

Area 38: An AI-based Business Intelligence Website

Group 38: Qiupu Shen, Wencheng Zhang, Xiaohang Tang, Yixing Lu, Yongyin Yang, Yunfan Shi

1 Project Description

Business Intelligence (BI) is an integrated technology-based solution, providing useful insights into business data and empowering business decisions accordingly. However, most existing BI platforms are suitable for large companies and require professional training to use, which seems too complicated and unfriendly for small business owners. Therefore, in this project, we propose to build a lightweight but just right AI-based Business Intelligence Website for everyone. Our website will provide visualized analysis for user uploaded business data, such as financial time-series spreadsheets and target customer persona with customer survey data. In addition, we provide interactive guidance for pricing strategy to optimize profits. Finally, our platform enables customized business metrics forecasting based on historical performance. We hope our project can facilitate the wide application of BI into more business cases.

2 Aims & Objectives

2.1 Aims

We aim to build a lightweight, all-visualized and data-safe BI website, which enables businesses to improve their data-driven decision making. Specifically, we expect to:

1. Provide customer persona analysis to locate optimal/potential customer group
2. Provide optimized pricing strategies for a higher profit or customized goal
3. Provide concise and clear business metrics forecasting

2.2 Objectives

1. Implement user account operations (register/login/change password/delete account data) for website service

2. Implement a real-time interactive demand curve to provide optimized pricing strategy for a higher profit or customized goal

Why do you need
a user account?
What does this
mean in practice?

3. Implement algorithms such as Recency, Frequency, Monetary Value (RFM) analysis to find optimal/potential customer group

4. Implement a business metrics forecast dashboard derived from Statistical/Machine Learning (ML)/Deep Learning (DL) models

5. Enable users to upload required files. If there is a upload failure like network or browser issue, remind users to upload again and clear the incomplete buffer data.

6. Enable users to download detailed report about the corresponding service

7. Enable administrator to monitor database user accounts and transactions

8. Enable service engineer to upgrade BI services, such as Statistical/ML models

9. Enable Product Manager to modify contents in report templates

10. Enable database to store analysis history of a certain account for user to reference

3 Key Literature & Background Reading

Business decisions currently rely more on data analytics [1]. Developed AI libraries and frameworks enable the training and analysis of mass data where most analytic method based [2] and help put complex Mathematical models into practical uses [1]. In success of former studies [3], existing AI business tools are either too expensive or too complex for some small-scale individual businesses. Thus, a website is decided to be especially designed for small individual business use. It offers efficient business analysis which especially focus on price related decision optimization and customer analysis. Besides, to realize the clearance and accuracy of the website, implementation inspiration and two aspects of technique issues will be discussed.

Some sources inspired the logic and implementation of the website. Previous research [4] enhances the

thoughts of using various data and visualization techniques to support the business service in the website. Another platform [3] provide basic knowledge to help locate potential users and applied field of this website. Others experience gives range to compare and choose proper tools during different stages of development and Python is recommended. Some papers [5] provide technique supports with regard to the core coding task: Machine Learning.

Firstly, technical literature related to front-end development is investigated and summarized. Vue.js typically works as a progressive JavaScript (JS) framework for building user interfaces. In terms of data visualization, however, Apache Echarts and Antv provide a more integrated and reliable solution compared to naive JS.

Secondly, back-end development takes server, functional logic and database into consideration. Based on web server pricing and capacity [6], the back-end team evaluates every related functional requirement accordingly. Hence, Django is selected as a Python based scalable and secure back-end framework which enables DL financial models to make business metric predictions. MySQL with Django API supports back-end database while online file service APIs are used for external data fetch. Sklearn with Pandas is a feasible library combination for basic machine learning prediction and data processing. R language is used to implement basic statistic data visualization, demand curve related service and target customer persona, embedded in Python to balance between efficiency and back-end compatibility.

4 Development & Implementation Summary

Our implementation mainly consists of five stages: requirement engineering, project design, project implementation, system testing, deployment and maintenance. After investigating our project requirements, we designed our system to include four main services for building a BI website. For project implementation, we firstly divide our team into three separate teams: Frontend, Backend and Testing/Product. We implement our system in a waterfall-agile mixture fashion. Specifically, in an iteration, our frontend and backend teams work on the same version of one milestone service in parallel, during which unit/component testing is done by the same developers. Then we integrate frontend and backend together into one piece and a separate testing team conducts

integration testing. During testing, the development team can move on to the next milestone service. Once feedback is generated by test team, that version will be fixed accordingly and archived. After all milestone services are implemented, tested, verified and validated for a certain number of iterations, we conduct system testing on the whole system. Finally, the system is deployed on server and maintained. For more details of the workflow, please refer to the Gantt Chart in Section 10.

Our development environment is briefly summarized as below:

IDE: Visual Studio Code

Frontend:

1. Use Vue.js architecture to develop basic HTML and CSS
2. Use jQuery to develop dynamic effect of interface
3. Use Echarts and Antv to integrate different kinds of charts into the architecture

Backend:

We intend to utilize Django back-end framework, MySQL DBMS and cloud storage APIs to implement local file upload and file fetch from cloud storage sources. For target customer group visualization, algorithms such as RFM analysis will be used. Besides, to achieve primitive data visualization, demand curve fitting and business metric forecasting, we plan to leverage techniques such as R & Python integration approach, Python with sklearn & Pandas and Django + PyTorch deep learning.

What strategies will you use for visualization?
Why would someone use your system rather than Excel?

5 Data Sources

We use 11 Kaggle datasets for testing our product and assume that users will only upload files from these datasets during software tests. A wide range of time series datasets is contained. Specifically, there are ones on Microsoft stock, Bitcoin price, business sales and household energy. These datasets will be used for data visualization and prediction. Besides, we have one dataset on customer surveys, which is used for target customer group visualization. All these datasets are obtained from Kaggle's public datasets [7], licenses, permissions and legal statements provided. Hence, we confirm that they are used with permission and personal privacy is respected. Additionally, we intend to use

quite high level many systems will have the same model

questionnaires to record users' feedback in interface testing and let beta testers document any problems that they encountered during link testing. Both will strictly comply with the ethical guidelines. To better assure its confidentiality and anonymity, we are going to anonymize personally identifiable data.

6 Testing & Evaluation

With respect to final testing and evaluation, we mainly focus on four areas which are functional testing, interface testing, performance testing and safety testing. When it comes to functional testing, we take link testing into major consideration. In the first stage, we are going to test whether all links correspond to the correct page as indicated. Here, we let beta testers document any problems which they may encounter during testing and report them back to developers. In the UI interface testing, we decide to design a manual testing measure that comply in testing techniques which are based Five Second Test. To be specific, participants are allowed to complete specified tasks using questionnaires in order to identify whether the homepage content is clear and concise. For performance testing, we mainly concentrate on two aspects which are stress testing and volume testing in order to inspect the responsiveness and stability of the system performance under a regulated load. In stress testing, we decide to use boundary value analysis to identify the maximum performance of our web pages under high traffic or data processing. In addition, we use equivalence class partition method for our volume testing to identify the website performance of varying amounts of data under our specifications. For safety testing, we mainly use equivalence class partition method to check whether the username and password on the registration and login page comply with limitations. Additionally, we will also test whether overflow errors occur, the website will result in system crash or personal information leakage.

this is all very generic

7 Ethical Considerations

We have carefully read through ethical guidelines and will strictly follow them. When it comes to the application of data in public domain, we ensure that all our data are obtained from Kaggle's public datasets with legal permission. Concerning the generation of new data, both beta test and Five Second Test guarantee that

participants are granted in full knowledge of possible consequences and all information before they fill in questionnaires. Worth to mention, participants are free to opt in or out of the investigation at any point of time. In aspect of the results of testing and evaluation, we remove all identifying information from our report, thus enabling us to prevent from personal information disclosure. Users can choose to delete all account data any time during our normal website service. Furthermore, at the end of semester, all user data will be wiped out from the server.

8 BCS Project Criteria

8.1 An ability to apply practical and analytical skills gained during the degree programme

We started to learn more technical details by self-studying, which is an important skill gained during the degree programme.

8.2 Innovation and/or creativity

We aim to build a BI platform for everyone with light weight but just right service, which is different from most existing BI platforms and indicates our creativity.

8.3 Synthesis of information, ideas and practices to provide a quality solution together with an evaluation of that solution

We evaluated related work about BI platforms and proposed our final project plan.

8.4 That your project meets a real need in a wider context

Our project will produce a BI platform which is suitable for everyone, including small business owners.

8.5 An ability to self-manage a significant piece of work

We self-scheduled our work according to the group plan and have a weekly group meeting to report the progress.

8.6 Critical self-evaluation of the process

Every week we evaluate the work done in the past week and discuss the problems with group members.

9 UI/UX Mockup

For UI/UX mockup, please refer to Figure 1. Note that we only include key UIs in our project proposal for clarity.

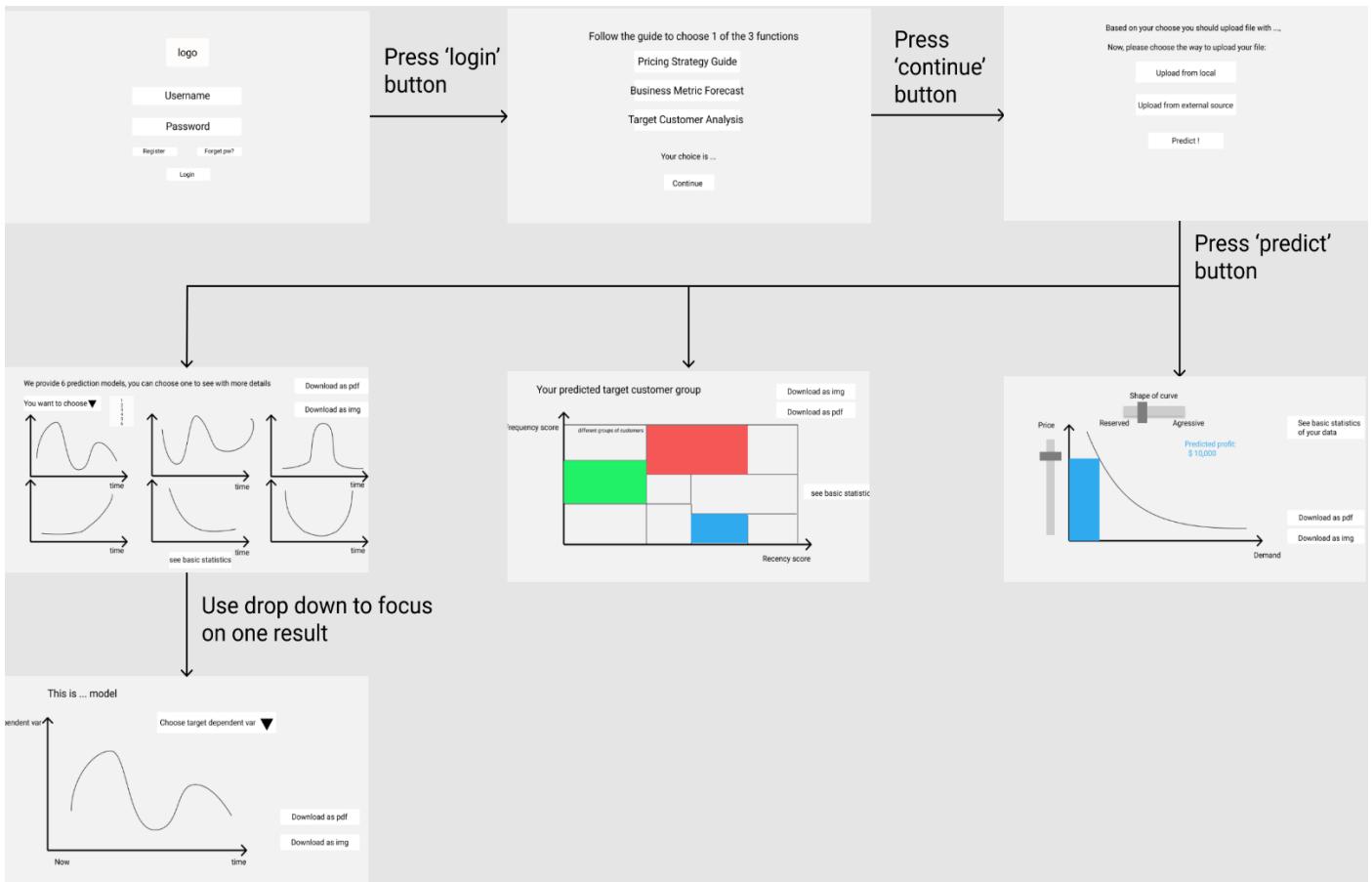


Figure 1: UI/UX Mockup

10. Project Plan including Risks & Contingency Plans

Considering practical situations of our project, we decided to develop our system in a waterfall-agile mixture fashion. Our project plan is shown below as a table (see Figure 3) as well as a Gantt chart (see Figure 4).

Risks & Contingency Plans are shown below as a table (see Figure 2). In terms of task dependencies, our project design, which decides view script, depends on requirement analysis. In addition, user interaction depends on function coding.

Risk	Probability	Effect	Solution
Financial problem	low	Catastrophic	Find alternative approaches to reduce expense
Hardware failure	low	Catastrophic	Backups
Group member lack of technical skills	moderate	Catastrophic	Do extra study or ask for help
Legal issues occur in our system	low	Catastrophic	Seek legal support
Group member's illness at critical point	moderate	Serious	Rebalance work allocation
Change of requirements lead to extra work	moderate	Serious	Modify existed process and reschedule the plan
Time arrangement problems	moderate	Serious	Evaluate progress and reschedule
Undiscovered bugs in our system	moderate	Serious	Halt our service and fix the issues
Unsatisfactory service quality	high	Tolerable	Ask for user feedback and improve our system

Figure 2: Risks & Contingency Plans

Task	Planned Start Date	Planned End Date	Duration	Participant	Label	Notes
Project Meeting	Every Wednesday, Start from Feb 2nd		<1 day, a meeting	All		
Technique Meeting	Every Saturday, Start from Feb 12th		<1 day, a meeting	All	Weekly Meeting	For emergency situations like unexpected bugs
Decide project subject	2022.2.2	2022.2.2	<1 day, a meeting	All		
Documentation Writing	2022.2.2	2022.5.11	Through the whole project	All		Documentation, daily updated
Former research searching	2022.2.3	2022.2.5	3 days	Qilupu Shen, Yongjin Yang		
Technique searching	2022.2.3	2022.2.5	3 days	Yixing Lu, Wencheng Zhang, Yunfan Shi, Xiaohang Tang		
Requirement analysis (Ass 1)	2022.2.3	2022.2.9	7 days	All		Till the last updating time, 90
UI Design	2022.2.10	2022.2.13	4 days	Yixing Lu, Wencheng		
View script of UI design	2022.2.13	2022.2.14	2 days	Yunfan Shi, Xianghong		
Specify Component	2022.2.13	2022.2.14	2 days	Yixing Lu, Wencheng		
Functional Coding	2022.2.14	2022.2.20	7 days	Yunfan Shi, Xianghong		
Vue Framework	2022.2.14	2022.2.16	2 days	Yixing Lu, Wencheng		
Backend Test	2022.2.14	2022.2.20	6 days	Yunfan Shi, Xianghong		
Add interaction Function	2022.2.18	2022.2.20	3 days	Yixing Lu, Wencheng		
Frontend Test	2022.2.20	2022.2.21	2 days	Yixing Lu, Wencheng		
Backend test	2022.2.21	2022.2.22	1 day	Yunfan Shi, Xianghong		
Joint Test	2022.2.22	2022.2.24	3 days	Qilupu Shen, Yongjin Yang		
UI Design	2022.2.25	2022.2.28	4 days	Yixing Lu, Wencheng		
View script of UI design	2022.2.28	2022.3.1	2 days	Yunfan Shi, Xianghong		
Specify Component	2022.2.28	2022.3.1	2 days	Yixing Lu, Wencheng		
Functional Coding	2022.3.1	2022.3.7	7 days	Yunfan Shi, Xianghong		
Vue Framework	2022.3.1	2022.3.5	5 days	Yixing Lu, Wencheng		
Prototype	2022.3.7	2022.3.8	2 days	Yunfan Shi, Xianghong		
Add interaction Function	2022.3.10	2022.3.12	3 days	Yixing Lu, Wencheng		
Frontend Test	2022.3.12	2022.3.14	2 days	Yunfan Shi, Xianghong		
Backend Test	2022.3.14	2022.3.16	2 days	Yunfan Shi, Xianghong		
Joint Test	2022.3.16	2022.3.11	3 days	Qilupu Shen, Yongjin Yang		
UI Design	2022.3.12	2022.3.16	4 days	Yixing Lu, Wencheng		
View script of UI design	2022.3.15	2022.3.16	2 days	Yunfan Shi, Xianghong		
Specify Component	2022.3.15	2022.3.16	2 days	Yixing Lu, Wencheng		
Functional Coding	2022.3.16	2022.3.22	7 days	Yunfan Shi, Xianghong		
Vue Framework	2022.3.16	2022.3.20	5 days	Yixing Lu, Wencheng		
Prototype	2022.3.20	2022.3.21	2 days	Yunfan Shi, Xianghong		
Add interaction Function	2022.3.20	2022.3.22	3 days	Yixing Lu, Wencheng		
Frontend Test	2022.3.22	2022.3.23	2 days	Yixing Lu, Wencheng		
Backend Test	2022.3.23	2022.3.24	2 days	Yunfan Shi, Xianghong		
Joint Test	2022.3.24	2022.3.29	5 days	Qilupu Shen, Yongjin Yang		
UI Design	2022.3.27	2022.3.30	4 days	Yixing Lu, Wencheng		
View script of UI design	2022.3.30	2022.3.31	2 days	Yunfan Shi, Xianghong		
Specify Component	2022.3.30	2022.3.31	2 days	Yixing Lu, Wencheng		
Functional Coding	2022.3.31	2022.3.31	2 days	Yunfan Shi, Xianghong		
Vue Framework	2022.3.31	2022.4.4	5 days	Yixing Lu, Wencheng		
Prototype	2022.4.6	2022.4.7	2 days	Yunfan Shi, Xianghong		
Add interaction Function	2022.4.4	2022.4.6	2 days	Yixing Lu, Wencheng		
Frontend Test	2022.4.6	2022.4.7	2 days	Yunfan Shi, Xianghong		
Backend Test	2022.4.7	2022.4.8	2 days	Yunfan Shi, Xianghong		
Joint Test	2022.4.8	2022.4.10	3 days	Qilupu Shen, Yongjin Yang		
UI Design	2022.4.11	2022.4.14	4 days	Yixing Lu, Wencheng		
View script of UI design	2022.4.14	2022.4.15	2 days	Yunfan Shi, Xianghong		
Specify Component	2022.4.14	2022.4.15	2 days	Yixing Lu, Wencheng		
Functional Coding	2022.4.15	2022.4.21	7 days	Yunfan Shi, Xianghong		
Vue Framework	2022.4.15	2022.4.19	5 days	Yixing Lu, Wencheng		
Prototype	2022.4.19	2022.4.20	2 days	Yunfan Shi, Xianghong		
Add interaction Function	2022.4.19	2022.4.21	3 days	Yixing Lu, Wencheng		
Frontend Test	2022.4.21	2022.4.22	2 days	Yixing Lu, Wencheng		
Backend Test	2022.4.22	2022.4.23	2 days	Yunfan Shi, Xianghong		
Joint Test	2022.4.23	2022.4.25	3 days	Qilupu Shen, Yongjin Yang		
Read and conclude requirements	2022.4.25	2022.4.25	0 days	All		
Prepare Functional Testing	2022.4.26	2022.4.28	2 days	All		
Prepare Acceptance Testing	2022.4.28	2022.4.30	2 days	All		
Prepare Performance Testing	2022.4.16	2022.4.30	15 days	All		
Prepare Safety Testing	2022.4.1	2022.4.15	15 days	All		
Start Testing Finished	2022.4.16	2022.4.25	10 days	All		
End and Conclude	2022.4.26	2022.4.30	5 days	All		
General Web Test	2022.4.26	2022.4.30	5 days	All		
Present Demonstration	2022.5.1	2022.5.3	3 days	All		
Emergency Time	2022.5.4	2022.5.10	7 days			
Finish and Submission	2022.5.11					
Stand for Team Work						
Stand for Backend						
Stand for Frontend						
Stand for Testing Work						
Stand for Core						

Figure 3: Project Plan Table

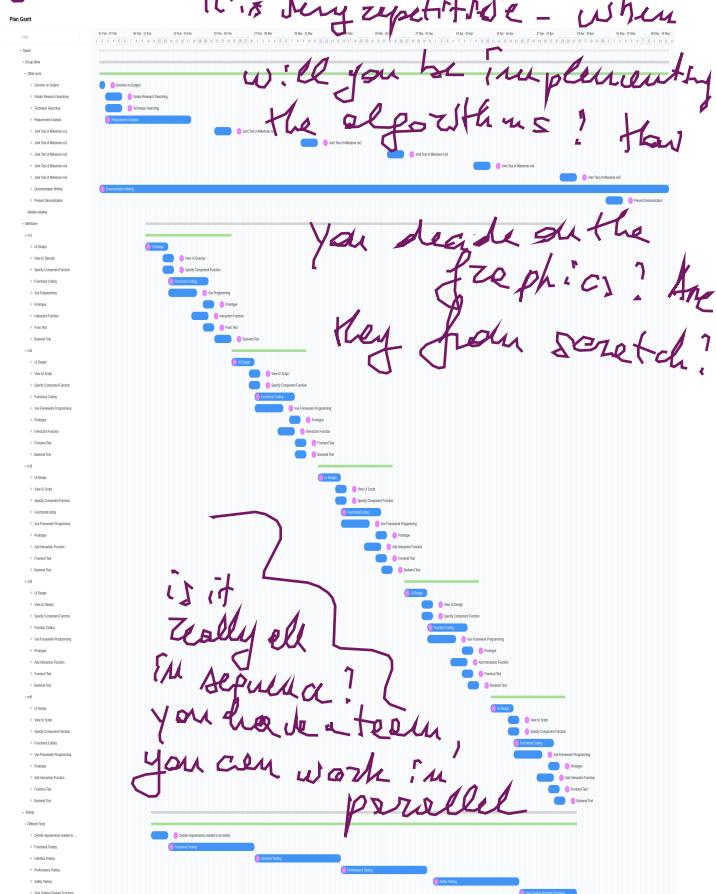


Figure 4: Project Plan Gantt Chart

11 References

- [1] B. Eastwood, "What Is Business Analytics? Common Uses, Core Skills, and Career Paths," [northeastern.edu](https://northeastern.edu/graduate/blog/what-is-business-analytics/), July 14, 2020. [Online]. Available: <https://www.northeastern.edu/graduate/blog/what-is-business-analytics/>. [Accessed: Feb. 09, 2022].
- [2] C. Bessega, "Four ways business analysts can use AI to work smarter," itproportal.com, Apr. 02, 2020. [Online]. Available: <https://www.itproportal.com/features/four-ways-business-analysts-can-use-ai-to-work-smarter/>. [Accessed: Feb. 09, 2022].
- [3] Business Intelligence Market, "BUSINESS INTELLIGENCE TOOLS," businessintelligencemarket.com. [Online]. Available: https://www.businessintelligencemarket.com/business-intelligence-software-comparison?gclid=Cj0KCQiA9OjPBhCOARIsAI0y71AVBBf7cyzOfjZ7RWJyEf8yx2MOKFtVwlM1zhFoazdXSUck79SFYsaAjfbEALw_wcB. [Accessed: Feb. 11, 2022].

- [4] S. Troy, "How to Integrate Web Development with Business Intelligence?" ictsd.org, Jan. 30, 2022. [Online]. Available: <https://www.ictsd.org/how-to-integrate-web-development-with-business-intelligence/>. [Accessed: Feb. 13, 2022].
- [5] A. Krishna, A. V. A. Aich and C. Hegde, "Sales-forecasting of Retail Stores using Machine Learning Techniques," *3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, pp. 160-166, Dec. 22, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8768765> [Accessed: Feb. 15, 2022].
- [6] N. Drake and B. Turner, "Best cloud computing services in 2022," itproportal.com, para.1, Jan. 04, 2022. [Online]. Available: <https://www.itproportal.com/guides/best-cloud-computing-services/>. [Accessed: Feb. 11, 2022].
- [7] A. Adam et al., "Kaggle Trending Datasets," kaggle.com. [Online]. Available: <https://www.kaggle.com/>. [Accessed: Feb. 16, 2022].