Requirement Analysis Document

1.Introduction

1.1 Vision

Labeled data is a group of samples that have been tagged with one or more labels. A labeling system typically takes a set of unlabeled data and augments each piece of it with informative tags. After obtaining a labeled dataset, machine learning models can be applied to the data so that new unlabeled data can be presented to the model and a likely label can be guessed or predicted for that piece of unlabeled data.

1.2 Scope

The aim of this project is to design and implement a data labeling system in object-oriented manner. The data labeling system can have multiple labeling mechanisms such as random, machine learning, etc. In our program, instances of the dataset will be labeled based on the labeling mechanism provided by the user.

2. Functional Requirements

- The system must read the input file.
- The system must parse the file and construct the dataset.
- The system must configure itself according to the configuration file.
- The system must determine the corresponding labeling mechanism and start to assign the labels.
- Users should be able to assign one or more labels to an instance.
- Users should be able to assign labels to multiple instances.
- The system must print its actions to the command line and the log file.
- The system must store the assignments in the dataset.
- The system must print the labeled dataset to an output file.

3. Nonfunctional Requirements

3.1 Usability

- Outputs and logs of the system should be printed with an organized manner to be easily understood by the user
- The system should be a multi-user system.

3.2 Flexibility:

- The system should support easily pluggable labeling mechanisms.
- The system should be able to integrate to a user interface with minimal changes.
- The system should support different labeling mechanisms.
- The system should support scenarios in which words/terms in a document can also be labeled.

3.3 Performance:

- The system must give an output at a reasonable time.

3.4 Reliability:

- Components of the project code will be tested alongside the implementation phase to ensure that they are functional.

3.5 Stability:

- System should be stable in a sense that it should work with different inputs.

3.6 Supportability:

- The application must not be platform dependent, i.e., it should be able to run on any platform.

4. Glossary

Instance: An example of a particular type.
Label: A word or phrase indicating that the instance belongs to a particular category or class.
Dataset: A group of instances.
User: Someone who interacts with the system.
Labeling Mechanism: A mechanism to assign labels to an instance.
Assignment: The allocation of some labels to a particular instance.

5. Stakeholders

Customers:

- Murat Can Ganiz
- Lokman Altın

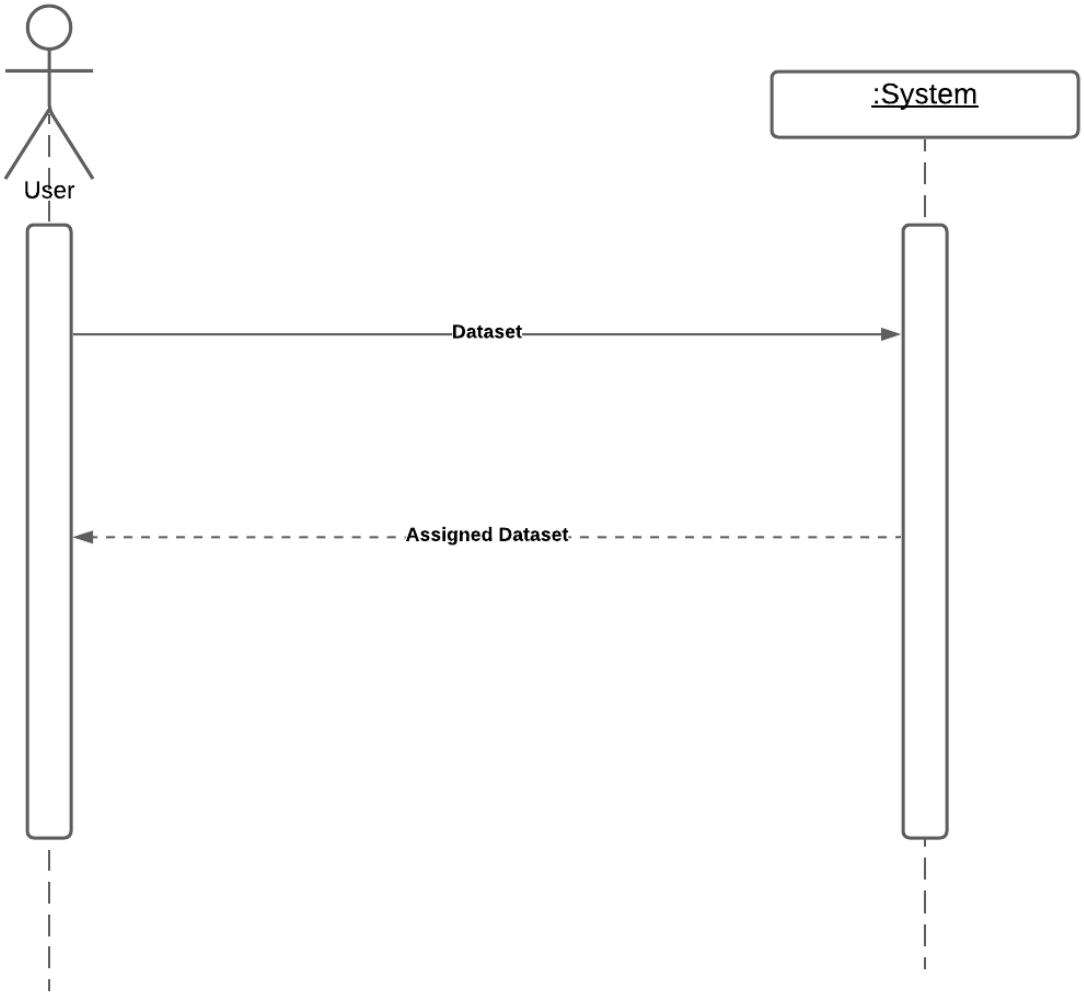
Developers:

- Eymen Topçuoğlu
- Berkay Deniz
- Muhammed Raşit Ayaz
- Ahmet Emirhan Bakkal
- Yunus Yıldırım
- Vahap Gözenelioğlu
- Ahmet Faruk Yılmaz
- Ubeydullah Günay

6. Use Cases

Use Case: Simulation
Actors: User, Data Labeling System
Precondition: User must provide an input.json.
1) User starts the system.
2) System parses input.json and constructs the dataset.
3) System determines the corresponding data labeling mechanism based on the user type.
4) System starts to assign the labels to the instances.
5) System outputs the labeled dataset to output.json.

7. System Sequence Diagram



8. Domain Model