Cereal Analysis Based On Ratings By Using Machine Learning Techniques With IBM Watson

1. INTRODUCTION

1.1 Overview

The project objective is to find the high dietary food that is predicted on the basis of rating of the food. A customer wants to buy some food items with high dietary benefits so that he wants to know which food item has high dietary benefits. It is so difficult to choose an item. Usually a customer expects to consume dietary cereals with high proteins, fiber and low sugars, fats. Predicting a brand with high dietary cereals became a big issue.

1.2 Purpose

The purpose is to find the high dietary food that is predicted on the basis of rating of the food. A customer can get high dietary food by the rating of the food given to it from the cereals and ingredients present. The model can predict the rating of the food more accurately by giving the inputs which are the cereals and ingredients present in the food. it is important to know the relationships between things like calories, sugar, vitamins, and how these factors related to the rating of a cereal. It would be very useful for manufacturers to be able to receive higher ratings on their cereals while at the same time offering healthier options for consumers.

2. LITERATURE SURVEY

2.1 Existing Problem

A customer wants to buy some food items with high dietary benefits so that he wants to know which food item has high dietary benefits. It is so difficult to choose an item. Usually a customer expects to consume dietary cereals with high proteins, fiber and low sugars, fats. So it is difficult to find the beneficiary cereals by manual examination. Therefore, Predicting a brand with high dietary cereals became a big issue.

2.2 Proposed System

We use machine learning algorithms to predict the food with a high beneficiary diet. The model can predict the rating of the food more

accurately by giving the inputs which are the cereals and ingredients present in the food. The most notable of findings is that healthier cereals are generally disposed to receiving better ratings. It would be very useful for manufacturers to be able to receive higher ratings on their cereals while at the same time offering healthier options for consumers. The rating is predicted using the neural networks model.

.

3. THEORETICAL ANALYSIS

3.1 Block Diagram

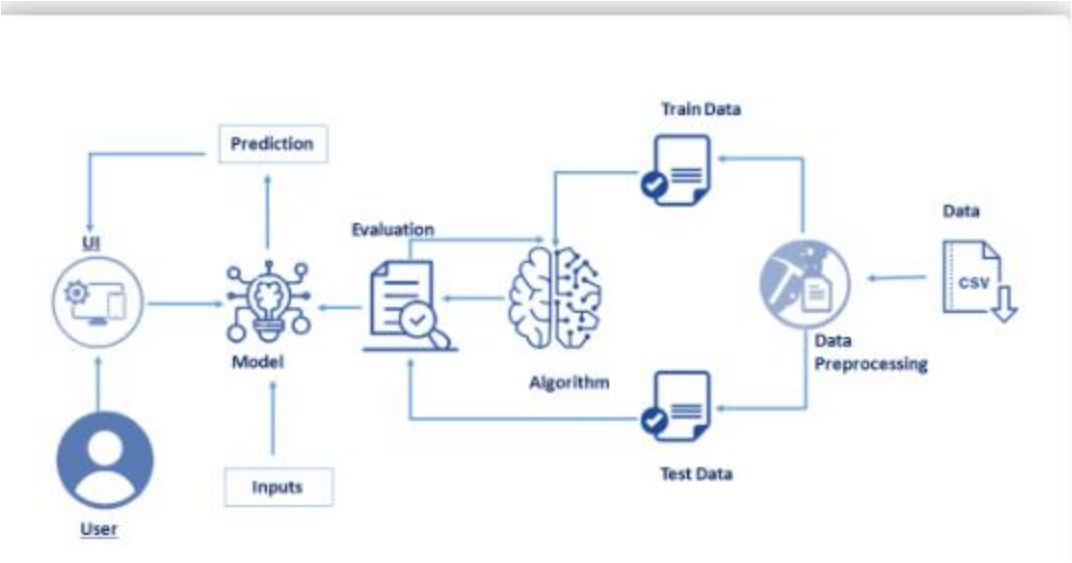

Fig 1: Block Diagram

3.2 Hardware/Software Designing

- **IBM Watson Studio:** IBM Watson studio provides tools for data scientists, application developers and experts to collaboratively and easily work with data to build and train models at scale. It gives you the flexibility to build models where your data resides and deploy anywhere in a hybrid environment.
- **IBM Watson Machine Learning:** Watson Machine Learning provides a full range of tools and services so that you can build, train, and deploy Machine Learning models.
- **IBM Cloud Object Storage:**

A service offered by IBM for storing and accessing unstructured data. It allows us to store limitless amounts of data, simply and cost effectively.
Anaconda Navigator: Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning-related applications. It can be installed on Windows, Linux, and MacOS, Conda is an open-source, cross-platform, package management system. Anaconda comes with great tools like JupyterLab, JupyterNotebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code. For this project, we will be using Jupyter notebook and Spyder.

4. EXPERIMENTAL INVESTIGATIONS

Our aim is to build a machine learning model that estimates the ratings of food. For this purpose, we collect the dataset for training purposes. From the dataset we have chosen the necessary attributes that contribute to our prediction. The dataset consist of 16 attributes and has no missing values. There are 15 input variables and 1 output variable. Here we are going to build a machine learning model that find the high dietary food that is predicted on the basis of rating of the food. After some preprocessing on the dataset, we are building our model.

In this analysis, we will disclose the main attributes that influence cereal ratings. We will predicting ratings using neural networks model.

5. FLOWCHART

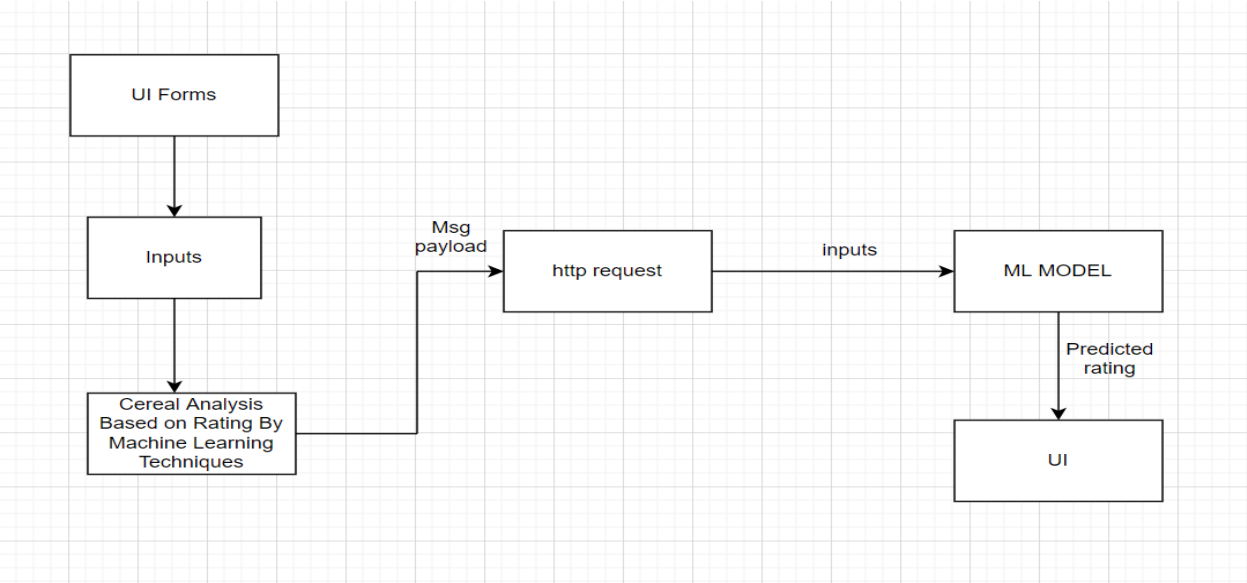

Fig 2: Flowchart

6. RESULT

Fig 3: Home Page

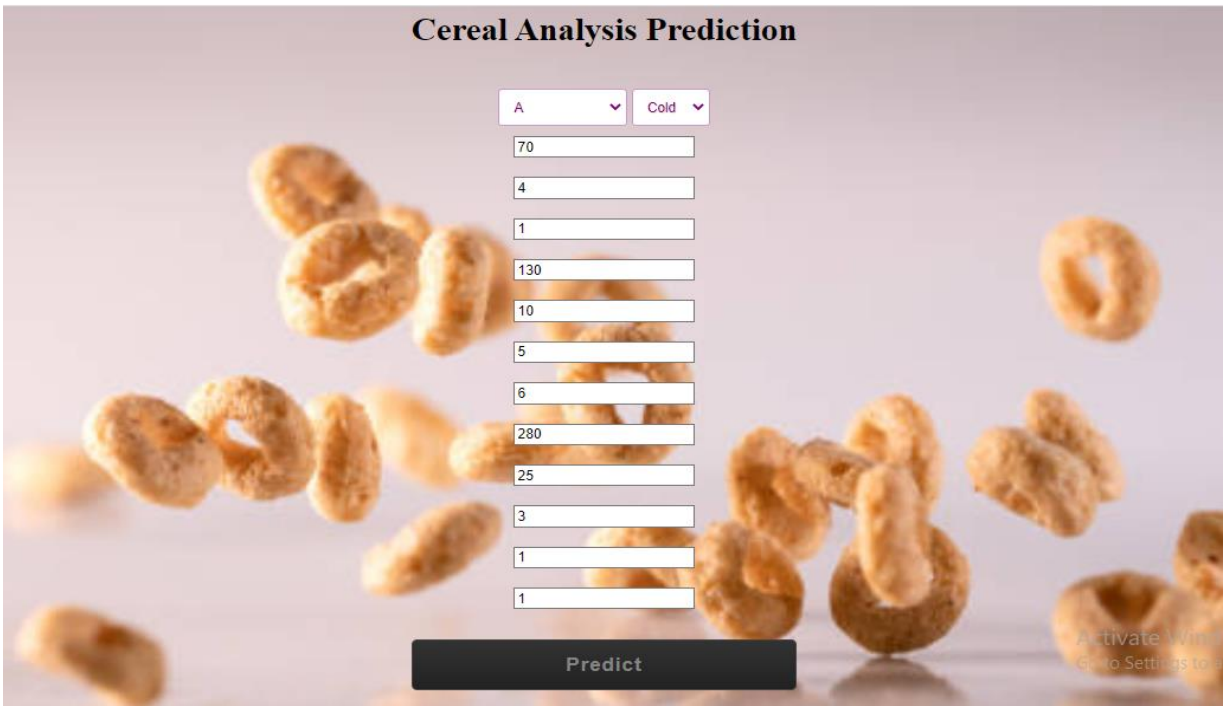

Fig 4: Predict Page

Fig 5: Result Page

7. ADVANTAGES & DISADVANTAGES

7.1 Advantages

- Help to choose high beneficiary cereal
- Critical role in human diets
- Encouraging progress toward better nutritional quality
- Toward a sustainable dietary revolution

7.2 Disadvantages

- Attributes and dataset collected should be correct. Otherwise it will affect the accuracy of our model.
- Require more dataset to increase the accuracy of model.

8. APPLICATIONS

An individual can use this application as a cereal rating calculation to choose high dietary food and improve their health. It can also be used in the medical industry to provide high beneficiary diet food to the patients for improve their health. It will reduce pressure on medical staff and save time, providing great

convenience in medical treatments. Helps to keep track on contents of food intake on a daily basis which helps in health maintenance.

9. CONCLUSION

The more calories that a cereal has, the less likely it is to receive a high rating. Manufacturers that want to bring in high ratings should create cereals that are high in fiber, protein, and potassium and avoid creating cereals with high calorie counts or lots of sugar or fat. Cereals with high ratings are more likely to be placed on the first or third shelf, because that is generally where the consumers' eyes gravitate.

10. FUTURE SCOPE

In future, we can include the aspects to accurately predict the cereal ratings more effectively. An improved form of this application developed and used in the health sector for better suggestions of food and thereby we can maintain health.

11. BIBLIOGRAPHY

https://www.kaggle.com/code/kianwee/analysis-on-cereal-prediction-on-ratings/notebook

12. APPENDIX

Source code

```
from flask import Flask, render_template, request

import requests

import json

    # NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.

API_KEY = "bJSrKnWH-dYvye3Ig4-MxTGXynisAZKTwdMhOaIXTXsk"

token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey": API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})

mltoken = token_response.json()["access_token"]


header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

app = Flask (__name__)
```

```
import  pickle
model  = pickle.load(open('cerealanalysis.pkl','rb'))

@app.route('/')
def helloworld():
    return  render_template('base.html')

@app.route('/assesment')
def prediction  ():
    return  render_template('index.html')

@app.route('/predict',  methods  = ['POST'])
def admin():
     a=request.form["mfr"]
     if  (a == 'a'):
        a1, a2, a3, a4, a5,  a6, a7=1,0,0,0,0,0,0
     if  (a == 'g'):
        a1, a2, a3, a4, a5,a6,a7 = 0,1,0,0,0,0,0
     if  (a == 'k'):
        a1, a2, a3, a4, a5,  a6, a7=0,0,1,0,0,0,0
     if  (a == 'n'):
        a1, a2, a3, a4, a5,  a6, a7=0,0,0,1,0,0,0
     if  (a == 'p'):
        a1, a2, a3, a4, a5,  a6, a7=0,0,0,0,1,0,0
     if  (a == 'q'):
        a1, a2, a3, a4, a5,  a6, a7=0,0,0,0,0,1,0
     if  (a == 'r'):
        a1, a2, a3, a4, a5,  a6, a7=0,0,0,0,0,0,1
```

```
    b= request.form["type"]
    if (b=='c'):
        b=0
    if (b== 'h'):
        b=1
    c= request.form["Calories"]
    d= request.form["Protien"]
    e= request.form[  "Fat"]
    f= request.form["Sodium"]
    g= request.form[  "Fiber"]
    h= request.form["Carbo"]
    i= request.form["Sugars"]
    j= request.form["Potass"]
    k= request.form[  "Vitamins"]
    l= request.form[  "Shelf"]
    m= request.form["weight"]
    n= request.form["Cups"]


    t=[[int (a1), int(a2), int(a3), int(a4), int(a5), int(a6), int (a7), int (b), int(c), int(d), int(e), int(f)
,int(g),  int(h),int(i),int(j),int(k),int(l),int(m),float(n)]]
    y = model.predict(t)
    return render_template("prediction.html",  z = y[0][0])


 # NOTE: manually  define  and pass the array(s)  of values  to be scored in the next  line
    payload_scoring  = {"input_data":  [{"field":
[["mfr","type","G1","G2","G3","G4","G5","G6","calories","protein","fat","sodium","fiber","carb
o","sugars","potass","vitamins","shelf","weight","cups"]],  "values":  t}]}


    response_scoring  = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/d18d52cb-55ec-40df-9e62-
```

```
b8de982c3585/predictions?version=2021-10-28', json=payload_scoring,
headers={'Authorization': 'Bearer ' + mltoken})

    print("Scoring response")

    predictions=response_scoring.json()

    print("Final Prediction:")

    #print(predictions['predictions'][0]['values'][0][0])

    print(predictions)

if __name__ == "__main__":

   app.run(debug=False)
```