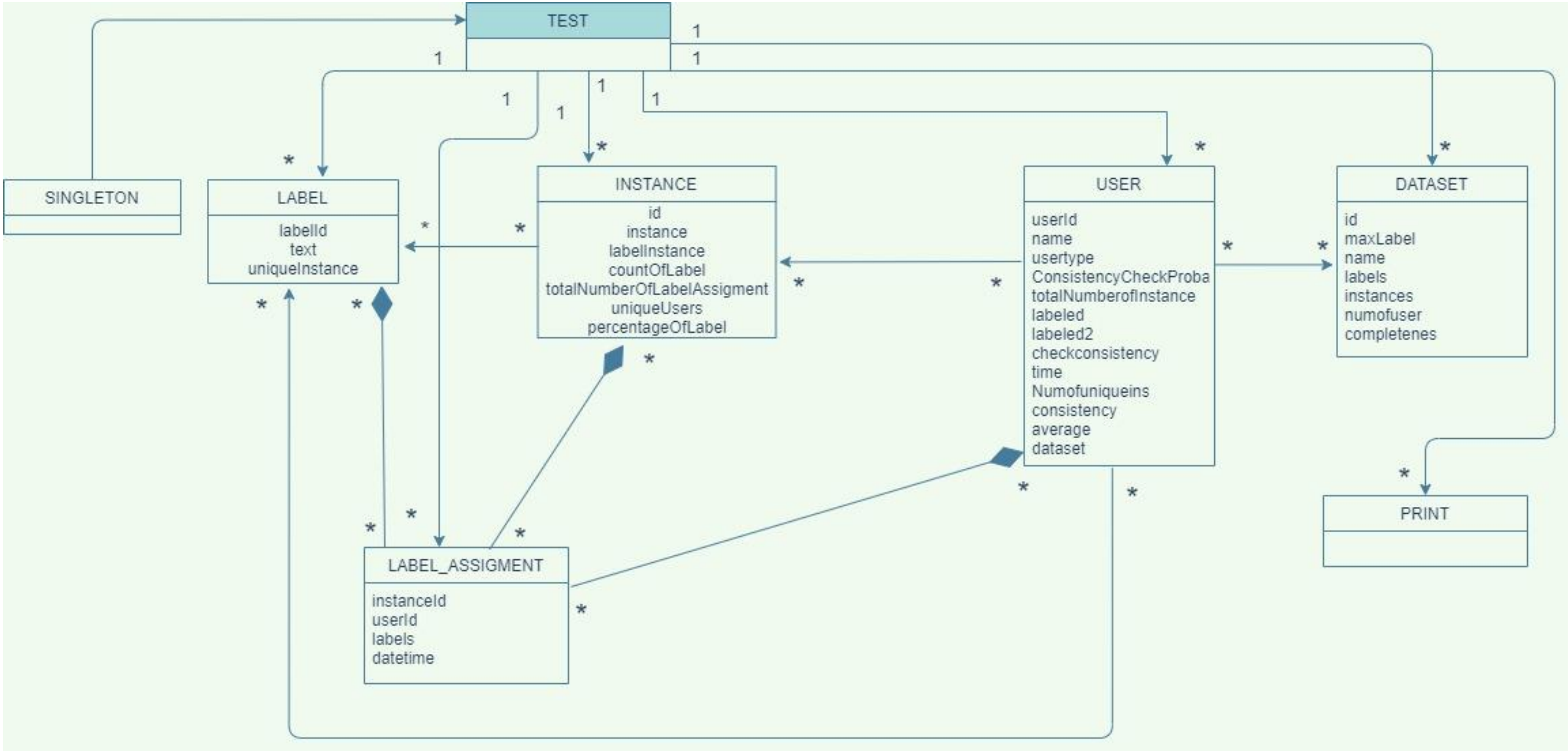
***Dataset completeness percentage* –** It shows the how many instance are labeled in the dataset.

***Consistency percentage* –**It shows the percentage when users labels the previous instances with same labeling.

***Unique instances* –**Group of instances which are labeled by the user uniquely.

***Unique labels* –**Group of labels which are in the instance uniquely.

Domain Model :



CREATED BY GROUP:

- 150117034 Berra Mercan
- 150117510 Elif Gökpınar
- 150117042 Ezgi Doğruer
- 150117061 İsra Nur Alperen
- 150117057 Rümeysa Öztürk
- 150116010 Şükriye Soyer
- 150114036 Ömer Kaya