COMP2043.GRP

FINAL GROUP REPORT

PROJECT 14

AUTOMATED VALUATION MODEL

TEAM 08

Supervisor:

Anthony Graham Bellotti

Members:

Liwei Fu (scylf1) Jiazhen Ruan (scyjr1) Yixuan Huang(scyyh1)
Wenfeng Li(scywl1) JaeSung Park(shyjp1) Jingwen Zhou (scyjz3)

Date: 2020/04/23

Words: 9285

CONTENT

1.INTRODUCTION AND BACKGROUND 1

1.1 OUTLINE 1
1.2 A BRIEF INTRODUCTION OF THE PROJECT 1
1.3 BACKGROUND 2

2.REQUIREMENT SPECIFICATION 11

2.1 INTRODUCTION 11
2.2 USER TYPES AND CHARACTERISTICS 11
2.3 ACTORS AND USE CASE 13
2.4 USE CASE DIAGRAM 13
2.5 SYSTEM REQUIREMENT 14

3.OVERVIEW OF THE PROTOTYPE 16

4.THE USER INTERFACE OF THE FINAL SYSTEM 20

5.IMPLEMENTATION OF THE SYSTEM 31

5.1 INTRODUCTION 31
5.2 FRONT-END 31
5.3 BACK-END 34
5.4 MODEL 36
5.5 RESULTS OF IMPLEMENTATION 41

6.SUMMARY OF ACHIEVEMENTS 48

7.REflECTIVE COMMENTS ON THIS PROJECT 49

7.1 A BRIEF REVIEW 49
7.2 TECHNICAL PERSPECTIVE 50
7.3 PROJECT MANAGEMENT PERSPECTIVE 52

8.TIME SCHEDULE 56

APPENDIX I

I. MEETING MINUTES I
II. TEST REPORT XVII
III. USER MANUAL XXIII
IV. REFERENCE XXX

1. Introduction and background

1.1 Outline

This final report describes the implementation of the project from the following content:

- A brief introduction of the project
- Background of the Automated Valuation Model (AVM).
- Market research on AVMs
- Some researches of machine learning algorithms
- Users requirements and functional requirements of this project
- Overview of the prototype
- The user interface of the final system
- Implementation of the front-end, back-end and model construction
- Summary of achievements
- Reflective comments on this project

1.2 A brief introduction of the project

Throughout this project, our goal is to build an automated valuation model (AVM) to estimate properties sale price. In this project, our team learned different machine learning algorithms and select the most suitable one, training it and evaluating its performance. For the preparation step, data processing was the most important work, all processes based on existing data from the US property market. After dealing with data, the project went to the modelling process, and our works involved some machine learning algorithms selection and programming language learning. When modelling and testing finished, our model was finally built into a website that can be used for property valuation by different users. Also, the final system should be able to measure the uncertainty in the estimate.

1.3 Background

1.3.1 Introduction to AVMs

Automated valuation models (AVMs) are used to provide predicted properties valuation in a computer-based way based on statistics, data mining and computing methodologies [1]. AVMs aroused by financial institutions and property listing sites to help value properties for sale and are now widely applied for users such as residential buyers, mortgage providers and other financial organizations. According to George [1], AVMs are the convenient, fast and cheap substitution of full valuation. Using AVM to generate a predicted value can only cost a few minutes, whereas it will take a human valuer a couple of hours or even days to complete this task. The costs of applying AVMs is also much lower [2]. Traditionally valuations are undertaken by some surveyors and human experts. However, the varies professional level of people often leads to many errors in the valuation results. Also, the artificial valuation process takes much time and may lead to a decrease in efficiency. Therefore, to optimize these weaknesses, the valuation system which based on AVM will be applied to yield transparent, objective and more accurate prediction.

1.3.2 Marketing research and current situation of AVMs

Last thirty years have witnessed the development of AVMs. They are now widely used by real estate professionals, government and the general public. They have been used for appraising local taxes, estimating relocation compensation and determining capital adequacy ratios in banks. However, the accuracy of the AVMs is still unconvincing to many professionals and average users. This is mainly because that AVMs are considered as lacking the experience and judgement of a human expert. Also, AVMs are not suitable for unique property valuation, as the model requires non-standard data [2]. Therefore, AVMs still need time to obtain the professionals' trust. Robson and Downie reported that 71% of the valuers suggested that AVMs were not an adequate tool for loan valuations due to the lack of physical inspection. Besides, nearly 90% of the valuers believe physical valuations have better accuracy than AVMs [1].

Figure1.1 shows the percentage of a US AVM valuations falling within 5% and 20% of sales prices. What stands out for the lower valued properties, less than US$ 50K, only 7% of the valuations were within +/-5% of the achieved sales prices, rising to 23% being within +/- 20% of the sales price. However, several factors account for the difficulty in estimating values for such low-priced properties. Consequently, looking at the raw inaccuracy figures without an understanding of the data context can result in misleading conclusions. For

property values over US$ 100K, the figure shows that over 80% of values are within +/- 20% of the sales price [1].

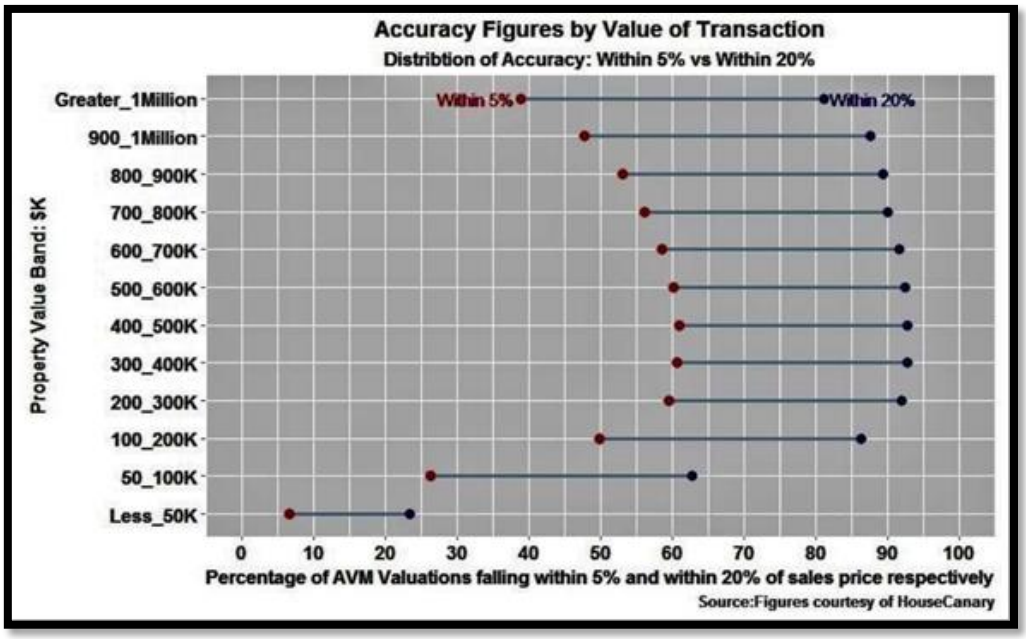


Figure 1.1 Accuracy by the value of the transaction

Although the US models provide results which could be considered as acceptable levels for AVM valuations, a purely derived statistical or data-mined valuation risks being widely off the mark, there remains a requirement for professional judgement to augment model-based valuations, thereby arriving at a more broadly considered valuation estimate.

Facing these current shortcomings of AVMs, our team made some improvements. We applied sevral accurate machine learning algorithms in the valuation model. Also, the dataset was processed so that it contains less unique properties. The section of project implementation (chapter 5) will describe this in more details.

1.3.3 Algorithms research

Linear regression

Linear regression, whose principle is the least square, is one of the useful models to build AVMs.

Following is the detailed explanation:

Suppose there exists a set of data

$$D = \{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), \dots, (x_n, y_n)\}$$

$$x_i = \{x_{i1}, x_{i2}, x_{i3}, x_{i4}, \dots, x_{in}\}$$

The linear regression model can be represented as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \cdots + \beta_n x_{in} + \varepsilon \,, \varepsilon \sim N(0, \sigma^2)$$

$$\sigma^2 = \frac{SSE}{n-2} = \frac{S_{yy}}{n-2}(1 - r^2)$$

$$S_{yy} = \sum (y - \bar{y})^2$$

Where $\beta_0, \beta_1, \beta_2, \dots., \beta_n$ are regression parameters and ε is deviation.

In the matrix form

$$y = Xb + \varepsilon$$

Where X is

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ x_{21} & \cdots & x_{2n} \\ x_{31} & \cdots & x_{3n} \end{bmatrix}$$

Where b is

$$b = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}$$

Moreover, ε is the deviation.

To get the result b:

$$L(b) = \varepsilon^T \varepsilon$$

$$= (y - Xb)^T (y - Xb)$$

$$= y^T y - 2y^T Xb + b^T (X^T X) b$$

$$\frac{\partial L(b)}{\partial b} = -2y^T X + 2b^T (X^T X) = 0$$

$$b^T (X^T X) = y^T X$$

$$(X^T X)b = X^T y$$

$$b = (X^T X)^{-1} X^T y$$

Although linear regression analysis has its shortcomings, including the inability to adequately handle interactions between variables, independence of errors and multicollinearity [1], it is still widely used. According to Su, Yan and Tsai [3], as the base of many modern modelling tools, linear regression is famous for several reasons. For instance, the model parameters are easily interpretable. Besides, linear model theories are well established with mathematical elegance.

Neural Network

Artificial Neural Network (ANN) is one of the most powerful and extensively used in the subfield of machine learning called deep learning [4].

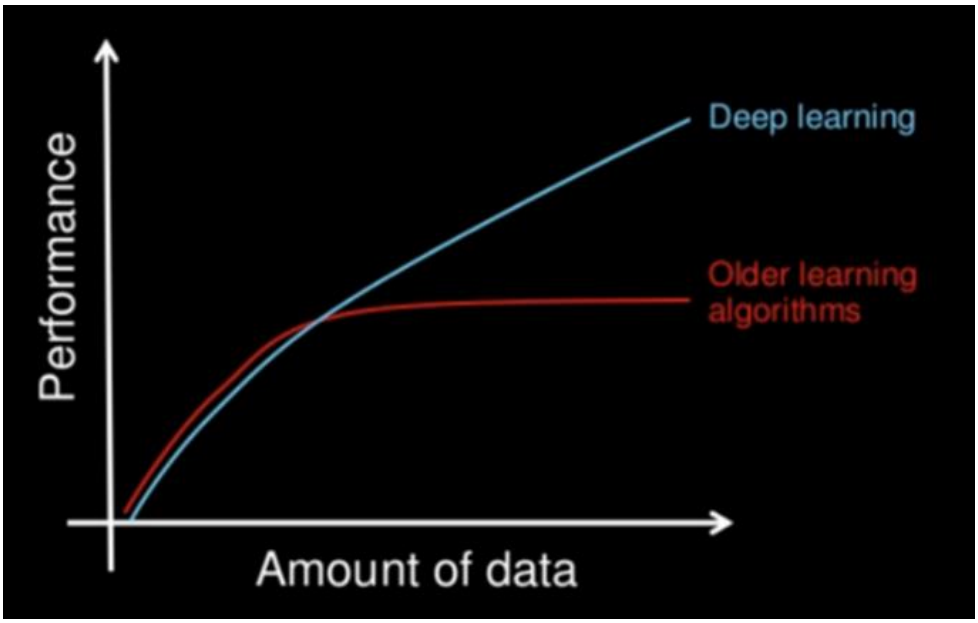


Figure 1.2 Deep learning is powerful.

Briefly, a Neural Network is defined as a computing system which is consisted of several simple but highly interconnected elements called neurons, and they are organized in layers to process information using dynamic state responses to external inputs.

The ANN is typically organized in layers which are being made up of many interconnected neurons. Usually, a Neural Network may contain three layers:

<u>Input layer</u>

The input layer is used to receive as input the values of the descriptive attributes for each observation. In general, the number of input nodes is dependent on the shape of training data which means the number of input nodes in an input layer is equal to the number of explanatory variables. However, to capture the bias term, there will be one additional node sometimes [5].

$$Number\ Of\ Neurons = Number\ Of\ Training\ Data\ Feature + 1$$

Figure 1.3 Number of neurons in the input layer.

Nodes of the input layer are passive, which means they do not change data, and they receive a single value on their input and duplicate the value to their many outputs.

<u>Hidden layer</u>

The hidden layer will connect to output nodes or other hidden nodes by outgoing arcs. The actual processing of the hidden layer is done via a system of weighted connections.

The number of hidden layers could be one or more, all of that will multiply the values entering a hidden node by weights which is a set of predetermined numbers stored in the program. The weighted inputs are then added to produce a single number.

<u>Output layer</u>

The output layer is responsible for producing the result. And there must always be one output layer in a Neural Network.

Output layer receives connections from hidden layers or the input layer. An output value that corresponds to the prediction of the response variable will be return by the output layer.

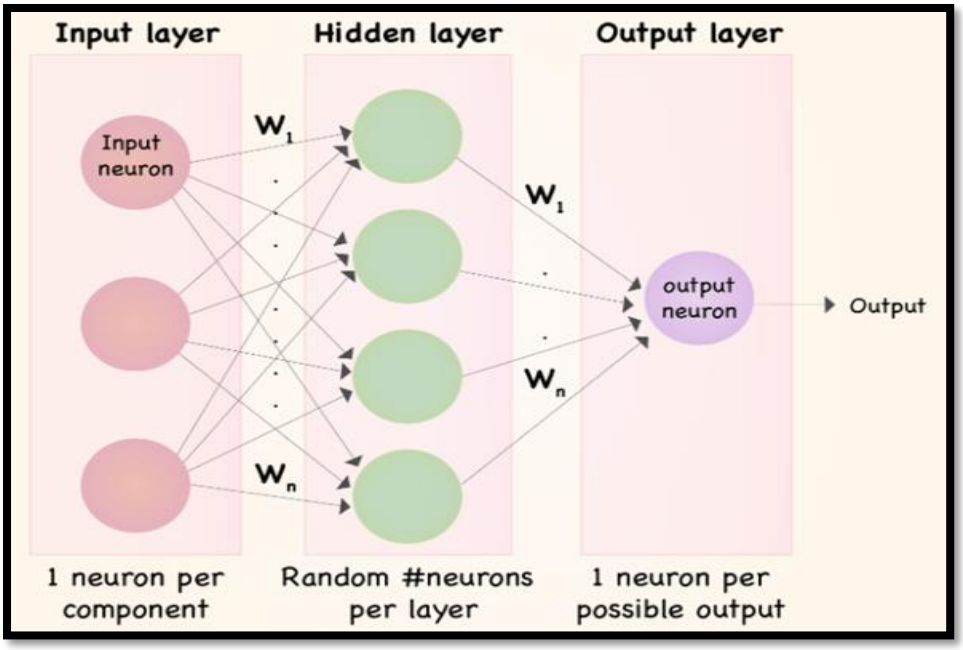


Figure 1.4 Neural Network layer

<u>Training a Neural Network</u>

(1). Loss function

In training, the loss function can be used to estimate the error and to compare and measure the predicted result against the correct result. There are many loss functions, and one of the most common ones is the Mean-Square Error (MSE).

MSE is calculated as follows:

$$MSE = \frac{1}{n}\sum_{i}^{n}(y_i - \widehat{y_i})^2$$

Where y_i is the desired output, and y_i hat is Neural Network output.

(2). Backpropagation

The information of error or loss, which is calculated by loss function will be propagated backwards, starting at the output layer, the loss information will be propagated to all the neurons of the hidden layer. However, the neurons of the hidden layer only receive a fraction of the total signal of the loss information. This process repeats until all the neurons in the Network have received a loss signal that describes their relative contribution to the total loss.[6]

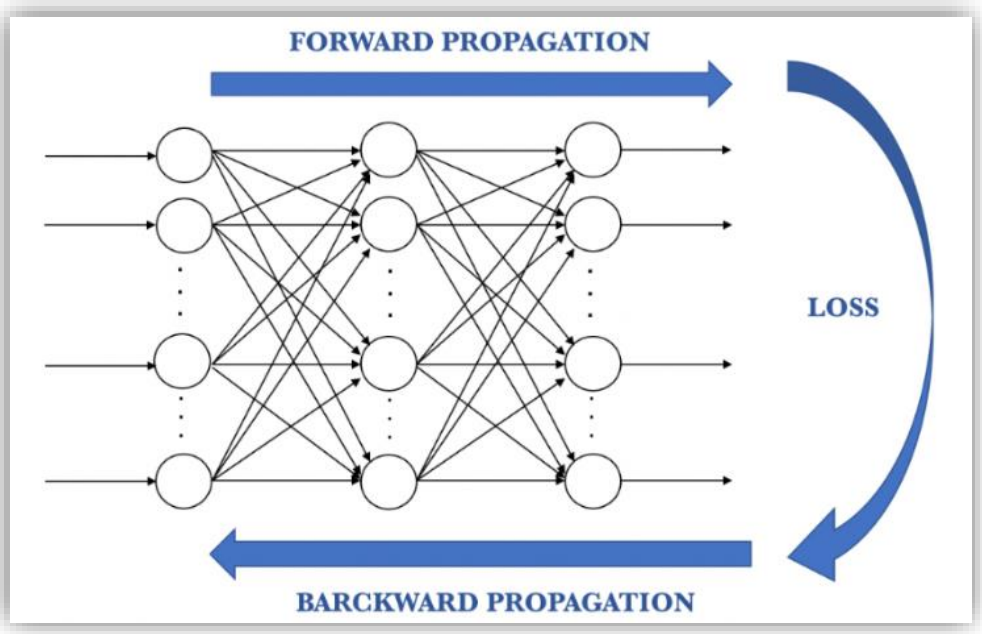


Figure 1.5 Backward propagation

(3). Gradient descent

The weights of connections between neurons should be adjusted when spread the loss information back. The purpose of this is making the loss as close as possible to zero before the Neural Network is used to prediction. For that, gradient descent can be used to changes the weights in small increments with the calculation of the gradient of the loss function. But calculating the value of the loss function for all samples may take much time, and it is inefficient to do that every iteration [6]. Therefore, Mini-batch gradient descent can be used in Neural Network Learning. The method for adjusting the weights is not different, but it will save much time from calculating the exact derivative.

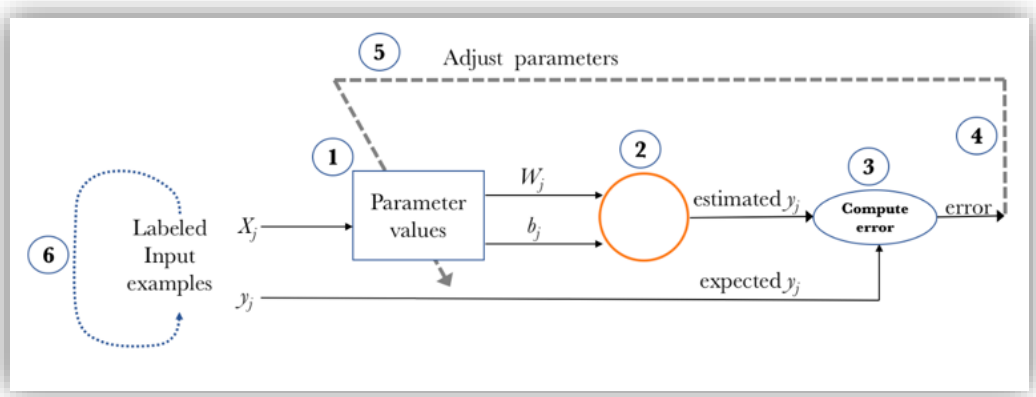


Figure 1.6 The process of Neural Network learning

Random Forest

Random Forest is a flexible algorithm with plenty of advantages. According to Leo Breiman and Adele Cutler [7], the accuracy of the Random Forest algorithm was unexcelled at that time. It is capable of handling a large number of input variables without eliminating a variable. Moreover, an internal unbiased estimate of the generalization error is generated in the building process.

Random Forest is a type of ensemble learning, which organizes hundreds of decision trees to complete the classification. The building of the Random Forest model involves two processes.

The first process is the construction of decision trees. Leo Breiman and Adele Cutler [7] introduced that the construction of each tree involves bootstrap sample, which suggests that the training set (Randomly pick 66.6% of the total data set) of each tree is slightly different to the others. The left data set (33.3% of the data set that is not picked as the training set) are called out-of-bag (OOB) sample. Put the cases in the OOB down the decision tree and get a classification. Suppose most of the decision trees voted class j for case n. The proportion of times that j is not equal to the correct class of case n averaged over all cases is the OOB error estimate. In this way, a running unbiased estimate is completed during the process of constructing decision trees.

The second process is to calculate the proximities. Put all the cases down the tree, and if two cases occupy the same terminal node, their proximity is increased by one. At the end of this process, proximities are capable of being normalized by diving by the number of trees [7].

According to Philipp Probst, Marvin Wright and Anne-Laure Boulesteix [13], the hyperparameters of the random forest algorithm are the number of the trees, the number of variables drawn randomly for each split and the splitting rule. The increase in the number of trees can improve the performance of the entire model. However, the training process is also increased. Thus, the selection of the number of trees is related to the balance between time and accuracy. The number of variables drawn randomly for each split is relevant to the tree constrains. Each tree needs to consider more samples at each node as the number of the variables increase.

<u>How are features chosen as candidates for each tree?</u>

a) Arrange all the feature variables in Random Forest model in descending order of VI (Variable Importance).

b) Determine the proportion of deletion variables. Excluding a corresponding portion of the unimportant features from the current feature set to get a new feature set.

c) Build a new random forest model with the new feature set. Count the VI of each feature and order them.

d) Repeat steps a to c until there are only m features left.

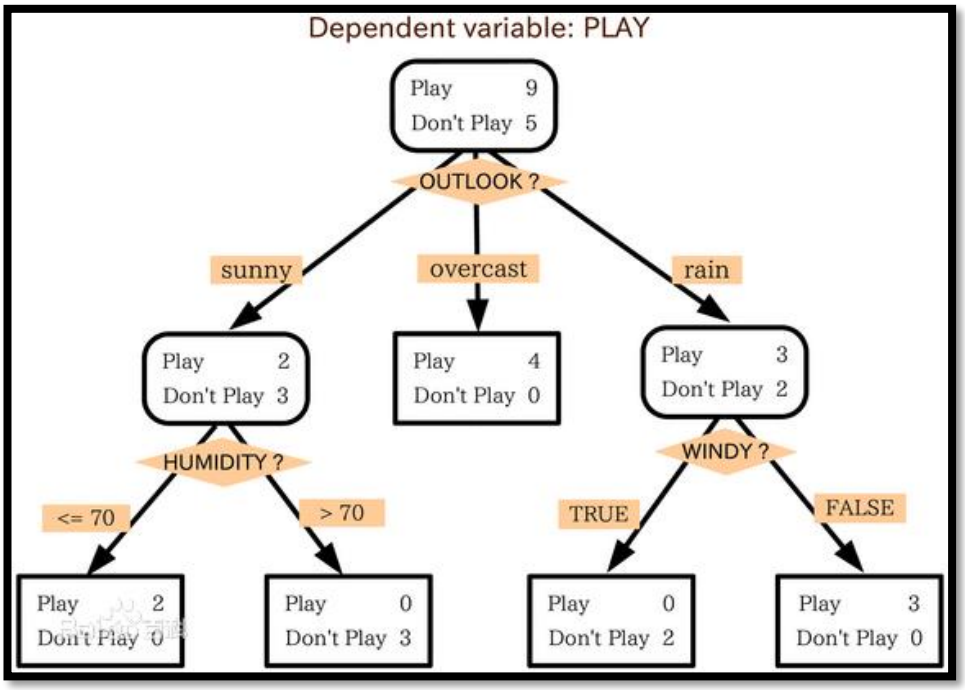


Figure 1.7 Decision tree sample

Available at:

https://baike.baidu.com/item/%E5%86%B3%E7%AD%96%E6%A0%91/10377049?fromtitle=Decision%20Tree&fromid=11319683&fr=aladdin

2. Requirement specification

2.1 Introduction

Product perspective

The final system can be used as an external function in the sales website or bank loan business. Also, it can be completely independent of other systems and form a complete valuation system for different users.

Product function

The product is an automatic valuation system based on AVM, which enables users to obtain the forecast of property price by submitting the property description form. The product is aimed at a wide range of users, and according to the different valuation purposes of users, the system provides multiple valuation entrances to provide different degrees of valuation results. Following is a detailed description of different user types.

2.2 User types and Characteristics

The system is mainly used for the following categories of users

1）Real estate agency

Real estate companies need to provide accurate prices to customers while ensuring their interests. They need to rapidly compare price differences in different regions to achieve precision marketing. Providing customers with an accurate price is their pursuit; gaining a good reputation in the industry is their goal. Therefore, they need AVM to help them achieve these goals.

2）Property Buyer

To compare the valuation from experts with the system valuation，ensuring personal benefits. Some individual buyers entrust private institutions to do the valuation or entrust an intermediary to buy a property. However, these channels may not guarantee the maximum range of interests of individual buyers. Therefore, these individual buyers can use the AVM to compare prices on all sides, making sure they don't lose too much.

3）Mortgage lender

1) Pre-evaluation before loan

Through real-time automatic valuation results, directly communicate with customers about loan limit, improving work efficiency and customer experience.

2) Internal appraisal and risk control

Compared with the external appraisal company, more objective to judge the market price of properties and minimize the loan risk.

3) Revaluation after loan

The property is collateral, and the value of it influences the amount of loan. If property value changes beyond the limit after the loan, the mortgage lender should detect the potential regional risks earlier and timely do emergency treatment to reduce the loss of assets.

4）Evaluation authority

In the process of the rapid development of financial technology, the banking industry requires traditional evaluation companies to provide an efficient and low-cost evaluation in response to the acceleration of the banking business process. Faced with the high requirements of partners, evaluation institutions can provide informationized and standardized evaluation results with the help of AVM.

5）Tax official

To strengthen the administration of tax holdings and raise the level of refined management of tax holdings. Also, it is essential to reduce disputes and achieve tax fairness effectively.

6）Judicial auctioneer

To provide accurate property valuation result for the auction, increasing the auction premium rate, improving the transaction rate, reducing the time wasted in the auction process. Before the auction, if bidders want to know the current market value of the property through the necessary information of the property, they can evaluate the property through AVM.

2.3 Actors and Use case

Actor	Use case
Real estate agency	▪ Login ▪ Logout ▪ Register ▪ Check the account information ▪ Choose the purpose of a valuation ▪ Fill the website form of the property description ▪ Download the complete file of the property description ▪ Upload the complete file of the property description ▪ Get the valuation results
Property buyer	
Mortgage lender	
Evaluation authority	
Tax official	
Judicial auctioneer	

2.4 Use case Diagram

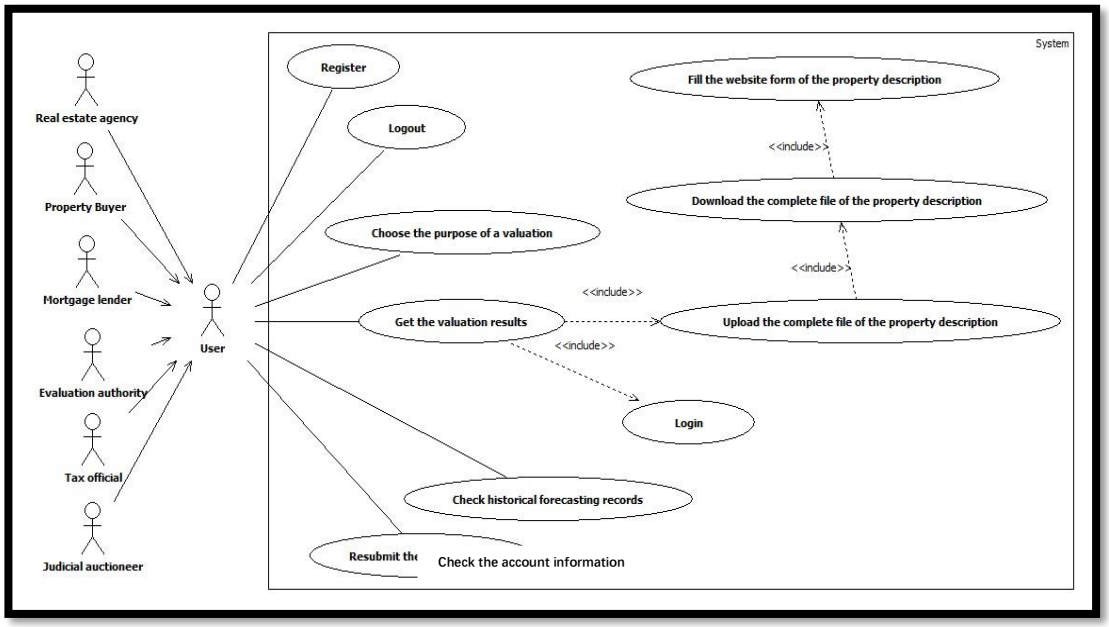


Figure 2.1 Use case diagram

2.5 System requirements

Functional Requirements

1) Register & Login

    - The frontend page needs to allow all the users to register and login to the system.
    - The system should allow each user to register with a unique username and matching the password.
    - The system should validate the login information from users.
    - If any login error occurs, for example, invalid username or password, the system should allow the user to give another attempt.
    - The system should ensure the security and privacy of personal information of users.
    - The system should allow users to log out.

2) Data & File uploading
    a) The web page should allow users to download a template file contains all the properties of the house which are needed for valuation
    b) The system should allow users to upload their house properties forms and pass them to the valuation model.
    c) The web page should also allow users to upload their data by filing a provided questionnaire.

3) Frontend Appearance & Required Display:
    a) The system should show some necessary background information about the program and links to the corresponding method page on the homepage.
    b) In the page of analyzing, the system should allow the users to upload their properties forms or fill in the questionnaire.
    c) The system should display a final result of analyzing the house to the user, which includes the predicted value and the most influential effect on the value of the provided properties.

4) Model
    a) The model should provide different calculating deviation measure to different roles, which is using to give every user

5) User management
    a) The system should allow users to check their account information.

6) Database

    a) The database should store all the login information of users.
    b) The database should store all the house information which is uploaded by users.
    c) The database should store the query record of users.

Nonfunctional Requirements

1) Reliability
   a) The system should be stable for getting the correct information, for example, the house properties.

   b) The system should be supportive for expected around 300,000 users.
   c) The system should prompt for input and check out the data, avoiding abnormal data.
   d) The probability of not completing the service due to the failure of the software system is less than 5‰.
   e) The system is robust, should be able to deal with the system in the process of running a variety of abnormal conditions, such as human operation error, illegal input data, hardware failure, the system should be able to handle correctly, appropriate avoidance.
2) Usability
   a) In 95% of failures, the system takes up to 20 seconds to restart.
   b) The system should support 40,000 users to access at the same time.
3) Maintainability
   a) After receiving the modification request, the ordinary modification should be completed within one to two days; Major requirements or design modifications after assessment should be completed within one week.
   b) Any method code should not go over than 200 lines.

4) Security & Privacy:
   a) Information should be secure and reliable, avoiding privacy leak.
5) Compatibility
   a) The system should support a database to store all the information of users.

3. Overview of the prototype

Introduction

The purpose of prototyping is to give all developers a general understanding of the functionality and framework of the system in the early stages of development. In a subsequent development, we continued to improve the system as problems arise. Therefore, there are some deviations between the prototype and the final user interface. Chapter 4 will show the final user interface, while this chapter mainly shows the initial design of the system.

Overview of the design

1) Figure 3.1 shows the homepage of the website. When the prototype was designed, the plan here was allowing the user to choose their purpose of valuation. Besides, users can learn some knowledge about AVM and machine learning.

2) Before users go to the valuation part, they must login to the website, and the system also provides the register page for users that do not have the account. Figure 3.2 and figure 3.3 show the login page and register page.

3) In this page (Figure 3.4), users can fill the form which describes the characteristic of the properties. Instead of filling the form online, they can also download the form directly, complete it and submit the document to the website, then get the prediction results.

4) Figure 3.5 is a page which displays the valuation result, and it shows the final price of the user's property.

Figure 3.1 Homepage

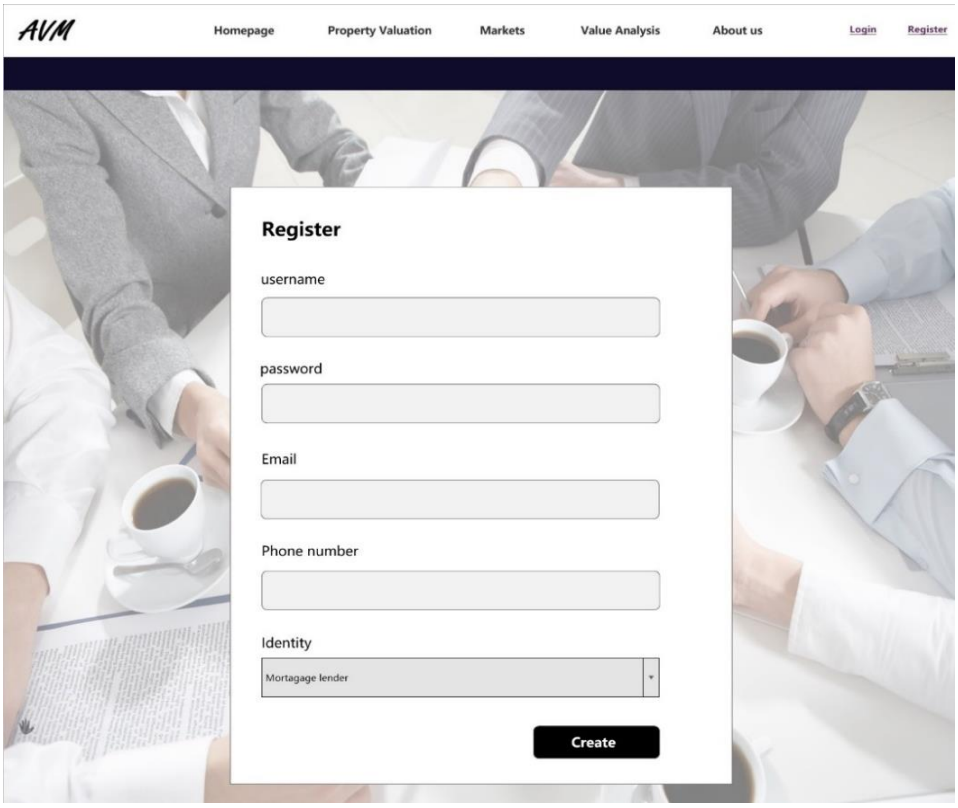


Figure 3.2 Register Page

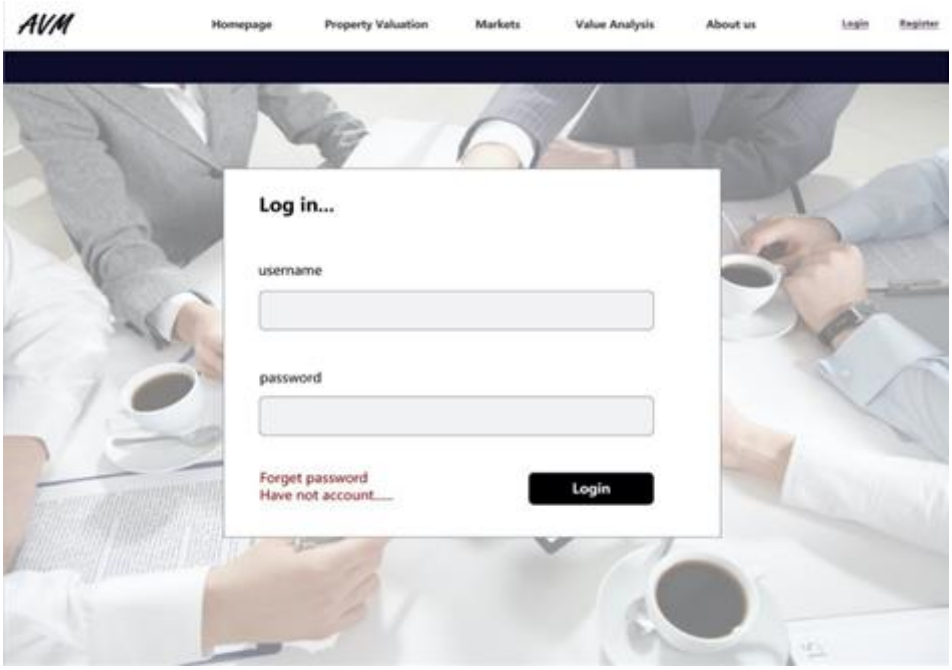


Figure 3.3 Login Page

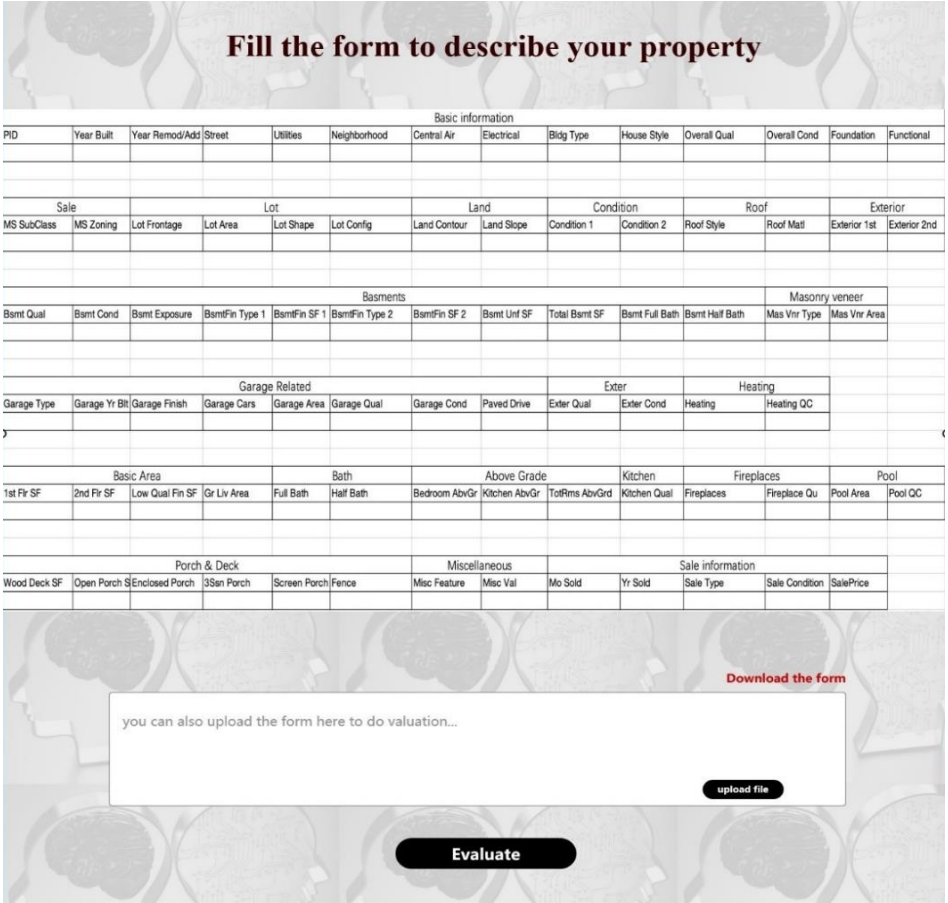


Figure 3.4 Properties form page

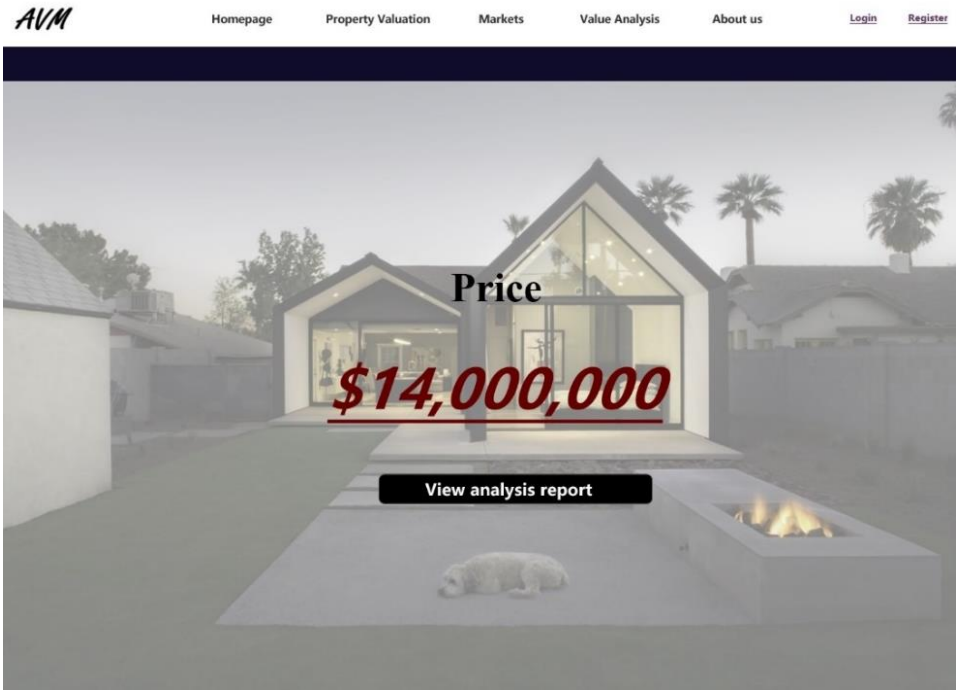


Figure 3.5 Result page

4. The user interface of the final system

This chapter mainly shows the user interface of the system. The details of each page will be described in the 'Front-end' part of chapter 5. Figure 4.0 is a diagram showing the flow between web pages, where all the grey boxes represent pages.

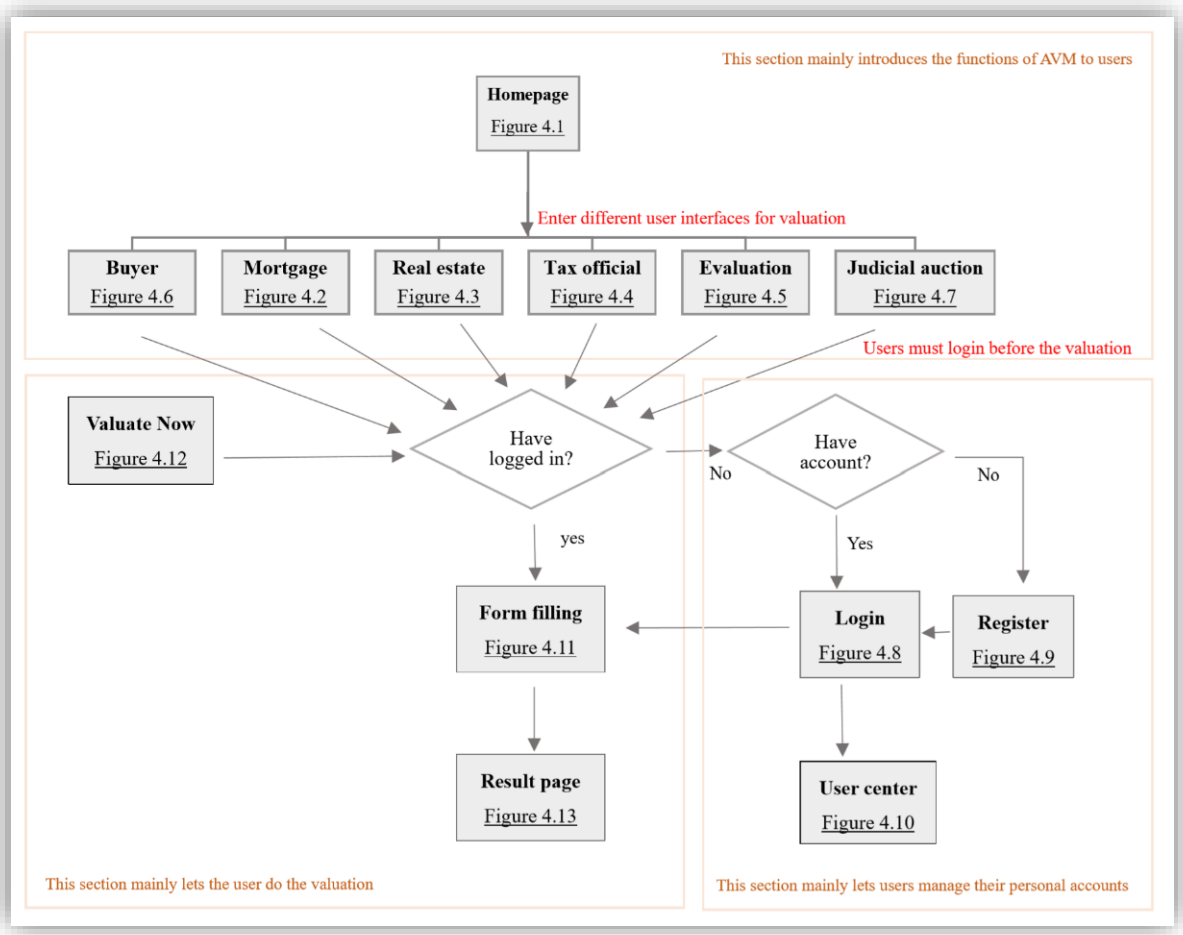
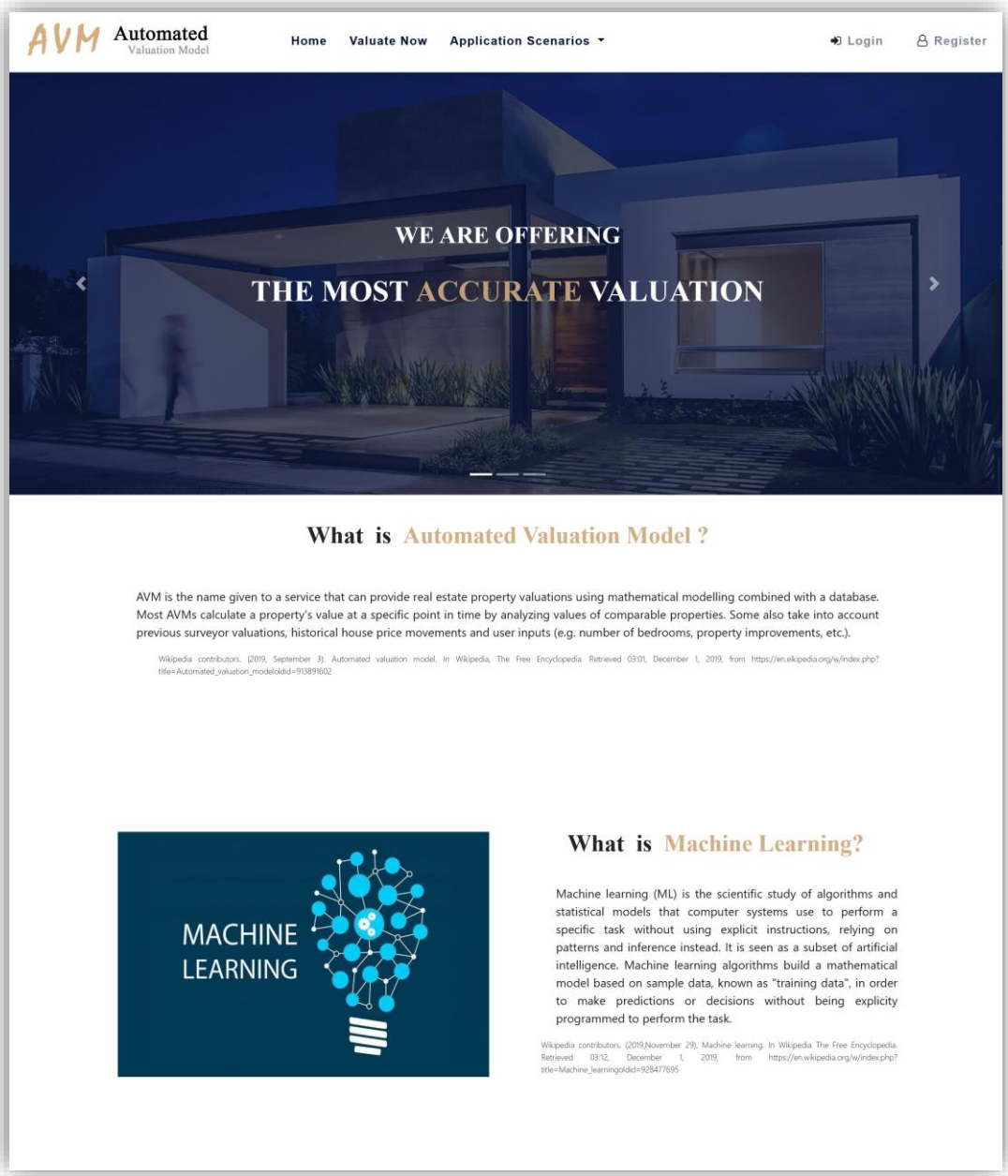


Figure 4.0 Overview of the website


AVM Automated Valuation Model
Home
Valuate Now
Application Scenarios
Login
Register
WE ARE OFFERING
THE MOST ACCURATE VALUATION
What is Automated Valuation Model ?
AVM is the name given to a service that can provide real estate property valuations using mathematical modelling combined with a database. Most AVMs calculate a property's value at a specific point in time by analyzing values of comparable properties. Some also take into account previous surveyor valuations, historical house price movements and user inputs (e.g. number of bedrooms, property improvements, etc.).
Wikipedia contributors. (2019, September 3). Automated valuation model. In Wikipedia, The Free Encyclopedia. Retrieved 03:01, December 1, 2019, from https://en.wikipedia.org/w/index.php?title=Automated_valuation_model&oldid=913891602
MACHINE LEARNING
What is Machine Learning?
Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicity programmed to perform the task.
Wikipedia contributors. (2019,November 29). Machine learning. In Wikipedia The Free Encyclopedia. Retrieved 03:12, December 1, 2019, from https://en.wikipedia.org/w/index.php?title=Machine_learning&oldid=928477695

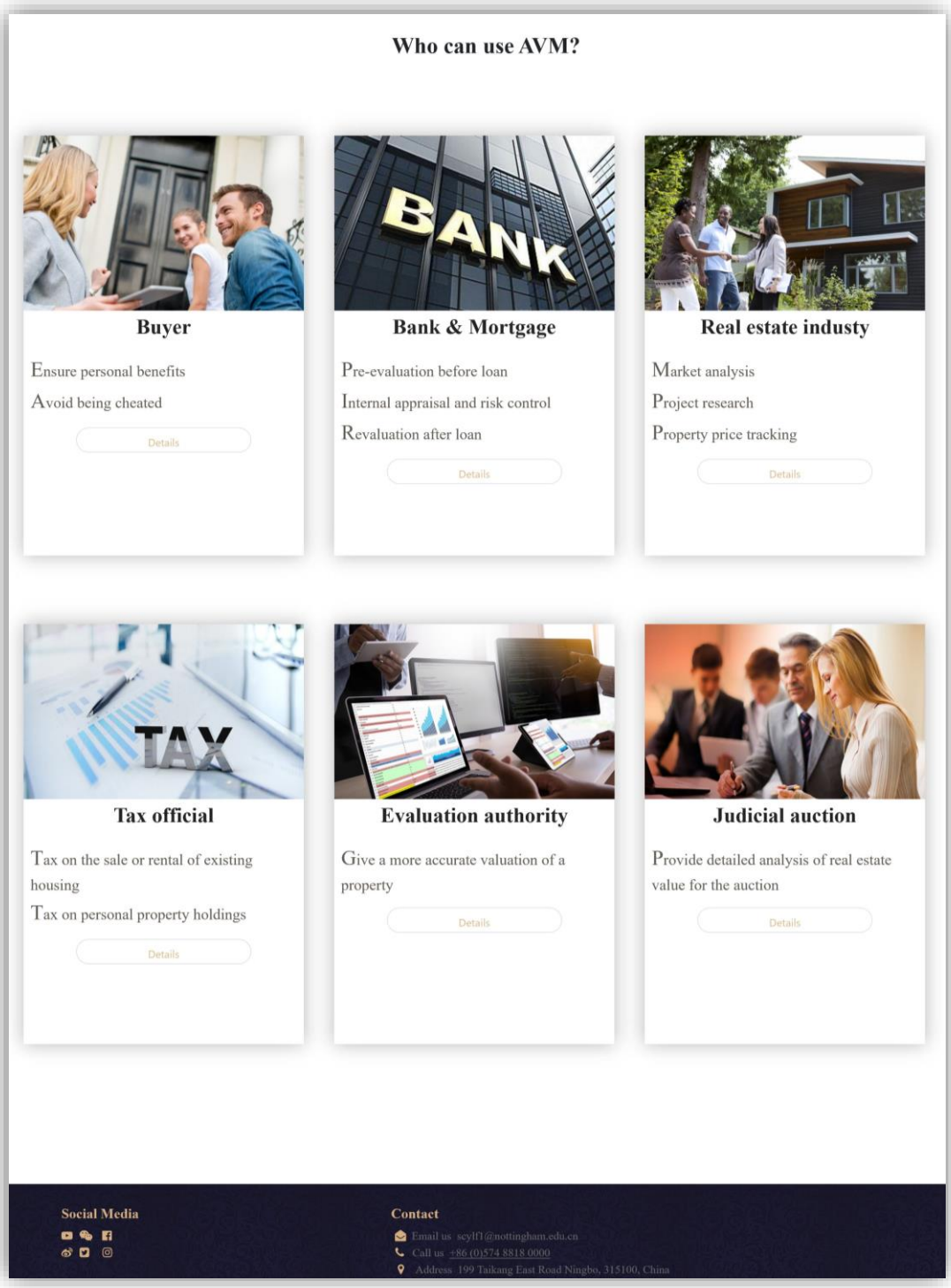


Figure 4.1 Homepage

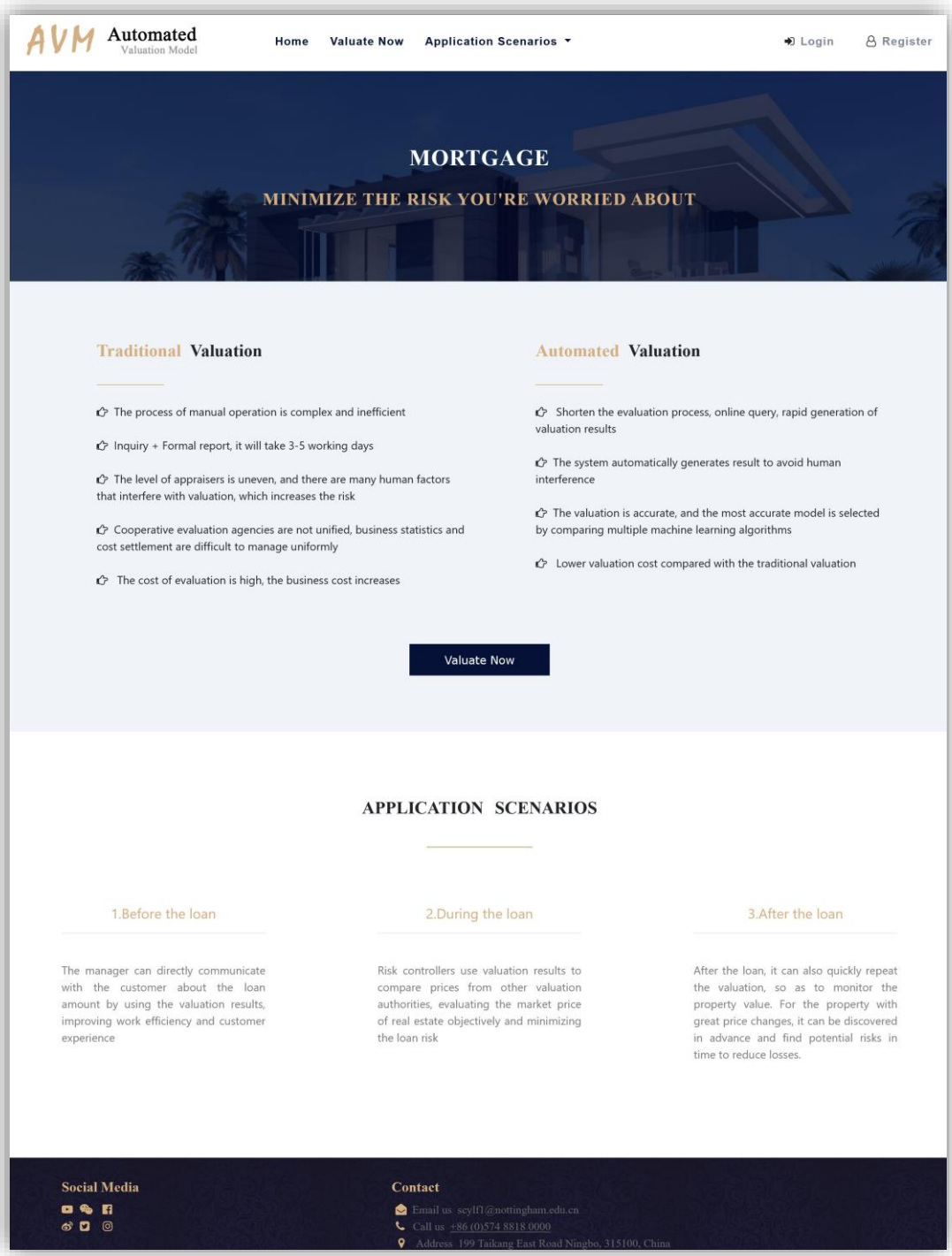


Figure 4.2 Product Applications Scenarios-Mortgage

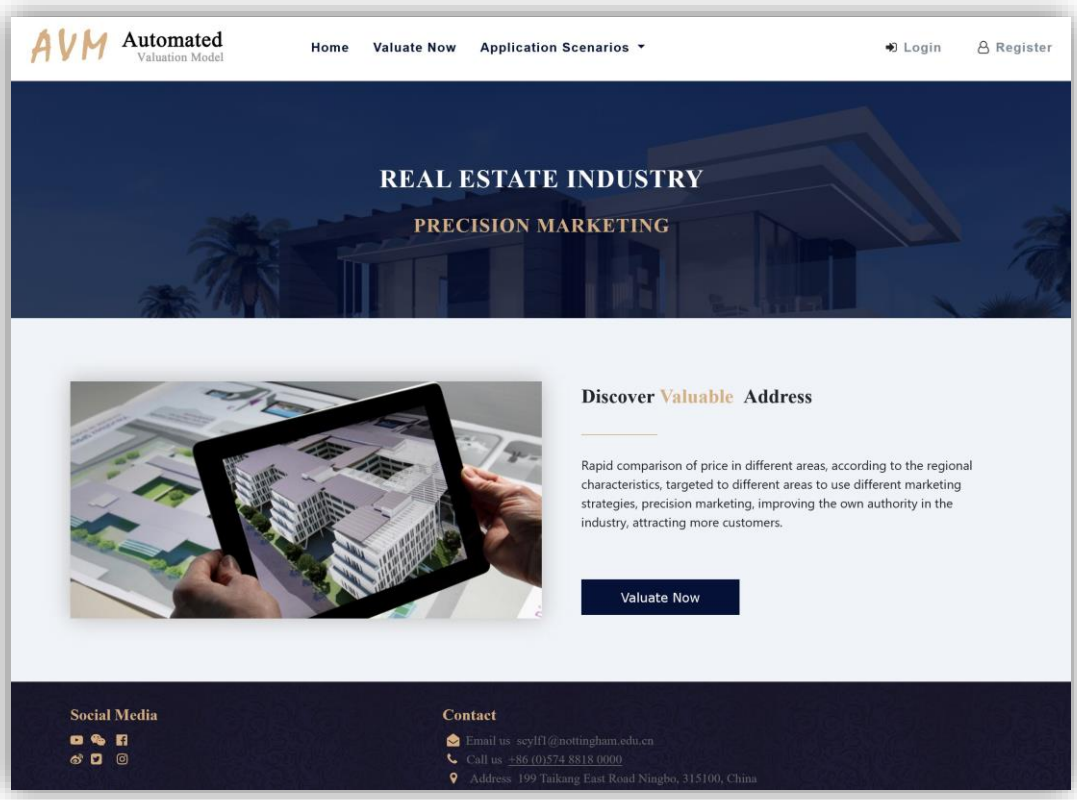


Figure 4.3 Product Applications Scenarios-Real Estate

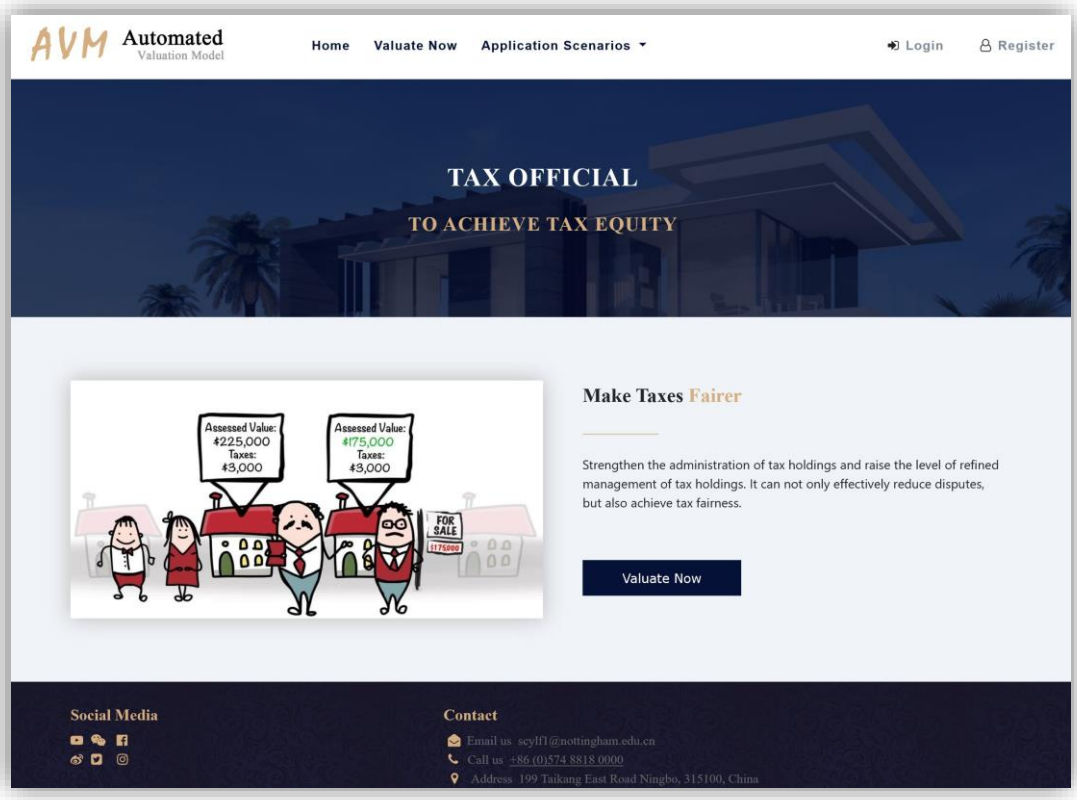


Figure 4.4 Product Applications Scenarios-Tax official

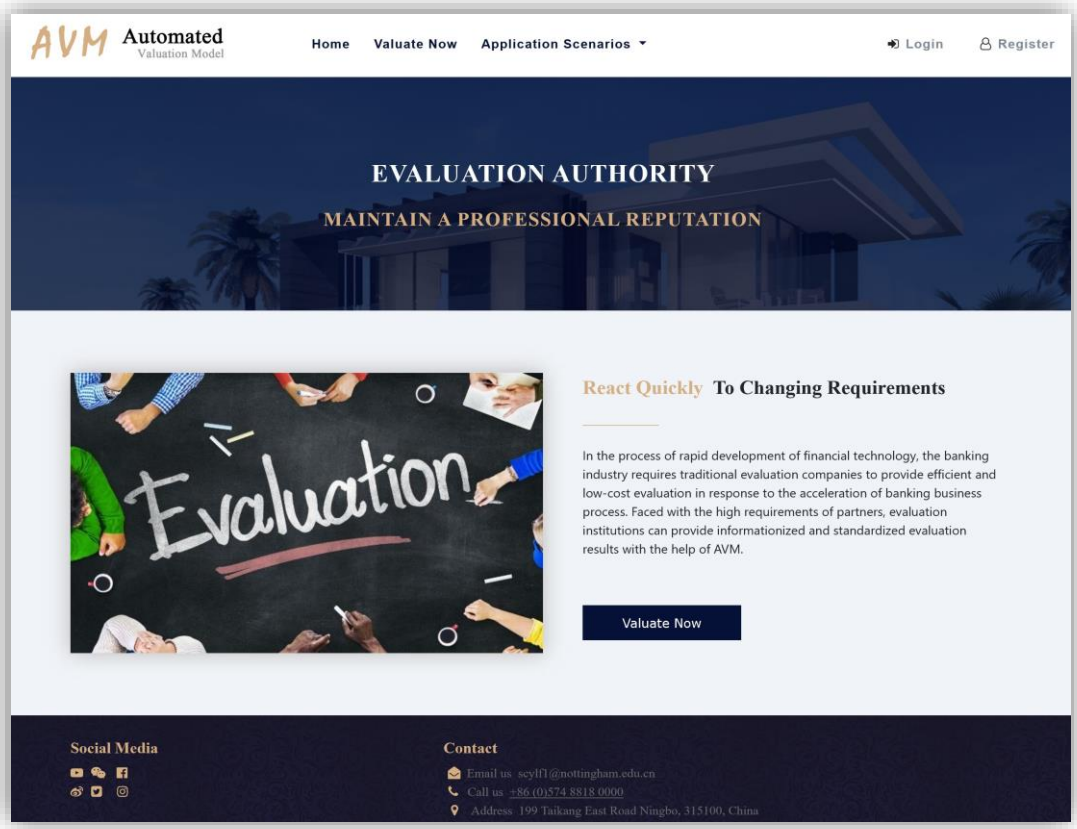


Figure 4.5 Product Applications Scenarios-Evaluation authority

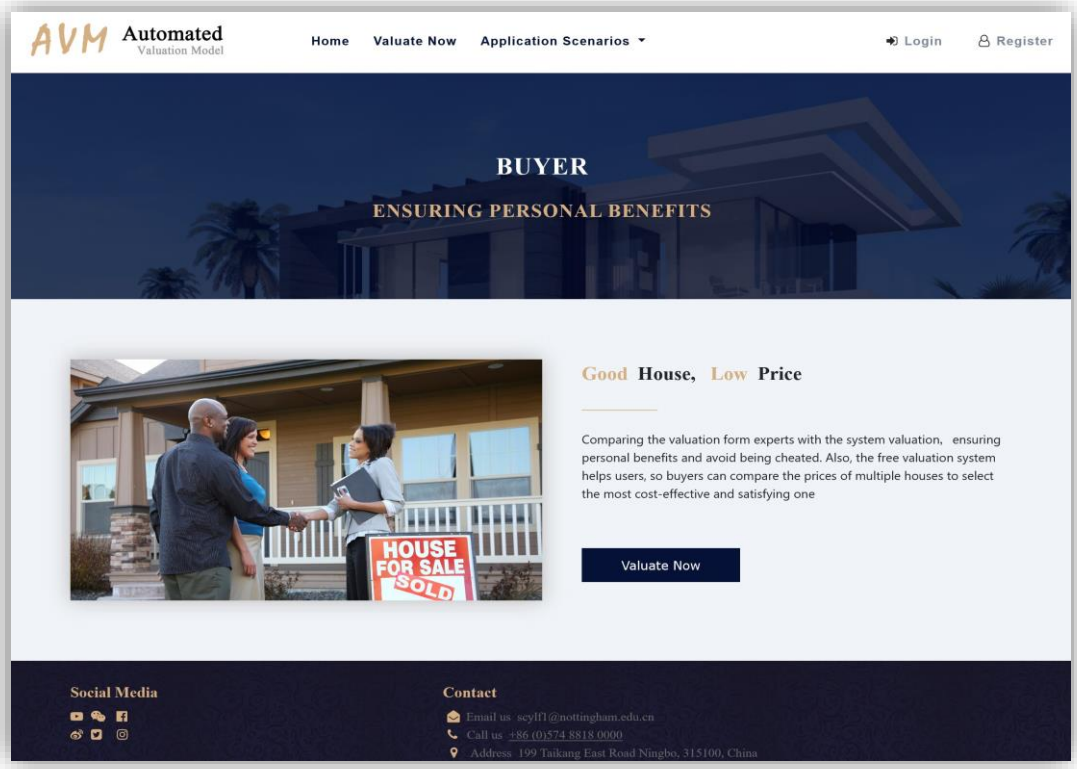


Figure 4.6 Product Applications Scenarios-Buyer

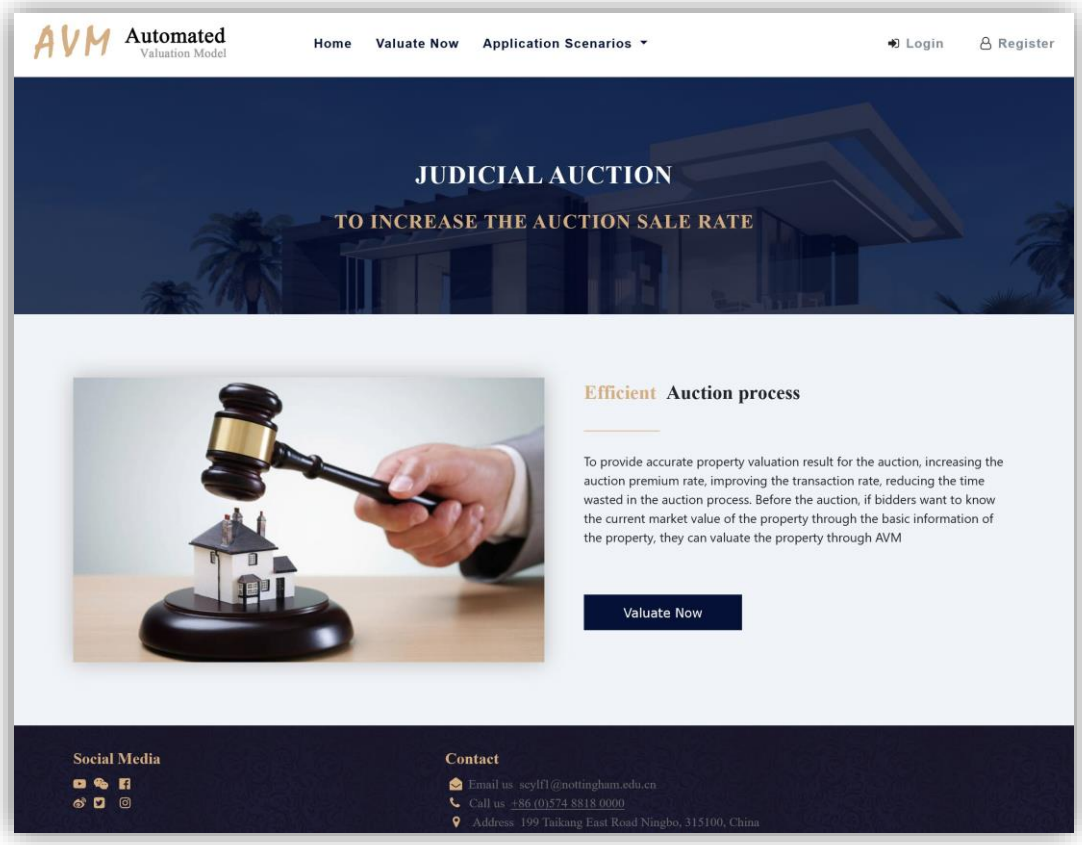


Figure 4.7 Product Applications Scenarios-Judicial auction

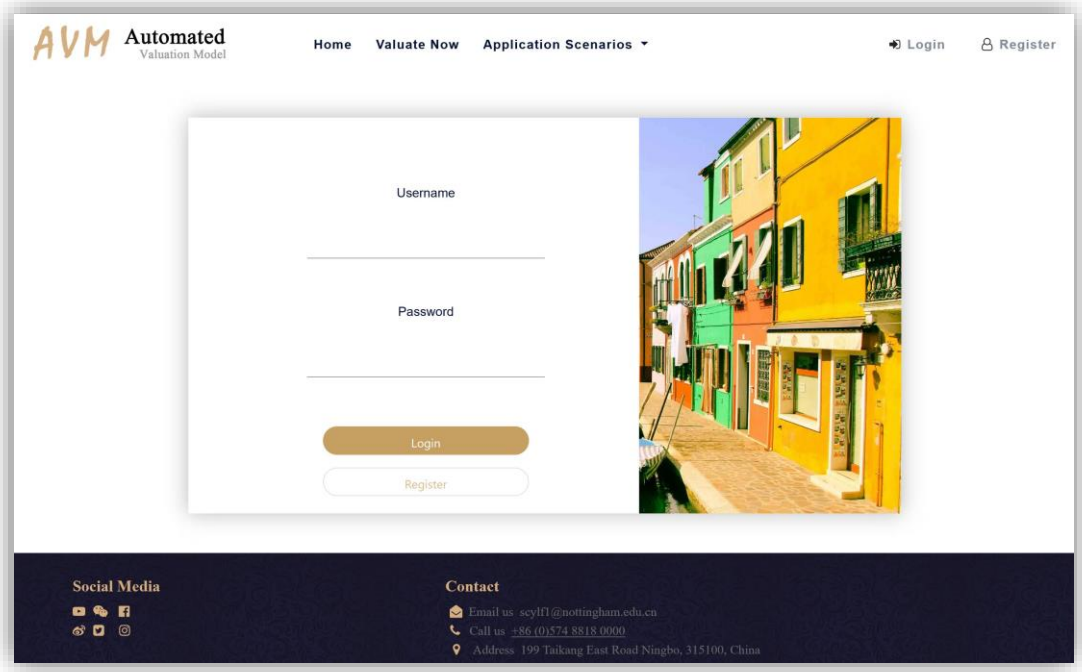


Figure 4.8 Login

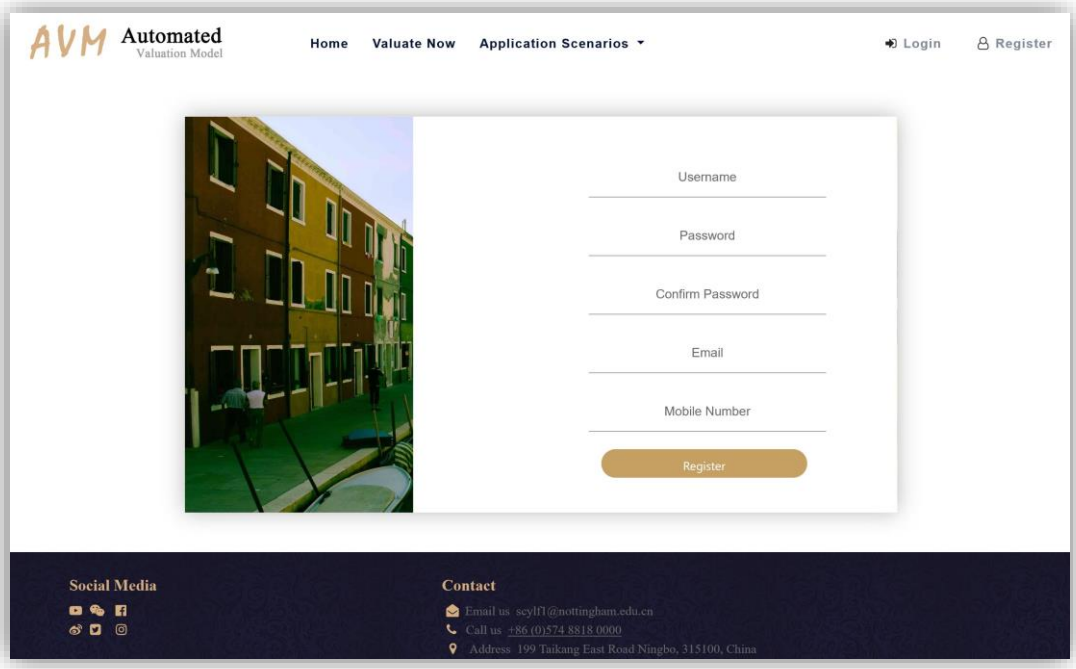


Figure 4.9 Register

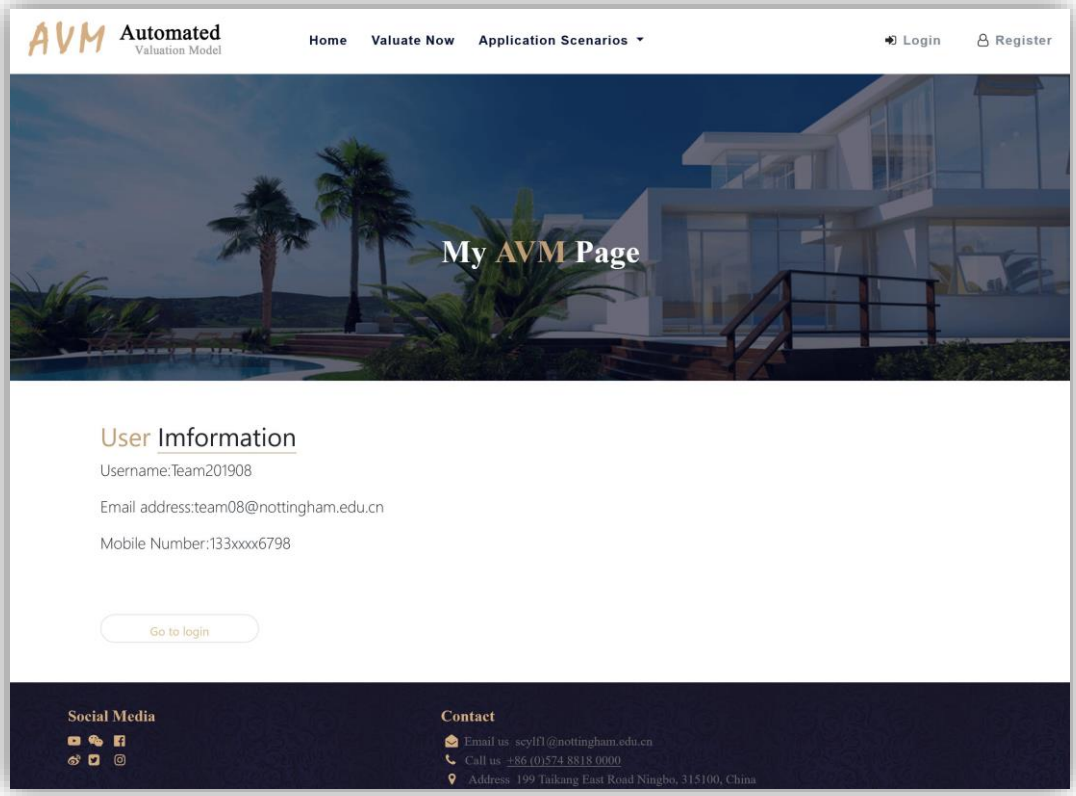
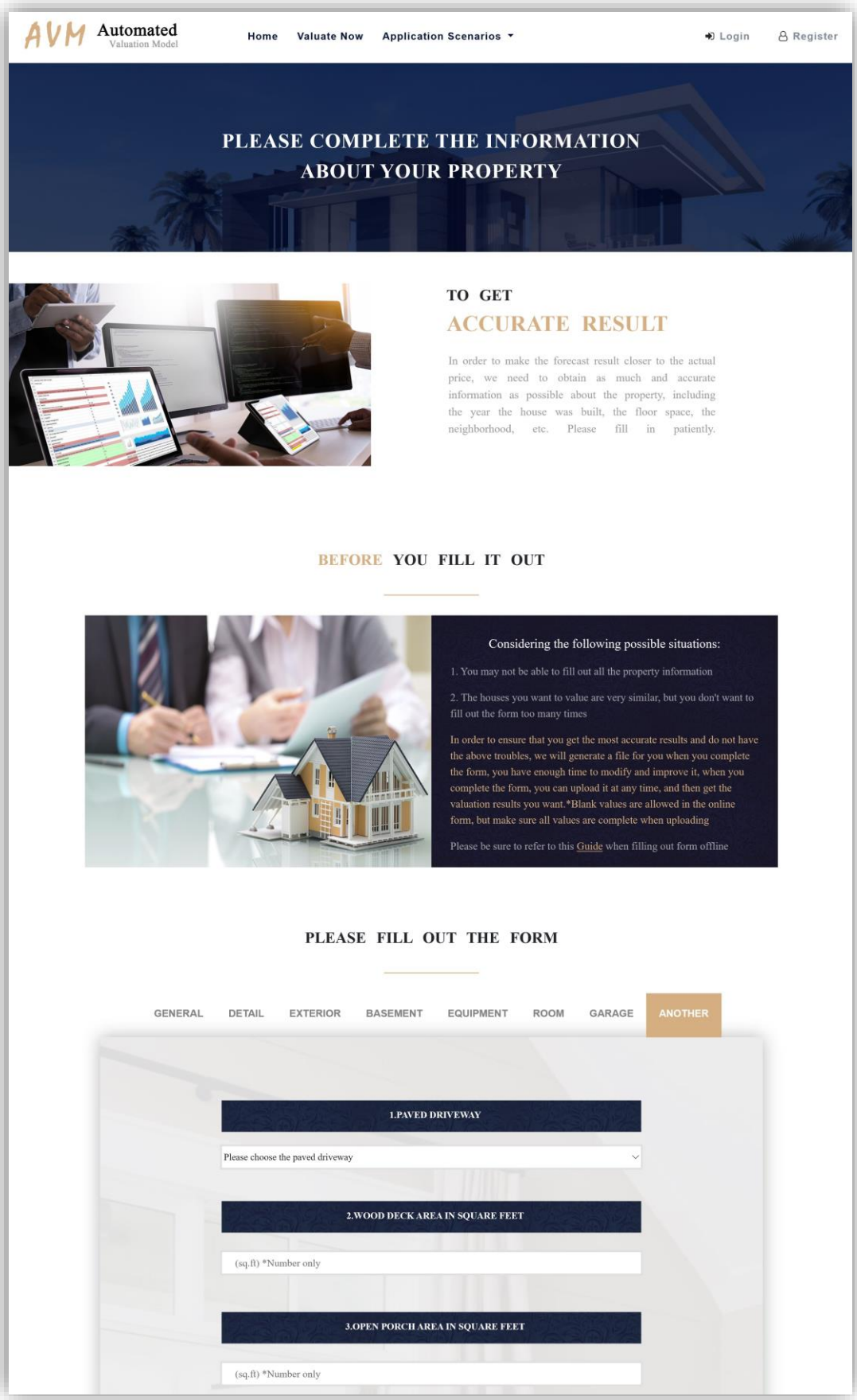


Figure 4.10 User centre


AVM Automated Valuation Model
Home
Valuate Now
Application Scenarios
Login
Register
PLEASE COMPLETE THE INFORMATION ABOUT YOUR PROPERTY
TO GET
ACCURATE RESULT
In order to make the forecast result closer to the actual price, we need to obtain as much and accurate information as possible about the property, including the year the house was built, the floor space, the neighborhood, etc. Please fill in patiently.
BEFORE YOU FILL IT OUT
Considering the following possible situations:
1. You may not be able to fill out all the property information
2. The houses you want to value are very similar, but you don't want to fill out the form too many times
In order to ensure that you get the most accurate results and do not have the above troubles, we will generate a file for you when you complete the form, you have enough time to modify and improve it, when you complete the form, you can upload it at any time, and then get the valuation results you want.*Blank values are allowed in the online form, but make sure all values are complete when uploading
Please be sure to refer to this Guide when filling out form offline
PLEASE FILL OUT THE FORM
GENERAL
DETAIL
EXTERIOR
BASEMENT
EQUIPMENT
ROOM
GARAGE
ANOTHER
1.PAVED DRIVEWAY
Please choose the paved driveway
2.WOOD DECK AREA IN SQUARE FEET
(sq.ft) *Number only
3.OPEN PORCH AREA IN SQUARE FEET
(sq.ft) *Number only

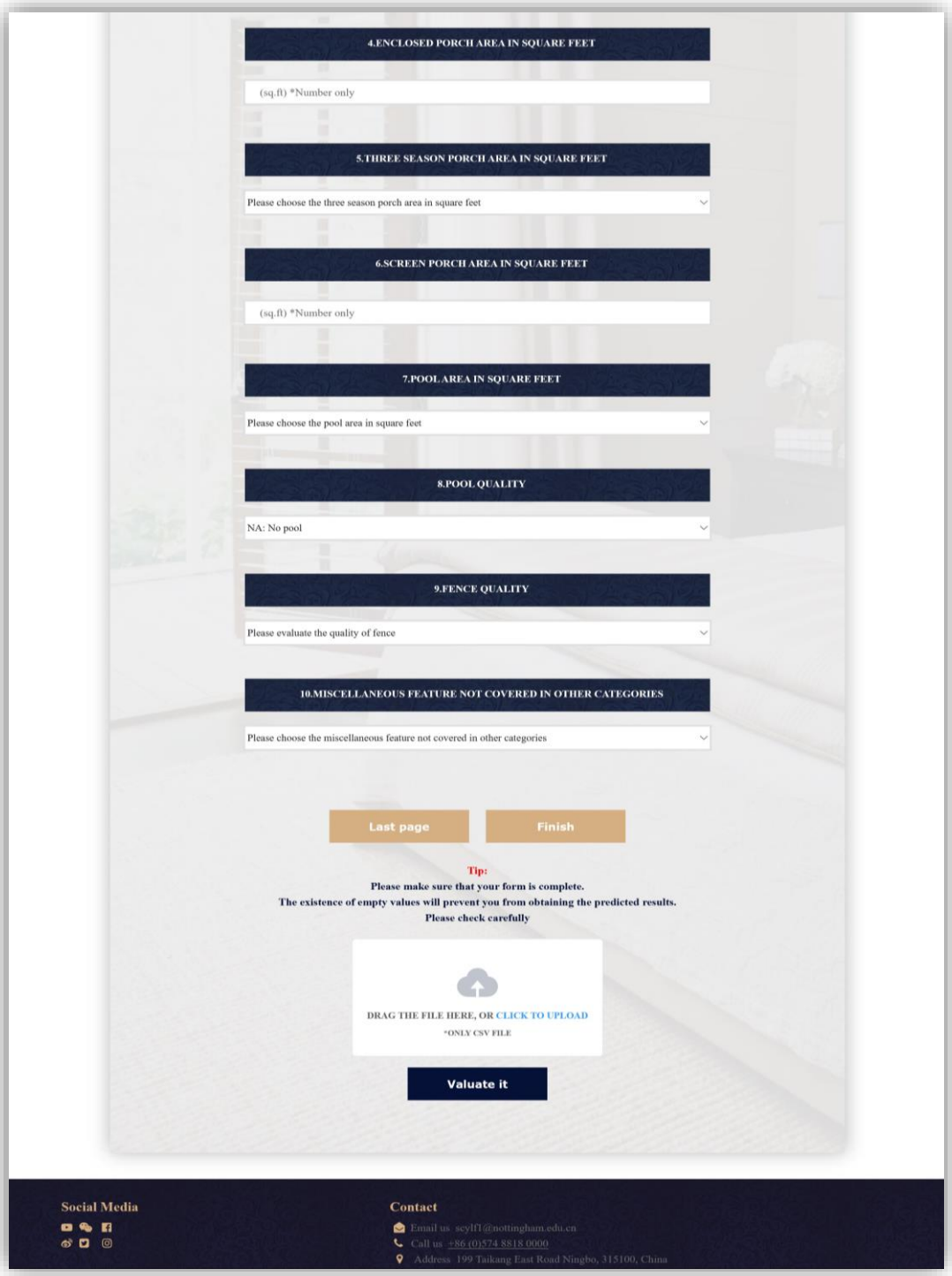


Figure 4.11 Form filling

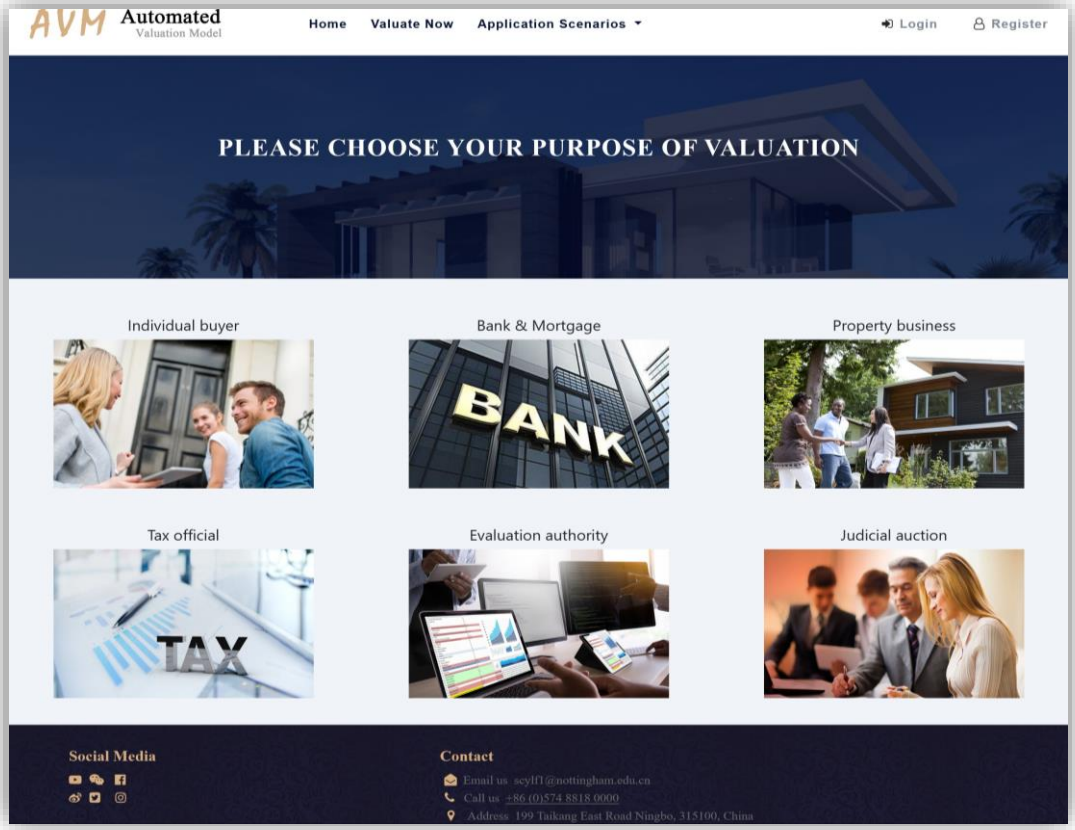


Figure 4.12 Valuate Now

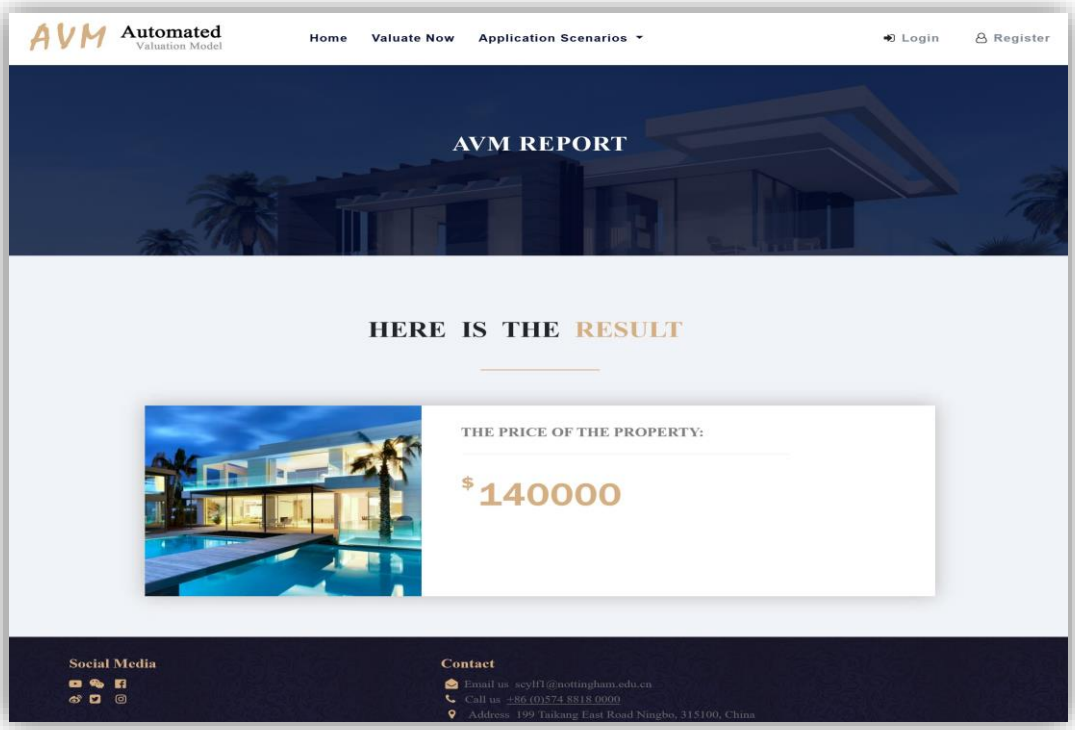


Figure 4.13 Result page

5. Implementation of the system

5.1 Introduction

The implementation of the system can be divided into three parts. The first part is the construction of the front-end, which is the website that interacts directly with the user. The website receives property information from the user and shows the user the result of the valuation. The second part is the server, which passes the data to the models. The last and most important part is the model that does valuation depending on the data. Following is our report on these three parts.

5.2 Front-end

Framework

In this project, we used the Vue.js framework, which is a progressive front-end framework for building user interfaces. It provides Model-View-ViewModel (MVVM) data binding and a composable component system with a simple, flexible API. MVVM (see Figure 5.0) consists of three layers. The View layer represents the view of the system, and it is responsible for transforming the data model into the UI presentation. The Model layer represents the data in which you can define the business logic for data modifications and operations. The ViewModel layer connects the Model to the View. In the architecture of MVVM, the View layer and the Model layer are not directly related but interact through the ViewModel layer, which connects the View layer to the Model layer through bi-directional data binding, making the synchronization between the View layer and the Model layer completely automatic. Therefore, when the system maintainer needs to modify the data, they only need to focus on the business logic, no need to operate the Document-Object-Model (DOM) manually, and complex data state maintenance is left to the unified management of MVVM. Using this architecture enables better maintenance of the system and makes data modification easier and faster. Also, since the Vue's core libraries only focus on the view level, it's easy to integrate with other libraries or existing projects.

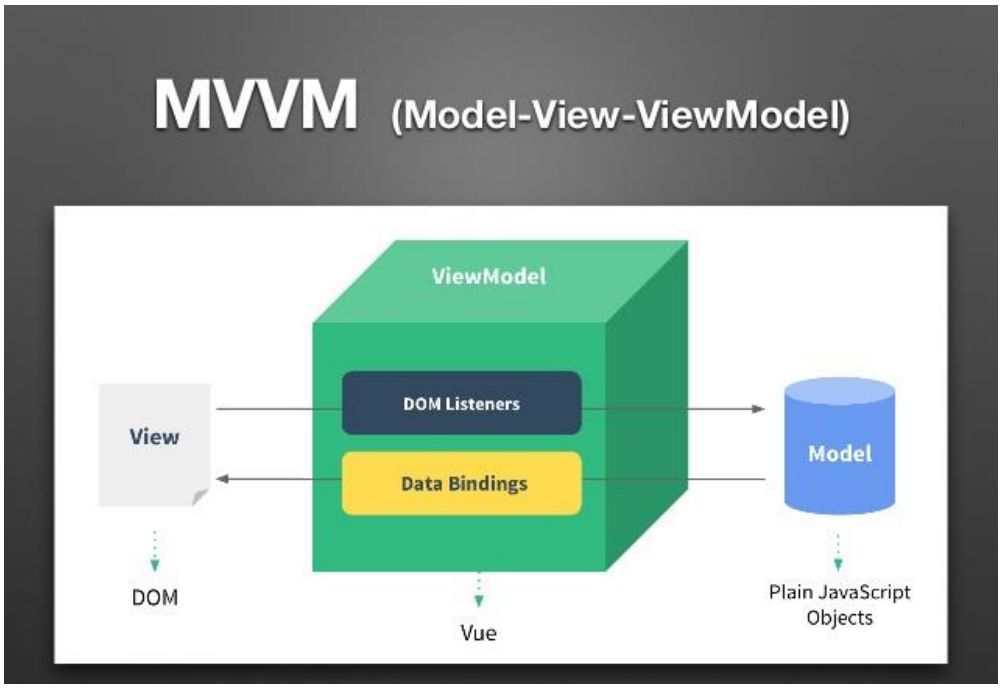


Figure 5.0 Model-View-ViewModel in Vue.js

Available at: https://012.vuejs.org/guide/

GUI design

The website provides the user with a comfortable environment to interact with the AVM model, and users can enter property information on the web page to get the estimated value of the property. In line with the goal to improve the user experience and increasing the reliability of the product, for the design of the overall style of the page, we used generous and straightforward colour and frame, making the pages look stable and professional. When interacting with the page, users should be able to focus on the task with minimal distraction from the interface [8], and this is why we did not use too garish colours for the interface.

For this project, the core function is valuation, which is also the essential user requirement. Therefore, it is very crucial to provide users with a comfortable interface to enter information and obtain results. According to Shneiderman [8], too much text entry tasks take up time and are likely to cause impatient; less text entry tasks mean higher operational efficiency. As a result, only a few fields in the form require the user to type directly, and most of the fields that require property information are implemented by drop-down boxes.

Functions of interface

The homepage (see Figure 4.1) presents a brief introduction of the automated valuation model and machine learning, which can help users quickly understand the marketing and technology background of our model. Users can also check the different scenarios for which this model is applicable and can choose different character scenarios according to their requirements (see Figure 4.2-4.7).

The user can create an account on the registration page (see Figure 4.9). Username and password are only available with numbers, letters and underscores. Username should be unique. Email address and mobile number should also be in the correct forms.(e.g. Email address: xx@xxx.xx; mobile number: 1xxxxxxxxxx (in all numbers)).

The login page (see Figure 4.8) provides the user with an interface to log into the website. The username and the password should be registered in advance. If the user inputs a wrong user name or password, the website will give a warning and allow users to try another time.

After logging in, the ‘Login’ and ‘Register’ entries will be hidden from the navigation bar. Instead, the username of the user will be shown. Users can access the user centre page (see Figure 4.10) to check his/her account information. User can also log out from here.

Considering the diversity of users, we designed different guidance pages (see Figure 4.2-4.7) for users in a variety of industries, which describe how the AVM model help them. The main purpose of this design is to let users know more about the AVM model and use it more clearly with their purpose.

Besides, users can simply click on 'Valuate Now' to do the valuation (see Figure 4.12). They need to choose their purpose of valuation, and then they can jump to the form filling page.

When users come to the form filling page (see Figure 4.11), the page will not immediately ask them to fill out the form. Instead, it introduces the method of filling out the form and explains how the system can help them better complete the valuation process, which also plays a good guiding role for users.

For the form filling part of this page, our team discussed the design of it for a long time. Due to the lack of data and technical support, users cannot directly locate a property for valuation. To get an accurate result, users can only enter the property information in this way. How to make the user patiently fill out

more than 70 data is a problem. Initially, we put all the information on one page. The problem is obvious. The page is too long, and the information is scattered, the user can't concentrate on the task, resulting in impatience. Therefore, we changed the design scheme, we classified these property characteristics into eight categories and displayed them on different pages, so that users could know which piece of content they are editing. Breaking a broad goal into smaller ones, this effectively improves the user experience.

Finally, Figure 4.13 shows the page that displays the results of the valuation, showing the price of the house to the user.

5.3 Back-end & Server

The main function of the website is to provide an interface for the users to input the information of their properties. The back-end server is the tool that links the models and the front-end website. Customary websites use PHP files to link the front-end and the server. However, due to the development of the website building tools, the website can connect to the Mysql database without the support of the PHP files, which increases the efficiency of the website. Due to the usability, node.js is applied as the after-end server in this project.

```
InterfaceName( req, res )
{
        Load database operations;
        Load req data;
        Print req data;
        Excute database operations;
        If(Error occure)
        {
                Print Error message;
        }
        If(Excute successed)
        {
                Print Result;
        }
}
```

Figure 5.1 APIs

The principle of node.js is quite simple. The first step is to define interfaces. These interfaces are responsible for database interaction and server interaction (Provide download or upload serves). Since the server is capable of connecting to the Mysql database, these APIs can execute database operations such as insert, search and delete. Then, the front-end designers use these APIs as the submitted URL. When files are submitted to the after-end server, the code defined in the interface will execute. Plenty of information is added to the submitted file in order to transit between the website and the interface. Therefore, it is necessary to extract the required data before executing database operations.

The link between Python models and the website is implemented by using the observers (pyinotify). The observers keep watch on the upload file folder. Once the new file is uploaded, the observers notify the model to evaluate the file. After evaluation, the predicted value is uploaded to the Mysql database by the python program. The website can obtain these predict value through APIs.

```
MariaDB [Team201908]> desc LoginInformation
    -> ;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| user_id       | varchar(20) | NO   | PRI | NULL    |       |
| user_password | varchar(20) | NO   |     | NULL    |       |
| user_email    | char(20)    | YES  |     | NULL    |       |
| user_number   | char(20)    | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
```

Figure 5.1.1 Database Table Information(1)

```
MariaDB [Team201908]> desc PredictedValue;
+----------------+----------+------+-----+---------+-------+
| Field          | Type     | Null | Key | Default | Extra |
+----------------+----------+------+-----+---------+-------+
| Value          | int(11)  | NO   |     | NULL    |       |
| user_id        | char(20) | NO   |     | NULL    |       |
| residual_value | int(11)  | NO   |     | NULL    |       |
+----------------+----------+------+-----+---------+-------+
```

Figure 5.1.2 Database Table Information (2)

5.4 Model

Data process

The original data consists of 2930 rows with 83 columns, including both numerical and categorical data. Data modelling is one of the most crucial processes in machine learning. There are three approaches that we have been accomplished in modelling the data.

1) Step 1. Categorical values to Numerical values.

```
pd.get_dummies(df[['price', 'color', 'size']])
```

	price	size	color_blue	color_green	color_red
0	10.1	1	0	1	0
1	13.5	2	0	0	1
2	15.3	3	1	0	0

Figure 5.2.1 Example of 'get_dummies()'

The first step is to change categorical values to numerical values. There are several ways to change categorical values to numerical values, and 'one-hot encoding' is one of the common ways. The use of 'get_dummies()' function in pandas allows us to transform categorical variables into dummy/indicator variables.

2) Step 2. Dropping one of the columns after used a one-hot encoding

Leaving out one column from one-hot encoding leads to output higher accuracy. In other words, after One Hot encoding processed, retaining extra column does not add any new information for the modelling process, it would be a proper practice to drop one column.

3) Step 3. Creating one new level with few numbers of variables.

	Roof Style
Gable	2171
Hip	537
Gambrel	21
Flat	19
Mansard	11
Shed	5

	Roof Style
Gable	2171
Hip	537
Other	56

Figure 5.2.2 Example of changing a few variables to one new variable, 'Other'

In last semester, the dataset had shortened by deleting levels that have blank or has few numbers of variables. In fact, by deleting levels, it shortened dataset more than half. By solving the problem, the dataset has remodelled by creating a new one level with levels with blank and few variables. In this way, the dataset kept its original size and got better accuracy.

Machine learning

1) Sampling

One important key point in machine learning is to choose the right ratio between the training set and test set. Below is the learning curve of testing with different sizes of the test set.

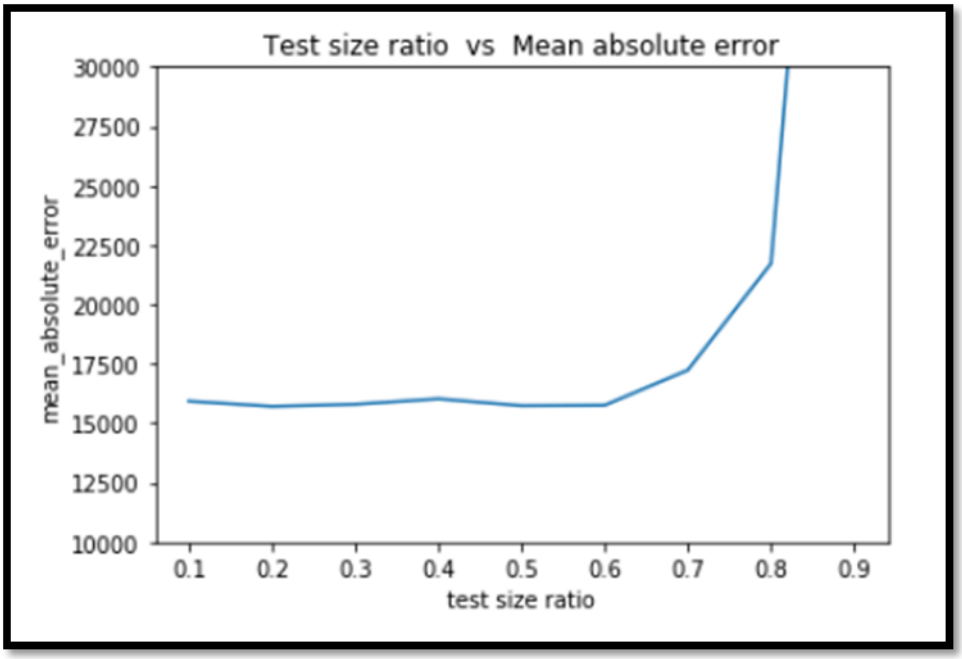


Figure 5.2.3 Learning Curve

In Figure 5.2.3, 0.1 test size ratio means 0.1 test set / 0.9 training set. The experiment was processed with a linear regression model. The learning curve made to find the right test and training set size for machine learning. The learning curve shows that with the test set size from 0.1 to 0.6, machine learning gives out stable results.

2) <u>Algorithm decisions</u>

i. Linear Regression

Linear Regression was the first choice as a starting learning algorithm. Linear Regression seemed a great algorithm to start because it is relatively straight forward compared to other algorithms.

$$\hat{y} = \hat{f}(x) + \varepsilon$$
$$\hat{f}(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

Figure 5.2.4 Linear Regression with multiple variables.

Where e is the error term, is the estimation of the function of x, y the predicted result, β is the unknown parameters that we want to find, x the independent variable.

ii. Neural Network

Artificial Neural Networks are based on the human brain's structure and function. However, the actual implementation in machine learning is way far from reality, it takes in multiple inputs and produces a single output.

The Neural Network is made up of neurons connected to each other, and each connection of the Neural Network is associated with a weight that dictates the importance of relationship in the neuron when multiplied by the input value.

iii. Random Forest

Random Forest is an interpretable model because it makes classifications much like human does: asking a sequence of queries about the available data we have until it arrives at a decision. It builds multiple decision trees and merges them together to get a more accurate prediction.

3) <u>Regression Model Accuracy</u>

- Mean Absolute Error (MAE)

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

Figure 5.2.5

The MAE is a linear score which means that all the individual differences are weighted equally. MAE penalizes huge errors that not as that worse as MSE does.

- Mean Squared Error (MSE)

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

Figure 5.2.6

MSE measures the average squared error of our predictions. For each point, it calculates the square difference between the predictions and the target and then averages those values. The higher this value, the worse the model is.

- Root Mean Squared Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} = \sqrt{\text{MSE}}$$

Figure 5.2.7

The optimized version of MSE. MSE is often used more because it is easier to work with.

Residual Error Model

In most cases, there is a gap between predict price and the real price, which is called residual error. The residual error suggests the reliability of the predicted value. Also, the training dataset is the property characteristics, which suggests the potential influences that the characteristics can cause the predicted price.

$$Residual\ Error = Abs(p_{real} - p_{predict})$$

Therefore, providing a model that predicts the residual error can help the user get a better understanding of the range of the potential values.

The training features of the model are the property characteristics. The training labels are residual errors from the Random-Forest model's predicted results (Since the Random-Forest model has the best performance among all models we have).

5.5 Results of implementation

According to Data Science for Business by Foster Provost and Tom Fawcett [9], *"Going through the process once without having solved the problem is, generally speaking, not a failure. Often the entire process is an exploration of the data, and after the first iteration, the data science team knows much more. The next iteration can be much more well-informed."*

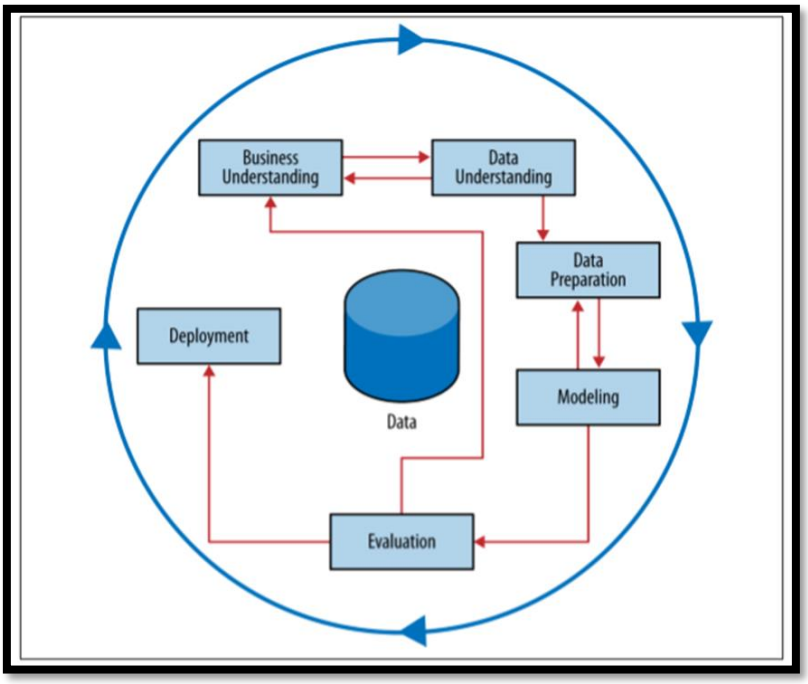


Figure 5.3.1 The CRISP data mining process

The data reprocessing and applying/testing models were followed by the CRISP data mining process. The whole practices were going back and forth repeatedly cause of errors, misunderstandings, wrong steps, and more to get the pleased data and model.

In this project, three types of the algorithm have been used to build the model. The first model is the linear regression model, the second model is the Neural Network model, and the last model uses Random Forest algorithm. Following is the discussion of these three models.

i. <u>Linear Regression</u>

The linear regression model relies on Python and a Python open-source library called scikit-learn. This model act as a testing model for AVM. Also, linear regression model on scikit-learn doesn't provide enough hyperparameter compare to other models. Thus, AVM used default

parameter. However, since the model was easy to handle, it was the best model to gain experience for the first-time machine learning programmers.

This model was the first tested model on AVM. People tend to say its accurate model when mean absolute error less than 10% of the mean value of the y values, which is the sale price for this case [1]. The result came out with MAE (mean absolute error) 17,257.77, which is less than 10% of the mean of the sale price, 18,100. The parameter was set up with scikit-learn default setting. The difference among different parameter set on Linear Regression wasn't significant, except for having different test ratio size. Yet, applying different test ratio size isn't just a matter of the linear regression model.

```
print('Data Shape:', df3.shape)
print('Mean of SalePrice', np.mean(df3['SalePrice']))
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
```

```
Data Shape: (2785, 245)
Mean of SalePrice 181007.59317773787
Mean Absolute Error: 17257.77613718342
Mean Squared Error: 622081446.991339
Root Mean Squared Error: 24941.56063664299
```

Figure 5.3.2 Linear Regression model result

ii. <u>Neural Network</u>

The Neural Network model relies on Python and a Python open-source Library called scikit-learn.

```
df= pd.read_csv('C:/Users/admin/Desktop/17final.csv',keep_default_na = False)
df = pd.get_dummies(df,drop_first=True)
df2 = df[df.columns.difference(['SalePrice','PID','Order'])]
df3 = df['SalePrice']
x_data = df2.values
y_data = df3.values
```

Figure 5.3.3 Data Modify

The above code changes the format of the data, which makes the data recognizable by the Neural Network model.

```
mlp = MLPClassifier(hidden_layer_sizes=(15,15,15,10,10,10,10,10,10,10,10), max_iter=450,
learning_rate='adaptive', learning_rate_init = 0.00126,batch_size = 64)
```

Figure 5.3.4 Model Building

The testing of the Neural Network model was done by splitting the input data into the test group and training group. 80% of the data will be used for training the data, and the rest 20% will be used for evaluating the trained model. The evaluating standard is based on mean absolute error and mean squared error.

The result of the first version Neural Network model (mean absolute error) is beyond 50,000. After modifying the parameters, the performance of the model increased.

The process of training is to adjust the parameters shown in figure 5.3.4. The main efforts were paid to decide the hidden layer sizes and the number of cells. Since the number of the cells is related to the features of the input data, the size of the first hidden layer is set to be 80. The other layers were decided by trial and error. The initial size of the new layer is set to be the size of the last layer. Then, the size is gradually decreased until the best value of the size is found. The learning rate is set to be adaptive, which means the learning rate will gradually decrease during the training process.

```
In [174]: predictions = mlp.predict(x_test)
          print('Mean Absolute Error:')
          print(mean_absolute_error(y_test, predictions))
          print('Mean Squared Error:')
          print(mean_squared_error(y_test, predictions))

          Mean Absolute Error:
          26275.575322812052
          Mean Squared Error:
          1677748469.609756
```

Figure 5.3.5 The performance of the updated Neural Network model

```
In [351]: predictions = mlp.predict(x_test)
          print('Mean Absolute Error:')
          print(mean_absolute_error(y_test, predictions))
          print('Mean Squared Error:')
          print(mean_squared_error(y_test, predictions))

          Mean Absolute Error:
          23327.39167862267
          Mean Squared Error:
          1135043709.3974175
```

Figure 5.3.6 The best result of the updated Neural Network model

iii. <u>Random Forest</u>

The Random Forest model relies on Python and a Python open-source Library called scikit-learn. The first test was driven with the default setting from scikit-learn, which as follows on Figure 5.3.7.

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, criterion='gini', max_depth=None, min_samples_split=2,
min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False
class_weight=None, ccp_alpha=0.0, max_samples=None)
```

Figure 5.3.7 default parameter set

After obtaining unsatisfied result than the linear regression model, Random Forest hyperparameter tuning proceeded (figure 5.3.8).

```
n_estimators=[500,800,1500,2500,5000]
max_features = [ 'sqrt','log2', 'auto']
max_depth = [10,20,30,40,50]
max_depth.append(None)
min_samples_split = [2,5,10,15,20]
min_samples_leaf = [1,2,5,10,15]

grid_param = {'n_estimators' : n_estimators,
              'max_features' : max_features,
              'max_depth' : max_depth,
              'min_samples_split' : min_samples_split,
              'min_samples_leaf' : min_samples_leaf}

from sklearn.model_selection import RandomizedSearchCV
from sklearn.ensemble import RandomForestRegressor
RFR = RandomForestRegressor(random_state = 1)
RFR_random = RandomizedSearchCV(estimator = RFR,
                                param_distributions = grid_param, n_iter = 500,
                                cv = 5, verbose = 2, random_state=42,
                                n_jobs = -1)
```

```
RFR_random.fit(train_features, train_labels)
print(RFR_random.best_params_)

Fitting 5 folds for each of 500 candidates, totalling 2500 fits

[Parallel(n_jobs=-1)]: Using backend LokyBackend with 6 concurrent workers.
[Parallel(n_jobs=-1)]: Done  29 tasks      | elapsed:   28.9s
[Parallel(n_jobs=-1)]: Done 150 tasks      | elapsed:  3.3min
[Parallel(n_jobs=-1)]: Done 353 tasks      | elapsed:  8.3min
[Parallel(n_jobs=-1)]: Done 636 tasks      | elapsed: 16.7min
[Parallel(n_jobs=-1)]: Done 1001 tasks     | elapsed: 24.6min
[Parallel(n_jobs=-1)]: Done 1446 tasks     | elapsed: 34.0min
[Parallel(n_jobs=-1)]: Done 1973 tasks     | elapsed: 46.5min
[Parallel(n_jobs=-1)]: Done 2500 out of 2500 | elapsed: 57.9min finished
```

Figure 5.3.8 hyperparameter tuning (parameter testing)

Due to successful hyperparameter tuning, the MAE went down by 1.000 and time taking went down to less than 30 seconds from 2 minutes. The Random Forest model got the best accuracy among all models with the following parameter set (Figure 5.3.9).

```
rf = RandomForestRegressor(n_estimators = 1000, min_samples_split = 2, min_samples_leaf = 1, max_features = 'sqrt', max_depth= None, random_state = 0)
rf.fit(train_features, train_labels);
```

Figure 5.3.9 final parameter set for the Random Forest model

There were a consideration choosing max-features between ‘sqrt’ and ‘auto’ on parameter. Setting max-features ‘auto’ brought better result, but longer time taking. Because the result difference between those two wasn’t significant compared to long time taking, our model set with ‘sqrt’.

Lastly but not least, the Random Forest first tested model brought over MAE 18,000 with a default parameter set. After trying a different set of parameters and running parameter testing, MAE went down to 16,085.02, which is by far the best result (Figure 5.4).

```
# Use the forest's predict method on the test data
predictions = rf.predict(test_features)
# Calculate the absolute errors
errors = abs(predictions - test_labels)
# Print out the mean absolute error (mae)
print('Mean Absolute Error:', round(np.mean(errors), 2))
```

```
Mean Absolute Error: 16085.02
```

Figure 5.4 The best result of the updated Random Forest model.

iv. Choice of the algorithm: Random Forest

Algorithm	Linear regression	Neural Network	Random Forest
MAE	17,257.77	23,327.39	**16,085.02**

Figure 5.5 MAE comparison among three algorithms

The Random Forest has its own significant strength. For example, RF is less prone to overfitting compare to another algorithm.

Among three different algorithm tested in model; the Random Forest model got the lowest MAE. Thus, the **Random Forest** is selected to be used on our AVM.

v. Hyperparameters of Random Forest

- N_estimators

  N_estimators represents the number of trees in the Forest. Higher the number of trees the better result. However, it causes to slow down the training process.

- max_depth

  Max_depth represents the depth of each tree. The deeper the tree, the more splits, meaning that it captures more information.

- min_samples_split

  Min_samples_split represents the minimum number of samples required to split an internal node. When we increase

min_sample_split, each tree becomes more constrained as it has to consider more samples at each node.

- min_samples_leaf

  Min_samples_leaf is the minimum number of samples required to be at a leaf node. It describes the minimum number of samples at the leaves, the base of the tree.

- max_features

  Max_features represents the number of features to consider when looking for the best split.

vi. Overfitting

Overfitting refers to a model that models the training data too well. Although the Random Forest model is less prone to overfitting, it doesn't mean that it can never happen. Our model shows no big difference in accuracy (about 1~3%) between training and test data sets and in MAE less than 10,000(mean of the sale price is 181,007). To remove overfitting, listed ways were carried out: increasing the size of n_estimators, decreasing the size of max_ features, and increasing the size of max_depth.

vii. Feature importance

```
importances = list(rf.feature_importances_)
feature_importances = [(feature, round(importance, 2)) for feature, importance in zip(feature_list, importances)]
feature_importances = sorted(feature_importances, key = lambda x: x[1], reverse = True)
[print('Variable: {:20} Importance: {}'.format(*pair)) for pair in feature_importances]
```

```
Variable: Bldg Type            Importance: 0.33
Variable: Overall Cond         Importance: 0.15
Variable: BsmtFin Type         Importance: 0.14
Variable: Year Remod,          Importance: 0.05
Variable: Bsmt Exposure        Importance: 0.05
```

Figure 5.6 Feature importance

Each feature has a different ratio of importance that affect the result. Figure 6 shows the top five that has high importance. It indicates that building type affects the sale price mostly by 33%.

Values in each feature

- 'Bldg Type_new' is the type of building which consists of '1Fam' single-family, 'Twnhs' Town-house, and 'Duplex' two-family.
- 'Overall Cond' is an overall condition, which consists of numerical values from 1 to 10.

- ‘Bsmt Exposure’ is basement exposure, which consists of ‘Gd’ good, ‘Av’ average, and more that explains how well basement exposure has been done.
- ‘BsmtFin Type 1’ is finished basement type, which consists of ‘GLQ’ good loving quarters, ‘ALQ’ average living quarters, ‘LwQ’ low quality, ‘Unf’ unfinished, and more.
- ‘Year Remod/Add’ is the year of remodelled, which consists of remodelled years. For example, ‘1990’, ‘2000’, ‘2010’, and more.

6. Summary of achievements

Referring to the stated user requirements, the system has realized all the basic functions required by users. Figure 6 shows all functions that have been implemented. The system functions are mainly divided into three parts. The first part contains the introduction to AVM and some descriptions of the usage scenarios. This part mainly aims to let users know the background and functions of the product, and let different users know how AVM helps them. The second part is about users' personal accounts. Users can create their unique personal accounts and log into the system. In addition, they can check their account information in their user centre. The last and most important part, which is the primary user requirement, is to value the property. The system allows users to fill in their property information and download the information form, and then users can upload the form. Once the upload is complete, users can click the button to view the final valuation results. After the results are displayed on the page, the user completes a property valuation. This is all the user requirements that the system satisfies.

Introduction of AVM	Personal account (user center)	Valuation
•Understand AVM usage scenarios in different industry areas •Learn about the basic principles of AVM	•Log into the website •Log out of the website •Create an account •Check the account information	•Fill the website form of the property description •Download the complete file of the property description •Upload the complete file of the property description •Get the valuation results

Figure 6 System achievements

7. Reflective comments on this project

7.1 A brief review

From September 25, 2019, the day our project team was formed, to this day, we have completed a successful project, nearly 200 days, we have created a significant experience together. The first time we felt the power of teamwork was the day we got this project, which we were interested in. We are very grateful to our supervisor for giving us this opportunity to try this beautiful project and patiently helping us during the whole project. Words can hardly express our gratitude to him. In our team, some were interested in machine learning, others in connecting the front and back ends, and still others were fascinated by the subject of the project. Fortunately, this project included almost all the contents we wanted to try. Therefore, when we got this project, we made up our minds to do it well, so as not to let everyone in the team leave a regret, and not to let our supervisor disappointed. Now the whole project is close to the end, although there are still many shortcomings in our final product. Also, some of the first plan requirements are not implemented, but we believe that the most precious thing we have already gained. We have accumulated project experience and learned how to team cooperation. Also, we gained much valuable knowledge from our team members and supervisor. Therefore, we would like to thank this project, our supervisor, the convenor of this course and all team members. The success of the project is inseparable from everyone's efforts.

Below we will describe in detail the problems we encountered in the process of the project and our reflections.

7.2 Technical perspective

1) At the beginning of the project, we had our first technical problem, which is about presenting the final model. This project is web-based. The model which built by Python needs to be implemented on the website finally. Therefore, our first main task to discuss was how to put the model on the website, and let users input their data to the model.

   As the project progressed, we improved our automated valuation model and designed user interfaces with frameworks. Users can go to the website and upload the information form of properties. The front-end sends the parameters to the server, and then the model automatically catches the variables which have been changed. After detection, the model calculates the result as the sale price and return the result to the front-end. We tested the model and followed the instructions from supervisor step by step. Finally, we solved our initial concerns. When we had any ideas, we discussed together and shared our ideas. Also, we timely discussed the problems with our supervisor. Because of this, our project could proceed smoothly.

2) When we started this project, we knew that we need to build a major part of our work by Python. Python is a powerful language in handling data analysis. However, only one of us had the experience of using Python at the beginning. Thus, most of the data analysis job was done by this member at first. We all realized that this was an unreasonable decision. Therefore, we immediately switched to a collaborative way, where we split the dataset into six parts and distributed them evenly among each member. We learned how to modify the data in Python and communicate problems in the group. At last, we gathered all the code to one person to integrate the data. As a result, we all learned new skills and improved ourselves from this part of our work. From then on, we understood that sometimes cooperation could greatly improve work efficiency, and teamwork was essential in a project.

3) At the first time, after comparing the initial data set and modified data set, the result of the linear regression model did not satisfy our expectation. The result of the initial data set was better than that of the modified data set. Our supervisor suggested that the result of the modified data set may be affected by overfitting. We tried to solve this problem by adjusting the size of the training data set.

   In the first semester, we used the wrong data set to train the linear regression model and got an unsatisfactory result. Therefore, we handled the original data set again and used the more professional data to test. The result performed better than the former one. We took many detours when we met with some

problems. It's very normal to get in some troubles, but the significant thing is to find to true reason causes the problems. Find the reason and then solve the problem; those were the things we learnt from the project.

4) What we realized, at one point, was that AVM needed more understanding of methods than the skill of coding. Thus, teammates, including team leader, started to study all different parts of machine learning plus math while coding their own part. After getting a sense of the background of machine learning, progress went more swiftly.

5) Different from expectation at the beginning, the data pre-processing took the longest time among the work with data, modelling, and server. All different errors from making categorical variable to numerical variable, combining fewer variable levels to one new level, and so on. Thanks to of official and unofficial article, "Hands on machine learning" textbook, and also thanks to our supervisor, coding issues could succeed well.

6) According to the requirement, the system should provide a range of value. The first method that we discussed was related to the reliability of the model.

$$Reliability = \frac{\sum[1 - \left[\frac{[abs(RealValue - PredictedValue)]}{RealValue}\right]]}{number\ of\ the\ data}$$

If the real value is 9000 and the predicted value is 8500, the reliability is 1-(500/9000) = 0.94. Then the value can be represented as PredictValue±[PredictResidualError]. But this method highly relies on the performance of the model. Besides, some extremely inaccurate values will have negative influences.

We asked our supervisor for advice, and he suggested that we can model the residual error from the first model using property characteristics as predictors. And the secondary model can then be used to construct a prediction interval. Finally, using 'price prediction ±[resideual prediction]' to provide a range of price. Then, we successfully built the residual model and applied it to the system so that different type of users could get different results.

We understand that our level of knowledge is limited, and when we try some new ideas, there always are many obstacles and difficulties, so it is essential to timely communicate with the supervisor to discuss ideas, in the future projects, we should also keep this in mind.

7.3 Project management perspective

Project process introduction

The project development process is as follows:

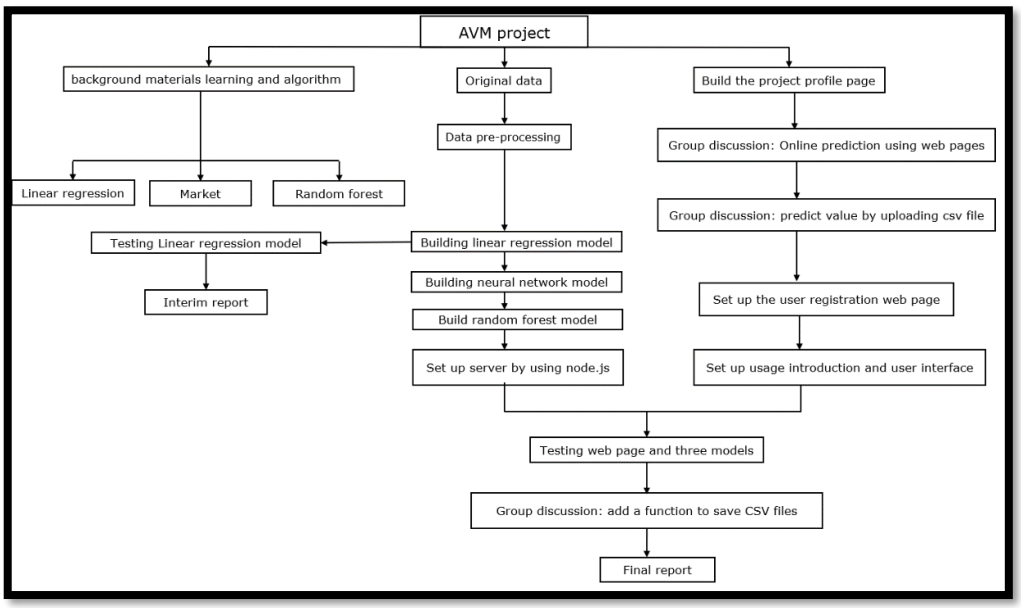


Figure 7.1

i. Preparatory phase

- Relevant knowledge preparation

  For the first few weeks, there were some background materials and basic knowledge related to the project. Market research was beneficial for understanding AVM (Automated Valuation Model), which was used to predict the value of house price. The learning algorithms provided much help for later work.

  To improve the efficiency of learning, we tried to distribute learning tasks. Two of the group members learned linear regression and made notes. After they understood most of the knowledge, we had a meeting and let them explain all the points to others.

- Data preparation

Initial data could not be used directly before pre-processing. Therefore, getting an available set of data was significant preparation before the project starts.

Eighty-three columns of variables in this data set were divided into six-part for each member to filter one part of data for compelling content and turn some null or unreadable variables into some values, which were easily understood for program else.

ii. Build models and web pages

- Linear regression model
- Building website
- Neural Network model
- Random Forest model
- Model testing and compare
- Server setup and connection

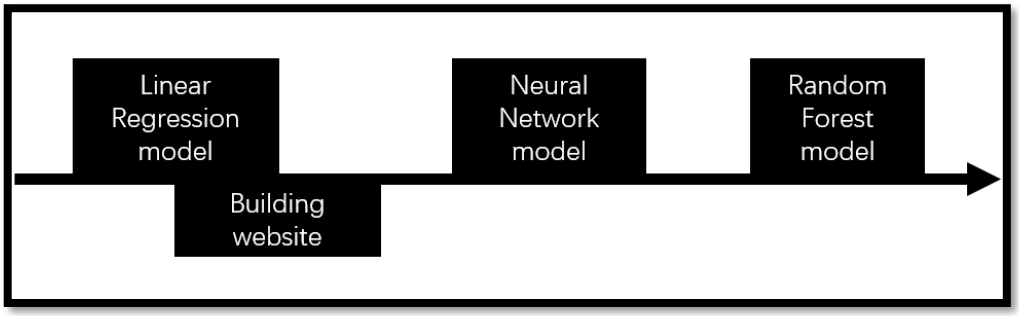


Figure 7.2

Linear Regression model is built first as a testing model because of its usability. Therefore, building a Linear regression model and the website was the first step of AVM. After completing all the models, these three models were used to independent testing of accuracy and comparison of errors between models. As the final step, available models were connected to web pages.

iii. Testing and improvement

The model could work well at this stage. However, there were still some deficiencies in the use. After discussions among team members and user surveys, there were some usages changed.

Such as:

(1). Add the ability to generate CSV files.

(2). Add a guideline for user

The testing of AVM provided much useful experience, which helped to upgrade and improvement of AVM a lot.

Management issues

1) The first problem was about communication. It was the first time for all of our team members to finish a large project in a group. We might be lack of experience of teamwork and communication skills. We tried to solve this by using more time in communication. However, we encountered a very special situation in the second semester. Due to the coronavirus spreading, we could not go back to campus or have offline meetings as much as we did as usual. As we know, the online meeting was not as efficient as an offline meeting, because of the poor connection quality and some arrangement conflicts. However, the online meeting was not as good as offline meeting with the supervisor. To be honest, the progress was slow. We didn't have other choices but worked hard to finish the job before the deadline.

   Afterwards, this problem had been solved during team working. We tried to have more efficient communication within the group, no matter it was online or offline. We explained our points in our WeChat group and sorted out the minutes quickly after our meeting, in case we lost some important contents in the meeting. Besides, we have increased the frequency of online communication through WeChat and Team. Although we could not go back to the campus, there was no big problem in communication. The supervisor asked us to submit a progress report every once in a while. We thought this was a perfect way to report our progress to the supervisor and let him know how our system was being built. We could get valuable advice from the supervisor, which was very important for the progress of our project.

2) The second problem is that the efficiency of our group meeting was not high. The weekly group meeting took nearly three hours. Most of the time was used to summarize the things that had been learned in that week. However, with the technical requirements of the project increasing gradually, the time for the group members to communicate what they had

learned should be decreased. More time should be arranged for planning future work.

We had a tight schedule in the first semester; everything was in a hurry. We fixed this problem by using the more free time to learn and communicate. Also, we had a long winter holiday. Therefore, we had enough time to improve our project and communication.

3) The third manage issue was still about schedule management. Unexpectedly, all coursework was released in the same month, which undoubtedly increased the pressure for each member and made everyone feel challenging to manage the time. Therefore, time spent on the project is forced to be reduced. We repeatedly adjusted our work schedule. Although there are no major changes, but also caused the delay of a part of works. However, every member still completed the tasks carefully every week, which made our project progress smoothly.

   Although we had a long winter holiday, the time for us to complete the project was still limited. We had to hurry up and work hard on the project. There were many deadlines. Because of the delay of the term, we had the less time left. Every one of us did a great job on the project to overcome the difficulties.

4) The fourth issue was about work allocation. Each member was likely to have a different understanding of the same task. For instance, each member was required to process one part of the existing data. However, the standard of data was not declared clearly, which resulted in an unavailable dataset at last.

   In the first semester, we allocated the data work on different members. The result was not expected. Therefore, we handle the data set again. One member dealt with the data alone, unified the standard. Therefore, we understand that some works needed to be allocated, but some should be done by the same person. Also, having good communication with team members was always important in the teamwork.

5) The last problem we found was in code generation. Since we had less experience in using git at the beginning of the project, we did not choose to use it to organize the code, but to exchange the latest version of code among the team members. As a result, we encountered several incidents where the latest version was overwritten with an older version. This reminds us that in the future cooperation work, we must pay attention to the clear labelling of each version and file it separately.

8. Project schedule

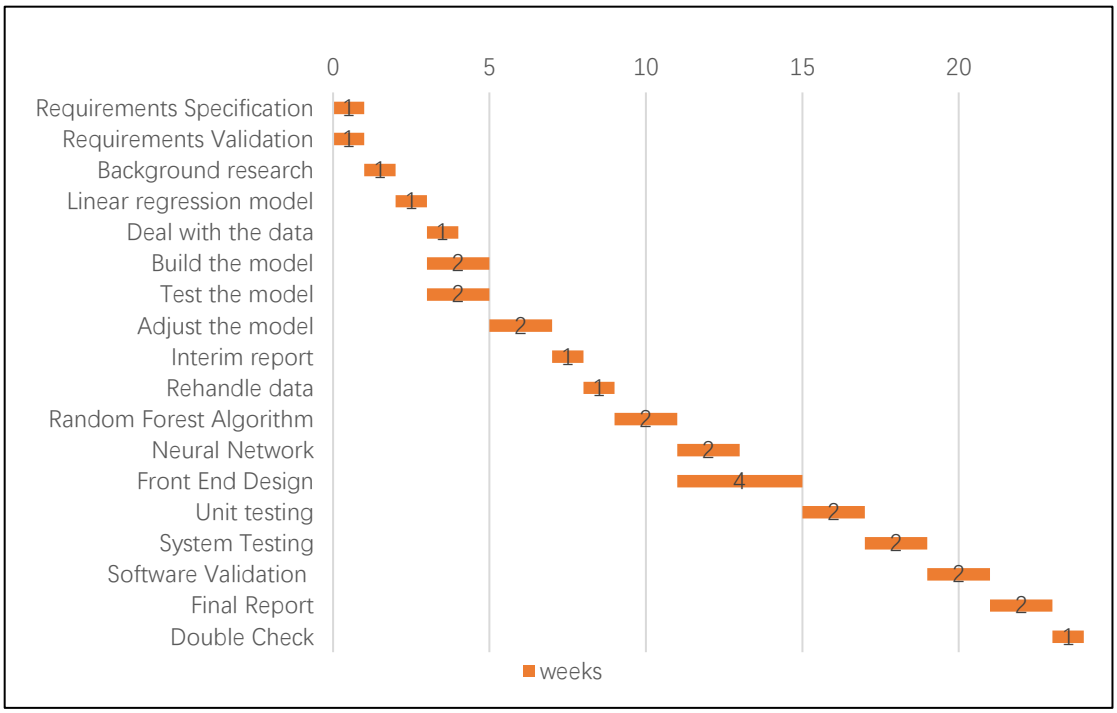


Figure 8.1 Gant chart

This is the timetable of the whole project. There are several conflicts with the original timetable. In general, the overall time is 24weeks, rather than 22 weeks, because of the coronavirus. Therefore, we spent more time on front-end development and system testing. Also, because we didn't deal with the original data in the right way, we have to spend another week to handle data.

APPENDIX

I. Meeting Minutes

1st Meeting Record (Informal)			
Date	19.10.11	time	14:00
Place	PB hall	recorder	Yixuan Huang
Attendee	Liwei Fu、Jaesung Park、Jiazhen Ruan、Wenfeng Li、Yixuan Huang、Jingwen Zhou		
absentee	No		
Late	No		
Meeting content			
Chair : Liwei Fu Secretary : Yixuan Huang Discuss about future tasks: 1. Build the group site; 2. Analyze the marketing background and find some present template Discuss what should we talk about in the first formal meeting: 1. Development model 2. Data format Liwei shared a website which introduces the machine learning in AVMs.			
Tasks			
Task	People in charge	deadline	requirements
Front end of the site	Yixuan Huang, Jingwen Zhou	10.30th	
Site content	All	10.30th	
Do background research	All	Before the first formal meeting	Write down the questions and points
Email the supervisor	Liwei Li	This weekend	Confirm the time of the first formal meeting

2nd Meeting Record (Formal)			
Date	19.10.14	time	15:00
Place	PMB 419	recorder	Yixuan Huang
Attendee	Liwei Fu, Jaesung Park , Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Professor Anthony Bellotti		
absentee	No		
Late	No		

Meeting content

Chair : Professor Anthony Bellotti Secretary : Yixuan Huang

- Brief Introduction of the industry
- Why he chose us
- Things we need to do this week

Note: We can select the variables in the dataset

- Starting points:

1. property market variable
2. multivariate Linear regression (first)
   Random Forest
   Artificial Neural Network (further)
3. Take the dataset 2940
4. organize the tasks

- More about the data:
  Data frame → dummy variables
  Training bias → divide the dataset into training dataset and test dataset
- Organize a team : communicate & allocate tasks
- 2 articles: one about linear regression & one about the market
- Next Meeting !!! 15:00 next Monday, October 21, 2019 @ PMB419

Tasks

Task	People in charge	deadline	requirements
AVM & property markets	All	This week	Find out about aged data Look out for some weird data
Multivariate linear regression	All	This week	
Python	All	This week	useful packages: pandas → load the data & Scikit-learn → machine learning
machine learning	All	This week	

3rd Meeting Record (Formal)

Date	19.10.21	time	15:00
Place	PMB 419	recorder	Yixuan Huang
Attendee	Liwei Fu, Jaesung Park , Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Anthony Bellotti		
absentee	No		
Late	No		

Meeting content

Chair : Liwei Fu
Secretary : Yixuan Huang

- Linear regression
  1. Write on the board to explain what optimization linear regression is
  2. Build a linear regression model

- Data:
  1. Load up the data into python
  2. List top three variables in each category as common variables (using a function in pandas)
  3. Get data description.

Next Meeting !!! 15:00 next Monday, October 28, 2019 @ PMB419

Tasks

Task	People in charge	deadline	requirements
Build site	Jingwen, Yixuan	Oct 28th	
Load up the data into python	Jaesung	Oct 28th	
Learn about linear regression	Wenfeng, Jiazhen, Liwei	This week	Be able to introduce it on the whiteboard next week
Learn about machine learning & python	All		

4th Meeting Record (Informal)			
Date	19.10.28	time	14:00
Place	PMB hall	recorder	Yixuan Huang
Attendee	Liwei Fu, Jaesung Park, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou		
absentee	No		
Late	No		
Meeting content			
Chair : Liwei Fu Secretary : Yixuan Huang Discuss about what we learned about linear regression and data processing Specialize the tasks of next week			
Tasks			
Task	People in charge	deadline	requirements
Improve the website	Jingwen Zhou & Yixuan Huang	10.29th	Add more details about the technology background
Build the model	Jaesung Park	10.29th	
Test the model with some categories and apply dummy variables	all	11.4th	Each member takes one part of data

5th Meeting Record (Formal)			
Date	19.10.28	time	15:00
Place	PMB419	recorder	Yixuan Huang
Attendee	Liwei Fu, Jaesung Park, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Anthony Bellotti		
absentee	No		
Late	No		

Meeting content

- Check the website ------- supervisor's advice : add reference & more about machine learning & Show the clearer example about the input and output of the product
- Check the loaded data ------ supervisor's advice: use Jupiter, try different approaches to use the data
- Check the understanding of LR (SLR & MLR)--------- supervisor's advice: pay attention to the difference between Least-Squares Regression (continues) and Least Absolute Value Regression(discontinues) & partial derivatives
- Some teaching about matrix and derivatives
- Covert all the char variables into numbers ------ dummy variables (get.dummy from panda)
- Remove the variables with small numbers
- No ethical problems

Tasks

Task	People in charge	deadline	requirements
Improve the website	Jingwen Zhou & Yixuan Huang	10.29th	Add more details of the technology background (e.g. Machine learning)
Build the model	Jaesung Park	10.29th	
Test the model with some categories and apply dummy variables	all	11.4th	

6th Meeting Record (Informal)			
Date	19.11.07	time	16:00
Place	PB hall	recorder	Yixuan Huang
Attendee	Liwei Fu, Jaesung Park, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou		
absentee	No		
Late	No		

Meeting content

Chair : Liwei Fu Secretary : Yixuan Huang
Divide the tasks of the interim report.(As is shown in Tasks table below)
Update some new progress:

- Finish dealing with the data
- Finish building the basic model

Tasks

Task	People in charge	deadline	requirements
Discussion of any problems	Liwei, Jiazhen, Wenfeng	11.29th	For interim report
Write the SRS doc. Build the prototype	Jingwen, Yixuan	11.29th	For interim report
Results of initial implement efforts	Jaesung	11.29th	For interim report
Enrich the Background information & make a complement time plan	all	11.29th	For interim report

7th Meeting Record (Formal)			
Date	19.11.11	time	15:00
Place	PMB 419	recorder	Yixuan Huang
Attendee	Liwei Fu, Jaesung Park, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Anthony Bellotti		
absentee	No		
Late	No		

Meeting content

Chair : Liwei Fu Secretary : Yixuan Huang

Events:
Show the linear model to Tony

Discussion & advice:
About our new dataset:
1. Not to delete the properties but change the level. Join the small groups of values into one new category.

Missing values:
1. Remove the whole observation(listwise deletion)/ column– depends on the number of missing values
2. Try to guess the value
3. Add “NaN” to a new category

Machine Learning:
Build the model on the training and test on the test set. Get the evaluation
Read about :
Testing
Overfitting
Data Science of Business: chapter 2 (normal to go backwards)
Learning curve find the peak

Consider as different roles to find out which kind ofmeasure will be more appropriate (MAERMSE R^2)

Interface:
Make the impact of each properties visible.
Next meeting @ next Friday 1pm 11.22

Tasks

Task	People in charge	deadline	requirements
Find the learning curve	Jaesung	11.18	

8th Meeting Record (Formal)			
Date	19.11.22	time	13:00
Place	PMB 419	recorder	Yixuan Huang
Attendee	Liwei Fu, Jaesung Park, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Anthony Bellotti		
absentee	No		
Late	No		

Meeting content

Chair : Liwei Fu Secretary : Yixuan Huang

Events:
Show the latest progress to Tony

Discussion & advice:

The new dataset causes larger error:
1. Suggested: overfitting.

The prototype:
1. Maybe use the linear regression to show some impact

Some examples for different users:
Buyer – MSE
Bank (no over estimation)– MAE
Read about :
FICO – a US website as a template
Grid Search

Interface:
Make the impact of each properties visible.
Next meeting @ next Monday 3pm on 12.2!!

Tasks

Task	People in charge	deadline	requirements
Adjust the size of the test data set and find the result	all		

9th Meeting Record (Formal)			
Date	19.12.2	time	15:00
Place	PMB 419	recorder	Yixuan Huang
Attendee	Jaesung Park, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Professor Anthony Bellotti		
absentee	Liwei Fu (illness)		
Late	No		

Meeting content

Chair : Anothony, Secretary : Yixuan Huang
Events:

Show the latest progress to Tony

Discussion & advice:
<u>About the data</u>
An unordinary peak in the learning curve.
Implement machine learning
Random Forest : set number of trees & depth
Divide training dataset into “true training” and “validation dataset” – use cross-validation
<u>About the report:</u>
Using “we” is ok
Try to write scientifically.
Things like: “Why to do this” , “Why is this important.”
Think about more about what we want to do to this project.

Tasks

Task	People in charge	deadline	requirements
Finish the report	All	12.6	Add more details, complete the gant chart
Finish linear regression	All	This semester	Start Random Forest
Find some ideas about the project	All	This semester	
Read the book about data science	All		

10th Meeting record			
Date	19.12.16	time	15:00
Place	PMB 419	recorder	Yixuan Huang
Attendee	Jaesung Park, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Professor Anthony Bellotti		
absentee	Liwei Fu (illness)		
Late	No		

Meeting content

Chair :Professor Anthony Bellotti Secretary : Yixuan Huang

Events:

Discuss for the next step

Discussion & advice:

About the Interface

Link the page to python- JavaScript - jQuery

About the data:

Clean up the data and rerun the model to check if it can work.

Neural Network

About the uncertainty of the model

Ways to measure the uncertainty

Build the model to predict the errors e.g. says: properties with larger bedroom are difficult to predict.

Meta modelling approach

Others

Think about more about what we want to do to this project.

Do some more research about this project

Tasks

Task	People in charge	deadline	requirements
Finish the report	All	12.6	Add more details
Finish linear regression			
Start random forest			
Find some ideas about the project			
Read the book	All		

11th Meeting record (Conclusion of Team Dialog)			
Date	20.03.13	time	15:00 (Beijing)
Place	Online	recorder	Yixuan Huang
Attendee	Liwei Fu, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Anthony Bellotti, Jaesung Park		
absentee			
Late	No		

Meeting content

Events:

Progress Report:

Model:

1. Replace the blank/unpopular items
2. Set the criteria for unpopular items as 3% of all items
3. Ask for book recommendation for data remodeling

Frontend:

1. Show the basic pages

Backend & sever:

1. Successfully register and login
2. Test NN model

Discussion & advice:

Model:

1. It may be smart to write generic code to take and replace variables
2. You can remove some columns which unlikely to have impact
3. 3% is reasonable, no special number for this
4. Google Data pre-processing

e.g. http://arxiv.org/pdf/1906.08510.pdf

Teamwork:

Get more involve in coding of ML and modelling decisions, such as testing.

Tasks

Task	People in charge	deadline	requirements

12th Meeting record (Formal online)			
Date	20.03.20	time	15:00 (Beijing)
Place	Online	recorder	Yixuan Huang
Attendee	Liwei Fu, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Anthony Bellotti		
absentee	Jaesung Park		
Late	No		

Meeting content

Events:
Gather all the progress
Discuss about the next step

Discussion & advice:
Model:
Neural network: normalize the data
Website:
Connect files and model to the webpage -> JS & jQuery
Design

Tasks

Task	People in charge	deadline	requirements

13th Meeting record (Informal)			
Date	20.03.21	time	15:00 (Beijing)
Place	Online	recorder	Yixuan Huang
Attendee	Liwei Fu, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Jaesung Park		
absentee			
Late	No		

Meeting content

Events:
Gather all the progress
Discuss about some unsolved problems:

1. Model test
2. Interface design
3. How to upload files through our website

Tasks

Task	People in charge	deadline	requirements
Model test	Jae, Liwei	03.25	Test RF and NN
Interface update	Jingwen, Yixuan	03.29	
File uploading function	Jingwen, Liwei		

14th Meeting record (Conclusion of Team Dialog)			
Date	20.04.01	time	15:00 (Beijing)
Place	Online	recorder	Yixuan Huang
Attendee	Liwei Fu, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Jaesung Park, Anthony Bellotti		
absentee			
Late	No		

Meeting content

Events:

Model:

1. Have tested NN and RF. MAE: Linear: 17,257 NN: 23,000 RF: 16,085
2. Some parameters in RF have no importance to the valuation
3. Use **joblib** to export the trained model successfully.

Front-end:

1. Add user center page
2. Add 6 guide pages for different characters

Discussion & advice:

Model:

1. Report MAE on both training and testing set
2. Keep going with NN

Website:

1. Linking with the Python server should be the first priority

Make a big effort on report

Tasks

Task	People in charge	deadline	requirements
Keep going with current work	All		
Report	All		See the picture below

15th Meeting record (Informal)			
Date	20.04.03	time	15:00 (Beijing)
Place	Online	recorder	Yixuan Huang
Attendee	Liwei Fu, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Jaesung Park		
absentee			
Late	No		

Meeting content

Events:

Discuss about the final report updating

1.Updated design of the system and its user interface: a.Interface part: Zhou & Huang
b.System: Fu & Jason

2.Discussion on the implementation and testing of the system. This must include a list of all major system components, which of these were written by the group and where the others come from, and an overview of the developed source code hierarchy.
: a.System: Fu& Jason & Li & Ruan
b.Interface: Huang & Zhou

3.Summary of what was achieved, referring to the stated requirements.
a.System: Fu& Jason & Li & Ruan
b.Interface: Huang & Zhou

4.Reflective comments on the success of the project, both from a technical and a project management perspective, including group working issues etc
a.System: Fu& Jason
b.Management Issues: Li & Ruan

Tasks

Task	People in charge	deadline	requirements
Report modifying	all		See the chart above

16th Meeting record (Informal)			
Date	20.04.05	time	15:00 (Beijing)
Place	Online	recorder	Yixuan Huang
Attendee	Liwei Fu, Jiazhen Ruan, Wenfeng Li, Yixuan Huang, Jingwen Zhou, Jaesung Park		
absentee			
Late	No		

Meeting content

Chair: Liwei Fu

Events:

Discuss some unsolved problems:

1. CSV File uploading issue (form error & no auto uploading)
2. The implementation of Different models for different user
3. Model testing errors

Solutions:

1. Change the way of user uploading
2. Make 6 different detail pages for different users and connect to responding models

Tasks

Task	People in charge	deadline	requirements

II. Test report

1 Introduction

This is the test report of the Automated Valuation Model project. The purpose of this test report is to summarize the test processes in the test phase and analyze the test results, and then describe whether the system meets the requirements.

2 Outline of test

2.1 Test Case design

This test case design mainly adopts the black box test method, the function module and the integration test adopt the specific test case design method has the equivalence class division method, the boundary value division method and the error guess method.

2.2 Test environment and configuration

Server configuration： https://localhost:8081

Operating System： Linux

Database： MySQL

Browser： Firefox 68.7

2.3 Test method

Black Box Testing

Black box testing, also known as functional testing, tests to see if each function is working correctly. In black box testing, we see the program as a black box which is unable to be opened and test it on the program interface without considering its internal structure and internal features. The black box testing only checks whether the program functions can run correctly according to the requirements specification and whether the program can receive the input data and give correct output message to the user. It is mainly aimed at testing the software interface and functions.

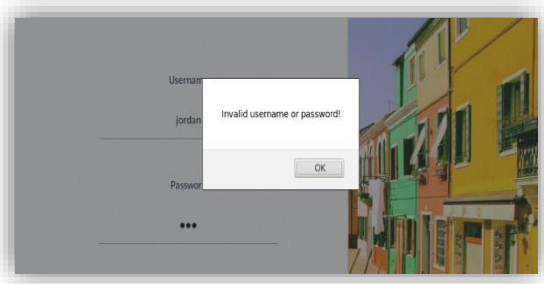
3 Test results and defect analysis

3.1 Login

Test items and features	Test case	Test data	Expected result	Actual result
Invalid operation	Username: NULL Password: String	Username: Password: 123abc	Login failed	Red Alert: Username must be filled.
Invalid operation	Username: jordan Password: NULL	Username: jordan Password:	Login failed	Red Alert: Password must be filled.
Invalid operation	Username: string Password: String	Username: jordan Password: mary123	Login failed	Pop-up Window: Invalid Username or Password
Valid operation	Username: String Password: String	Username: jordan Password: 123jordan	Login succeed	Turn to HomePage

Deviation： None

Test result： Ability to determine if user inputs correct username and password.



Deficiency： None

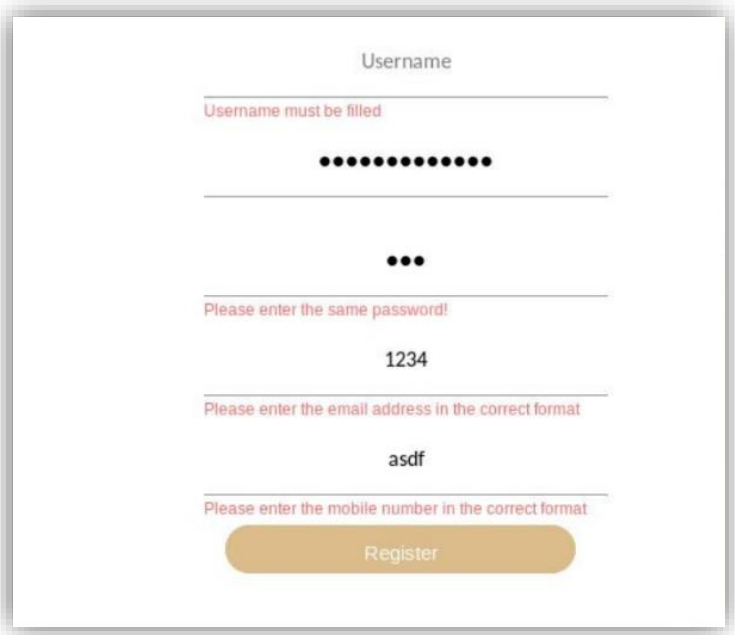
3.2 Register

Test items and features	Test case	Test data	Expected result	Actual result
Invalid operation	Username: NULL Password: String Confirm Password: String Email: String Mobile Number: Integer	Username: Password: 123abc Confirm Password: 123abc Email: 123@136.com Mobile Number: 12345678912	Register failed	Red Alert: Username must be filled.
Invalid operation	Username: String Password: String Confirm Password: Integer Email: String Mobile Number: Integer	Username: jordan Password: jordan Confirm Password: 123456 Email: 123@136.com Mobile Number: 12345678912	Register Failed	Red Alert: Please enter the same password!
Invalid operation	Username: String Password: String Confirm Password: String Email: Integer Mobile Number: Integer	Username: jordan Password: jordan123 Confirm Password: jordan123 Email: 123456 Mobile Number: 12345678912	Register Failed	Red Alert: Please enter the email address in the correct format
Invalid operation	Username: String Password: String	Username: jordan Password: jordan123	Register Failed	Red Alert:

	Confirm Password: String Email: String Mobile Number: String	Confirm Password: jordan123 Email: 123@136.com Mobile Number: 123451asdv		Please enter the mobile number in the correct format
Valid operation	Username: String Password: String Confirm Password: String Email: String Mobile Number: Integer	Username: jordan Password: jordan123 Confirm Password: jordan123 Email: 123@136.com Mobile Number: 12345678901	Register Succeed	Turn to log in page.

Deviation： None

Test result： Ability to determine if user input is in a correct way.



Deficiency： None

3.3 Form filling

Test items and features	Test case	Test data	Expected result	Actual result
Invalid operation	Choice Box: Choice Input Box: Integer	Choice Box: Choice A Input Box: 1000	Generate a csv file	Generate a csv file
Valid operation	Choice Box: Choice Input Box: Integer	Choice Box: Choice A Input Box: 1000	Generate a csv file	Generate a csv file

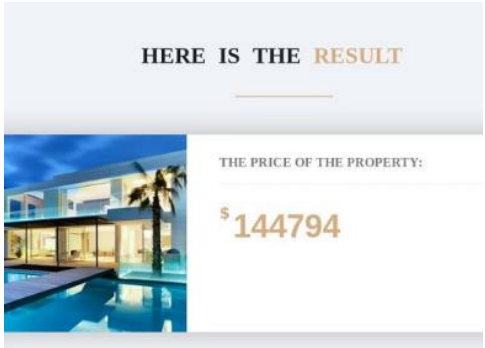
Deviation： None

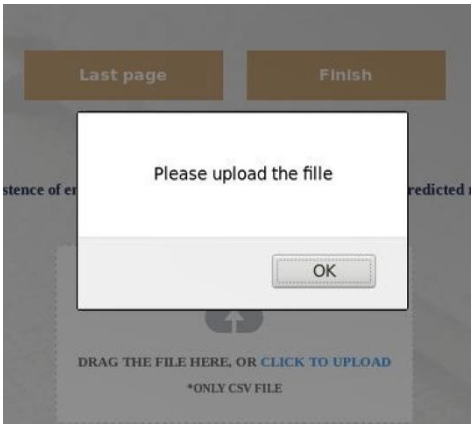
Test result： No matter what user inputs, it will download a csv file for user to fill.

Deficiency： None

3.4 Upload CSV file

User uploads the csv file to the webpage.





Test items and features	Test case	Test data	Expected result	Actual result
Invalid operation	No file	No file	No results	The dialog box pops up and prompts the user: "Please upload the file"
Invalid operation	Wrong file name	The filename is the same as the previously uploaded file	No results	The page records the last valuation result, resulting in the display of the wrong result
Invalid operation	Csv file with blanks	Csv file with blanks	No results	The dialogue box pops up and prompts the user: "Please check the form for empty values"
Valid operation	Correct(complete) Csv file	Correct Csv file	Get the result	Show the result See Figure x

Deviation： None

Test result： The system can do the main functions, but when the file name is the same as the previous file, we can't overwrite the previous file by using the operating system's command because we can't get the cslinux permission.

Deficiency： The user has to change the file name and make sure the file name contains their user name every time they upload. This somewhat reduces the user experience, which is a deficiency of the system.

4 Summary Evaluation

The system basically meets the requirements in the requirements specification and can be delivered for use.

III. User manual

1.0 General Information

1.1 System Overview

This is a web-based system which provide the automated valuation of properties. Users can log into this system and get their own properties valuation by uploading the required property information.

Users can choose to valuate by different users. The system will provide the results within different ranges according to the required scenarios.

2.0 Getting Started

2.1 Run the front-end & server & model

2.1.1 Open one terminal, change directory to team08-v2 folder. Run 'npm run dev' in command line. The terminal will show the running address. Turn to this link in any browser, e.g. Firefox. The homepage will be presented.

```
[Team201908@CSLinux team08-v2]$ npm run dev
```

```
 DONE  Compiled successfully in 500ms

 I  Your application is running here: http://localhost:8083
```

2.1.2 Open another terminal, change directory to team08-v2/server. Run 'node index' in command line. The server will run as the figure below.

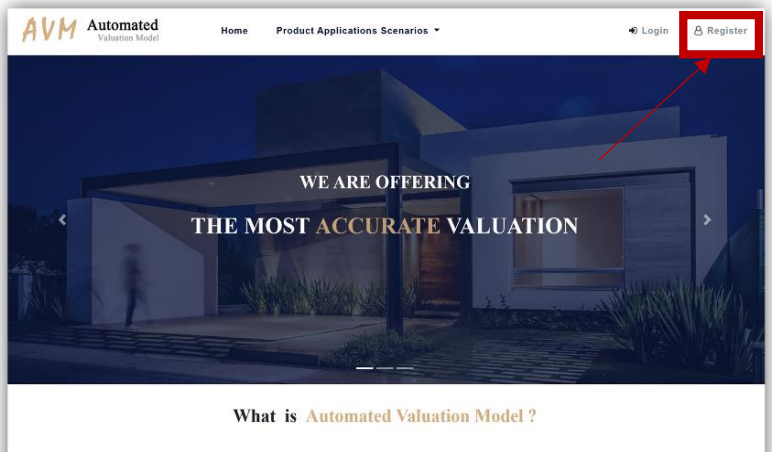
```
[Team201908@CSLinux server]$ node index
success listen at port:3000......
```

2.1.3 Open another terminal, change directory to 'models' folder on the desktop. Run 'python3 randomforest.py' in command line.

```
[Team201908@CSLinux Models]$ python3 RandomForest.py
```

2.2 Registering and logging

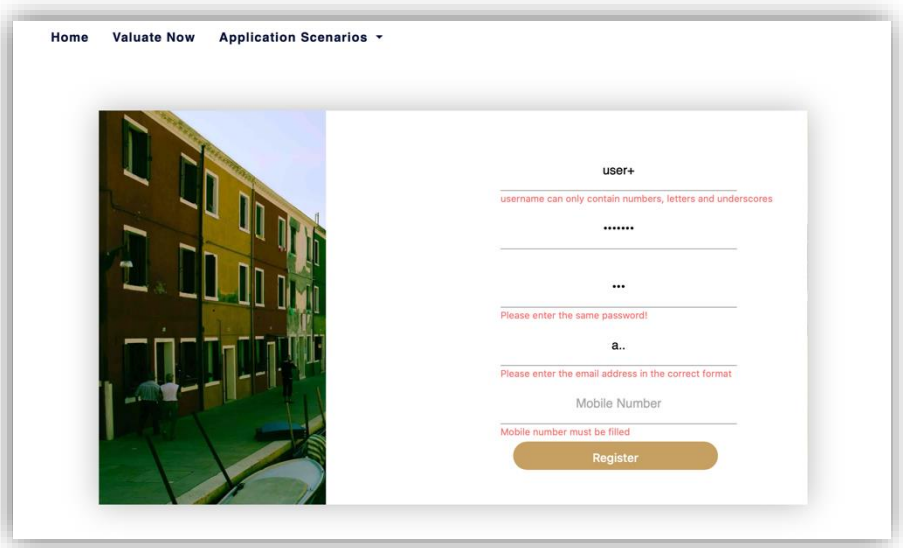
2.2.1 New users can register through the link on the navigation.



The user can create an account in the registration page.

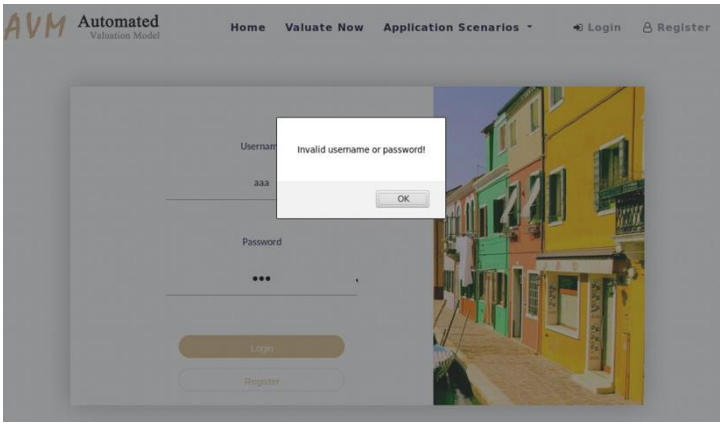
Username and password are only available with numbers, letters, and underscores. Username should be unique. Email address and mobile number should also be in the correct forms. (e.g. Email address: xx@xxx.xx, mobile number: 1xxxxxxxxxx (in all numbers)). Click 'Register' button, if the information passes all the validations, the webpage will pop out an alert window saying "Submit!" and jump to the login page.

If any value does not pass the validation, the page will present the responding warning below the block.



2.2.2 Users can login in the login page (see figure.)

The username and the password should be registered in advance. If user input a wrong username or password, the website will give a warning and allow users to try another time.

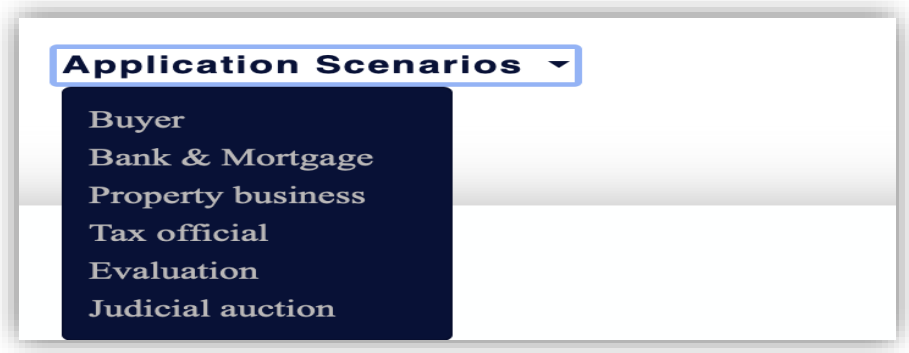


Click 'Log in' button, if the information passes all the validations, the webpage will pop out an alert window saying, 'Login successfully!' and show two buttons to jump to different pages. At this moment, the 'Login' and 'Register' entries should be instead by the user's username.

2.3 View the application for different characters.

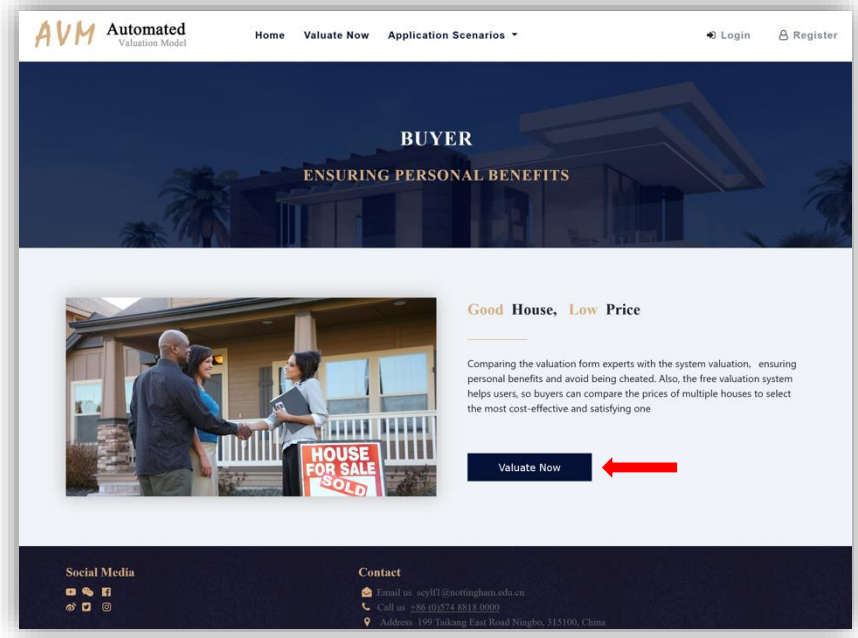
2.3.1 Users can view the brief introductions of different user characters in the homepage and click "Details" button to jump to the character details page.

2.3.2 Users can also directly go to the responding character pages through the navigation bar.

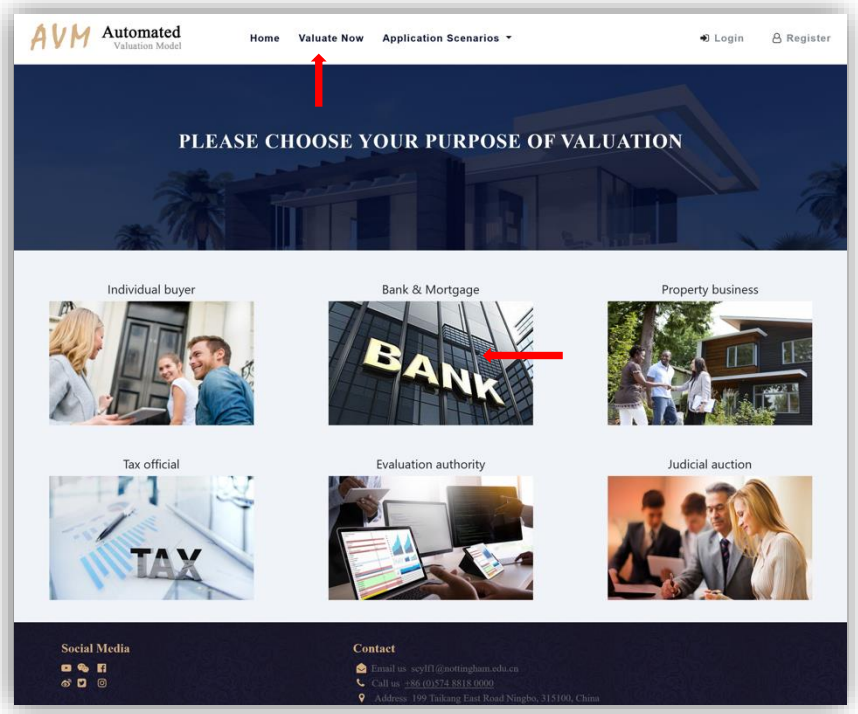


2.4 Get the valuation result

2.4.1 Users can jump to the valuation (form filling) page by clicking the button in each character page. **Notice: This step requires the user to login, or it will jump to the login page**.



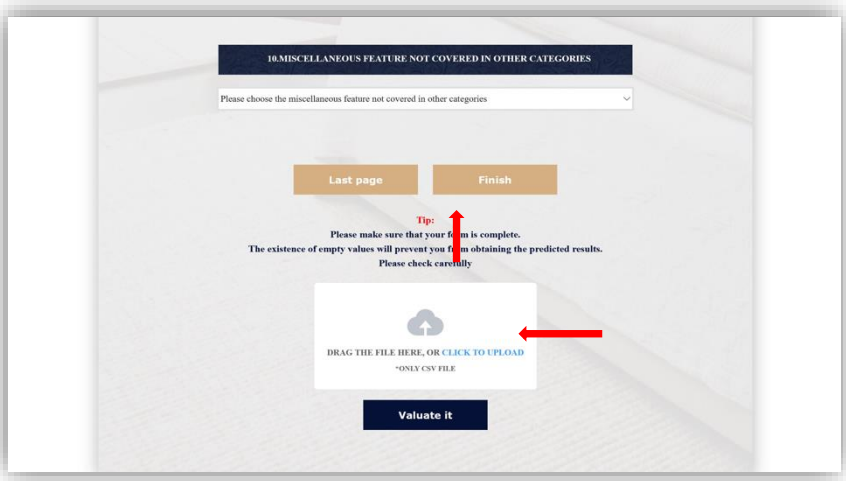
2.4.2 Users also can click on 'Valuate Now' to start the valuation.



Then, they can click on one of the pictures (their purpose of valuation) to jump to the valuation (form filling) page. **Notice: This step requires the user to login, or it will jump to the login page**.

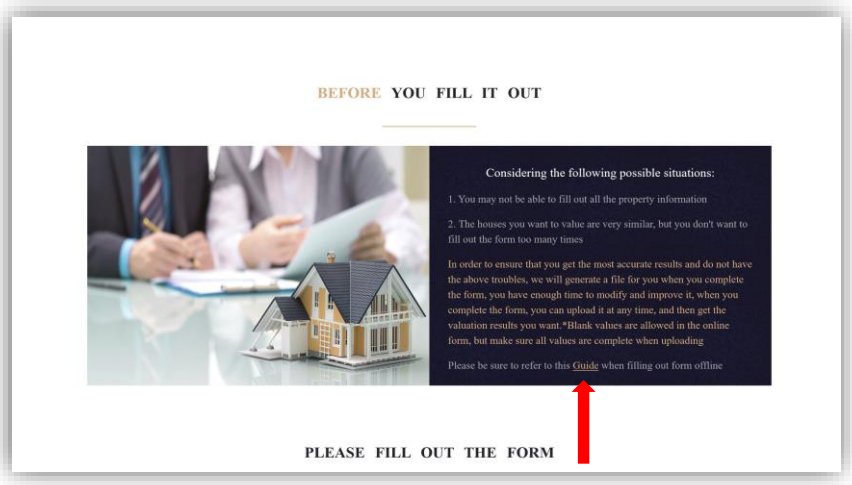
When the user comes to the form filling page, they should fill out the information about their property. When they have finished filling out the form, they can click 'Finish'. At this time, the system will generate a CSV file for them, which contains the information they filled out. Users can modify and improve it offline. Once they've filled it out completely, they can upload the file and click 'Valuate it' to get the final result.

Notice: This step requires the user to upload a complete form without any empty values, and the file name must be the same as the username. Otherwise, they won't get results.

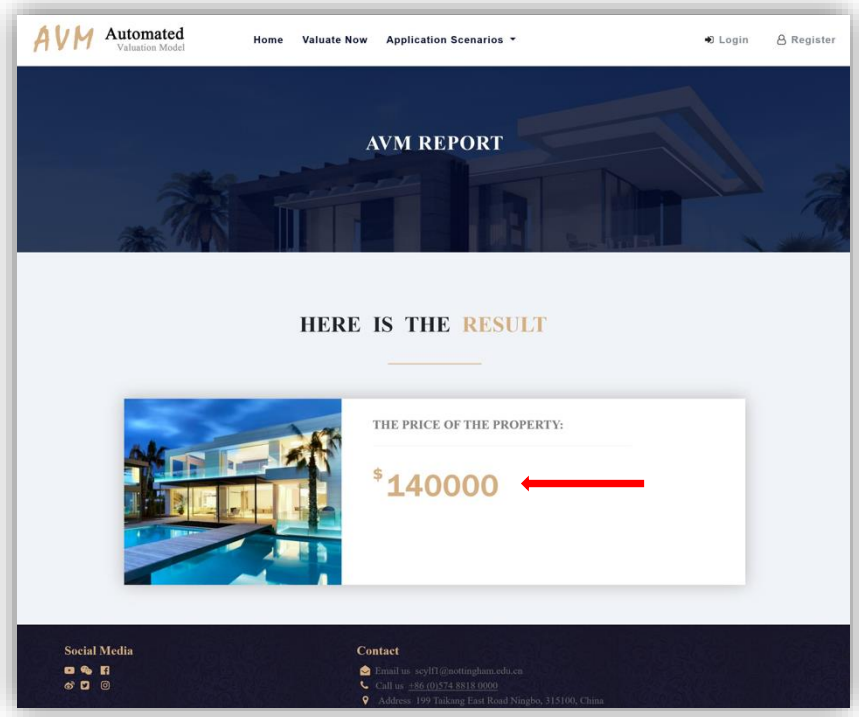


When users fill out the form offline, we provide them with a guidance manual that tells them how to fill in the form correctly. Users can

download it by clicking 'Guide'.

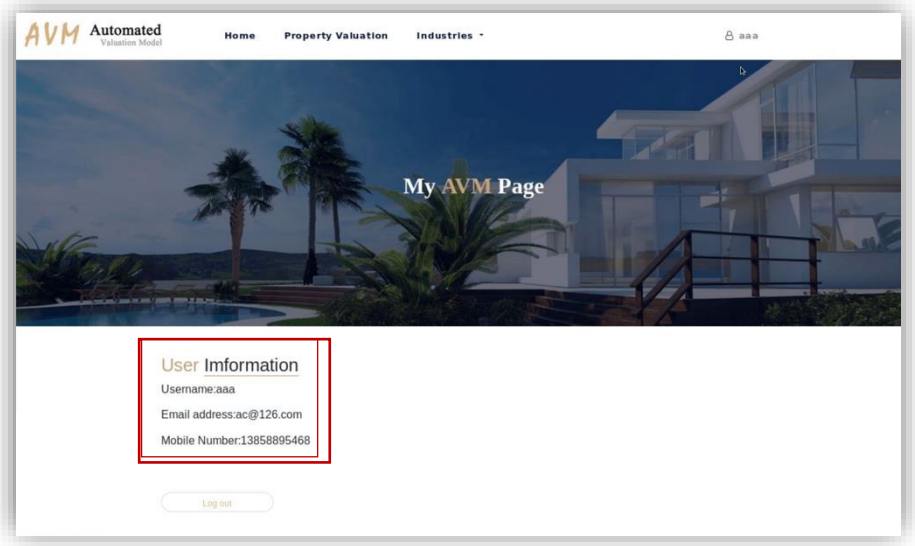


2.4.3 When the user clicks' Valuate it ', they jump to the results page, where they will see the final price of the property.

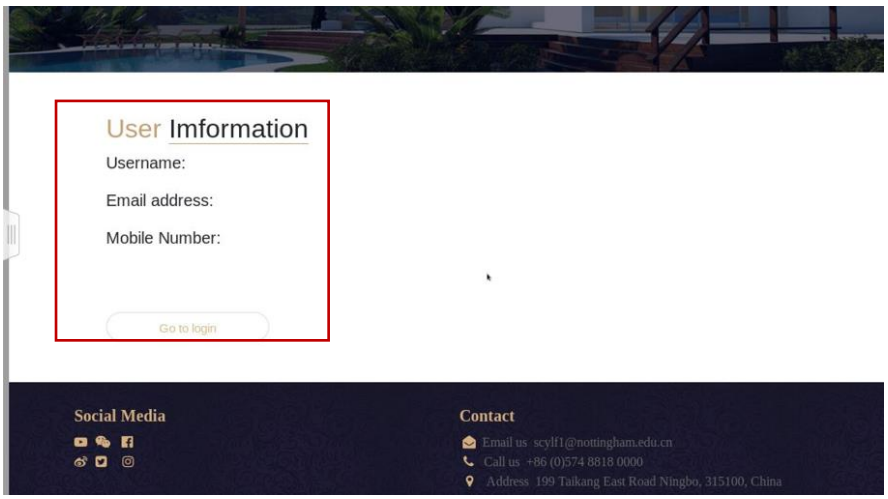


2.5 Check user information and log out

2.5.1 Users can get their account information by clicking their username on the left-top corner.



Once the user clicks log out button, the cookies of user's account information will be deleted, and the page will present a button to login again.



IV. Reference

[1] George M (2017), “Automated Valuation Models (AVMs): A brave new world?”, Wroclaw Conference in Finance 2017. Cracow University of Economics, September.

[2] Faishal Ibrahim, M., et al. (2005). "Automated valuation model: an application to the public housing resale market in Singapore." Property Management 23(5): 357-373.

[3] Su X, Yan X, Tsai C (2012), “Linear regression”, Volume 4, issue 3, pp. 275-294. Doi: https://doiorg.ezproxy.nottingham.edu.cn/10.1002/wics.1198

[4] Euge Inzaugarat (2018). Understanding Neural Networks: What, How and Why? Available at: https://towardsdatascience.com/understanding-Neural-Networks-what-how-and-why-18ec703ebd31 (Accessed: 9 April 2020).

[5] Farhad Malik (2019). What Are Hidden Layers? Available at: https://medium.com/fintechexplained/what-are-hidden-layers-4f54f7328263 (Accessed: 9 April 2020).

[6] Jordi TORRES.AI. (2018) Learning process of a Neural Network Available at: https://towardsdatascience.com/how-do-artificial-Neural-Networks-learn-773e46399fc7 (Accessed: 22 April 2020).

[7] Breiman, L and Cutler, A (2004). Random Forests Leo Breiman and Adele Cutler. Available at: https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm (Accessed: 10 December 2019).

[8] Shneiderman, B., & Plaisant, C. (2004). *Designing the user interface : Strategies for effective human-computer interaction* (4th ed.). Boston: Pearson/Addison Wesley.

[9] Provost, Foster, and Tom Fawcett. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. OReilly, 2013. Pp.27

[10] Drakos, Georgios. “How to Select the Right Evaluation Metric for Machine Learning Models: Part 1 Regression Metrics.” Medium, Towards Data Science, 5 Dec. 2018, https://towardsdatascience.com/how-to-select-the-right-evaluation-metric-for-machine-learning-models-part-1-regrression-metrics-3606e25beae0.

[11] Bellotti, A. (2017). Reliable region predictions for automated valuation models. Annals of Mathematics and Artificial Intelligence, 81(1-2), 71-84.

[12] De Cock, D. (2011). Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project. Journal of Statistics Education, 19(3).

[13] Philipp Probst, Marvin Wright, Anne-Laure Boulesteix (2018). Hyperparameters and Tuning Strategies for Random Forest. DOI:10.1002/widm.1301