AN INTELLIGENT MOBILE APPLICATION TO ASSIST IN DETECTING METALS IN FISH AND SEAFOOD USING MACHINE LEARNING AND OBJECT DETECTION

Zixiu Qiao[1], Alan Xu-Zhang[2]

[1]St. Margaret's Episcopal School, 31641 La Novia Ave, San Juan Capistrano, CA 92675 [2]Computer Science Department, California State Polytechnic University, Pomona, CA 91768

ABSTRACT

The fish we eat everyday may well likely contain heavy metals, which are detrimental for one's health [1]. There are many solutions out there, such as machines and sensors that could detect the amounts of metals in a fish, but for the average person, these solutions are not the most convenient. My app, however, is an easy and convenient way to figure out if your fish has metals. All you need to do is take a picture of your fish and the app will use an AI system to inform you about the levels of metals in the fish and some tips and facts on how to cook the fish. I experimented with the AI accuracy and the accessibility of my app. I can say that the AI is quite accurate and my app is accessible to all IOS devices [2]. Ultimately, my app should be used by everyone because it's quick, easy to use and access, and doesn't require any professional knowledge about fish.

KEYWORDS

Machine Learning, Data Science, Object Detection, Mobile Application

1. INTRODUCTION

The problem I am trying to solve is the detrimental impact of the metals in the fish we eat, which can cause major health issues. Fish has always been a part of my family's diet as it provides good sources of protein and vitamins. Even though myself do not like to eat fish, my family members, especially my dad, eat fish almost every day. However, a few years ago, we found out that many fish contain harmful metals such as Mercury, Cadmium and Lead, leading to human poisoning [3]. Furthermore, Mercury in fish can cause neurological and developmental disorders, particularly harmful to young children. Wild Salmon, a commonly eaten salmon in the United States, is found to have "56.3 ± 12.9 μg/kg of Mercury … which is three times higher than compared to farmed salmon". It is important to take this problem into account because "more than 3 billion people around the world rely on wild caught and farmed seafood". If people continue to eat fish that contain harmful metals, it can lead to a downfall of global health [4]. It's really important to choose fish with lower levels of metals and follow guidelines on the portion and frequency of fish consumption [5]. My problem affects everyone in the long run as people will continue to eat fish. Humans have been eating fish for about 1.95 million years and today, with the right technology, we can help ourselves to become healthier by eating fish with minimal amounts of metals.

The first methodology uses a Conveyor Metal Detector, which is a machine to detect metals in fish. The shortcomings to this solution are that the machine is not yet ready to be sold, and it is a fairly large machine, making this solution not as convenient as the other two. The second methodology uses colorimetric sensors to visually detect the amount of metals in a fish. The shortcoming to this solution is that for average people, using sensors and modeling data could be quite tricky. My app does not require professional knowledge of anything and gets you the information you want quicker. The last methodology utilizes an image processor to examine the outside of a fish, especially its eyes, to figure out if it is a healthy or metal exposed fish. This solution is very similar to my solution so there aren't any major limitations except for the fact that it cannot examine the insides of a fish.

My method of solving this problem is to create an app where it can detect the amounts of metals in certain fishes just by taking a picture. My solution solves my problem by easily detecting the fish and the levels of metals in it. It will also provide useful tips and facts about the fish and furthermore, it will give you methods on how to cook the fish in order to be as healthy as possible. Thus, people will know which fish are healthy to eat, which are not, and how to cook the fish of their choice. This is an effective solution because you wouldn't have to go online and search for the different levels of metals in fish, as it could be quite tedious if you are searching for more than one fish. Moreover, it is more convenient since you just take a picture on your phone and the app will tell you whatever you need to know. Other methods such as searching up information on google, is not the best method as it lacks the information you need and it is difficult to find information on a specific species of fish. Why go on Google and search for countless fish facts when you can just take a picture on our app and get information at a way faster rate. Not only does our app contain fish, but it also contains seafood like shrimp, crab and lobsters, which is also commonly eaten by people all around the world.

In Section 4, I was trying to test the AI accuracy and the accessibility of my project. For testing the AI accuracy, I first researched a few facts about fish and asked Chat GPT the questions that I researched [6]. If it outputs something similar or very close to my findings, then I can say that the AI is accurate. For testing the accessibility of my project, I tried to access my app on IOS and the Android Studio Emulator. If both systems can access all the features of my app, then my app is easily accessible for various systems. My findings for experiment one was the Chat GPT got most of the questions partially correct and some correct. As for experiment two, IOS could easily access all the features of my app while the Android Studio Emulator could not access the camera feature [7]. The reasons for my results were that Chat GPT did not have enough information and data to get every question correct and the Android Studio Emulator was operated using a Macbook and did not have access to the camera.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Making the App Available on iOS devices

One challenge I had with my project was making the app available on iOS devices. Since I used Android Studio to create my app, it was only available to Android users. It was very difficult to make the app available to iOS users and I had to download many other programming apps to solve this problem. One way to resolve this problem could be to connect your iOS phone to your laptop using a charging cord. By doing this, it adds a new device to the Andriod Studio app and allows you to access the app on your iOS phone.

2.2. The Coding Language

The coding language for this app has been a challenge for me as it was one that I've never encountered before called flutter. I've previously only used java and python so I had no experience at all with this new language. It was challenging at first to understand and utilize the language effectively, which caused many errors in my code. Also, the Andriod Studio app was confusing to maneuver around at first, which often slowed down my progress. However, after I gained some experience with the coding language, I was able to create and debug most of the pages on my app.

2.3. Designing the Pages on the App

Designing the pages on the app was a major challenge since it was my very first app. Choosing the right colors and organizing the page was probably the most difficult part of the app. Also, the different titles, fonts, and headers for each page were a challenge as there were so many options to choose from. One solution could be to go online and search for popular app designs or take inspiration from already existing apps like Instagram and Twitter [8]. By doing so, we would be designing and learning how to design from successful apps at the same time.

3. SOLUTION

The main structure of my program would be the homepage, where you can access your profile, news, daily tips and facts, and a camera page. The three major components that my program links together are the Home, News, and Camera pages. To construct this, we used the programming language flutter [9]. When you first enter the app, a splash screen will appear, showing the logo of our app. Once you've entered, you will directly go to the homepage, where you can see buttons for the analyze page, news page, camera page, and profile page. The analyze page is the app's main function as it's the page where you take a picture of a fish. After you take a picture, the app will take you to a new page where the information about the fish will be given back to you. The news page acts as an informational page, where you can see the latest fish-related news from many different news article companies. The profile page is just like any other regular profile page, but it will ask for your medical information to further assist you with your selection of healthy fish. Each page is linked together, creating a flow between them. The buttons to transfer to another page are located at the very bottom of the screen. This makes it easier and more efficient to get where you want. The back button is located on the top left of the screen, where it will take you to the previous page you clicked on.
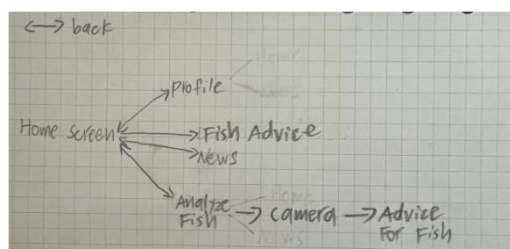

Figure 1. Overview of the solution

The purpose of the home page is to introduce you to the app and the features of the app. Various shapes were used to construct this page, which was later turned into buttons so that the user could navigate to where they wanted. I also used color to decorate the background of this page.

Figure 2. Screenshot of the main page

Figure 3. Screenshot of code 1

The code on top is the code for the Analyze page button. This code is written in the homepage file as the button is located in the homepage. The first few lines represent the position of the Analyze page. The button is adjusted to on towards the top of the screen. The 7th line creates the white padding about the text, making it look like a button. Inside the child: ElevatedButton() is where the actual button is created. The child: Text() creates the text Analyze Fish and gives the option to choose the textstyle and the fontsize of the text. The second screenshot shows the code for the background of the homepage. The background is divided into 3 sections and the blue and green section is shown in the code. It is essentially 3 rectangles combined together to make the background three different colors. The child: Container allows the code to adjust the color, height and width of the background shape they create.

The purpose of the news page is to give users news relating to fish and seafood. The page will have a short summary of a recent news article and the link to the article will be provided below the summary. The link is connected with a URL that will take you online.

Figure 4. Screenshot of the news page

Figure 5. Screenshot of code 2

This block of code creates the link button. The first few lines of code operate the same as the code in the analyze page, adjusting the position of the link. The child padding creates the padding on the text and the child text creates and adjusts the text. The onTap: _launchURL line makes it so that when you take on the link, it will launch the URL to the article you are trying to read about [10]. The second screenshot shows the code for the splash screen, which appears right as you open the app. The Image.asset line allows us to input an image on the screen and adjust its properties. One property inside of Image.asset is fit. This property keeps the image inside of the parent widget. The reason the code for fish background is in front of the code for fish logo is that we need to make sure the logo fits right on top of the background.

The purpose of the analyze page is to take in a picture of your fish and give you information and facts about the fish. One Service that was implemented in this system is the Drop Down Menu which gives you the location of different fish so that you can be more specific.
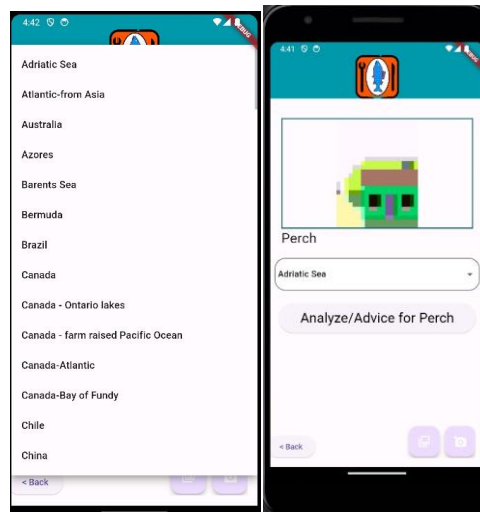
Figure 6. Screenshot of the functions

Figure 7. Screenshot of code 3

The block of code inside the child container creates the shape of the drop down menu, much similar to the codes shown before. The borderRadius however, makes the corners of the rectangle circular, instead of rectangular. The block of code inside the child padding, creates the features of the drop down menu. The first line of code inside Padding creates the padding for the drop down menu. The second line hides the underline and the third line states that the values inside the drop down menu have to be strings. The value: _selectedLocation line stores the values inside the drop down menu inside selected Location. The line after that changes the value whenever you click on a new value. The block of code under the items: LocationOptions line names the lines inside of the drop down menu. Value is stored inside location and location is converted to text, which is then placed inside the drop down menu.

4. EXPERIMENT

4.1. Experiment 1

One blind sport I want to test out in my program is the AI's accuracy. Since I used AI to identify the types of fish, there can be problems with its accuracy due to a lack of information provided to the AI.

To test the AI, I will research a few particular types of fish and get as much information as I can. I will then ask the AI about the type of fish and their attributes. If it outputs the data that I'm looking for, then the AI is accurate. The experiment is set up this way because it's important for the creator himself to understand the fish in his app first and test the AI to see if it is an accurate resource to be used. An expert at something is always more knowledgeable on the topic than an AI.
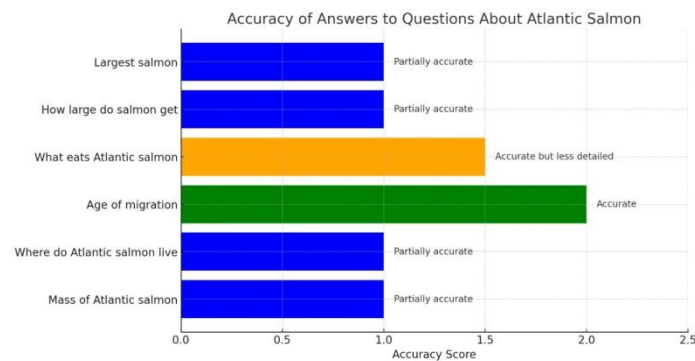

Figure 8. Figure of experiment 1

The mean of the data is 1.25 and the median of the data is 1. The lowest value in the graph is 1. The highest value is 2. The data met my expectations because most of the responses that chat gpt gave were partially accurate. I did not expect chat gpt to answer everything accurately because it isn't developed to the highest level yet. I think it turned out to be this way because AI systems like chat gpt still need more information to perform actions flawlessly. I have been using chat gpt for about a year and a half and I notice that sometimes it outputs wrong answers to simple questions. The biggest effect on the results was realizing that AI systems for apps are not always reliable and it's better to use the information from researching on the internet instead of AI for some parts of the app.

4.2. Experiment 2

The accessibility of the app on different platforms and how well the features work was another blind spot. The app couldn't open the link to some websites and the camera feature does not always work.

To test out the accessibility of the app, I will open the app on different platforms such as iOS and the Android Studios emulator. I set it up this way to see which platforms can access the app and all its features and which platforms cannot. It is important for people to access the app from any platform and not be limited to only one. The thing that will separate if a platform can and cannot access the app is the camera feature, making it the main goal of this experiment. If a platform cannot access the camera feature, then people can't use the app in the intended way.
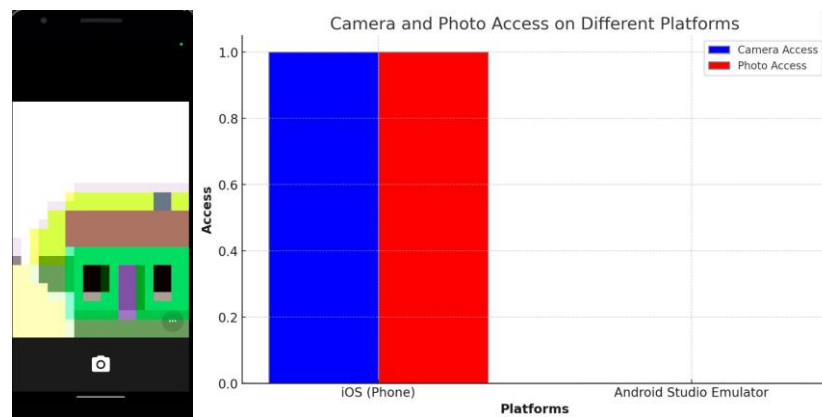
Figure 9. Figure of experiment 2

The mean of the data is 1. The median of the data is also 1. The lowest value of the data is 0 while the highest value is 1. The data met my expectations because running the app on an iPhone usually allows it to access the camera and photos app. The emulator ran on a computer and depending on if the computer has a camera or not, the app may not have access to the camera feature. My emulator was an Android phone but on a computer, therefore not have access to the camera feature. The first picture is a picture of my emulator when I press the option to take a picture. Instead of having access to a camera, a pixelated figure appears. The biggest effect on the results from this experiment is realizing that I have to make the app accessible for every platform in the future.

5. RELATED WORK

The company Easy Weigh has a product called the Conveyor Metal Detector [11]. According to their website, the product "Detect metal contaminants during food processing with high sensitivity and stability for ferrous metals, non-ferrous metals, and stainless steel." This solution is pretty effective, as it can straight up detect the metals in a fish. The limitations to this product are that it is difficult to purchase and it's a fairly big machine. One thing that it ignores is the fact that not all people are willing to purchase a machine just for detecting metals in a fish. One thing my project does better than their product is that my app is easier to access and we provide facts on the fish and how to cook them as healthy as possible.

The National Natural Science Foundation of China uses colorimetric sensors to visually detect the amount of metals in the forty eight fish samples they tested [12]. They then used an extreme learning machine to model the data from the sensors. This solution is effective because it is simple and gets straight to the point. However, one limitation to this solution is that it is not as convenient as my project. Using sensors and modeling the data it received is far more tedious than just taking a picture of a fish and receiving information instantly. Despite being less efficient, this solution is still a smart and innovative solution.

Another solution I found was where an image processor was used to examine the exterior of a fish, especially a fish's eyes, to determine if it is a healthy or heavy metal exposed fish [13]. Their solution is very much similar to mine, as they also take a picture of a fish and examine it based on what they see. I don't see any limitations other than the fact that they cannot examine the insides of the fish. My project improves on what they have by including information and facts like how to cook and eat the fish after the user has taken a picture of their fish.

6. CONCLUSIONS

One limitation that my project has is the lack of information available due to the insufficient information provided to the AI system [14]. This can cause the app to give out little information or sometimes incorrect information. Another limitation is that the camera feature can only identify one fish at a time. If you were to take a picture of 5 fishes, then the app cannot identify all 5 fishes at the same time. This needs to be fixed as it can help the app become more efficient. If I had more time with my project, I would gather a lot more information to put in the AI system to ensure that the information it provides it true and detailed [15]. As for the camera, I don't have anything in mind that can enhance the camera to identify more than one fish. However, with the advancements of technology, there will surely be a feature like that in the future.

I think that this app is designed pretty well, considering it's the first app that I've created. All the essential pages like the news page and analyze page have been implemented in the app and it is easy to maneuver around to the pages you like. I'd say that the app is a success, but there are still more features that can be added.

REFERENCES

[1]   Tchounwou, Paul B., et al. "Heavy metal toxicity and the environment." Molecular, clinical and environmental toxicology: volume 3: environmental toxicology (2012): 133-164.
[2]   Liu, Chang, et al. "Status and trends of mobile-health applications for iOS devices: A developer's perspective." Journal of Systems and Software 84.11 (2011): 2022-2033.
[3]   Vallee, Bert L., and David D. Ulmer. "Biochemical effects of mercury, cadmium, and lead." Annu. Rev. Biochem.;(United States) 41 (1972).
[4]   Chen, Jingwei, et al. "Review on metal dissolution characteristics and harmful metals recovery from electronic wastes by supercritical water." Journal of Hazardous Materials 424 (2022): 127693.
[5]   Kris-Etherton, Penny M., William S. Harris, and Lawrence J. Appel. "Fish consumption, fish oil, omega-3 fatty acids, and cardiovascular disease." circulation 106.21 (2002): 2747-2757.
[6]   Fuchs, Kevin. "Exploring the opportunities and challenges of NLP models in higher education: is Chat GPT a blessing or a curse?." Frontiers in Education. Vol. 8. Frontiers Media SA, 2023.
[7]   Xu, Fenghao, et al. "Android on pc: On the security of end-user android emulators." Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. 2021.
[8]   Laor, Tal. "My social network: Group differences in frequency of use, active use, and interactive use on Facebook, Instagram and Twitter." Technology in Society 68 (2022): 101922.
[9]   Tashildar, Aakanksha, et al. "Application development using flutter." International Research Journal of Modernization in Engineering Technology and Science 2.8 (2020): 1262-1266.
[10]   Le, Hung, et al. "URLNet: Learning a URL representation with deep learning for malicious URL detection." arXiv preprint arXiv:1802.03162 (2018).
[11]   Yoo, Jae-Sang. "A Study on Development of Metal Detector on Belt Conveyor in Material Plant." Proceedings of the KIEE Conference. The Korean Institute of Electrical Engineers, 1997.
[12]   Wu, Ying, et al. "Colorimetric/Fluorescent Sensing of Metal Ions and Advanced Molecular Information Safety Based on Fish Scale-Derived Carbon–Gold Nanocomposites." ACS Applied Nano Materials 6.20 (2023): 18767-18779.
[13]   Martignac, François, et al. "The use of acoustic cameras in shallow waters: new hydroacoustic tools for monitoring migratory fish population. A review of DIDSON technology." Fish and fisheries 16.3 (2015): 486-510.
[14]   Van de Poel, Ibo. "Embedding values in artificial intelligence (AI) systems." Minds and machines 30.3 (2020): 385-409.
[15]   Van de Poel, Ibo. "Embedding values in artificial intelligence (AI) systems." Minds and machines 30.3 (2020): 385-409.