Automatic Food Labels Reading System

Diogo Pires[1], Vítor Filipe[1,2], Lio Gonçalves[1,2], and António Sousa[1,2](✉)

[1] School of Sciences and Technology, University of Trás-os-Montes and Alto Douro (UTAD), 5000-811 Vila Real, Portugal
al64273@alunos.utad.pt, {vfilipe,lgoncalv,amrs}@utad.pt
[2] INESC Technology and Science (INESC-TEC), 4200-465 Porto, Portugal

Abstract. Growing obesity has been a worldwide issue for several years. This is the outcome of common nutritional disorders which results in obese individuals who are prone to many diseases. Managing diet while simultaneously dealing with the obligations of a working adult can be difficult. Today, people have a very fast-paced life and sometimes neglect food choices. In order to simplify the interpretation of the Nutri-score labeling this paper proposes a method capable of automatically reading food labels with this format. This method is intended to support users when choosing the products to buy based on the letter identification of the label. For this purpose, a dataset was created, and a prototype mobile application was developed using a deep learning network to recognize the Nutri-score information. Although the final solution is still in progress, the reading module, which includes the proposed method, achieved an encouraging and promising accuracy (above 90%). The upcoming developments of the model include information to the user about the nutritional value of the analyzed product combining it's Nutri-score label and composition.

Keywords: Nutri-Score · Digital Image Processing · Artificial intelligence · Deep Learning · Image Classification

1 Introduction

Obesity is a worldwide issue that accounts for 8% of global deaths annually [1]. People with obesity are at higher risk for many serious diseases and health conditions, including high blood pressure, diabetes, cardiac arrest, and an overall low quality of life. Dietary intake is one of the major causes of excess weight and fat accumulation. Sometimes the problem is not the quantity of food ingested but the quality: the nutritional balance. The Nutri-Score is an alternative label that consists in a food product rating system which was initially created and implemented in the food market in France and is now applied to many food labels in Europe. The system classifies foods by the nutritional balance. In general, it shows the 'macro' balance between nutrients/ingredients that must be privileged and nutrients/ingredients that must be avoided. The problem is that many people don't pay attention to the score or do not know how to interpret it. Usually, the consumer doesn't know why a certain food has a better classification than other, when apparently is not correct. Although the classification is correct.

A. Cunha et al. (Eds.): MobiHealth 2023, LNICST 578, pp. 432–444, 2024.
https://doi.org/10.1007/978-3-031-60665-6_32

Nowadays deep leaning is being applied in several fields including food industry. Various architectures of convolutional neural networks (CNNs) such as LeNet, VGGNet, GoogleNet, and AlexNet have also been used for "food recognition". The structure of CNNs was inspired by neurons in human and animal brains. The CNN simulated the complex sequence of cells that forms the visual cortex [2]. Their filters and other components are used to read the main image features and learn from them. This is image-based recognition which uses computer vision techniques to analyze images and identify their contents, detecting features such as color, texture, and shape. In general, CNN networks tend to present high performance for image analysis [3].

At the time of writing of this paper and, to authors knowledge, there are no related work in the literature proposing the automatic extraction and evaluation of products nutritional value from their packaging. The main purpose of this paper is to develop an image classification model that can be implemented on a mobile application for future daily use to support user's choices when buying products.

This paper is divided in 5 sections. In Sect. 2, the research work is presented which includes detailed information related to Nutri-score labeling system. Next, in Sect. 3 the materials and methodology used during this work are described focusing on the dataset building and the development of the proposed model. In Sect. 4, the model evaluation is performed including the presentation and discussion of the obtained results and the developed mobile app prototype. Finally, in Sect. 5 the conclusions are shown with some final remarks and future work.

2 Research Work

The Nutri-score is based on official rules defined by regulatory authorities. The labeling consists in a set of rules used to generate new labels with algorithms.

2.1 European Union Food Labeling Laws

Firstly, some of the topics contained in the official regulations of the competent authorities developed for labeling are reviewed. The use of different information sources allows one to have a different perspective of the topic to get the most correct idea about this theme.

The regulation of an appropriate system for food labeling is a method of health safety and safeguarding the interests of consumers. This regulation governs the labeling of all types of foods (solid, liquid, or processed) in all the countries that integrate the European Union. In addition, it may be supplemented by rules to be defined by each country for application to the national system. For that, it requires the examination by the European committee deliberated for this purpose.

These are some of the general rules which define the parameters that must be displayed on a food label [4]: the name of the food; the list of ingredients; any ingredient or processing aid listed or derived from a substance or product causing allergies or intolerances used in the manufacture or preparation of a food and still present in the finished product, even if in an altered form; the quantity of certain ingredients or categories of ingredients; the net quantity of the food; the date of minimum durability or the "use by" date; any special storage conditions and/or conditions of use; the name or business name and address of the food business operator; the country of origin or place of provenance; instructions for use where it would be difficult to make appropriate use of the food in the absence of such instructions; with respect to beverages containing more than 1.2 (%) by volume of alcohol, the actual alcoholic strength by volume; a nutrition declaration.

Nutritional Declaration

Nutritional declaration incorporates some information about the nutrient compositions in a product. The percentage of nutrients in 100 g or 100 ml of food must be presented. Sometimes the values can be shown per portion or the consumption unit.

These are the nutritional elements that must be integrated into the Nutritional declaration: energy value; the amounts of fat, saturates, and carbohydrates; the amounts of sugars, protein, and salt.

2.2 Nutri-Score

As seen earlier the amount of information presented to the consumer can be overwhelming. In turn, the diversity of information can have a negative effect in the consumer capability of choosing the most adequate product. This section exposes the fundamentals at the base of the Nutri-Score classification system.

As mentioned earlier, the Nutri-Score is based in a food components balance. Components that must be privileged by this system are: fiber, protein, vegetables, fruits, legumes, nuts, olive oils, rapeseed, and walnut oils. On the other hand, the components to avoid are: energy, saturated acids, sugar, and salt. In accordance with the balance, the aliment can be classified into 5 categories (good to bad, respectively): A (dark green); B (light green); C (yellow); D (light orange); E (dark orange/red) as seen in Fig. 1.

Fig. 1. The Nutri-Score label (Color figure online)

The Nutri-Score Main Algorithm

The classification of the food is based on the calculation of the final score. The score results from the sum of negative and positive points. The negative points (Table 1) are assigned according to the composition of the components to be avoided (per 100 g). The positive points are assigned by the quantity (per 100 g) of good components that exist on aliments. The algorithm regulates three different types of categories: The main algorithm general aliments; fats, oils, nuts and seeds category and beverages [5].

Table 1. Negative points attribution [6]

Points	Energy(KJ/100 g)	Sugars (g/100 g)	Saturates (g/100 g)	Salt (g/100 g)
0	≤335	≤3.4	≤1	≤0.2
1	>335	>3.4	>1	>0.2
2	>670	>6.8	>2	>0.4
3	>1005	>10	>3	>0.6
4	>1340	>14	>4	>0.8
5	>1675	>17	>5	>1
6	>2010	>20	>6	>1.2
7	>2345	>24	>7	>1.4
8	>2680	>27	>8	>1.6
9	>3015	>31	>9	>1.8
10	>3350	>34	>10	>2
11		>37		>2.2
12		>41		>2.4
13		>44		>2.6
14		>48		>2.8
15		>51		>3
16				>3.2
17				>3.4
18				>3.6
19				>3.8
20				>4

The main algorithm will be explained in detail. The other groups follow the same logic but different reference values and calculation algorithms [6]. For the general aliments, it is necessary to reference Table 1and Table 2.

The aliment's label must be analyzed, and the values of the different components must be retrieved. Then the tables must be consulted and added the corresponding points to each parameter. For example, if the protein quantity in the food was 10 g per 100 g, is added 4 points to a positive score according to Table 2. It is necessary to do the same for all positive parameters. For negative parameters, for example, if the amount of sugar is 2 g per 100 g, it isn't added negative points to a negative score. Is the same logic used for the good points. At the end of the process, exist a positive point score and a negative point score.

With the final score calculated, it is necessary to attribute the corresponding letter following Table 3.

Table 2. Positive points attribution [6]

Points	Proteins (g/100 g)	Fibers (g/100 g)	Fruit, vegetables and legumes (%)
0	≤2.4	≤3.0	≤40
1	>2.4	>3.0	>40
2	>4.8	>4.1	>60
3	>7.2	>5.2	
4	>9.6	>6.3	
5	>12	>7.4	>80
6	>14		
7	>17		

Table 3. Final Nutri-score thresholds points attribution [6]

Final NScore Points	Class/category	Colour
Min to 0	A	Dark Green
1 to 2	B	Light Green
3 to 10	C	Yellow
11 to 19	D	Light Orange
19 to max	E	Dark Orange or Red

Nutri-Score Other Algorithms

As referred previously, the main algorithm has some variances that must be adjusted to different groups like beverages and other specific aliments. The tables are different for the fats, oils, nuts, and seeds category [6] and beverages [5]. The conditions to calculate the final score are also different. For fats, oils, nuts, and seeds category: a) if previous negative points sum ≥7, so the final score is (negative points - (fruit, legumes, and vegetables)); b) if previous negative points sum <7, so the final score is (negative points - total positive points).

For beverages, the algorithm is equal to the main algorithm but is necessary to adjust the sugar parameter (add 1.5 g/100 g at each tabled level) and energy (add 30 kJ/100 g at each tabled level) [5]. The final score is calculated by subtraction of positive points to negative points. Note that the classification ranges are different between the different categories, depending on whether it is food in general, beverages, or another group mentioned.

3 Materials and Methodology

To run the code, a laptop with the following main specs was used: AMD Ryzen 5, 8 GB RAM, and Nvidia Geforce RTX 2 GB. The laptop programs employed and fundamental to the project are the latest Matlab version Matlab 2023a, Python, Flutter Dart, and Visual Studio Code.

Matlab 2023a was used to treat the dataset and to create the deep learning model. Flutter used for the mobile application development. For app development, an Android smartphone was used to receive and test the depurated app.

3.1 Dataset

Image Capture

An iPhone 13 camera with 12 megapixels and an f/1.2 lens opening was used to capture the dataset images. These were captured in 3024 × 3024 × 3 format. Processing technologies already incorporated in the smartphone were used. The main technologies are active rapid capture prioritization that intelligently adapts image quality to shutter press speed. Furthermore, the smartphone has an active lens correction that corrects lens distortion on the front camera and an ultra-wide-angle camera. The image capture quality can be optimized by considering lighting conditions, capture angle, and other relevant factors like the camera specs, for example. It's important to focus the camera on the scale of letters and capture the minimum objects/details in the background.

Fig. 2. Examples of captured images

Random products chosen in a hypermarket were captured resulting in 500 Nutri-score images (100 images for each category A, B, C, D, E). Figure 2 shows some examples of the captured images.

Image Pre-processing and Data Augmentation

In order to increase the number of images in the dataset, data augmentation was performed. To perform this procedure, the "imageDataAugmenter" function from Matlab [7] was used. The code allows the generation of new dataset images from the original images. The new images are generated by applying rotation and scaling transformations to the original images and applied randomly. This procedure resulted in a dataset with a total number of 5000 images (1000 images for each category). Due to GPU's limited performance images resizing to 512 $\times$ 512 $\times$ 3 was applied.

3.2 Model Development

Dataset Division

The dataset was divided into train/validation set (4000 images): 560 images for each category (70% of the total) to train and 240 images for each category (30% of the total) to validate the model. The test set was constituted by the remaining 1000 images.

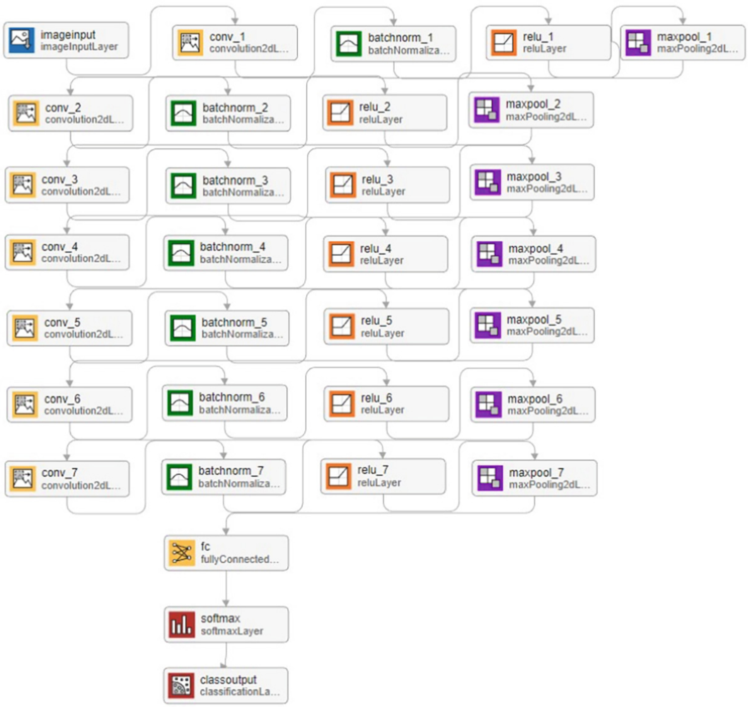

Fig. 3. Network architecture

Neural Network Architecture

The proposed neural network architecture (Fig. 3) was obtained thru experimentation and finetuning based on a simple image classification network using deep network designer available in Matlab. The model is composed by the input layer which receives the image and then, has six convolution layers with 3×3 filters. The number of filters in each one of the convolutional layers are 8, 16, 32, 64, 128, 512 respectively. After applying a convolution layer, the model always has a batch normalization layer, a Relu layer, and a max pooling layer (pool size 2, and stride 2). The last layer is composed by a fully connected layer, a softmax layer and a classification layer that classifies the images in one of the 5 classes.

Training Options

Relatively to the parametrizations of the training process, the hyperparameter values were obtained heuristically based on experimentation. It was established that the initial learning rate was 0.01 and the train epochs were 15. The minibatch size was 16 which means 16 images were shown to the model each time. Additionally, an Adam optimizer was applied since it obtained better results than similar optimizers (sgdm for example).

4 Results and Discussion

The model performance in the dataset was evaluated using several metrics. To integrate the proposed Nutri-Score classification method, for assisting the consumer, a mobile application was developed.

4.1 Model Evaluation

Performance Metrics

To analyze the performance the accuracy, confusion matrix, recall, precision, and F1-score were used, applying the following equations for each of the categories.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1\text{-}Score = 2 * \left(\frac{Precision * Recall}{Precision + Recall} \right) \tag{3}$$

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{4}$$

P represent true positive cases; FN represent false negative cases; FP represent false positive cases and TN represent false negative cases.

Evaluation Results

Figure 4 shows the graphic of accuracy during the train and loss of accuracy to validation data. In the graph, it is possible to see the existence of a curve of high growth in an initial phase. After a specific moment, the graphic growth stabilizes, originating a horizontal approximately straight line. The graph showed few oscillations, which indicates that the training process was stable. Accuracy was 97.25% to the train/validation dataset, and 95.00% to the test dataset. When the model was applied to the test dataset, a drop of two percentage points was denoted.

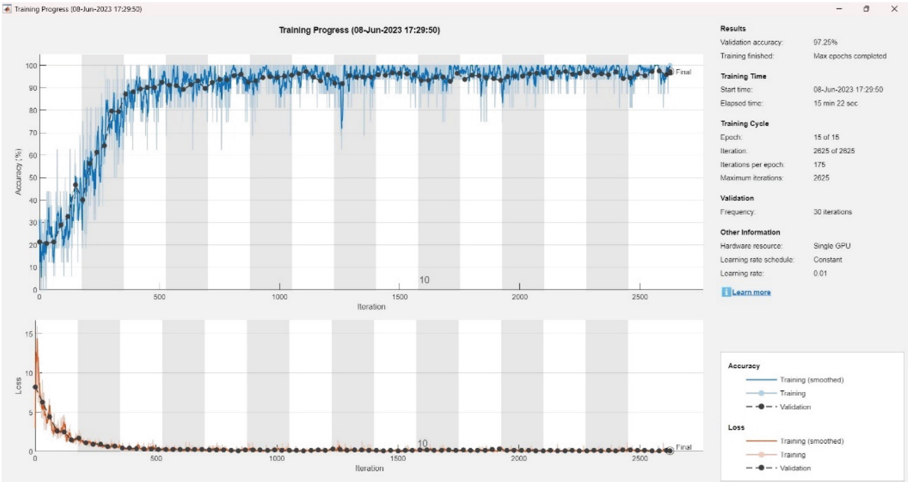

Fig. 4. Model Performance

Evaluation Metrics

Performance metrics were analyzed for two different situations: when the model was applied to training and validation data and when it was applied to test data (never seen before by the neural network). Confusion matrices (Fig. 5) show a comparison between true results and model results. They are commonly used to identify true positives (TP), false negatives (FN), false positives (FP), and true negatives (TN). For model performance purposes, it is more relevant to observe the performance metrics and confusion matrix regarding the test data. This is because the test dataset are images never seen by the model before and, therefore, it shows the real performance of the model comparable to a real-world scenario using images captured and processed using a mobile app.

The model's performance was evaluated using the metrics precision, F1-score, and recall which were calculated for each of the categories. On the training set (Table 4), all the average values of the performance metrics are all greater than 97%, which translates into a good performance of model for training the letter classification.

When the model was evaluated on the test dataset (Table 5), a drop of two percentage points was registered for each of the average values of the metrics, approximately. The drop is natural because the test dataset is compounded by images that were not seen before to the model. So, in this case, is normal to observe a small performance reduction.

These metrics have been calculated in accordance with the confusion matrices as shown in Fig. 5, applying the metric formulas.

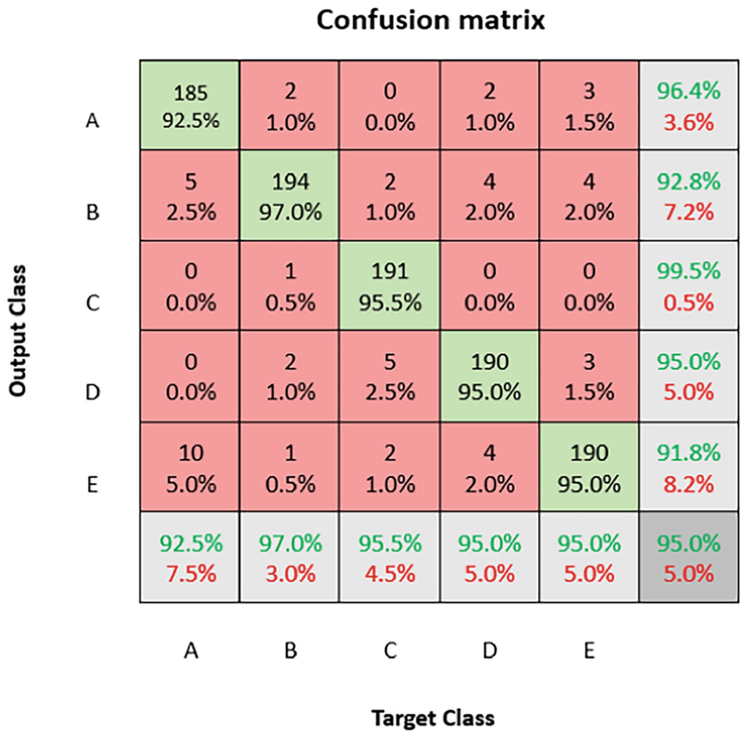

Fig. 5. Confusion Matrix calculated for testset

Table 4. Performance metrics to train and validation dataset

Class (Category)	Precision	Recall	F1-Score	Accuracy
A	97.84%	94.58%	96.19%	94.60%
B	96.71%	97.92%	97.31%	97.90%
C	100.00%	99.58%	99.79%	99.60%
D	97.85%	95.00%	96.41%	95.00%
E	94.07%	99.17%	96.55%	99.20%
Mean values	97.29%	97.25%	97.25%	97.25%

Analyzing each of the performance metrics it was found that the accuracy, recall, and F1-score values are around 95%. Individually, it is worth notice that there were some values below average. It is the case of the precision calculated for category E or the F1-score or accuracy of the same category. These lower values observed for category E may have resulted from the lack of sharpness of the calculated images and the presence of some noise in the images. Everything indicates that the problem could be with the images because this was the group with the lowest values. The same was verified for

Table 5. Performance metrics to test dataset

Class (Category)	Precision	Recall	F1-Score	Accuracy
A	96.35%	92.50%	94.39%	92.50%
B	92.82%	97.00%	94.87%	97.00%
C	99.48%	95.50%	97.45%	95.50%
D	95.00%	95.00%	95.00%	95.00%
E	91.79%	95.00%	93.37%	95.00%
Mean values	95.09%	95.00%	95.02%	95.00%

A group recall when the model is applied to the test dataset. Also, the fact that the test images were not set apart before applying data augmentation may have decrease the overall system's accuracy. This issue will be further investigated in the future.

4.2 Mobile App

A prototype mobile application was developed to integrate the proposed Nutri-Score classification method for assisting the consumer in real time. The application was implemented using the Flutter Dart in a Visual Studio Code IDE. To incorporate the trained model it was necessary to convert it to a ".tflite" file format. First, the original format ".mat" was converted to a Tensorflow model using the "Deep Learning Toolbox Converter for TensorFlow Models" [8], after installing the add-on, a code was created to convert the format model. Finally, in the Spyder-Python environment, a new code was developed to convert the TensorFlow model into a ".tflite" file ready to be incorporated into an Android app.

The application has a simple interface. The user must touch a button to open the camera and, then it is possible to capture the images. Furthermore, it was introduced a space delimiter to help the user to capture just the relevant objects (Nutri-score label). The captured image with the camera has $512 \times 512 \times 3$ default dimensions. The Android app interface can be shown in Fig. 6. This is a prototype of an Android application to implement the trained model. Although not fully functional the mobile app allows the user to capture images at a predefined size. Despite the attempts it was not possible, at this stage of development, to incorporate the model fully functional due to problems with the "dart.io" packages.

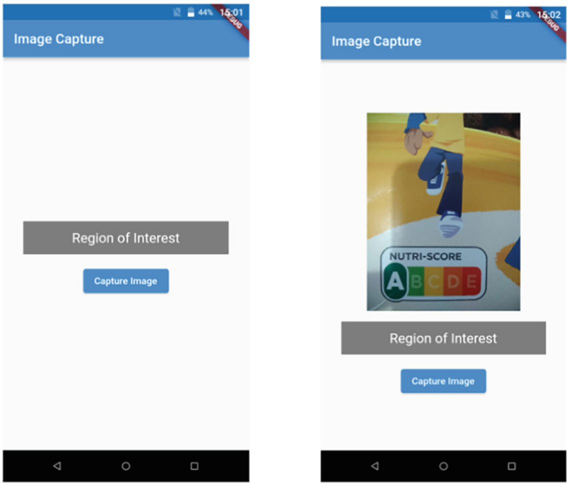

Fig. 6. Mobile app interface

5 Conclusions

In this work a method capable of automatically reading food Nutri-Score labels to simplify its interpretation was proposed. This method was intended to support users when choosing the products to buy based on the letter identification of the label. To accomplish this, a dataset was created, and a prototype mobile application was developed incorporating a deep learning model. The proposed method achieved good accuracy (above 90%). The obtained results were encouraging and are a great incentive to improve the model and the functionalities of the mobile application.

As a future work, it is possible to try to improve the model performance even further by adjusting parameters in the network architecture or increasing the training dataset. Also, the test images will be set apart before applying data augmentation to evaluate the overall system's accuracy improvement. It is also intended to improve the mobile prototype application to present to the user the general scientific information. The upcoming developments include information about the nutritional value of the analyzed product combining it's Nutri-score label and composition to advise the user to consume or not the analyzed product. That will help inform the consumer about the macro nutritional composition of the food, within reach of a photograph, with real impact on consumers health.

References

1. Okunogbe, A., Nugent, R., Spencer, G., Powis, J., Ralston, J., Wilding, J.: Economic impacts of overweight and obesity: current and future estimates for 161 countries. BMJ Glob. Health **7**(9), e009773 (2022)
2. Alzubaidi, L., et al.: Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. J. Big Data **8**, 1–74 (2021)

3. Nadeem, M., Shen, H., Choy, L., Barakat, J.M.H.: Smart diet diary: real-time mobile application for food recognition. Appl. Syst. Innov. **6**(2) (2023). https://www.mdpi.com/2571-5577/6/2/53
4. E. Parliament and T. C. of 25 October 2011. Regulation (eu) no 1169/2011 of the European parliament and of the council of 25 october 2011. Off. J. Eur. Union (304), 18–63 (2011)
5. T. N.-S. S. Comite. V2-update of the nutri-score algorithm for beverages (2022). https://www.santepubliquefrance.fr/determinants-de-sante/nutrition-et-activite-physique/documents/rapport-synthese/update-of-the-nutri-score-algorithm-for-beverages.-second-update-report-from-the-scientific-committee-of-the-nutri-score-v2-2023
6. T. S. C. of the Nutri-Score. Update of the main nutri-score algorithm (2023). https://www.asan.gob.es/AECOSAN/docs/documentos/Nutri_Score/2022_main_algorithm_report_update_FINAL.pdf
7. MATWORKS. Imagedataaugmenter (2023). https://www.mathworks.com/help/deeplearning/ref/imagedataaugmenter.html
8. Helper, M.: Deep learning toolbox converter for tensorflow models (2023). https://www.mathworks.com/matlabcentral/fileexchange/64649-deep-learning-toolbox-converter-for-tensorflow-models