**Introduction**

**Surround**

Surround is a framework used for machine learning pipelines which are used to help automate the work flow in terms of cleaning, visualizing and interpreting data within the analysis in the python script. The framework is flexible and developed in a way that it could help the data scientists to render the problem within their scope rather than understanding any cryptic code.

**Surround was developed in consideration to following problems:**

* Previous to surround, there was no standard pipeline used for handle the configuration, run the scripts which made it harder for the data scientists to handle the code.
* Previous techniques were vague and were not able to evolve machine learning pipeline without reverse engineering the end outcome

**Working**

**Surround works as follows:**

* The input data is processed by three stages which can be manipulated by the users using the config, the first stage involves processing the raw input data to meaningful data.
* The surround data is an object which includes useful information which is carry forwarded from one stage to another. Users can also add additional variables for transforming the input data as per their preference. Mainly variables are used for error handling, time required for execution, transformation analysis etc.
* Finally, the processed data is implemented in the final stage. In the final stage the surround data is altered to achieve the desired result. Each stage in the process is designed for performing relational activities with the other stages, like the first stage prepares the data which is processed in the last stage of the process.

**Applications of surround**

* Combining multiple models into a single surround solution
* Integrating existing API with the surround solution
* Building an flexible data pipeline without writing hectic C/C++ code using python

**Installation of Surround:**

* The framework of surround is built in python so users need to have a running python application on their systems

**Installation of python**

* Python can be downloaded from the official python website, the software is free to use and can be used for creating frameworks for surround.
* Pycharm a software created by an independent company can be used to write and implement the code which will be used for developing the framework.

**Minimum requirement for using the surround framework**

**Pyaml** (Python module)– Version 17.12.2

**Doit . PyPI** (automated tool designed for to build open source frameworks) - Version 0.31.1

**Installation requirements**

* Before you use the Surround framework a running python application must be present in the system, recommend version is 3.0
* Download the appropriate version of python with the interpreter such as pycharm for your operating system using the python official website.
* For all cases the path where the framework exists must be clearly given, otherwise the framework wont work.
* In many OS the configuration for installing python is automatically detected, if not users can use the system config for manual configuration.

**Running the files using Surround framework**

* Once the appropriate version of python is installed in the OS, users can use the surround framework implemented in python code
* For that, firstly users have to write the code in python and save the code in .py which is an extension for python.
* For executing the code using surround users need to use the following command:
* For instance, the name of the file containing the code is myproject.py. The file needs to be compiled first then executed.

**Installation Requirements:**

**macOS**

**Linux**

**Windows (x64)**

* Before installing the surround framework to windows operating system you will need to be having python v3 in your OS.
* Download the appropriate python version for your OS and you should be able to use the framework.
* Once python is installed and the basic framework setup as shown in the **Running the files using surround framework** you should be able to run the files on the framework on windows.
* For executing the file following command is used:
* Cd/myfolder/myproject surround run dev

**Download and install**

1. **Clone surround AI**

* First install python version 3.6.5 or higher
* Clone the GitHub repository in which the code exists
* Navigate the root of the cloned repository and run the following command:

**Python3 setup.py install**

To run tests following command should be used

**Python3 setup.py test**

1. **Install python 3**

* For installing python 3
* Download the python3 installer from the official python website [www.python.org/downloads](http://www.python.org/downloads)
* Run the setup for python 3 on your OS simply by double clicking on the installer file.
* Once the installation is finished you are good to go.

1. **Surround installation**

**Installation of surround framework: (basic setup)**

The installation of surround framework is done by writing a code in python for executing the surround files in the OS or mainly on the web- interface.

In the code the surround package is created which includes the OS version, minimum executable requirement, author, author details, configuration details etc.

This code should be executed for using the surround framework to the users OS, once the framework is configured to the users system, Surround framework can be used.

**For installing surround framework following steps should be followed:**

* For more information surround –h command is used.
* To compile or initialize the file surround init <path-to-dir> command is used.
* -P command is used for specifying the name and command used is surround init <path-to-dir> -p sample
* -D command is used for the description of the project the command used is surround init <path-to-dir> -p sample -d sample-project
* To run a surround project surround run <project doit task> command is used, for each running program a to-do file with python extension should be provided for functioning instructions.

**Getting started**

**Hello world example:**

* First write a desired code which will have a output of “hello world” and save it in .py extension.
* Save the file in the surround root folder
* In the code file define the operation method of hello stage in the operate() function inherited by surround data.
* The surround data will be initialized only in one stage as surround = surround ([hello stage]).
* The output calls the operate () method of the hello stage and prints hello world on the screen.

**Run hello world:**

Once the code file is saved in the root directory of the framework in python extension the file can be executed using the following command:

* python3 world/init-stage-with-data/main.py

**Troubleshooting:**

* **Syntax error**

This message occurs when a syntax or part of any code is written wrong.

* **Name error**

This message occurs when any variable is defined incorrectly

* **Index error**

You get indexing errors when you are searching for a variable which does not exist

* **Indentation error**

If you forget spaces in the code this error happens

* **For updates or repair patches request the developers.**