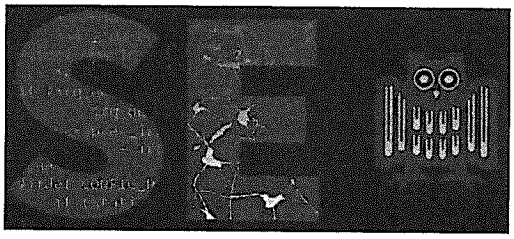


Requirements Specification:

Microstructure: Unsupervised Learning

Tutorial in Software Engineering

Winter Semester 2019/20

Contents

1. Requirements Specification

2. Objectives

The objective of the project is to develop an automated tool for unsupervised learning for microstructures using clustering algorithms of machine learning.

2.1 Must-Have Criteria

/M1/ The project should have a Web User Interface with a simple UI like selecting radio buttons as well as files from the pc.
/M2/ The interface will be general and not specific to the dataset provided by the client as the client intends to use it on other datasets as well.
/M3/ The interface should have the option of uploading dataset.
/M4/ The interface should have the option of Quit to stop the algorithm.
/M5/ The interface must have the option of selecting the clustering algorithm from the list of algorithms provided.
/M6/ The interface must have the option of selecting the number of clusters.
/M7/ After all selections are made the user can click on the Submit button to proceed.
/M8/ The interface will have the option to save the output of the corresponding algorithm in the form of an excel file in the pc.
/M9/ The interface must keep a default cluster settings (a default clustering algorithm in the preference order of k-means, hierarchical clustering, fuzzy c-means and an optimal number of clusters) in case no option is selected.
/M10/ The maximum number of clusters that can be made will be 50
/M11/ At least two out of the six clustering algorithm (k-means, k- medoids, Hierarchical clustering, self-organizing map, fuzzy c-means, Gaussian Mixture Model) mentioned must be implemented on the provided dataset.
/M12/ The output should appear in an excel sheet with the corresponding object name, sample name and cluster.
/M13/ The clustering algorithms will train and test on the dataset provided.
/M14/ The dataset could be split into shorter versions depending on the cpu load needed by the algorithm to process the dataset.

/M15/ A documentation of the final interface will be provided to the client during submission of the final project.

/M16/ The project will be up for running on localhost of the client pc.

2.2 May-Have Criteria

/O1/ We may implement a dimension reduction algorithm over the dataset.
/O2/ This dimension reduction algorithm may be a part of the interface.
/O3/ All algorithms, i.e. k-means, k- medoids, Hierarchical clustering, fuzzy c-means, and Gaussian Mixture Model, may be implemented.

/O4/ Code for Clustering of image data provided by the client might be provided to the client but will not be added to the application as the dataset uploading feature will not support such a huge file and also the clustering algorithm needs GPU support which at the moment client can not provide.
/O5/ The program may write a short summary on the clustering result, including information such as the number of data points in each cluster.

2.3 Must-Not-Have Criteria

/N1/ Apart from the algorithms mentioned in Must Have and May Have, others can not be implemented.
/N2/ The interface must not have a comparison section that compares the clustering result of the algorithm and that of manual clustering.
/N3/ The application will not have a very interactive interface.

3. Product Use

3.1 Application Area

The product should be easily accessible by all members of the chair of functional materials at Saarland University. It is then hosted on a local server.

3.2 Target Groups

The direct stakeholders of the product are the research assistants, PhD students, Master students, and professors of the chair of functional materials at Saarland University are the indirect stakeholders as they use the clustered data for future research.

3.3 Operating Conditions

The product is a local host based. It may have multiple instances on different PC's by accessing the local server.

4 Product Functions

4.1 Upload

/F10/ The client can upload a dataset by pressing the "Upload File" button (/M3/)
/F20/ The client can upload an image (/O4/)
/F30/ The application may provide a short summary about the uploaded data.

4.2 Choose parameters

/F40/ The client can select a clustering algorithm from the list of algorithms provided in the drop-down menu "Clustering Algorithm". (/M5/)
/F50/ The client can select the number of clusters by entering a value in the box.(/M6/)
/F60/ The client can (un)select a dimension reduction algorithm.(/O1/)
/F70/ The client can use the default cluster settings in case no option is selected.(/M9/)

4.3 Cluster/Classify

/F80/ After uploading and selecting all the parameters, the client can click on "Submit/Classify" button to proceed. (/M7/)

/F90/ The client can stop the process by pressing the "Quit" button. (/M4/)

4.4 Save/Download

/F100/ The client can save the output of the algorithm in the form of excel. (/M12/)
/F110/ The client can preview the result.

4.5 Short summary

/F120/ The client can read a short summary on the clustering result, including information such as the number of data points in each cluster.
/F130/ The client can read information about the application.

5 Product Data

The following data is stored on the device after each time a clustering algorithm is run:

/D10/ Each sample along with its corresponding cluster obtained when run, is saved altogether in an excel file (.xlsx format). This size is almost equal to the size of the input excel file. (/M12/)

6 Non-Functional Requirements

NFR explain how a system must behave, also called as Quality attributes

a. **Usability :** The usability of application is simple that users will upload the dataset containing objects and select the parameters (no.of clusters, algorithm) and run the algorithm to check the output(which will be generated as Excel).
b. **Efficiency of use:** The average time of task depends on the input dataset size and algorithm.
c. **Intuitiveness:** The Interface will be very simple. The headings and buttons will be labeled for user comprehension.
d. **Security:** The application access is made available to users as the application is directly installed on user server(PC).The users having access to the server(PC) can only use the application.
e. **Reliability:** The application will be reliable to work for any input data features to cluster. Reliability may decrease in case of hardware failures, problems with server or bugs.
f. **Performance:** The application will load in less time(less than a minute) and access the functions to generate the output.Clustering performance is made available at the end.
g. **Availability:** The application will be available online and supported by major browsers when launched locally.

h. **Scalability:** As this is a locally hosted application, the number of users will be one per system. Later on, when the application is hosted on a web server, it can serve multiple clients.

7 User Interface

The user interface consists of the product logo. Then there is the homepage which gives a description about the application. The data upload page gives the facility to upload the dataset to be clustered. The clustering page describes the clustering algorithm selected, gives the provision to select the number of clusters and select the option of dimensionality reduction. After clustering the Download page allows to download the output clustered files.

8 Use Case Diagram

The use case diagram shows all use cases of Microstructure: Unsupervised learning. There is only one actor and is described as a " Client". The client can be a bachelor, master or PhD student. The use cases are represented by ellipses.The blue use case is performed by the application internally. The application uses information from the uploaded file and chosen parameters to prepare a short summary about the uploaded file and clustering results.

9 Technical Product Environment

9.1 Hardware

a. System Model: X10DAi
b. System Type: x64-based PC
c. Processor(s): 2 Processor(s) Installed.
   [01]: Intel64 Family 6 Model 79 Stepping 1 GenuineIntel ~1200 Mhz
   [02]: Intel64 Family 6 Model 79 Stepping 1 GenuineIntel ~2400 Mhz
d. BIOS Version: American Megatrends Inc. 3.0a, 2/5/2018
e. Total Physical Memory: 65,403 MB
f. Available Physical Memory: 58,101 MB
g. OS Name: Microsoft Windows 10 Pro
h. OS Version: 10.0.18362 N/A Build 18362
i. OS Manufacturer: Microsoft Corporation
j. OS Configuration: Standalone Workstation
k. OS Build Type: Multiprocessor Free

9.2 Software

a. Machine learning programming language: Python
b. User interface: Javascript, CSS, HTML

c. Tools: Colab, Visual studio
d. Framework: Tensorflow, Pytorch, Flask, Klein

10 Quality Requirements

Quality Requirements	Very Good	Good	Normal	Irrelevant
Functionality	✓			
Usability		✓		
Reliability	✓			
Efficiency	✓			
Modifiability		✓		

Functionality
The application is designed for clustering in a faster way and to serve the students and research peer groups. Hence functionality should be "very good" and is of high importance.

Usability
The usability of the application should be deemed "very good". The application should serve the intended purpose of the user and also the user should be able to navigate and use it with ease.

Reliability
The application designed should have "very good" reliability as the main purpose of it is to reduce the manual effort and computational time for clustering.

Efficiency
The application is intended to reduce the computational time and manual effort by a greater margin. Hence it should have a "very good" efficiency.

Modifiability
The user should be able to switch among the various clustering algorithms and modify other parameters. It is not possible to choose the clustering algorithms and their corresponding parameters. So modifiability should be "good".

Glossary

Clustering algorithm

A machine learning technique which is used to classify data points into a specific groups/clusters without any external supervision.

Dimensionality reduction

It is the process of reducing random variables under consideration and selecting the principal variables. It is mostly used in case the dataset has a lot of unnecessary variables. This reduces the size overhead of the dataset before further processing with other algorithms.

Core use case

Microstructure: Unsupervised Learning

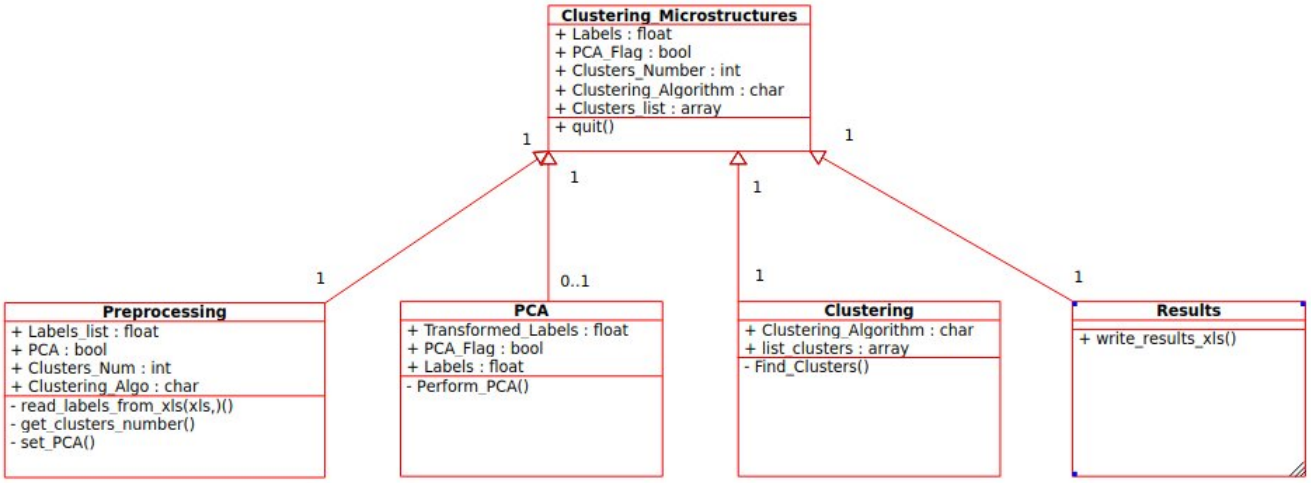
Use case 1: Clustering without dimensionality reduction

Use case 2: Clustering with dimensionality reduction

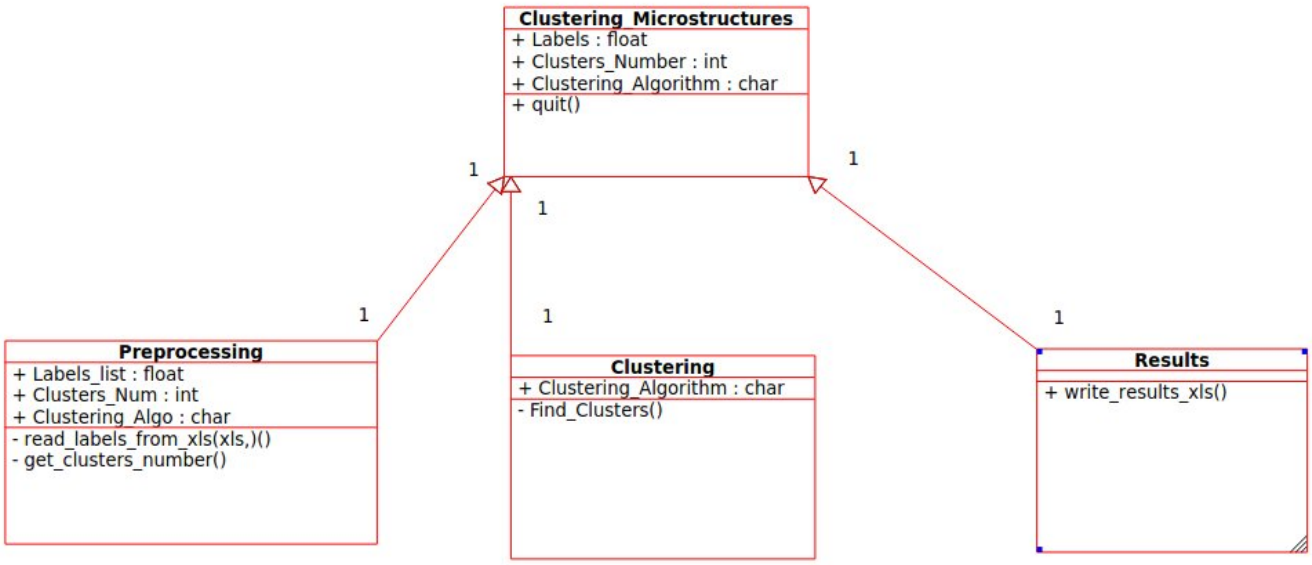
Use case3: Output preview and download functionality included

The use case diagram shows all use cases of Microstucture: Unsupervised learning. There is only one actor described as a "Client". The client can be a bachelor, master or phd student. The use cases are represented by ellipses.

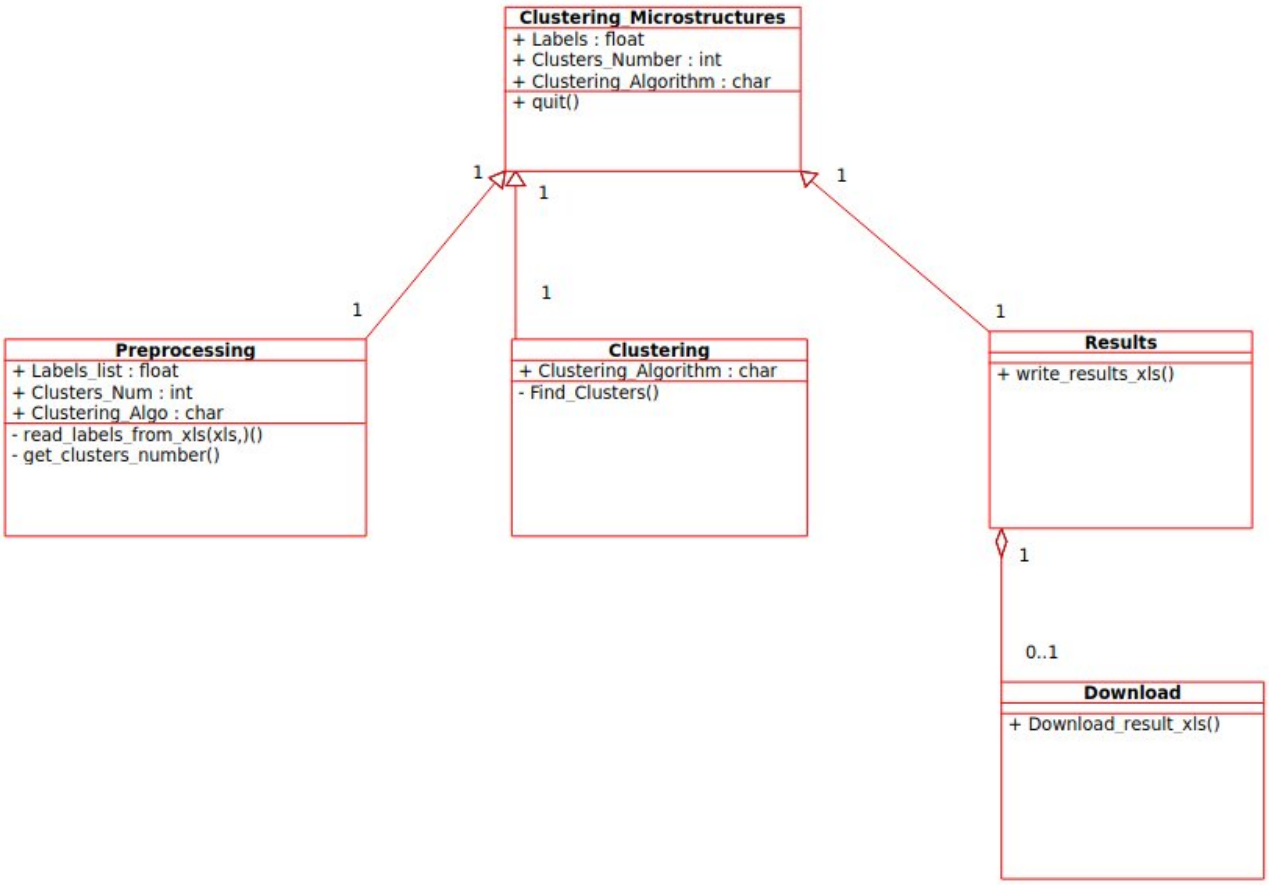
The blue use case is performed by the application internally. The application uses information from the uploaded file and chosen parameters to prepare a short summary about the uploaded file and clustering result.



Draft of UML class diagram for the use case where PCA is performed before clustering.

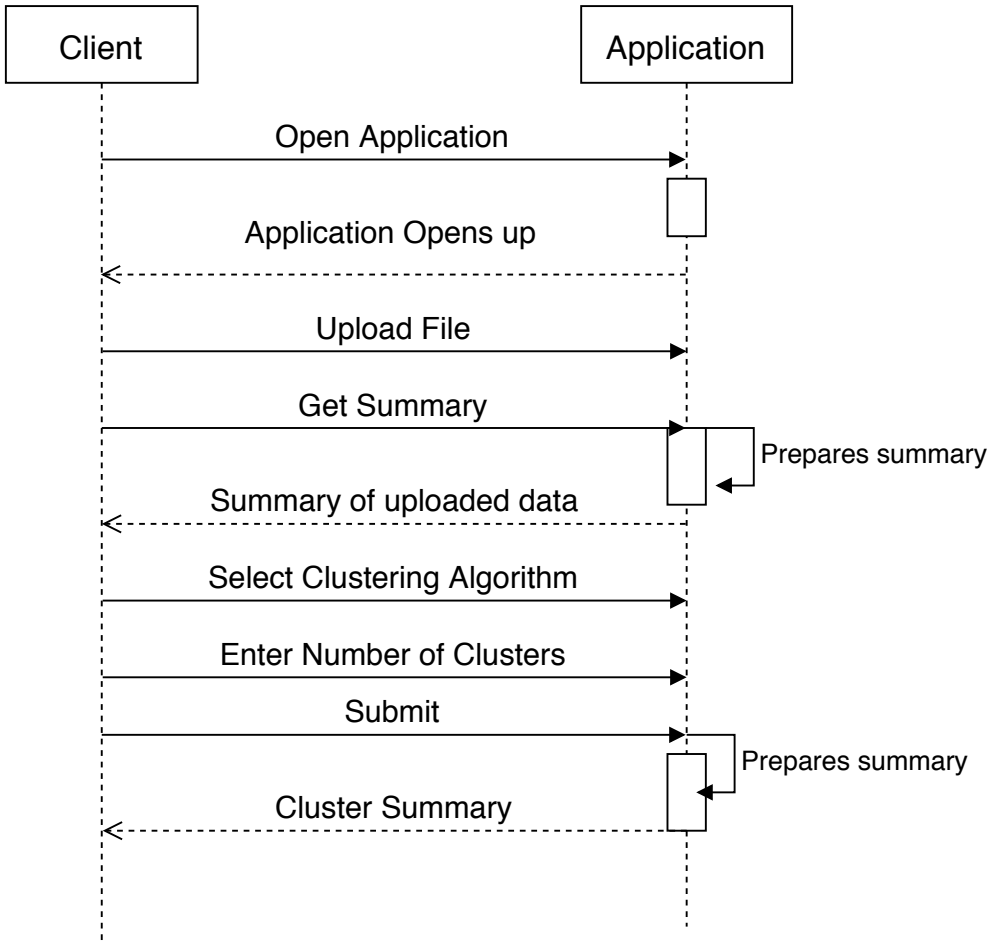


Draft of UML class diagram for the use case without PCA.

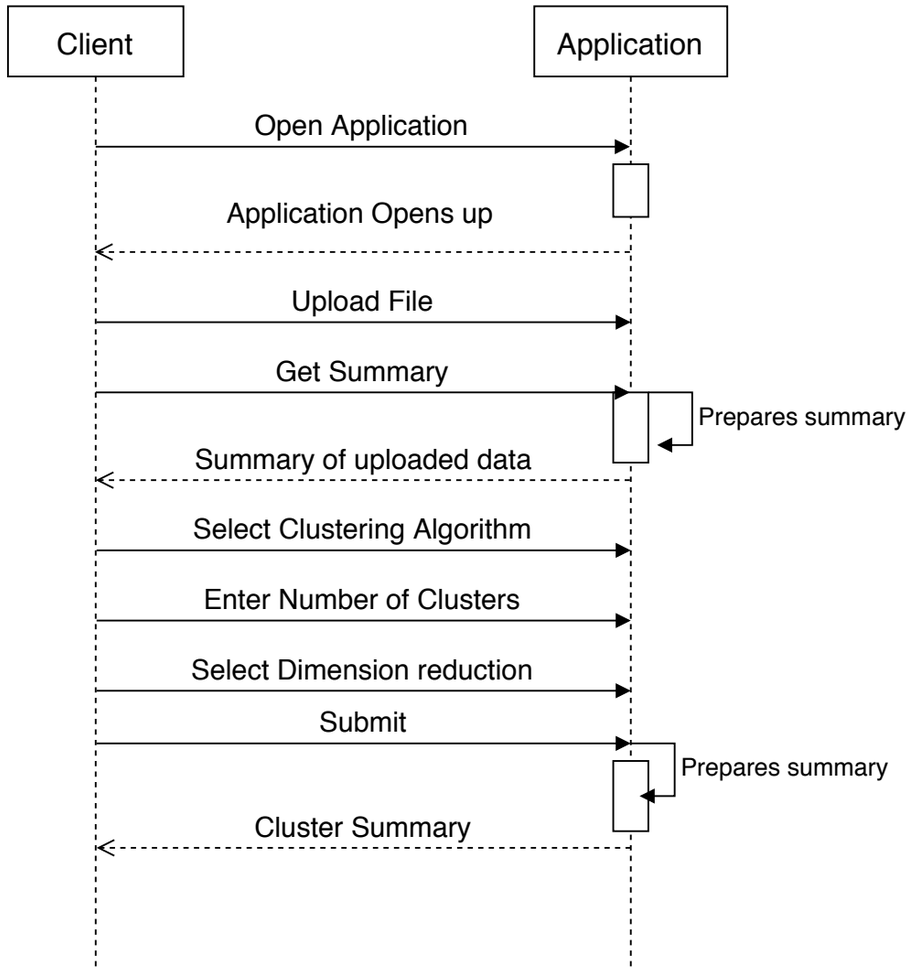


Draft of UML class diagram for the use case where results are downloaded.

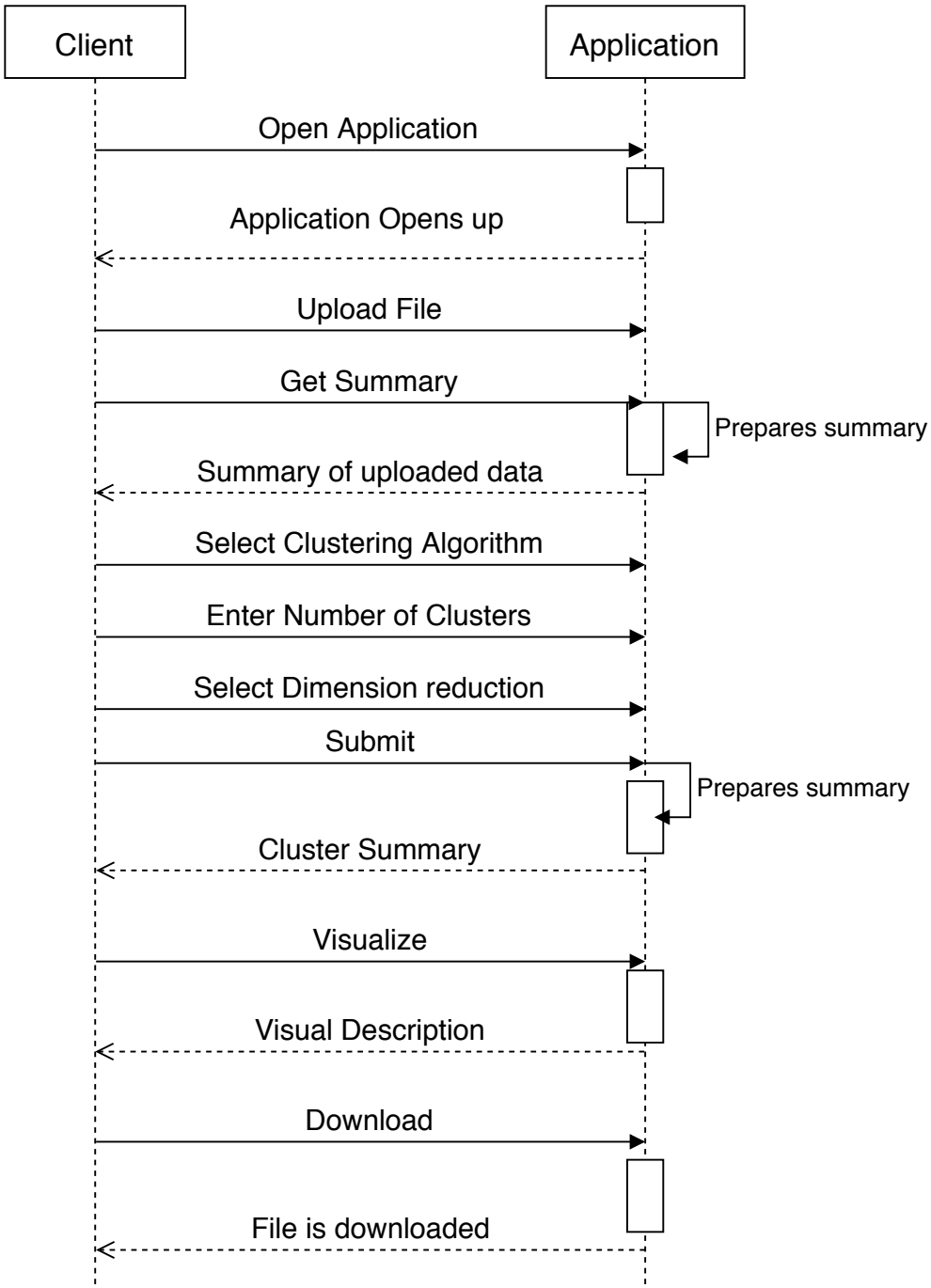
Usecase-1. Clustering without Dimensionality Reduction



Usecase2: Clustering with Dimensionality Reduction



Usecase3: Clustering with visualize and download functionality



Usecase4: During Quit