Deployment report

Loan Prediction by

Zhaklin Yanakieva

Date: 06/04/2021

Table of contents

Versioning table

Time	Changes	Comments	Version
Week 6 - 9	Creating the documents - initial changes	The document was created and the structure was done	1.0.0
Week 10 - 12	Adding the methods of deployment and explaining; adding graphs	The deployment will be with flask API and after that added a representation of the work	2.0.0

Abstract

Machine learning research typically focuses on optimization and testing on a few criteria, but deployment in a public policy setting requires more. For machine learning models to have real-world benet and impact, effective deployment is difficult. In this report, I describe the implementation of the deployment. My decision of how the project will be delivered to production is further explained and the evaluation is made after that .

Introduction

The goal of building a machine learning model is to solve a problem, and a machine learning model can only do so when it is in production and actively in use by consumers. As such, model deployment is as important as model building. [1] Deployment is part of the Machine Learning Lifecycle. Broadly, the entire machine learning lifecycle can be described as a combination of 6 stages:

Stage 1: Problem Definition

The first and most important part of any project is to define the problem statement. Here, we want to describe the aim or the goal of our project and what we want to achieve at the end.

Stage 2: Hypothesis Generation

Once the problem statement is finalized, we move on to the hypothesis generation part. Here, we try to point out the factors/features that can help us to solve the problem at hand.

[1] https://stackoverflow.blog/2020/10/12/how-to-put-machine-learning-models-into-production/

Stage 3: Data Collection

After generating hypotheses, we get the list of features that are useful for a problem. Next, we collect the data accordingly. This data can be collected from different sources

Stage 4: Data Exploration and Pre-processing

After collecting the data, we move on to explore and pre-process it. These steps help us to generate meaningful insights from the data. We also clean the dataset in this step, before building the model

Stage 5: Model Building

Once we have explored and pre-processed the dataset, the next step is to build the model. Here, we create predictive models in order to build a solution for the project.

Stage 6: Model Deployment

Once you have the solution, you want to showcase it and make it accessible for others. And hence, the final stage of the machine learning lifecycle is to deploy that model.

Modelling

1. Logistic regression:

   Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions

exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).[2]

2. Decision tree:

It is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements. [3]

3. Random forest:

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees.

Results:

1. Logistic Regression

```
MAE: 0.19796954314720813
MSE: 0.19796954314720813
RMSE: 0.44493768456628635
```

2. Decision tree classifier

[2] https://en.wikipedia.org/wiki/Logistic_regression
[3] https://en.wikipedia.org/wiki/Decision_tree

```
MAE: 0.18781725888324874
MSE: 0.18781725888324874
RMSE: 0.433378886060741
```

3. Random forest classifier

```
Mean Absolute Error: 0.238071065989847 76
Mean Squared Error: 0.12558375634517768
Root Mean Squared Error: 0.354377985130535
```

Evaluation of the models

In this paper, I used three machine learning algorithms: Logistic regression, Decision Trees and Random Forest to find the model with the best performance that will work for the loan prediction. The results of both the models are shown in the modelling part above and to get a better understanding of the scores of the two models, I will explain more about them here. The results show that the Random forest is the best fit for this project and should be used for the deployment.

```
From the exploring of the models accuracy:

* Linear Regression score:  0.73 (73%)

* Decision Tree score: 0.79 (79%)

* Random forest score: 91.91 %
```

All of the models showed RMSE values between 0.2 and 0.5 so that they show relatively accurate predictions of the data.

I evaluated the models performances with F1 score metric and the one that is overfitting the least is the Random forest. In the end, I tried three different models and evaluated them using Mean

Absolute Error. I chose MAE because it is relatively easy to interpret and outliers aren't particularly bad for this type of model. The one I will be using for the deployment is the **Random forest**.

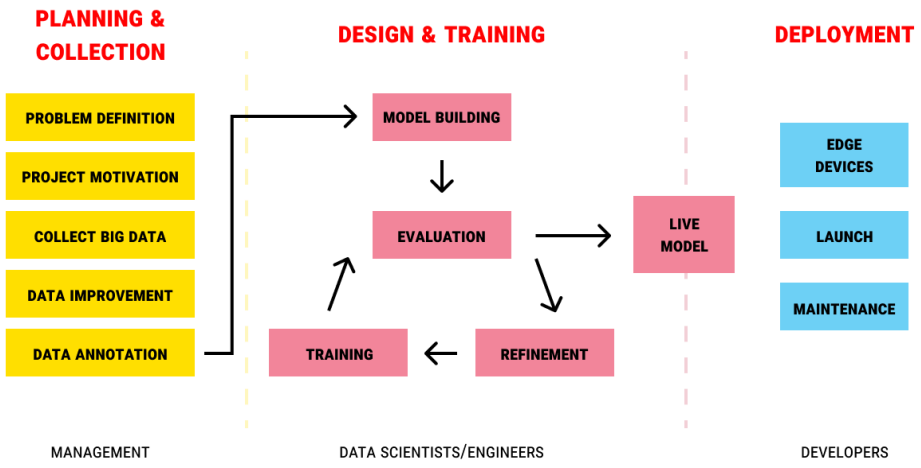
Deployment

The last step before the AI project is done is the deployment. After the training of models, one is selected, the one with the best performance. And to do the deployment, I will use Streamlit, which is a recent and the simplest way of building web apps and deploying machine learning and deep learning models.

First, I will explain the tool I am using for the deployment and after that, I will explain the deployment itself.

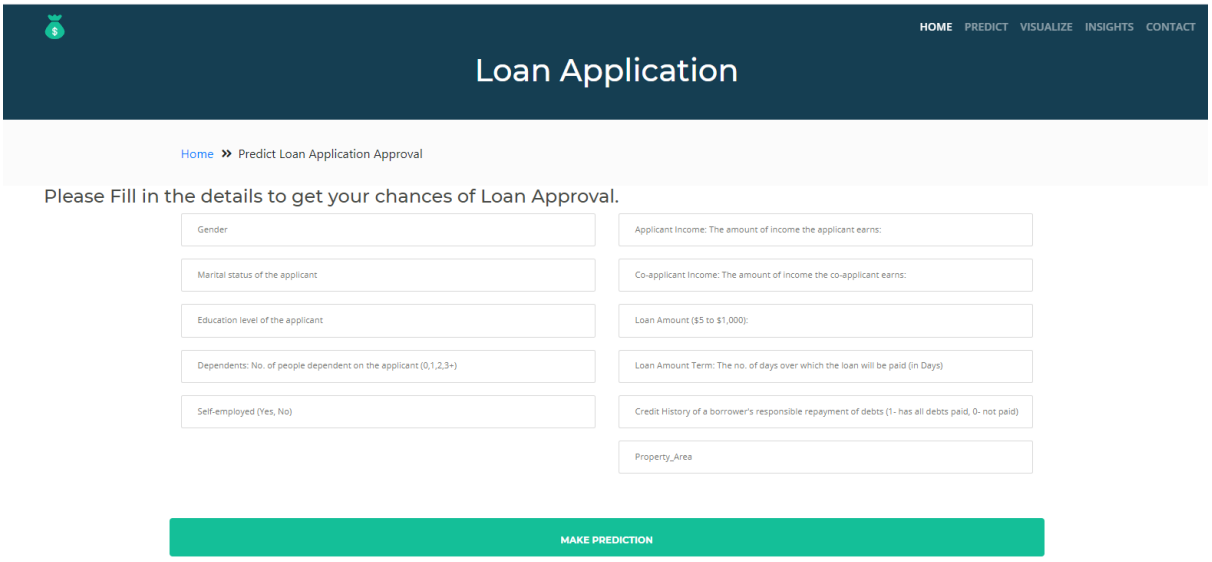
What is deployment?

Model deployment generally contains two parts, frontend, and backend. The backend is generally a working model, a machine learning model, which is built-in python. And the front end part, which generally requires some knowledge of other languages like java scripts, etc. I decided to deploy the project as a web app using Flask.



What is flask API?

Flask is a web framework for Python, meaning that it provides functionality for building web applications, including managing HTTP requests and rendering templates.[4] It is designed as a web framework for RESTful API development.



Deploying Flask App on Heroku:

Heroku is a container-based cloud Platform as a Service (PaaS). Developers use Heroku to deploy, manage, and scale modern apps. Our platform is elegant, flexible, and easy to use, offering developers the simplest path to getting their apps to market.[5]

Tools:

1. scikit-learn
2. pandas
3. numpy
4. flask

---

[4] https://programminghistorian.org/en/lessons/creating-apis-with-python-and-flask
[5] https://www.heroku.com/about

Summary

From a proper analysis of positive points and constraints on the component, it can be safely concluded that the product is a highly efficient component. This application is working properly and will meet all Banker requirements. This component can be easily plugged in many other systems. There have been numbers cases of computer glitches, errors in content and most important weight of features is fixed in automated prediction system, so in the near future the software could be made more secure, reliable and dynamic weight adjustment .In near future this module of prediction can be integrate with the module of automated processing systems. The system is trained on old training dataset in future software that can be made such that new testing date should also take part in training data after some fix time.