An End-to-End Deep Learning System for Recommending Healthy Recipes Based on Food Images

Ledion Lico[1], Indrit Enesi[2], Sai Jawahar Reddy Meka[3]
Goergen Institute for Data Science, University of Rochester, New York, USA[1, 3]
Electronic and Telecommunication Department, Polytechnic University of Tirana, Tirana, Albania[2]

***Abstract*—Healthy food leads to healthy living and it is a major issue in our days. Nutri-Score is a nutrition label that can be calculated from the nutritional values of a food and helps evaluating the healthiness of it. Nevertheless, we don't always have the nutritional values of the food, so it is not always easy identifying this label. In the same way, it is not easy finding the healthier option to a favorite food. In this paper an end-to end deep learning system is proposed to identify the Nutri-Score label and recommend similar but healthier recipes based on food images. A new dataset of images is extracted from the Recipe 1M and labeled with the Nutri-Score value calculated for each image. Pretrained models Resnet50, Resnet101, EfficientNetB2 and DensNet121 are tuned based on this dataset. The embeddings from the last convolutional layer of the input image are used to find its most similar neighbor based on KNN algorithm. The proposed system suggests recipes with the lowest Nutri-Score similar to the inputted image. Implementations show that the Resnet50 provides highest prediction accuracy.**

Keywords*—Deep learning; nutri-score; new dataset; healthy food; accuracy

I. Introduction

Nutritional food plays an important role in human health. There is an expression from nutritionists "We are what we eat". Nutri-Score is a nutrition label that converts the nutritional value of products into a simple code of five colored letters (A-E), where A is the healthier food [1]. Each product is awarded a score based on a scientific algorithm. It takes into account the negative nutrients with a high energy value, large amount of sugar, saturated fats, salt and the positive ones with fibers, proteins, fruits, vegetables, nuts, rapeseed oil, walnut oil and olive oil. A set of 50718 food images is extracted from the Recipe 1M dataset [2]. For every image of this dataset the nutri-score value is calculated using Nutri-Score system and the dataset is labeled with labels A, B, C, D, E where A is the healthier food with lowest nutritional value [3].

A new end-to-end deep learning neural network (DNN) architecture is proposed as a combination of existing deep learning systems with k-nearest neighbors (KNN) algorithm. Its input is a food image and the output of the system is a set recipes and their respective images classified with labels suggesting the user the closest foods regarding the input one. The DNNs considered are Resnet50 -Residual Network with 50 layers (ResNet50) [4], Resnet101-Residual Network with 101layers (ResNet101) [5], EfficientNetB2 [6] and Densely Connected Convolutional Networks with 121 layers (DenseNet121) [7]. These systems are pre-trained using ImageNet dataset. The systems are then tuned using the new labeled dataset with 50718 labeled images. The embeddings of the last convolutional layer are inputted to the KNN classifier to train it. Whenever a user inputs a food image to the proposed system, the embeddings for this new image are obtained to find its nearest neighbors. The k-neighbors with the lowest Nutri-score and the corresponding recipes will be returned to the user. The accuracy for the four DNNs is calculated yielding that the combination of ResNet50 with KNN algorithm is the most accurate one regarding the new food dataset. Based on our internet searches this is the only algorithm that proposes to the user a healthy recipe only from food images.

A novel approach for recommending healthy recipes based on food images using a combination of deep learning and k-nearest neighbors' algorithm is proposed. The proposed method can help people make healthier food choices by providing personalized recommendations that take into account the nutritional value of different food products. Moreover, the experiments show that the combination of ResNet50 with KNN algorithm is the most accurate approach for this task. The paper introduces a new and innovative approach for recommending healthy recipes. Based on the experimental results is shown that the proposed approach is effective and accurate.

The paper is organized as: Related works are treated in Section II. Section III describes the Nutri-Score evaluation system; Deep Neural Networks are described in Section IV; the new dataset is described in Section V; methodology is described in Section VI; experimental results in Section VII; conclusions and future work closes the paper with Section IX.

II. Related Work

Several datasets and approaches have been developed in the recent years to address the topic. The [8] introduced a novel method to automatically recognize dishes pictures using Random Forests which allowed them to mine for parts simultaneously for all classes and to share knowledge among them. They also introduced the Food-101 visual classification dataset of 101 food categories, with 101k images. Im2Calories was introduced in [9] in which a classifier was trained on food images taken from 23 different restaurant and used to predict which food are present on the plate and the compute the corresponding calories. The system first performs a food segmentation followed by a volume estimation. This is obtained by combining two Convolutional Neural Networks (CNNs) for the meal detection and food recognition, with a food image segmentation technique and Google's Places API used to recognize the restaurant. A bigger dataset was introduced in [10]. The dataset contained 720 639 training, 155 036 validation and 154 045 test recipes, containing a title, a list of ingredients, a list of cooking instructions and (optionally) an image. The [11] in collaboration with Facebook research introduced a recipe generation system that takes a food image as an input and outputs a sequence of cooking instructions, which are generated by means of an instruction decoder that takes as input two embeddings. The first one represents visual features extracted from an image, while the second one encodes the ingredients extracted from the image. A transformer-based instruction decoder is used to generate the cooking instructions. The [12] proposed RecipeNet, a system that returns to the user the recipe based on the inputted image. They use the embeddings obtained from the final convolutional layer of ResNet-50, ResNet-101, and DenseNet-121, then use K-NN to return the most similar images and recipes. Although, they don't take Nutri-Score into account when tuning the system and they cannot return the healthiest image. On the other hand they don't compare the models based on this classification task.

III. Nutri-Score

Nutri-Score is a system for categorizing foods based on their nutritional value [1][13]. It guides consumers towards healthier food choices preventing the wide range of nutrition-chronic diseases. It was selected from French government from March 2017 to be displayed on products. It relies on the computation of nutrient profiling system from the United Kingdom Food Standard Agency [14]. Its output is: five classification letters corresponding to five colored labels from A to E, where A is the healthiest food and E is the detrimental one. Calculation of the score is based on the quantity of seven nutrient components found on 100g of food, which are high content of fruits and vegetables, fibers, and proteins (based on a rule of 2019 are added and the healthy oils) for a preferable score, while high content of sugar, saturated fatty acids, sodium yields a detrimental score. Special rules are added for "cheese", "added fats" and beverages. The system doesn't take in consideration the degree of food processing, vitamins, antioxidants, fiber type of additives. The calculation consists of three steps. In the first step the nutritional score of the food is assessed.

Ingredients which affect negatively the Nutri-Score compounding the Total_N_Score value are: high energy density per 100 (g of l), high sugar content, high content of saturated fatty acids and high salt content. Ingredients which affect positively the Nutri-Score results compounding the Total_P_Score value are content of fruits, vegetables, nuts and legumes; fiber content; protein content; content rapeseed, walnut and olive oil.

$$Nutritional\_Score = Total\_N\_Score - Total\_P\_Score \tag{1}$$

In the second step, the Rayner's score is calculated for all the products in the same way, except cheese, vegetable, animal fats, oils, and drinks. In the third step, the two scores are used to classify the food in one of five levels of the Nutri-Score system.

Based on the Nutri-Score evaluation system algorithm, 0 – 10 points are assigned for energy value and ingredients which should be limited in diet; 0 - 5 points for beneficial ingredients. To determine the value of a product and its corresponding letter, the sum of points for beneficial ingredients subtracts the sum of points for nonhealthy ingredients, the final score varies from -15 to +40 and with corresponding letters from A to E, where the lower the score the better nutritional food.

Nutri-Score is easily computable by industrial and public stakeholders; it encourages the food industry to improve the nutritional quality of the food supply.

IV. Deep Neural Networks

Deep Learning is a subset of machine learning that automates part of the feature extraction step of the process, thus eliminating some of the manual human intervention and also enables the usage of big data sets [15]. It consists of processing nodes arranged in layers. The system processes data between input and output layers to solve or predict specific task. The neural network needs to learn all the time solving specific tasks in a more qualified way or providing better results. When new data are inputted, the system learns how to act according to new situation [16]. Deep neural networks use layers of nodes to create high level functions from datasets. Different types of neural networks are distinguished between them from working principles, action of schemas and application areas.

Convolutional Neural Networks (CNN) implements convolution in its structure. It reduces the memory using weight sharing and the number of network parameters avoiding the over-fitting problem. Shared weight as well as space or time down sampling implemented in CNN, provide a certain amount of translation, scaling, and distortion invariances [17][18].

Deep Neural Networks are widely used and with great impact in the computer vision field. They solve different problems like image and face recognition, object detection, image classification etc.

A. ResNet-50 and ResNet-101

The template is designed so that author affiliations are not repeated each time for multiple authors of the same affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization). This template was designed for two affiliations.

ResNet50 is a Residual Neural Network of type Convolutional Neural Networks (CNN) with 50 layers. It has very good results in image classification. Number of layers is very important in Deep Learning. Additional layers can improve solution of complex problems or having better results. But it also increases the risk of saturation of accuracy levels which may slowly degrade after some point. The performance of the system may be decreased in training as well as in testing [19].

To increase the accuracy, ResNet uses residual blocks. It introduces the concepts of “skip connections”, which add outputs from previous layers to the outputs of stacked layers, enabling the model to learn the identity function. Residual blocks make possible facilitation of in learning the identity function, minimizing so the percentage error [20].

The first ResNet architecture was ResNet-34, using shortcut connections to turn a plain network into a residual one. Comparing with VGG neural networks, ResNets have fewer filters and less complexity. Layers have the same number of features for the same output feature map size. To preserve time complexity per layer, number of filters is doubled when the feature map size is halved. The VGG plain network is improved with “skip connections” or “shortcuts”. Identity shortcuts are used directly if input and output dimension are the same. Otherwise, extra zeros are padded to increase the dimensions of shortcuts or the projection shortcut is used to match the dimensions [20][21]. ResNet-50 is based on the ResNet-34 model, but the building blocks are modified in a bottleneck design using a stack of three layers. Each block of two layers in ResNet-34 is converted in a three layers bottleneck block yielding ResNet-50 architecture. It has much higher accuracy than Resnet-34 [22]. ResNet-101 or ResNet-152 residual networks use more three layers blocks than ResNet-50.

B. EfficientNet-B2

Scaling up the ConvNets is one of visible ways to achieve better accuracy. For example, ResNets scale from Resnet-50 to ResNet-152 by using more layers. It is possible to scale the models by scaling one or more of dimensions like width, depth, and resolution. EfficientNet models carefully balance network width, depth, and resolution to get better performance. Using new type of scaling method called compound coefficient, EfficientNet models uniformly scale across all the dimensions of width, depth, and resolution [23]. This provides added benefit over ResNet models in terms of accuracy and size of model. These models start with a baseline network designed using neural architecture search and use this new scaling method to obtain the family of models called EfficientNets. Unlike traditional methods which scale network width, depth, and resolution arbitrarily, EfficientNet family of models is formed by scaling these factors uniformly with a set of fixed scaling coefficients. Out of several versions available, we choose to use EfficientNet-B2 keeping in mind the memory and space constraints. This model is comparable in performance but more efficient than ResNet-50.

C. DenseNet

It is shown in practice that residual connections especially short connections between layers help convolutional networks be deeper, more accurate and efficient to train. DenseNets introduces connection between each layer to every other layer of the network in a feed-forward fashion. ConvNets with L layers have L connections one between each layer and its subsequent layer whereas DenseNets have (L(L+1) /2) direct connections. All previous layers' feature-maps are used as inputs into each layer, and its own feature-maps are used as inputs into all subsequent layers. DenseNets alleviate the vanishing-gradient problem, improve feature propagation, and promote feature reuse. We use DenseNet121, the base model of DenseNet family keeping in mind space, computation and memory constraints [24].

D. K-Nearest Neighbors Classifier Algorithm

K Nearest Neighbors Algorithm is a supervised machine learning algorithm used to classify data. It a simple algorithm, easy to implement, its time execution increases significantly when data size increases. It finds the K smallest distances between the current data and the dataset, the output is the most frequent label in those K data. Usually, KNN algorithm is used for recommender systems, when the dataset size is not significantly high [25].

V. NEW DATASET

The images are extracted from the Recipe 1M dataset. There is a portion of the dataset that contains nutritional information. Also, it is populated with some data gotten from the web for the recipes without information. The Nutri-Score value is calculated for these recipes.

Each recipe is linked to several images. a new dataset of 50718 images with the respective nutri-score values is created.

In Fig. 1, the distribution of classes presented. It is seen that there is a little imbalance between classes, with class E being the minority class. The images are converted to 256x256 dimensions, normalized and augmented them. Next the dataset is split in a training and validation set with 90%-10% proportion in a stratified way to keep the ratios.

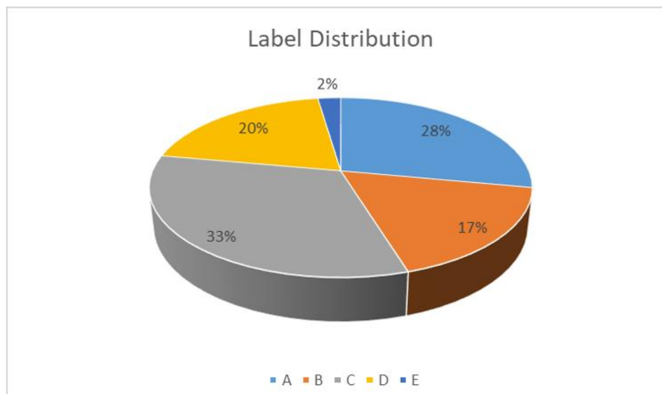

Fig. 1. Class distribution of the dataset.

VI. METHODOLOGY

Four deep learning neural networks, Resnet50, Resnet101, EfficientNetB2 and Densenet121, are considered for the healthy recipes system, which are pretrained using the ImageNet dataset.

The new dataset with 50718 labeled images from the Nutri-Score system is used to tune these pre-trained models. The deep learning pipeline is shown in Fig. 2.

The systems are evaluated and compared in terms of accuracy and loss. Cross-entropy loss and SGD optimizer are used for Resnet 50 and Resnet101 and Adam optimizer is used for EfficientNetB2 and DenseNet121. Servers equipped with GPU, are used for the training. Implementations are done using

Pytorch framework. Tensorboard was used for recording the models' loss and accuracies as well as for visualization.

Embeddings from the last convolutional layer are inputted to the KNN classifier to get the nearest neighbors with the lowest Nutri-Score value.

In our case, we need to return the nearest neighbors for the inputted image and it is returned based on the distance between the embeddings. KNN will evaluate the nearest neighbors based on the Minkowski distance, as it is a general form of Euclidean distance, as shown in equation 2:

$$d(x,y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{\frac{1}{p}} \quad (2)$$

KNN is a classifier that predicts the class of an element as the majority class among its K nearest neighbors.

The healthiest nearest neighbors for the input image are returned to the user together with the corresponding recipes.

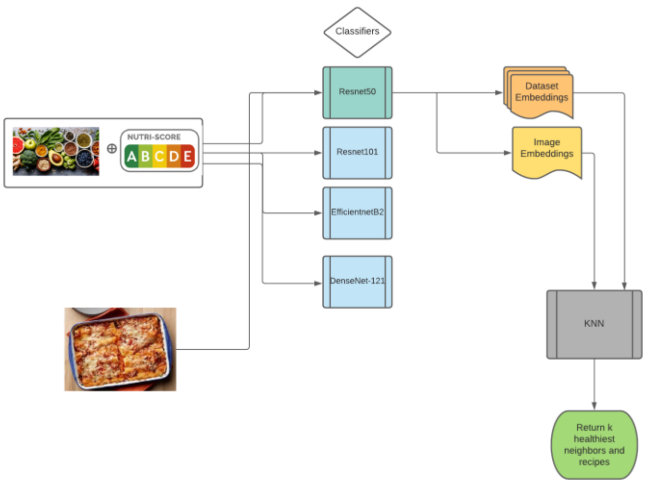

Fig. 2. Deep learning pipeline.

VII. Experimental Results

A. Results from Deep Learning Neural Network Models

The new dataset was created and labeled. The training and test dataset loaders were created and were ready for the modelling part.

At first the pre-trained models on ImageNet are downloaded and the convolutional

Layers are frozen. An initial train is performed getting the best accuracy around 70%. Then the convolutional layers are unfrozen and full training of the networks is performed.

The learning rate scheduler decays the learning rate by a factor of 0.1 every 7 epochs. The SGD optimizer is used for the Resnet50 and Resnet101 models and Adam optimizer is used for Densenet121 and EfficientNetB2. Models are trained for 100 epochs and the best model is selected. Fig. 3 shows the comparison of classification models.

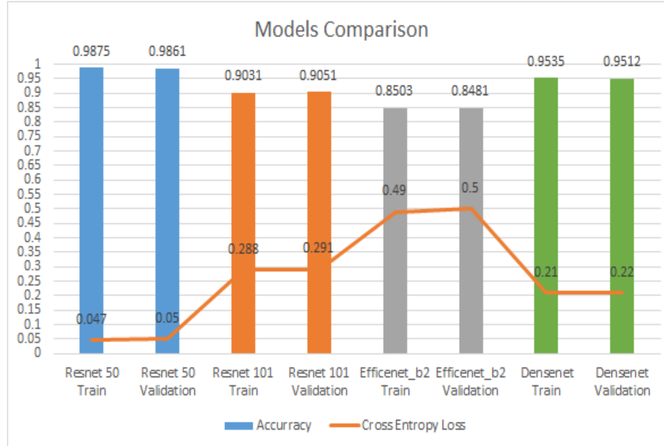

Fig. 3. Classification models comparison.

From the chart, it is shown that Resnet50 and Densenet121 are the best models giving respectively 98.61% and 95.12% of accuracy. Resnet 101 produced 90.51% of accuracy and performed poorer than Resnet50. This fact is explained that Resnet101 is a bigger model and our new dataset is relatively small. EfficientNetB2 is the worst performer with only 84.81%. Attempts are done for executions of EfficientNetB3 and EfficientNetB4 but we did not have the computational resources available to train these models so executions are limited ourselves to EfficientNetB2. To analyze the accuracy for every class, the confusion matrix for the two best models is printed, as shown in Fig. 4 and 5.

The results show that the model performs very well for minority classes also. Augmentation helped to deal with the imbalance. After defining best model, the next step is to get the embeddings and train a KNN model based on them.

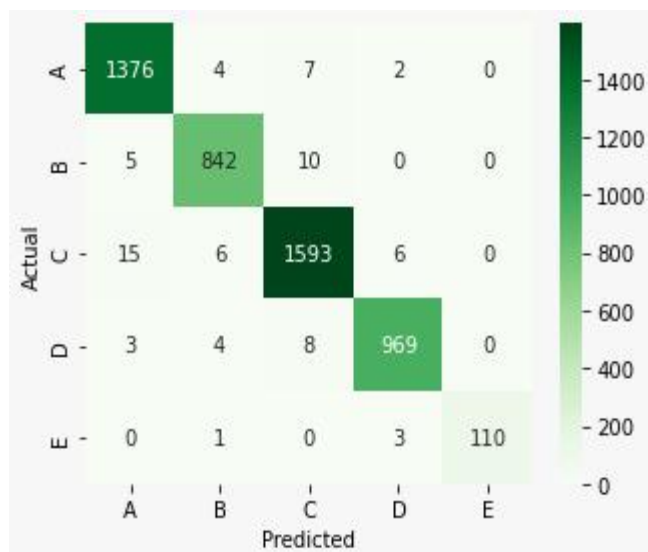

Fig. 4. Confusion matrix for resnet50.

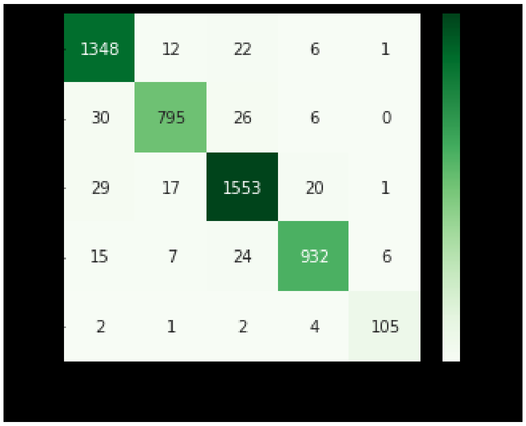

Fig. 5. Confusion matrix for densenet121.

B. *Results from KNN Model*

Since Resnet50 performed better than other models for the new labeled dataset, embeddings from the last convolutional layer of the Resnet 50 model are extracted.

The images based on the Euclidean distance between the embeddings in Tensorboard are plotted. The results are shown in Fig. 5.

As it can be seen from the plot images closer to the D label are also labelled with D, but some C-values in the surroundings exists. These images are the interested ones images they are similar but healthier. The number of neighbors to the KNN classifier is set to 24. The proposed system returns 24 similar food images according to the inputted one. This number is tunable and can be changed by the user.

Results of the experiment are shown in Fig. 6.

Fig. 6. Sample image input and its 24 outputs images recommended by the system.

As it is seen from the results, the images are similar to each other, and the class labels are very similar. Our main aim is to return the images and recipes with the lowest nutri-score value. The user might input the number of healthy recipes that he wants to get and the system will generate them. It is seen from the sample that the output is C-labelled has two B-labeled neighbors that are similar but healthier. These two images and the corresponding recipes will be returned to the user if he puts the recipe limit to two. According to the same logic, the first three neighbors will be returned for sample image and if the user inputs the recipe limit to three. As an example, we are generating the recipe for the second neighbor of the sample 2 image. The obtained recipe is shown in Fig. 7.

Recipe :Slow Cooker Fruit, Nuts, and Spice Oatmeal :

- Ingredient 1: {'quantity:2 cup': 'oats'}
- Ingredient 2: {'quantity:2 cup: 'cranberries, dried, sweetened'},
- Ingredient 4: {quantity 2 tablespoons: 'salt, table'},
- Ingredient 5: {'1/2 cup': 'nuts, almonds'},
- Ingredient 6: {' 1/2 cup': 'nuts, pecans'},
- Ingredient 7:{'3 cup ': 'water, bottled, generic'},
- Ingredient 8:{'1 cup: 'milk, fluid, 1% fat, without added vitamin a and vitamin d'},
- Ingredient 9:{'1 tablespoon': 'spices, cinnamon, ground'},
- Ingredient 10:{'1 teaspoon'pumpkin, raw'},
- Ingredient 11:{'2 teaspoon': 'butter, without salt'}],

Instructions: 'Combine the oats, apple, cranberries, almonds, pecans, water, milk, cinnamon, pumpkin pie spice, and butter in a slow cooker.Cook on Low overnight or 8 hours.'}

Fig. 7. Recipe for the second neighbor of sample 2 image , A-labeled.

It will be an A-class meal which is very healthy and it is similar to what is actually requested by sample 2 image. The proposed model is able to return the best and healthiest recipes for the input image. The user may decide the number of recipes that he wants to return.

User has also the possibility to set the maximum Nutri-score threshold that he can accept and the system will limit the result accordingly.

VIII. Discussion of Results

In this paper, a deep learning-based approach for food image retrieval and recipe recommendation is presented. Results show that ResNet50 and DenseNet121 achieved the highest accuracies among the models evaluated. This is consistent with previous studies that have reported the superior performance of these models in image classification tasks [26][27]. Results also highlight the importance of transfer learning, as pre-trained models on ImageNet were used as a starting point for our models, which significantly improved their performance.

One important aspect of the proposed approach is the use of data augmentation to deal with the imbalanced dataset. This allowed us to achieve high accuracies not only for the majority classes, but also for the minority classes. Data augmentation has been shown to be an effective technique to improve the performance of deep learning models, especially when dealing with limited datasets [28].

The use of KNN for food image retrieval and recipe recommendation is not a novel idea. However, the approach differs from previous studies in that we use the embeddings extracted from the last convolutional layer of the ResNet50 model as the feature vectors for KNN. This approach has been shown to be effective in several image retrieval tasks [29][30][31] and the obtained results confirm its effectiveness for food image retrieval as well.

One limitation of our study is the relatively small size of the dataset, which may affect the generalizability of our results. Moreover, our dataset contains only Western-style dishes, which may limit the applicability of our approach to other cuisines. Future studies should focus on collecting larger and more diverse datasets to improve the performance and generalizability of our approach.

The proposed approach provides a promising solution for food image retrieval and recipe recommendation. The high accuracies achieved by our models, especially for the minority classes, demonstrate the effectiveness of the approach. The use of KNN with embeddings extracted from deep learning models provides a fast and efficient way to retrieve similar food images and recommend healthy recipes. It has the potential to be useful for various applications, such as food logging, dietary analysis, and personalized meal planning.

IX. Conclusions and Future Work

In this paper, an end-to-end deep learning neural network model is developed that recommends healthy meals to the users based on the imputed food images. A new dataset with 50718 records is created by extracting food images from Recipe 1M dataset and labeling them with the Nutri-Score value.

This dataset can be expanded even more in the future and can be used for healthy food classification tasks. Deep learning models ResNet50, ResNet101, EfficientNetB2 and DensNet121 are compared on the new dataset. Experimental results show that Resnet50 gives the maximum accuracy of 98.61 followed by Densnet121 with 95.12%. Embeddings from the last convolutional layer are inputted to the KNN model providing the user the healthiest recipes of food similar to the input image but healthier in nutritional values.

The results demonstrate the feasibility of using deep learning neural networks for recommending healthy meals based on food images. The new dataset created in this paper provides a valuable resource for future research in healthy food classification. The comparison of various deep learning models on this dataset highlights the importance of selecting an appropriate model architecture for the task at hand.

However, there are still some limitations that need to be addressed in future work.

Firstly, the dataset used in this study is relatively small and limited to Nutri-Score values. Expanding the dataset with additional labels and nutritional information could further improve the performance of the model. Secondly, the proposed model is limited to recommending healthy meals based solely on food images, which may not be sufficient for some users. Future work could consider incorporating additional user information, such as dietary preferences and restrictions, to further personalize the recommendations.

References

[1] Hercberg, S., Touvier, M., Salas-Salvado, J., & on behalf of the Group of European scientists supporting the. (2021). The Nutri-Score nutrition label. International Journal For Vitamin And Nutrition Research. doi: 10.1024/0300-9831/a000722

[2] Marin, J., Biswas, A., Ofli, F., Hynes, N., Salvador, A., & Aytar, Y. et al. (2021). Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images. IEEE Transactions On Pattern Analysis And Machine Intelligence, 43(1), 187-203. doi: 10.1109/tpami.2019.2927476

[3] Moolya, D. et al. (2022) “Recipe generator using Deep Learning,” International Journal for Research in Applied Science and Engineering Technology, 10(5), pp. 846–851. Available at: https://doi.org/10.22214/ijraset.2022.42321.

[4] P. Yang, C. Dong, X. Zhao and X. Chen, "The Surface Damage Identifications of Wind Turbine Blades Based on ResNet50 Algorithm," 2020 39th Chinese Control Conference (CCC), Shenyang, China, 2020, pp. 6340-6344, doi: 10.23919/CCC50068.2020.9189408.

[5] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and

[6] S. Ethiraj and B. K. Bolla, "Classification of Astronomical Bodies by Efficient Layer Fine-Tuning of Deep Neural Networks," 2021 5th Conference on Information and Communication Technology (CICT), Kurnool, India, 2021, pp. 1-6, doi: 10.1109/CICT53865.2020.9672430.

[7] Ezzat, D., Hassanien, A.E. and Ella, H.A. (2021) “An optimized deep learning architecture for the diagnosis of COVID-19 disease based on gravitational search optimization,” Applied Soft Computing, 98, p. 106742. Available at: https://doi.org/10.1016/j.asoc.2020.106742.

[8] L. Bossard, M. Guillaumin, and L. Van Gool, “Food-101– mining discriminative components with random forests,” in European conference on computer vision. Springer, 2014, pp. 446–461. 1

[9] A. Meyers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K. P. Murphy, “Im2calories: towards an automated mobile vision food diary,” in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1233–1241. 1

[10] A. Salvador, M. Drozdzal, X. Giro-i Nieto, and A. Romero, “Inverse cooking: Recipe generation from food images,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 10 453–10 462.

[11] D. Raboy-McGowan and S. Lu, “Recipenet: Image to recipe/nutritional information generator,” 2020.

[12] Raboy-McGowan, D., & Lu, S. (2020). RecipeNet: Image to Recipe/Nutritional Information Generator.

[13] Vermote, M. et al. (2020) “Nutritional content, labelling and marketing of breakfast cereals on the Belgian market and their reformulation in anticipation of the implementation of the nutri-score front-of-pack labelling system,” Nutrients, 12(4), p. 884. Available at: https://doi.org/10.3390/nu12040884.

[14] Jürkenbeck K, Mehlhose C, Zühlsdorf A (2022) The influence of the Nutri-Score on the perceived healthiness of foods labelled with a nutrition claim of sugar. PLoS ONE 17(8): e0272220. https://doi.org/10.1371/journal.pone.0272220

[15] PirahanSiah, F. (2019) “Computer Vision, Deep Learning, Deep Reinforcement Learning.” Available at: https://doi.org/10.14293/s2199-1006.1.sor-uncat.clzwyuz.v1.

[16] Takagi, T., & Mizumoto, I. (2018). Parameter Optimization with Input/Output Data via DE for Adaptive Control System with Neural

Network. Journal Of Robotics, Networking And Artificial Life, 5(1), 19. doi: 10.2991/jrnal.2018.5.1.5

[17] Alzubaidi, L., Zhang, J., Humaidi, A., Al-Dujaili, A., Duan, Y., & Al-Shamma, O. et al. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal Of Big Data, 8(1). doi: 10.1186/s40537-021-00444-8

[18] Sirmorya, A., Chaudhari, M., & Balasinor, S. (2022). Review of Deep Learning: Architectures, Applications and Challenges. International Journal Of Computer Applications, 184(18), 1-13. doi: 10.5120/ijca2022922164

[19] Alaeddine, H., & Jihene, M. (2021). Deep Residual Network in Network. Computational Intelligence And Neuroscience, 2021, 1-9. doi: 10.1155/2021/6659083

[20] Patil, O.S. et al. (2022) “Deep residual neural network (resnet)-based Adaptive Control: A Lyapunov-based approach,” 2022 IEEE 61st Conference on Decision and Control (CDC) [Preprint]. Available at: https://doi.org/10.1109/cdc51059.2022.9992881.

[21] Image Recognition using Deep Learning Techniques. (2021). Journal Of Xidian University, 15(6). doi: 10.37896/jxu15.6/005

[22] Zhang, L., & Schaeffer, H. (2019). Forward Stability of ResNet and Its Variants. Journal Of Mathematical Imaging And Vision, 62(3), 328-351. doi: 10.1007/s10851-019-00922-y

[23] Tan, M., & Le, Q.V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. ArXiv, abs/1905.11946.

[24] Li, G., Zhang, M., Li, J., Lv, F., & Tong, G. (2021). Efficient densely connected convolutional neural networks. Pattern Recognition, 109, 107610. doi: 10.1016/j.patcog.2020.107610

[25] Zhang, X., Wang, S., Wu, Z., & Tan, X. (2022). Unsupervised image clustering algorithm based on contrastive learning and K-nearest neighbors. International Journal Of Machine Learning And Cybernetics. doi: 10.1007/s13042-022-01533-7

[26] A. Salvador, N. Hynes, Y. Aytar, J. Marin, F. Ofli, I. Weber, and A. Torralba, “Learning cross-modal embeddings for cooking recipes and food images,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp.3020–3028. 1

[27] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[28] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).

[29] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 60.

[30] Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2015). Neural codes for image retrieval. In Proceedings of the European conference on computer vision (pp. 584-599).

[31] Radenović, F., Tolias, G., & Chum, O. (2016). CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples. In Proceedings of the European conference on computer vision (pp. 3-20).