Conversational System About Traditional Portuguese Food with Image Recognition

FoodAids

Diogo Risca
1220186@isep.ipp.pt

Diogo Teixeira
1180893@isep.ipp.pt

Gonçalo Lopes
1220207@isep.ipp.pt

Guilherme Mendes
1190641@isep.ipp.pt

João Azevedo
1220259@isep.ipp.pt

Miguel Panda
1190896@isep.ipp.pt

Department of Computer Engineering
Instituto Superior de Engenharia do Porto (ISEP)

***Abstract*—A healthy diet is an important key to keep a good lifestyle. In Portugal, more than half of the population suffers from obesity, this indicates a more sedentary lifestyle and bad habits regarding eating. With the use of the application described in this study, it will be possible to identify the dish. With this information it will be possible to obtain a recipe, so it is possible to replicate that dish. This is import because it simplifies the process of managing a list of recipes for future dishes in order to get healthier eating habits. It will also be possible to get dishes to cook by giving the available ingredients, being a great asset to help in a daily basis and not worrying what kind of dish to cook while maintaining a great nutritional and healthy options. It will be a great asset for tourists when visiting Portugal, because they can photograph any dish that they like and get a recipe and information regarding that plate, being a great tool to know the traditional dish and being able to try to cook any time later with one of the given recipes.**

***Index Terms*—food, dataset, machine learning, CNN, prediction, NLP, ChatBot**

I. Introduction

A healthy diet is important to keep a good lifestyle [1].

In Portugal, more than half of the population suffers from obesity or are over-weight [2]. The causes for this are varied but it has been observed that people have a more sedentary lifestyle and have acquired bad eating habits over the past decades. The consumption of sugar and saturated fats have been favoured in detriment of more traditional diets. Obesity is considered by the World Health Organisation a direct impact in longevity and quality of life as well as increasing cardiovascular risk.

In parallel to this, the amount of cooking done at home is decreasing over the time [3]. Services like *UBER Eats*, *GLOVO* as well as traditional restaurants have all contributed to a decline of the home-cooked meals. All in all, this is not necessarily a "bad" thing, however, increasingly, the quality and healthiness of the meals has been compared to those cooked at home and has been found to be lacking [4]. There is an obvious convenience in eating out or having food delivered without the need to seek and acquire ingredients, storing them as well as the labour of making them into an edible meal. On the other hand, eating out or ordering food is still more expensive than cooking meals at home, on some of the latest information we found up to 3.4 times as expensive [5].

Everyone learns cooking in their own way, but everyone uses recipes at some point, and they are therefore essential when trying to cook.

In implementing the application described in this report, several things to achieve were aimed, among which are:

- Encourage users to cook for themselves, at home, using recipes they saved or already on the application.
- Allow users to quickly refer to a recipe they can cook with the specific ingredients they have in hand.
- Increase awareness as well as accessibility of Portuguese gastronomical tradition.

Another crucial purpose of this project is to provide an approach to food, or dish, classification. This is important as the identification of the ingredients of a dish allows for the tracking of the calories consumed on the daily basis. Being aware of what dishes that are eaten and what ingredients constitute them is crucial in keeping a healthy diet. Proper food classification will also allow for the provision of proper information for diabetics and allergic people on whether some ingredient of a dish might be dangerous to them, providing a good help in their diets.

With the use of a chatbot, it is possible to identify Portuguese dishes, it's ingredients and recipe in a casual and automatic way, another big asset is to make Portuguese food known to tourists, giving details regarding the dish and also its recipes, so when foreigners have some curiosity on Portuguese cuisine they can access this application.

II. State of the Art

Deep learning is a growing sub-area of machine learning and artificial intelligence. It uses algorithms to process data, with the objective of mimicking processes done by the human brain [6]. It is an example of biomimicry.
Image segmentation is a fundamental problem in computer vision, it can be described as the classifying of pixels with a semantic label or of individual objects.
Deep learning has already proved to be an excellent choice for data analysis, having a large number of successful applications in image processing, speech recognition and so on. This technology has also recently been introduced to food science namely in recognising and classifying good and bad food for this is an important task for human beings, helping their diets especially for people suffering from diabetes and allergies.
An image of food or dish can lead to knowing some of the most important information regarding their characteristics. Large variation in food shape, volume, texture, colour, composition and background creates many variations of the same dish making a great challenge for algorithms to do proper classification. At the moment, Convolutional Neural Networks (CNNs) are the most commonly used method for image analysis both in food recognition and classification [7]. A CNN is similar to a feed forward neural network being able to extract features from the input data with convolution structures as shown in figure 1 for better understanding). In a CNN, each neuron is no longer connected with all neurons of the previous layers, this is an asset in reducing parameters faster and allows for the sharing of weights and can down sample an image, reducing the amount of data while preserving the necessary information [8].

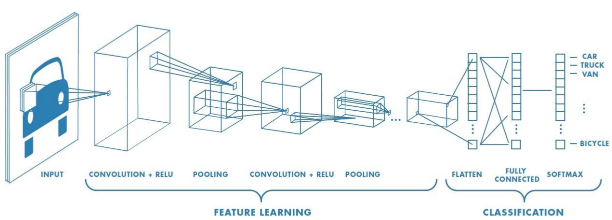

Figure 1. Convolutional Neural Network [9]

Natural Language Processing (NLP) systems began in the 1940s, after World War II by Joseph Weizenbaum, a known MIT researcher who made his first approach to NLP with the implementation of ELIZA, a computer program for the study of natural language communication between man and machine. ”ELIZA is a program that makes natural human language conversation with a Computer possible” [10], even if ELIZA was a very limited program, it is a good proof of concept that communication systems between man and computer are possible to implement by applying NLP concepts.
There are two main fields of NLP: syntax and semantics. The Syntax analysis is linked with tokenization, the first step of NLP [11]. During the tokenization phase the focus is identifying tokens, basic units that don´t need to be decomposed in a subsequent processing - The entity word is one kind of token for NLP, that consists on a group of characters - the most basic NLP token type . Although with this comes a downside, that is the complexity that will take for the computer to analyse and recognise some sort of expressions and tokens used in a defined order [12]. One answer to this concern is the second half of NLP, the semantics analysis [11]. Semantics analysis is about language significance study and the vocabulary used conveys the importance of the subject, because of the interrelationship between linguistic classes [13].

A. Applications in Food Science

Being able to do a proper binary classification comparing food to nonfood can be successfully resolved using deep learning. Singla, Yuan, and Ebrahimi (2016) [14], created a data based named Food-5K, it was selected 5000 images, 2500 being food images from three popular image sets for food recognition: Food-101, UECFood-100 and UECFood-256. A fine-tuned GoogLeNet model was implemented with an accuracy of 99.2%. McAllister, Zheng, Bond and Moorhead(2018) [15] reached the highest accuracy of 99.4% in validation and 98.8% for evaluation on Food-5K dataset, using Radial Basis Function (RBF) kernel-based SVM with ResNet-152. After being able to recognise images that contain food, the next challenge would be food classification, being a multiclassification problem instead of a binary one. There are open-acess datasets of food images with different categories such as Food-101(Bossard, Guillaumin, & Gool, 2014) [16], UECFood-256(Kawano & Yanai, 2015) [17], UECFood-100 (Matsuda, Hoashi & Yanai, 2012) [18]. Food-101 includes 1000 images of each class, being a prevalent dataset in food science. Bossard et al. (2014) [16] created the dataset and reached 50.76% of accuracy for classification. Tatsuma and Aono(2016) [19] by using covariances of features of trained CNN achieved a average accuracy of 58.65%. Yanai and Kawano(2015) [20] used the fine-tuned AlexNet and achieved an accuracy of 70.41%. Liu et al(2016) [21] presented a network named Dep Food and reached an accuracy of 77.40% and 93.70%. Fu, Chen, and Li(2017) [22] obtained a better result of 78.5% and 94.1% using a fine-tuned deep 50-layer ResNet.

On April 2020, three New York University researchers conducted an approach to generate recipes using a natural language model. Traditionally N-gram models are a good fit when producing similar documents to the training samples. Although, a Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM) layer is a good approach when the main objective is to capture information stored in long text sequences. In their study, the text generator was trained with the entire concatenated recipes information. Both N-gram language model and RNN with LSTM model were implemented to generate new recipes.
By the end, they concluded that for recipe generation with NLP Models the 3-gram models had five cooking steps and three ingredients on average. And recipes generated by LSTM

model had seven cooking steps and eight ingredients. Based on the users opinion they considered the cooking instructions of the recipes produced with the LSTM model more informative than the 3-gram model ones. They also concluded that the RNN model has a better performance if the dataset used has a large number of recipes [23].
Other research conducted in this area is dated from 2020. Oleksii Trekhleb experimented with the use of a model to generate cooking recipes. After experimenting with different technologies. The one that achieved the best results was the combination of tensorflow and a LSTM Recurrent Neural Network. Based on a dataset of multiple recipes he trained a model to generate recipes based on phrases or words input [24].

B. Available Common Technologies

1) Tensorflow: Is a software library originally created by Google allowing researchers to implement Machine Learning and Deep Learning techniques. It engages machine learning and artificial intelligence algorithms for picture captioning, pattern detection, feature classification, training, and other functions. Tensorflow library is also versatile allowing the users to use it in a range from image recognition to NLP [25] [26].

2) Gensim: Is an open source library also known as "generate similar" and is a python library for Topic Modelling, document indexing and similarity retrieval [27].

3) Word2Vec: Word embeddings are vector representations of words, that are used to train ML models. The most common way to represent words in vector format is by using one-hot encoding that maps each word to a one-hot vector. The main problem of using this technique is that the relations between words are not kept. This means that each word will be allocated in a huge sparse matrix representation, leading to an enormous consumption of memory and space.
Another approach is to use N-grams although this approach leads to similar problems since at the same time the "N" on N-grams increases, the size and allocated memory also increase exponentially. Apart from that, it can also lead to more data sparsity, which ,in the end, leads to the need of more data in order to train the model successfully.
In cases like the above mentioned, Word2Vec can be the solution since the words are represented in a continuous N-dimensional vector. Also, the words that have common contexts and semantics are located near each other. Word2Vec is used when training a neural network where the semantics part of NLP is important to be considered and where the objective is to maximize the probability of the next generated words making sense giving the previous ones [28] [29].

4) NLTK: Is the most used platform for building Python programs that work with human language data. It is a mix of libraries and different programs for symbolic and statistical NLP [29].

5) Spacy: Is a Python library known as a popular tool for NLP development since it is designed to help build applications that process and "understand" large amounts of text. Spacy includes advanced tokenization, parsing and entity recognition features. One characteristic that distinguish Spacy from other NLP libraries is the ability to use custom models for specific implementations, so developers can use the datasets they want for the topics they want to explore [30] [31].

6) Scikit-learn: It's used as data provider for multiple NLP problems, being a useful tool for to train models. Also it is commonly used in sentiment analysis tasks for example on Twitter datasets, [32] allowing to understand the user general feelings through his comments analysis. [33]

7) tensorflow-Keras: Keras is an open-source neural network library with user friendly features in Python platform that works on TensorFlow. Its an high-level API that compiles a model which is designed with a optimizer and a loss function, afterwards it handles the training process that is made with fit function. It also allows parallel data processing in which speeds up the training process for large volumes of data.

III. Dataset

The dataset used in this study to train the image recognition model was obtained on *Kaggle*, this dataset is composed by multiple classes with 6726 images of different Portuguese meals. Afterwards, some classes were removed because they were determined to not being relevant enough for this study, mainly those were not a typical Portuguese dishes, at the end it was added a few classes that were important to the Portuguese gastronomy. After this process described before, the dataset remained with 24 classes with a total off 4800 images.

One of the datasets used for NLP was obtained on *Kaggle*, it's composed by 13306 recipes. The other one was builded using Open-AI's Chat-GPT technology, because it was not found a good dataset of the Portuguese cuisine. The chatbot was asked to generate and retrieve multiple recipe variations from the existed classes presented in the dataset for image recognition and other Portuguese dishes. The recipes generated followed had the following structure: $title, ingredients, notes, region, recipe$. Afterwards the recipes were included in a csv file allowing it to be used for model training. This dataset is composed by 100 recipes.
The datasets used for Natural Language and Image Processing are about the same topic. The first described dataset for NLP consists exactly of 5 to 10 variations of the dishes represented by the image dataset. This allows the program to match the contents, between the image identification and the recipe suggestion.

A. Dataset Treatment

The dataset acquired from *Kaggle* for image recognition was not balanced, this may cause a problems while training the model. With this in mind, we set as the goal to have 200 examples for each class in order to balance the dataset and remove bias. Some of the classes didn't have 200 examples, so an extension was used to get more images for the specific class. Secondly, a python script was created to delete any image that couldn't be opened by python itself. As for the classes that had more than 200 images, another python script was created that randomly removed examples until each class reached the image goal.
Models with poor generalisation tend to overfit during the training and/or validation. The two graphs below (Figure 2 and 3) show how an overfitting model looks like in training/-validation accuracy and loss when plotting it.

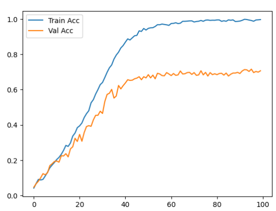

Figure 2. Overfitting Accuracy Plot

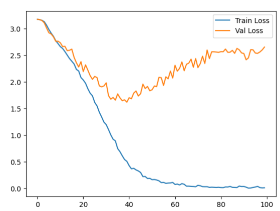

Figure 3. Overfitting Loss Plot

When building the model the validation accuracy and loss must increase and decrease, respectively, alongside with the training accuracy and loss. One really good way to help achieving this, although not the only way, is through data augmentation [34].
Data augmentation is crucial for image classification because it helps increase the variety and diversity of the training data, which in turn can improve the accuracy and generalisation of the model. Through data augmentation we can create more artificial training examples from the existing dataset, providing the model with a broader set of inputs to learn from in such a way that the label remains the same. This can help to prevent overfitting and improve the model's ability to recognise new and unseen images in the future [35]. So, in short, data augmentation is important for better accuracy and improved generalisation of the image classification model.
There are lots of different types of data augmentation. The ones used in this project were re-scaling, random zooms, horizontally and vertically shifts and flipping the image on a horizontal and vertical axis [36]. All these artificial transformations will increase the current dataset before starting the training.

The datasets used for the NLP had to go trough a script to remove words of the ingredient class that could add noise on the model training, such as: measurement words (kg, lb., teaspoon, gallon, etc...), and other random words (large, quality, plain, etc...). With this script the ingredient section of the dataset can be cleaned like this:

input = '['1 x 1.6kg whole duck', '2 heaped teaspoons Chinese five-spice powder', '1 clementine','6 fresh bay leaves', 'GRAVY', '', '1 bulb of garlic', '2 carrots', '2 red onions','3 tablespoons plain flour', '100 ml Marsala', '1 litre organic chicken stock']'

output = ['duck', 'five spice powder', 'clementine', 'bay leaf', 'gravy', 'garlic','carrot', 'red onion', 'plain flour', 'marsala', 'chicken']

The dataset from *Kaggle* had to go through other cleaning steps. It was removed some columns to keep just the title, ingredients and procedure, and it was removed rows empty/-faulty rows (rows without information regarding the dishes).

IV. Deep Learning Implementation

In this section, it will be explained the technologies in detail used in evaluating the performance of the segmentation model.

A. Technologies used

For the creation of the Convolution Neural Network an open-source library named TensorFlow for GPU was used. Using the GPU is an advantage since the training process will be faster. The first step was to divide our data into training and test, with 75% of the data going to training and the rest going to testing. The following step is the data augmentation process, in which an ImageDataGenerator function available in TensorFlow and Keras was used [37], in this function, the following parameters are available (rescale, rotation_range, zoom_range, width_shift_range, height_shift_range, horizontal_flip, vertical_flip and shear_range). The train and test data will have a target_size of (224,224), which means that the images will be the same size, and a batch_size of 32, which signifies that the network will be given 32 images to train in each epoch, and the class_mode is categorical because the given problem is a multi-class classification. The CNN was created with keras [38] having the following structure:

- Two Conv2D [39], this 2D convolutional layer will have 64 output filters (number of neurons) in the convolution, with a kernel size of (5,5), in the first layer it will be given an input shape for the images that will be 224x224px with 3 channels (RGB image). The activation function for each layer will be the rectified linear unit (relu).
- One MaxPoll2D [40], it will downsample the input image along its width and height, by taking the maximum value over an input window of 2x2 for each channel of the input
- One Dropout [41] layer that will set randomly the input units to 0 with a frequency rate of 0.01 at each step during the training time, which will help prevent overfitting from happening.

This structure will be repeated four times. Afterwards, a flatten layer will be performed [42] that will flatten the multidimensional input tensors into a single dimension from the output of the CNN constructed above, then this data will be passed to a dense layer [43] with a softmax activation function, in which the size of neurons is equal to the size of the given classes (24). A prediction is then made on the given input image.

An important factor in trying to avoid overfitting and reduce losses is by using an optimisation algorithm [44]. In which, for this project, it was used the Adam optimiser. The Adam optimiser is a widely used optimisation algorithm for training convolutional neural networks (CNNs). It optimises the learning rate specifically for each parameter, allowing for faster convergence towards the global minimum during training. It also includes momentum and adaptive learning rates, making it one of the most efficient optimisation algorithms for training deep neural networks [45] and the method is ”computationally efficient, has little memory requirement, invariant to diagonal rescaling of gradients, and is well suited for problems that are large in terms of data/parameters” [46].

The loss function that will be applied is the categorical_cross-entropy being useful in problems when there are two or more label classes to be classified. Then the train is performed when in this case it will be trained with 300 epochs, with the result of data augmentation in the training data, the parameters used for the training are the steps_per_epoch that will be equal to the length of the training data, the same goes for validation_step.

B. Models tested and evaluations

In evaluating the network segmentation results, the metric used is accuracy, it will return the accuracy of the model both in training and validation process. At first things weren't as good as expected, since it was still being performed a fine tuning in the hyper parameters, the first training got over fitting results as showed in figure 4.

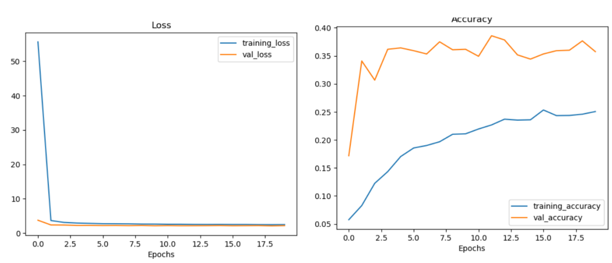

Figure 4. First training result

In order to reduce the overfitting an additional Dropout layer was added which also decreased the learning_rate, thus helping in reducing overfitting. The accuracy was improved by adding more convolutional layers until we got the structure already mentioned above returning an accuracy of 77% as shown in figure 5.

In the next two figures (Figure 6 and 7) it is shown the performance of the model in classifying different types of food.

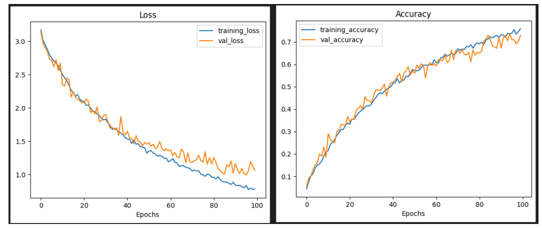

Figure 5. Model loss and accuracy

Figure 6. Prediction tripas moda porto

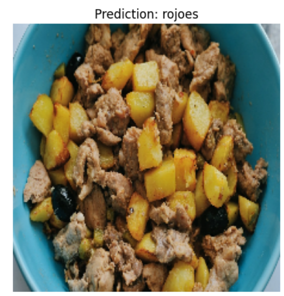

Figure 7. Prediction rojoes

V. NLP Implementation

This section will go over the implementation of NLP models and the technologies that were used. This project contains two NLP models:

The first receives as input (via the chatbot) a list of keywords such as ingredients, region, or dish name. With that information, the model will suggest a certain number of recipes containing the ingredients suggested by the user, the plate requested, various dishes from the region provided, or a plate combining those three categories.

The second model takes as input as a text document containing a recipe, extracts the ingredients from it, and modifies it so that it can be applied to an existing dataset. Further training of the two models indicated above will be possible using these extra recipes, which will improve model performance and result with a larger dataset.

A. Technologies Used

1) Model Word2Vec: The $keywords\_recipe$ model (first model mentioned on the section V) was built using *Word2Vec*, this NLP technique is ideal for word association problems, in this case reaching a solution given a certain word or a combination of words.

This model was trained using the Portuguese recipes dataset were the ingredients, region, and title are used as keywords to get the wished recipe. Every recipe will be a represented as a word embedding, where the keywords will be part of each recipe vector and will get a value, for example, if the keyword represents the dish then it will have a value close to 1, otherwise the value will be 0 (see Table I for better understanding). In the end, the model will give as an output the recipe with most similarities with the keywords given as

Recipe Embedding			
Keyword	Recipe 1	Recipe 2	(...)
Francesinha	1	0	-
ham	1	0	-
milk	0	1	-
Aletria	0	1	-
beef	1	0	-
cinnamon	0	1	-
egg	1	1	-
(...)	-	-	-

Table I
WORD EMBEDDING EXAMPLE

Keras Model	
Layer	Parameters
Spacial Dropout	0.2
LSTM	64
Spacial Dropout	0.2
LSTM	64
Sigmoid	1

Table II
LSTM MODEL

input, through a simple cosine similarity.

2) Model LSTM: The second described model was implemented using *Keras* library from *TensorFlow*. From that library the model was based on LSTM network, this layers can process sequences of data and are perfect for processing and predicting data. The model is composed by Dropouts with 20% rate of drop and 64 units of Bidirectional LSTM layers, represented in the Table II. The last layer is an unitary Sigmoid to map the output of the model between 0 and 1. This layer is commonly used for probability purposes.

For the training of the model, it was fed the tokenized recipes. For its' output, a vector of the size of the recipe is given with values instead of tokens, if the value is high it means that it's likely to be a ingredient. This vector of values is then compared with the true label (a vector with true or false, where true represents the presence of an ingredient). Loss is backpropagated so the model can adjust it's weights and get better results.
The *Kaggle* dataset was separated in 90% test and 10% evaluation, and it went trough the model 10 times (representing 10 epochs) in batchs of 256 recipes. The Adam optimizer was once again used with a step of 0.001, and a binary crossentropy as a loss function to track incorrect labelling of the data.

B. Models tested and results

The model LSTM was, by far, the model that needed more approaches and work put into until a reliable result was obtained. The first approach was to train a spacy model using the portuguese dishes dataset. The model was inappropriate and inconclusive as it had huge loss values. Considering this our next attempts were focused on the training of an RNN model. At the same time experiments were conducted to test and compare the results using a LSTM model. Both models reached an max accuracy value of 20%.
After being presented with these low results, an alternative approach was made by maintaining LSTM as the approach keeping the portuguese cuisine dataset. The results are displayed below (image 8):

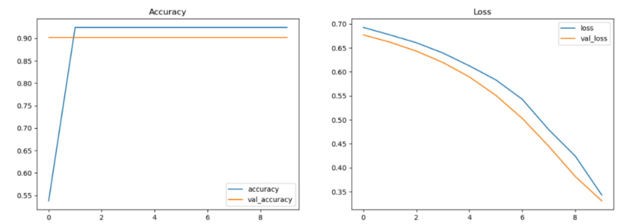

Figure 8. LSTM model result with Portuguese dataset

The outcomes were good, but far from flawless. The model's output had several words that weren't ingredients, representing a 33% loss. To address this issue, it was decided to obtain a larger dataset, which included 10,000 recipes, many of which were from other countries and origins.
As seen in the graphic 9, this dataset significantly improved the results, increasing accuracy to 92% and smoothing the loss value. The loss was reduced to 6.7%.

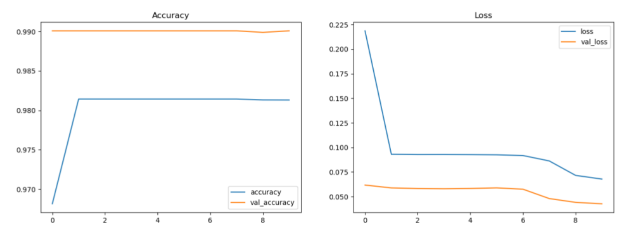

Figure 9. LSTM model result with big dataset

To get the ingredients of a recipe, it's filtered from the output of the model all the predicted words that have a probability greater than 10% (this value was chosen not calculated, it's a probability that works minimally well) to be an ingredient.
Example of a recipe not present on the dataset going trough the model:
input: Season the steaks with garlic, paprika, salt, and pepper. Heat the olive oil in a large skillet over high heat. Add the steaks to the skillet and cook them for 2-3 minutes on each side or until they reach your desired level of doneness. Remove them from the skillet and set them aside. In the same skillet, melt the butter and fry the eggs to your desired level of doneness. Add the red wine vinegar to the skillet and stir to scrape up any browned bits from the bottom of the pan. Reduce the heat to low and simmer for a minute. Serve the steaks on a plate with fries, rice, and salad. Place a fried egg on top of each steak and drizzle the pan sauce over the top.

output: garlic, paprika, salt, pepper, olive, oil, butter, wine, rice, egg, steak

It can be seen that the model identified almost every ingredient of this recipe.

VI. User Interaction

The FoodAids application contains a User Interface designed for Android mobile devices. It was chosen to target mobile devices because they offer a high quality camera and due to their small nature tend to be carried everywhere. The application components and the interaction the user can have with that are displayed at the images 10 and 11.

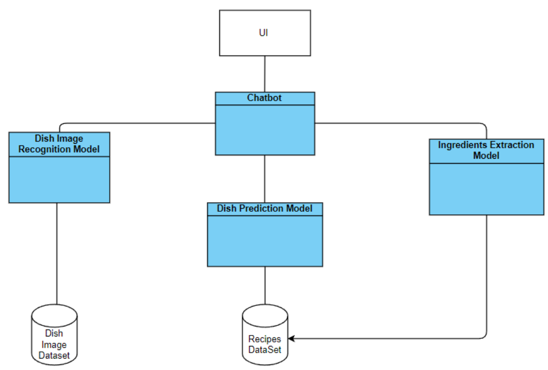

Figure 10. Application Components

The figure 11 shows what the application's use cases. The first use case requires an image to be taken by the user on his phone of any Portuguese dish, then a classification is made and if user asks it can return information regarding nutritional levels or some recipes of the dish. The other use case that can be performed is by giving an available ingredient list, title of the dish or a typical region and the bot will return dishes available with the given input, being more suitable for cooking purposes, since its deploying recipes. The last use case that can be made is letting the user adding a recipe that does not exist at the moment, so it can be trained after and being available for other uses.

A. Chatbot

For the purpose of interacting with the user a chatbot was also implemented. Chatbots provide an easy-to-understand way to interact, express doubts or questions as well as receive answers in real-time, providing a seamless user experience. This implementation allows the user to express his queries to the bot on if there exist dishes by a certain name or similar, as well as thorough the provision of an image of a dish get it identified. This was achieved through the identification of certain key words in the users queries that allowed for their intention to be devised. Then, through the use of the already described models, whose inputs were also inferred from the user's messages, answers are predicted and given back to the user.

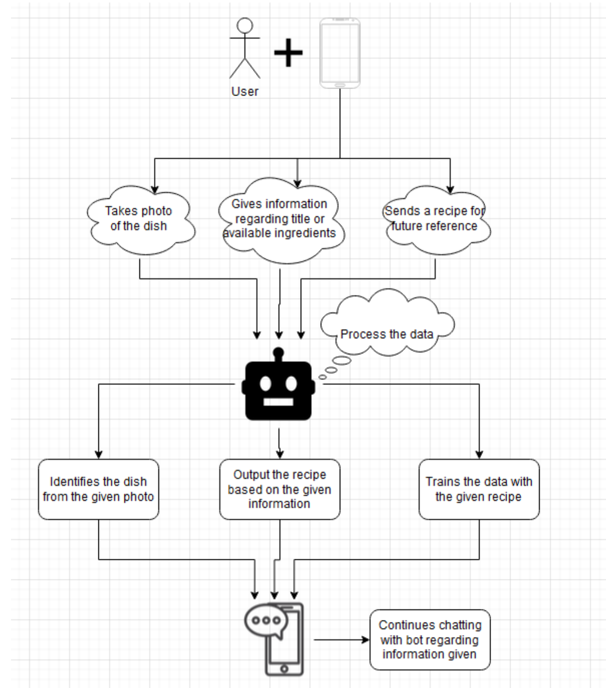

Figure 11. User application

VII. Future Improvements

One of the future improvements is regarding the UI, by enhancing even more the user experience. The UI is an important part of any product, an user-friendly UI makes the experience for the users better as well being easier to display the current available operations to do, with the objective to make the usage the easiest way possible.

Another future improvement is to expand the variety of dish's available in the dataset, at the moment its only available traditional Portuguese dish's, being perfectly possible to agglomerate more dish's so its possible to reach a wider selection of users, since it will be possible to cook any dish from different countries with correct selection of ingredients, being a valuable experience for the users since they can cook dish's that they never knew before with all information regarding that dish being displayed in one place, bringing a good variety of food in people diet and lifestyle also its always a good way to know different gastronomy across the world.

Another asset to be added is the possibility for an user to add new plates to the image recognition dataset, so if there is a plate that the dataset doesn't know, the user can add photos of the plate and when a quantity of images of the plate is reached the dataset will learn from it, being available for classification afterwards. With this in mind, the model can give the wrong prediction of the given photo of a plate, in which the user can give a feedback towards it, so the model can learn from it, correcting the wrong predictions and learn from it.

VIII. Conclusion

In this report an automatic method for identification of traditional Portuguese dish's in images taken from the phone and two NLP models with a chatbot were presented. The results obtained through image recognition showed a great potential in the proposed method. When looking at the state of the art regarding image classification in food science, it is observed that the obtained accuracy of 77% from the model is similar to the ones mentioned in this article. By the experiments performed, the model was successful in classifying the type of dish given, showing to his potential in future uses. The use of this application will be an advantage for home cooking since it can add variety to a person's diet with the available ingredients, being also a financially benefit since cooking at home is cheaper than home delivery or going out. With more development, a better user interface (UI) can be created, making it simpler for users to fully appreciate the program, contributing for a healthier lifestyle and better eating habits.

REFERENCES

[1] Denise de Ridder, Floor Kroese, Catharine Evers, Marieke Adriaanse, and Marleen Gillebaart. Healthy diet: Health impact, prevalence, correlates, and interventions. *Psychology and Health*, 32:907–941, 8 2017.

[2] Francisco Lima. As pessoas - 2021.

[3] Lindsey P Smith, Shu Wen Ng, and Barry M Popkin. Trends in US home food preparation and consumption: analysis of national nutrition surveys and time use studies from 1965–1966 to 2007–2008. *Nutrition Journal*, 12(1):45, dec 2013.

[4] Susanna Mills, Martin White, Wendy Wrieden, Heather Brown, Martine Stead, and Jean Adams. Home food preparation practices, experiences and perceptions: A qualitative interview study with photo-elicitation. *PLOS ONE*, 12(8):e0182842, aug 2017.

[5] Bonnie Ghosh-Dastidar, Deborah Cohen, Gerald Hunter, Shannon N. Zenk, Christina Huang, Robin Beckman, and Tamara Dubowitz. Distance to Store, Food Prices, and Obesity in Urban Food Deserts. *American Journal of Preventive Medicine*, 47(5):587–595, nov 2014.

[6] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44:3523–3542, 7 2022.

[7] Lei Zhou, Chu Zhang, Fei Liu, Zhengjun Qiu, and Yong He. Application of deep learning in food: A review. *Comprehensive Reviews in Food Science and Food Safety*, 18:1793–1811, 11 2019.

[8] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33:6999–7019, 12 2022.

[9] Sumit Saha. A comprehensive guide to convolutional neural networks — the eli5 way.

[10] Joseph Weizenbaum. Eliza- a computer program for the study of natural language communication between man and machine.

[11] Jonathan J Webster and Chunyu Kit. Tokenization as the initial phase in nlp.

[12] revathi B S, A Meena Kowshalya, First Year Scholar, AMeena Kowshalya, and Assistant Professor. The road map from artiicial intelligence, machine learning, deep learning techniques towards image captioning system. the road map from artificial intelligence, machine learning, deep learning techniques towards image captioning system. 2021.

[13] Dastan Hussen Maulud, Karwan Jacksi, Th Karzan, and Hussein Sharif. A state of art for semantic analysis of natural language processing.

[14] Ashutosh Singla, Lin Yuan, and Touradj Ebrahimi. Food/non-food image classification and food categorization using pre-trained googlenet model. *MADiMa 2016 - Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management, co-located with ACM Multimedia 2016*, pages 3–11, 10 2016.

[15] Patrick McAllister, Huiru Zheng, Raymond Bond, and Anne Moorhead. Combining deep residual neural network features with supervised machine learning algorithms to classify diverse food image datasets. *Computers in Biology and Medicine*, 95:217–233, 4 2018.

[16] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 - mining discriminative components with random forests. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8694 LNCS:446–461, 2014.

[17] Yoshiyuki Kawano and Keiji Yanai. Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8927:3–17, 2015.

[18] Yuji Matsuda, Hajime Hoashi, and Keiji Yanai. Recognition of multiple-food images by detecting candidate regions. *Proceedings - IEEE International Conference on Multimedia and Expo*, pages 25–30, 2012.

[19] Food image recognition using covariance of convolutional layer feature maps atsushi tatsuma †a) and masaki aono †b) , members.

[20] Keiji Yanai and Yoshiyuki Kawano. Food image recognition using deep convolutional network with pre-training and fine-tuning. *2015 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2015*, 7 2015.

[21] Chang Liu, Yu Cao, Yan Luo, Guanling Chen, Vinod Vokkarane, and Yunsheng Ma. Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9677:37–48, 2016.

[22] Zhihui Fu, Dan Chen, and Hongyu Li. Chinfood1000: A large benchmark dataset for chinese food recognition. In De-Shuang Huang, Vitoantonio Bevilacqua, Prashan Premaratne, and Phalguni Gupta, editors, *Intelligent Computing Theories and Application*, pages 273–281, Cham, 2017. Springer International Publishing.

[23] Yuran Pan, Qiangwen Xu, and Yanjun Li. Food recipe alternation and generation with natural language processing techniques; food recipe alternation and generation with natural language processing techniques. *2020 IEEE 36th International Conference on Data Engineering Workshops (ICDEW)*, 2020.

[24] Oleksii Trekhleb. Generating cooking recipes using tensorflow and lstm recurrent neural network: A step-by-step guide.

[25] Tensorflow. Introduction to the tensorflow models nlp library.

[26] Bo Pang, Erik Nijkamp, and Ying Nian Wu. Deep learning with tensorflow: A review, 2020.

[27] Gensim vs nltk.

[28] KENNETH WARD CHURCH. Word2vec. *Natural Language Engineering*, 23, 2017.

[29] Word embeddings in python with spacy and gensim.

[30] Joseph Robinson. Practical python: spacy for nlp.

[31] Yuli Vasiliev. *Natural Language Processing with Python and spaCy: A Practical Introduction*. 2020.

[32] Machine learning, nlp: Text classification using scikit-learn, python and nltk. — by javed shaikh — towards data science.

[33] Practical python: spacy for nlp. a beginner's guide to natural language… — by joseph robinson, ph.d. — towards data science.

[34] Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, 12 2019.

[35] Krishna Chaitanya, Neerav Karani, Christian F. Baumgartner, Ertunc Erdil, Anton Becker, Olivio Donati, and Ender Konukoglu. Semi-supervised task-driven data augmentation for medical image segmentation. *Medical Image Analysis*, 68, 2021.

[36] Liang Huang, Weijian Pan, You Zhang, Liping Qian, Nan Gao, and Yuan Wu. Data augmentation for deep learning-based radio modulation classification. *IEEE Access*, 8, 2020.

[37] tf.keras.preprocessing.image.imagedatagenerator- tensorflow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator.

[38] Module: tf.keras.layers-tensorflow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/layers.

[39] tf.keras.layers.conv2d — tensorflow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/layers/Conv2D.

[40] tf.keras.layers.maxpool2d — tensorflow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/layers/MaxPool2D.

[41] tf.keras.layers.dropout — tensorflow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dropout.

[42] tf.keras.layers.flatten — tensorflow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/layers/Flatten.

[43] tf.keras.layers.dense — tensorflow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dense.
[44] Roseline Oluwaseun Ogundokun, Rytis Maskeliunas, Sanjay Misra, and Robertas Damaševičius. Improved cnn based on batch normalization and adam optimizer. volume 13381 LNCS, 2022.
[45] Lydia Agnes and Francis Sagayaraj. Adagrad - an optimizer for stochastic gradient descent. 5 2019.
[46] Diederik P Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization.