Street food scene segmentation using Places features

Group 4

August 6, 2022

Abstract

Street food is one of the special cultures of Vietnam which attracts a lot of tourists. In this paper, in order to create more data for furthermore research, we study the relationship between places and the presence of food in a video. This relationship is figured out by segmenting the video into segments that contain food. Our work uses data collected from Youtube platform, which is a collection of street food videos. The project has 3 main steps. Firstly, we make the annotation file which contains the data of the street food videos. Second, we analyze the data to have a better understanding of data. Finally, we implement a lot of models to figure out which one has the best performance in segmenting the video into segments that contain food. The result shows that there is a weak relationship between places and the presence of food in a video with an IOU score in segmentation equal to 30.4%.

Key words: street food, scene segmentation, places segmentation

1 Introduction

Vietnam is widely known for its "sidewalk culture" and "sidewalk business culture", which were inherited a long time ago [1]. As a result, street food is considered to be one of the beauties of Vietnamese culture, making them a special feature when foreigners talk about Vietnam [2], and gradually gaining a strong reputation for its diversity and the delicious taste of the food. Based on the statistics from the General Statistics Office of Vietnam (2017), foreign tourists spend about 22,23% of their total daily expenditure on food and drinks in Vietnam, which means that foreigners give big attention to Vietnamese food [3]. Among many kinds of food, street food is considered to be a must-try for foreigners when they come to Vietnam [4]. The article will figure out the characteristics of street food and some related statistics.

In the new era, when science and technology are increasingly developing, more and more applications are constantly created to help to increase productivity in every aspect of human daily life [12]. Especially, artificial intelligence is developing rapidly, and its application to the assessment of people's behavior is necessary for the development of society [5]. With this evolution, we implement Artificial Intelligence in our project to:

- Create a premise for research projects related to the influence of where food is used on consumers' behavior.
- Analyze the relationship between eating places and consumer satisfaction and habits.
- Show the popularity of eating places that are chosen by many travelers and gives advice to sellers.

The research is widely carried out based on videos about street food across 3 main regions of Vietnam. Thanks to this wide range of data, the objectivity of the research will be guaranteed. This also helps our research become more accurate.

The data we use in this project is a collection of videos about street food in Vietnam. We choose YouTube to be a place to collect these videos. The reason why we choose it is that YouTube is the biggest and most popular video hosting platform in the world [6]. Therefore, it provides a huge collection of videos for us to do the research. However, the problem is that those videos have not had any annotation yet. Therefore, we have to do an additional step, which is to hand-label all videos in our collection to determine which segments contain food. Besides, we use data generated by a pre-trained model which consists of ratios of places that may appear in each frame of the videos. Based on these ratios, we can classify each frame of a video into 2 categories: food and not food. Thanks to that classification, the final result of this project is segmenting the video into several parts that belong to 1 of the 2 above categories. Having these parts of the video, we can implement other applications that are related to street food.

2 Project management plan

2.1 Team members

- Nguyen Trong Quoc Dat (Team Leader)
- Nguyen Xuan Duy
- Vu An Ninh
- Tran Thanh Duong
- Nguyen Tan Loc
- Nguyen Doan Hieu Nguyen

2.2 Project

In this project, we will focus on video scene segmentation using place information for street food videos that have been captured across Vietnam based on places probability which are exported by a pre-trained places classification model. The main scope of this project is to provide an overview of the machine learning method and some methods that are used to classify whether our frame has food or not and segment the video into small parts which have food at that time.

2.3 Planning

Using the input provided and the annotations created in the very first phase of the project, our team established a clear vision on what to do to proceed the project. We continue to do some research on the Internet for better resources of related works, related models, related projects,... to apply into this project. The requirement is to ensure the progress every week (introduction, project management, related works,...) and create the most sufficient model as well as a clear report and good experience at the end.

This project is continuously running from the start of semester Summer 2022 till the end of the semester. Divided into six phases as follow:

- Planning and project setup

- Collecting and exploiting data
- Collect and read related works
- Build base model
- Experiments and Improvement
- Writing paper

2.4 List of action

The project schedule is defined at the link. This is where we assign the member tasks as well as deadlines. All members have to check this link daily to know what to do to achieve the team's objective.

2.5 Risk Management

Any project has to deal with many risks, this project has no exception. Some of them are "not checking schedule", "tasks that only 1 member takes", etc. To get rid of them, we have to set some group rules. For instance, an important task must be assigned to 2 or more members. In case 1 member cannot take that task, the others can try to handle it. If some member wants to change the day of the meeting, he must announce it for earlier than 1 day. There are a number of group rules that should be set at first in order that the team does not have to face many obstacles.

3 Related Work

Have you ever asked yourself, what is the most popular food in Vietnam? If you once have that question in your mind, can you answer it? Undeniably, this is truthfully a very difficult question. It is because of the fact that Vietnam is a country with a rich tradition, especially in food variety. As a result, it is hard to tell which food is the most popular. However, nowadays, with the development of Data Science as well as Artificial Intelligence, we can at least find out top popular food in Vietnam. However, we have to collect more data for this research. You may notice that in Vietnam, street food is a style of business that attracts a lot of customers, which provides us with a large amount of data. Therefore choosing street food as a resource for extracting more data is a suitable approach.

In order to have more data for further research, we extract from videos only segments that have food. This is because these segments may have more valuable data. There are many criteria for segmenting videos. Among them, we choose using places in a video. And to find a way to carry out this implementation, we have searched several pieces of research related to our problem.

First of all, because our research works all places, the model should learn what places are there in a frame of a video. It might be very difficult to do this because our human being sometimes cannot do it somewhere. Fortunately, we find some ways to do this. One of them is using MACNet [7]. With the problem detection food place, the model MACNet improved its performance by using a self-attention mechanism. MACNet is created based on an atrous convolutional network and ResNet [10], it works on images but not use any time feature. Using the features that are extracted by MACNet, we fetch all features into LSTM [8] to create long-term dependencies of all images per event. Besides, the self-attention mechanism helps the model to

focus on the important features, which are the ratio of places in a frame.

After having a lot of columns representing the probabilities of each place in a frame, we try to determine whether there is food in a frame. In this case, we also refer to a lot of papers, for example, Food Log by Analyzing Food Images [9]. This paper uses other features from a frame of determination such as color histograms, DCT coefficients,... and SVM is used to detect if an image contains food or not. Using all of the features and methods (Kitamura, Yamasaki and Aizawa, 2008). create the model with accuracy up to 73%.

Another approach for scene segmentation problem, after partition a video sequence into shot, (Sidiropoulos et al., 2011) [13] was introduced the new method to concat the shot into scene. Scene transition graph STG and generalized scene transition graph GSTG were used in this situation. The methods were trained and tested in documentary and movie, GSTG achieved the F-Score equals 77.91% and 77.83%.

In the problem recognize the activity in the kitchen, (Bansal, Khandelwal, Gupta and Goyal, 2013) [14] was segment hand and recognize object in each frame as a feature to to train their model. SVM-HMM model was used to combine the structural feature and temporal video sequence information to jointly recognize the most likely cooking activity label. Their model was achieved the accuracy more than 72% on real work cooking dataset.

When video segmentation is done, we have to measure how good our work is. There are a number of ways to calculate the accuracy of a model's performance. One of them is used in the paper Temporary sequence modeling for Video Event Segmentation [10]. With this metric, we examine our model in order to improve it into a better version.

4 Data pre-processing

4.1 Data overview

Our data contains over 300 features, which are the probability of a place occurring in frames. For every row, all the columns will sum up to 1. The dependent variable is not very balanced: 383792 samples labeled 0, 201890 samples labeled 1. For simplicity, we will sample n records and do our exploration of data analysis. The data after sampling must keep the original ratio between classes, as well as the characteristics of the entire dataset. After trying with different values of n, we decided that 50000 is an appropriate sample size, which is not too large to handle and still keeps the population characteristic.

4.2 Exploratory data analysis (EDA)

After sampling, we will do some statistical exploration to have a better insight into our data. First, we are interested in which places are most frequently occurring in data. So there are 2 ways to nail it. First is to check in every row, find the top n places that have the highest probability, repeat this process for the entire data, collect the result and count the number of each place. However, this approach is not mathematically right, due to a place having low probability doesn't mean that place does not exist, however, if this place is low, this place will be discarded. So we

have another approach, instead of hard choosing n places, we will take the sum of all columns, across the row, the array after this step will be the sum of all probability of a place in all frames (¡ 1), after that we can take mean, these value will reflect the probability of a place in the entire dataset. The order at n after sort descending will be n placed most frequently.

Because most of the values in a column are 0, so to remove unnecessary columns, we will do a hypothesis test to verify which column really matters in classifying our target variable.

We first group all records by their label value, so all records have the same label value in the same set.

Now all columns are grouped by their name and label. We will do a hypothesis test that the distribution of probability occurring of place A in label 0 is different in label 1. Because the data in every column are right-skewed, most of them are 0 and non-negative, so this is exactly characteristic of Zero-inflated distribution. So we remove all 0 values and take their log, so we hope that data after transformation are approximately normal (show in Figure 1).

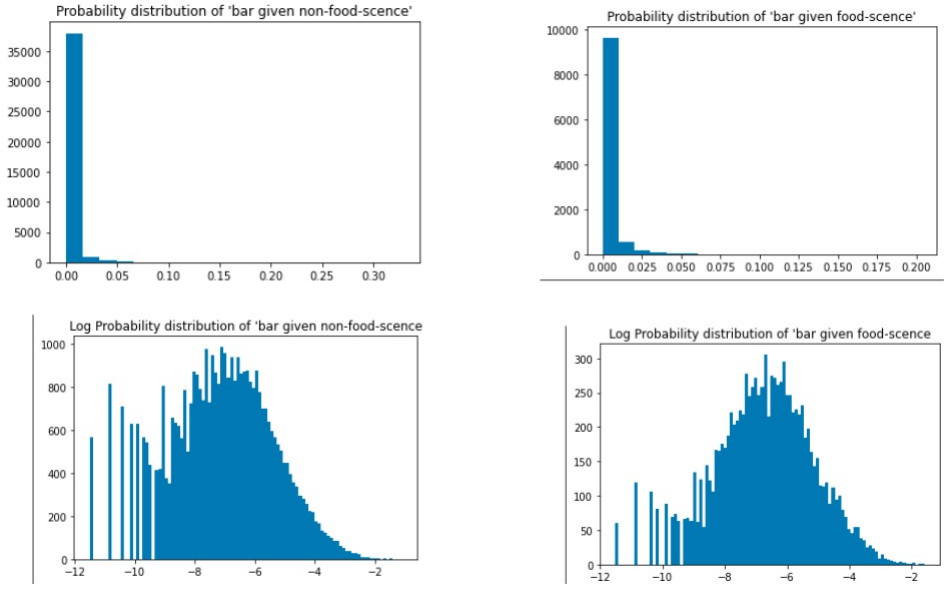


Figure 1: Probability Distribution

After having transformed data, we will do ANOVA F test to test means between group is different significantly, with number of groups is 2 (food and non-food scene) and variable to test is values of each feature within this group.After the test, we will get F-score, sort these score descending we will get features which is mostly separate 2 groups statistically

5 Our approach

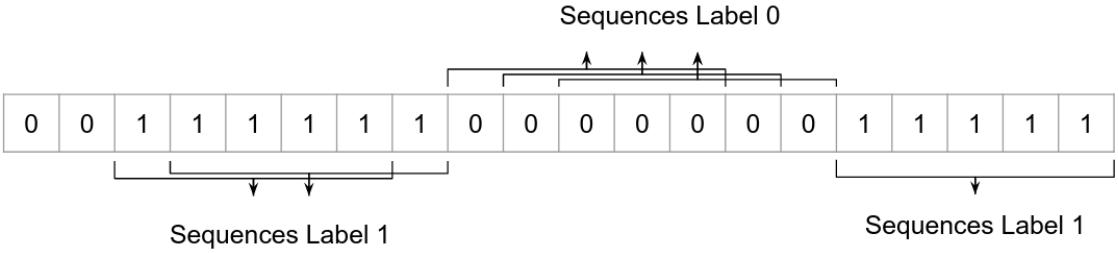
5.1 Method

It was quite easy for everyone to realize that a video contains the related information in continuous frames. For example, 2 frames in a video which are next to each other may be taken at the same place, with almost the same objects in each frame, containing almost the same colors as well as almost the same scene. Therefore they may have the same state (food or non-food). As a result, if we can make full use

of this characteristic, we may considerably improve the performance of the model. Based on this idea, using some well-known sequential built-in models in Tensorflow such as GRU, LSTM, we tried to implement a model that uses consecutive frames for classification.

5.2 Data preparation

Based on the idea cited above, we prepared needed data by merging 5 consecutive frames containing the same label (label 0 for non-food and label 1 for food). In order to have more data, we collected overlapped samples as in the graph below. The reason why we chose 5 to be the length of a sequence is that we thought this length could well maintain the information in consecutive frames. After the preparation step, we have a lot of tensors with shape (5, no_of_features) with "no_of_features" being the number of place features that we chose using SelectKBest in Scikit-learn. Having this data, we used it to train on multiple sequential models and tested which one is capable of carrying useful information of consecutive frames, so that it could perform as well as possible on classification tasks.



5.3 Loss function

Our model output is the label of each frame. So the loss function we are using here is binary cross entropy:

$$Loss = -\frac{1}{\mathrm{m}}\sum_{i=1}^{\mathrm{m}} y_i \cdot log\hat{y}_i + (1-y_i)log(1-\hat{y}_i)$$

With output size is the number of labels and this is equal to the number of frame data in input data. y_i is true label of frame and $\hat{y}_i$ is predicted label that we using our model to predict label of this frame

5.4 Evaluation protocol

Finally, our evaluation task is to measure the discrepancy between the predicted timestamps and the ground truth ones. In other words, we have to measure how exactly 2 periods of time (predicted one and ground truth one) match each other. This sound similar to the task of evaluation in image segmentation or object localization. In this situation, people use IOU [18] as a useful metric to score the accuracy of a model. Based on the idea of measuring how well two objects overlap each other, in this project, we use IOU as the main metric too. The idea is to calculate the ratio between the overlap of 1s and the 1s in the predicted list of the ground truth list.

$$IOU = \frac{A \cap B}{A \cup B}$$

In training process, we have trained our model in sequence data with the size is seq_length that we had combined before, our prediction is one label that is uniform for all frame that is in the sequence. And we will assign this label to all frame that

is included in the sequence so that we will get the predict label for each frame for current video

Because after the prediction and data reprocessing, each frame of our video will be labeled (i.e 1 for food existence, 0 for non-food existence). This is a binary form so we suppose that:

- Predicted video: pred_video
- True labeled video: true_video

First of all, to calculate the intersection between predicted video and true labeled video for IoU we are using bit-wise AND operation (i.e) because our label is only presented in bit number (i.e 1 and 0). And we focus on the food segment (a consecutive sequence of 1 label) so the intersection we calculate is the intersect between each food segment in the pred_video and true_video

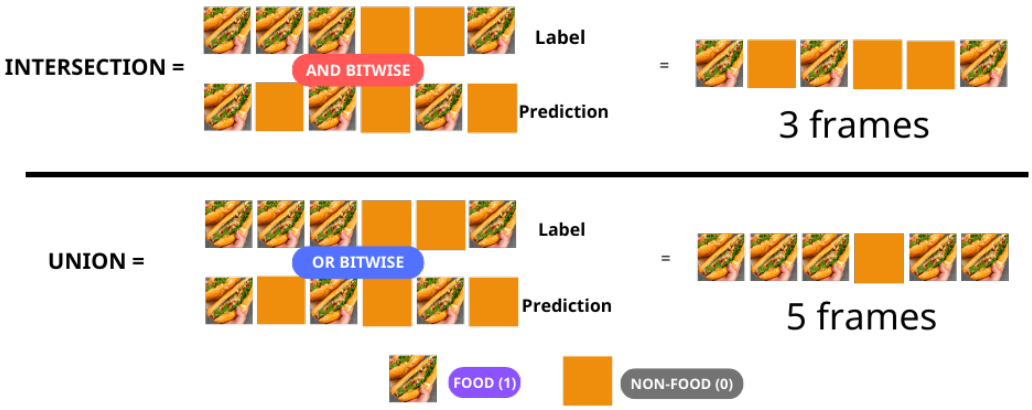
$$\textbf{Intersection} = \textbf{pred\_video AND true\_video}$$

Secondly, the union of food segment between predicted video and true labeled video we are using bit-wise OR operation so it will generate the union of ground truth segment and predicted segment

$$\textbf{Union} = \textbf{pred\_video OR true\_video}$$

Each intersection part and union part we will calculate the number of 1 label in each part. Finally, we have the formula of IOU score:

$$IOU = \frac{Intersection}{Union}$$



6 Experimental Results

6.1 Baseline models: Using built-in models for classifying frame-by-frame

In the very first step, we tried to use some state-of-the-art techniques in machine learning. In this stage, we only classify frame-by-frame for all the frames in a video. In order to apply this, we use a built-in model in Scikit-learn to create a baseline for the project. Specifically, we applied 3 well-known built-in classifiers: Gradient Boosting [15], Random Forest [16], and XGBoost [17]. The performances of these models are presented in Table 1.

Models	Precision		Recall		F1-Score		IOU
	0	1	0	1	0	1	
Gradient Boosting Classifier	0.75	0.5	1.00	0.01	0.86	0.02	0.009
Random Forest Classifier	0.76	0.38	0.95	0.09	0.84	0.14	0.078
XGBoost Classifier	0.83	0.29	0.38	0.75	0.54	0.40	**0.251**

Table 1: Performance of baseline models

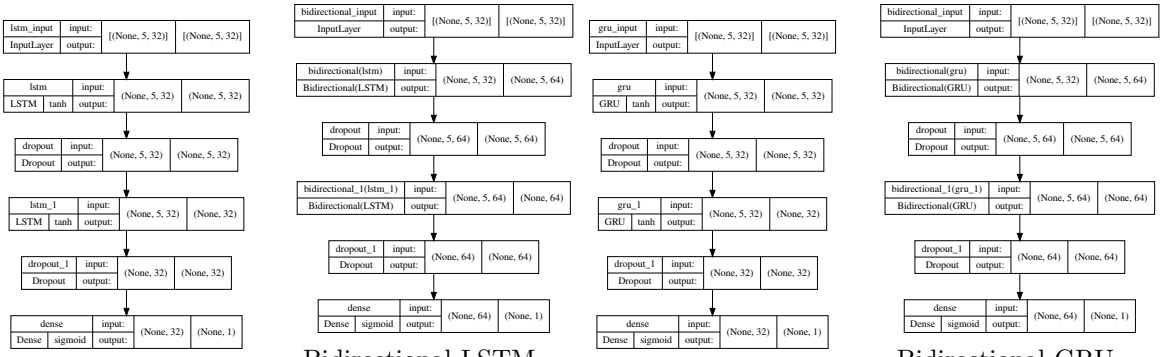


Figure 2: Model Architecture

6.2 Sequence modeling

The above results show that it is not appropriate to use the frame-by-frame methods when we are training on a dataset where the information of the frames may be interrelated. Frame-by-frame recognition will not create information coherence and may lead to model inefficiencies. Therefore, we prepared a new dataset with the length sequence of 5 consecutive frames and trained on two well-known sequences modeling LSTM [20], GRU [21], and two other variant models, Bi-GRU and Bi-LSTM, where Bi is Bidirectional because we believe that passing information in both directions of the sequence will bring better efficiency. The specific architecture of these models is in the Figure 2.

We used 527030 sequences with shapes (5, 32) where 5 the is number of frames and 32 number of features (the number of frames is the same as in the base models), we used Binary Cross-Entropy loss, AUC metrics, Adam optimizer [19] with learning rate 0.001, training on 1000 epochs with mini-batch size 4096, and keeping the model having the best validation AUC. However, because of the unbalanced dataset, we set class weight by the ratio of 1/3 with one for class 0 and three for class 1. The result of our training is shown in Table 2.

Models	Precision		Recall		F1-Score		IOU
	0	1	0	1	0	1	
LSTM	0.83	0.36	0.63	0.61	0.72	0.45	0.292
Bi-LSTM	0.83	0.36	0.63	0.62	0.72	0.46	0.294
GRU	0.83	0.37	0.65	0.61	0.73	0.46	**0.296**
Bi-GRU	0.84	0.31	0.45	0.75	0.58	0.44	0.283

Table 2: Performance of first sequence models

6.2.1 Try to solve Overfit

After training, we recognized the big problem with these models is that they are very easy to overfit, we have shown their learning curves in Figure 3. Because of that, we decided to increase the dropout rate to see if it would solve the problem of

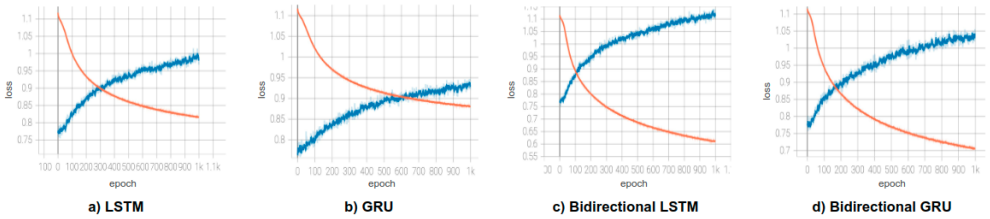


Figure 3: Learning Curve

overfitting. We increase the dropout rate from 0.3 to 0.5 and the number of features from 32 to 128. We obtained the result show in Table 3 and the learning curve in the Figure 4.

Sequences models' results							
Models	Precision		Recall		F1-Score		IOU
	0	1	0	1	0	1	
LSTM	0.82	0.38	0.69	0.56	0.75	0.45	0.288
Bi-LSTM	0.84	0.35	0.58	0.67	0.69	0.46	**0.294**
GRU	0.84	0.34	0.56	0.68	0.67	0.45	0.292
Bi-GRU	0.85	0.32	0.49	0.73	0.62	0.45	0.289

Table 3: Performance of first sequence models after changing dropout rate a

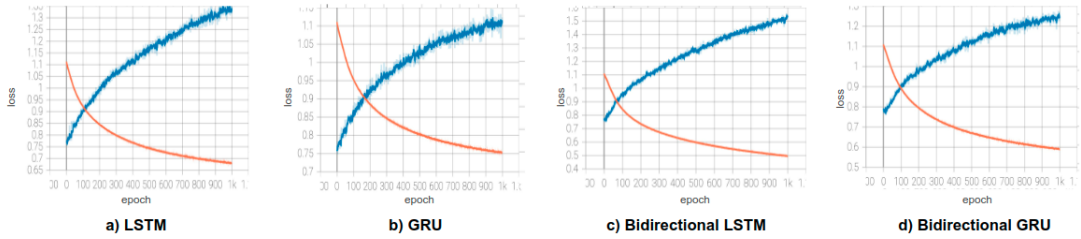


Figure 4: Learning Curve after Increase Dropout

In spite of having increased the dropout rate and having increased the number of features, models were still overfitted and the performance also did not improve. We think this happened because it still not enough dropout rate and also be noise because of too many features (bad features). So that we decide to use the grid search to find the appropriate parameters.

Sequences models' results - LSTM model					
Features \ Dropout Rate	0.3	0.5	0.7	0.9	Max
16	0.295/0.456	0.296/0.457	0.294/0.455	0.292/0.452	0.296/0.457
32	0.292/0.452	0.294/0.455	**0.298/0.459**	0.294/0.454	0.298/0.459
128	0.290/0.450	0.288/0.447	0.290/0.450	0.292/0.452	0.292/0.452
256	0.285/0.444	0.291/0.450	0.295/0.456	0.288/0.448	0.295/0.456
Max	0.295/0.456	0.296/0.457	0.298/0.459	0.294/0.454	0.298/0.459

Table 4: IOU/F1 score on LSTM using grid search of no. features and dropout rate

According to the results, we obtain that the higher features the worse performance, which is easy to understand because like what we analyzed in the previous section there very little data is significant so too many features will make data more noise. Like our assumption, 0.5 is not enough dropout rate to have better performance so 0.7 is clearly be the best choice, and also if the dropout rate too high it

Sequences models' results - GRU model					
Dropout Rate / Features	0.3	0.5	0.7	0.9	Max
16	0.295/0.456	0.294/0.454	0.295/0.455	0.295/0.455	0.295/0.456
32	0.296/0.457	0.294/0.454	**0.298/0.459**	0.288/0.447	0.298/0.459
128	0.291/0.451	0.292/0.453	0.289/0.449	0.294/0.454	0.294/0.454
256	0.283/0.441	0.284/0.443	0.287/0.447	0.291/0.451	0.291/0.451
Max	0.296/0.457	0.294/0.454	0.298/0.459	0.295/0.455	0.298/0.459

Table 5: IOU/F1 score on GRU using grid search of no. features and dropout rate

Sequences models' results - Bi-LSTM model					
Dropout Rate / Features	0.3	0.5	0.7	0.9	Max
16	0.293/0.453	0.295/0.456	0.294/0.455	0.293/0.454	0.295/0.456
32	0.294/0.455	0.296/0.457	**0.297/0.458**	0.296/0.457	0.297/0.458
128	0.291/0.451	0.294/0.455	0.296/0.456	0.295/0.456	0.296/0.456
256	0.286/0.445	0.288/0.448	0.290/0.450	0.288/0.447	0.290/0.450
Max	0.294/0.455	0.296/0.457	0.297/0.458	0.296/0.457	0.297/0.458

Table 6: IOU/F1 score on Bi-LSTM using grid search of no. features and dropout rate

Sequences models' results - Bi-GRU model					
Dropout Rate / No. Features	0.3	0.5	0.7	0.9	Max
16	0.295/0.455	0.291/0.451	0.294/0.454	0.294/0.454	0.295/0.455
32	0.283/0.441	0.292/0.452	**0.298/0.459**	0.295/0.455	0.298/0.459
128	0.293/0.454	0.289/0.449	0.289/449	0.292/0.452	0.293/0.454
256	0.285/0.443	0.287/0.447	0.287/0.446	0.291/0.451	0.291/0.451
Max	0.295/0.455	0.292/0.452	0.298/0.459	0.295/0.455	0.298/0.459

Table 7: IOU/F1 score on Bi-GRU using grid search of no. features and dropout rate

will make the model worse.

6.2.2 Fine-tuning the number of frames per sequence

After that, we thought the number of frames per sequence will also affect the performance of a video, so we decided to fine-tune the number of frames per sequence. We use the best parameters of the previous try which is a dropout rate of 0.7 and 32 features for 4 models and we tried many options: 1/2/4/5/6/8/12/14/16/18/20 frames/sequences, training the same configure of the previous try, and we obtained the result as in the Table 8.

Length \ Model	LSTM	GRU	Bi-LSTM	Bi-GRU	Max
1	0.290/0.450	0.289/0.449	0.280/0.483	0.289/0.449	0.290/0.450
2	0.292/0.452	0.294/0.454	0.293/0.454	0.292/0.452	0.292/0.452
4	0.295/0.456	0.295/0.455	0.295/0.456	0.295/0.456	0.295/0.456
5	0.298/0.459	0.298/0.459	0.297/0.458	0.298/0.459	0.298/0.459
6	0.298/0.459	0.296/0.456	0.297/0.458	0.294/0.455	0.298/0.459
8	0.298/0.460	0.298/0.459	0.297/0.457	0.292/0.452	0.298/0.460
12	0.301/0.463	0.302/0.464	0.300/0.462	**0.304/0.466**	0.304/0.466
14	0.297/0.458	0.300/0.461	0.295/0.455	0.297/0.457	0.300/0.461
16	0.300/0.462	0.299/0.460	0.297/0.457	0.298/0.459	0.300/0.462
18	0.299/0.460	0.295/0.456	0.296/0.457	0.297/0.458	0.299/0.460
20	0.295/0.456	0.295/0.456	0.295/0.457	0.297/0.458	0.297/0.458
Max	0.301/0.463	0.302/0.464	0.300/0.462	0.304/0.466	0.304/0.466

Table 8: IOU score/F1 score on different models according to length of a sequence (frames/sequence)

According to the results, we saw that 12 frames/sequence is the most appropriate frequency to get the best possible performance, either more or less will negatively affect the performance of the model.

7 Discussion

The result that our team has generated after many weeks of researching is shown in the tables above. We can observe that it is not a very good way to segment a video into scenes that contain food or do not contain food. Table 9 shows us that the best IOU score is 0.304 using Bi-GRU. This means that the final result is better than the best result on baseline models 0.053 in IOU score (the best result on baseline models is 0.251 in IOU score using XGBoost Classifier).

Contrary to the hypothesised association, it seems that the place dataset does not reflect too much the target variable. In other words, using places to determine whether there is food in a frame is not a very good approach. Instead of using place features, using other features such as object or transcript may result in a better performance.

Based on the results and the data preparation step we can conclude that the performance of our model is strongly affected by the data. First of all, the tabular dataset that we were provided had been created using a pre-trained model. This model was trained to predict the ratios of places that appear in any frames of a video. However, the problem is that this model was not trained in the same domain as the videos in the dataset that we were given, which creates unfair predictions. Therefore, the accuracy of our model is affected considerably. Moreover, the annotation data that contain starts and ends of many scenes in many videos is labelled by students, who have never been trained before taking the task of labelling. As a result, this

annotation data lacks the accuracy of each scene, which leads to false results. These data-related problems together with our result prove the importance of using quality data.

On the other hand, we find that our model is facing the problem of overfitting. In spite of having spent time trying to use a lot of model structures as well as a lot of fine tunings, we cannot get rid of this issue. This problem, in theory, can be solved if we use a bigger amount of data. However, we lack the necessary facilities for the research, for instance GPU or RAM. Because this project was carried out using free Google Colab, we are limited to using more resources that we need. Therefore, this is the best result that we can create now.

8 Conclusion

In this work, we have managed to find a way to divide a video into segments which contain food. We first created data by making the annotation to mark where the start and the end of a segment containing food is. Then we used a provided pre-trained model to create a tabular dataset containing ratios of places that appear in a frame of a video. The method that we implemented was using this tabular dataset to classify a frame into one of two categories, food or not food. Then we concatenate them to create continuous segments. Because of the sequential characteristics, we have applied sequential models such as GRU, Bi-GRU, LSTM, Bi-LSTM and got the best performance using Bi-GRU. The results showed that using places to segment the video is not a very ideal method with the best IOU score 30.4%. In our current study, we have tried many ways to reduce the overfitting problem, but it seems that the situation has not improved very much. However, our approach might become a general method that can be applied on another feature that may result in better performance for example objects or scenes in a video. We hope that our work has implications for furthermore research in the future. Based on what we had analyzed in the data-preprocessing section and the fine tuning steps, we came to a conclusion that the data which we were provided is too bad because it had been trained on different domains of places. As a result, we think that what we will do in the future is to do research on models like MACNet to create the dataset from scratch. This model will be trained on the same domain between places. The second problem, which is also a big problem, is about models. Because of the fact that low quality data leads to bad models, if we have quality one, we can try using other lighter models with sequential characteristics, for instance SVM, HMM, etc.

9 Appendix

Beside of what you saw on report above, we also trying some assumption that we think it can help us obtain a better result but it turned out that it was not as good as we expect.

9.1 Add ReLU layer

After reviewing our architecture, we had a hypothesis that our model has been slightly affected by gradient vanishing, so we decided to add a ReLU layer between layers in sequence models (LSTM, GRU) and the last dense layer. We used 2 options of the number of units for this ReLu layer, one used 32 units, the latter used 64 units. We choose using in the model the data with sequence length of 12 frames/sequences, dropout rate equal to 0.7 and select 32 features in the dataset, which is the best

result in four previous models. The specific architecture of these models is in the Figure 5.

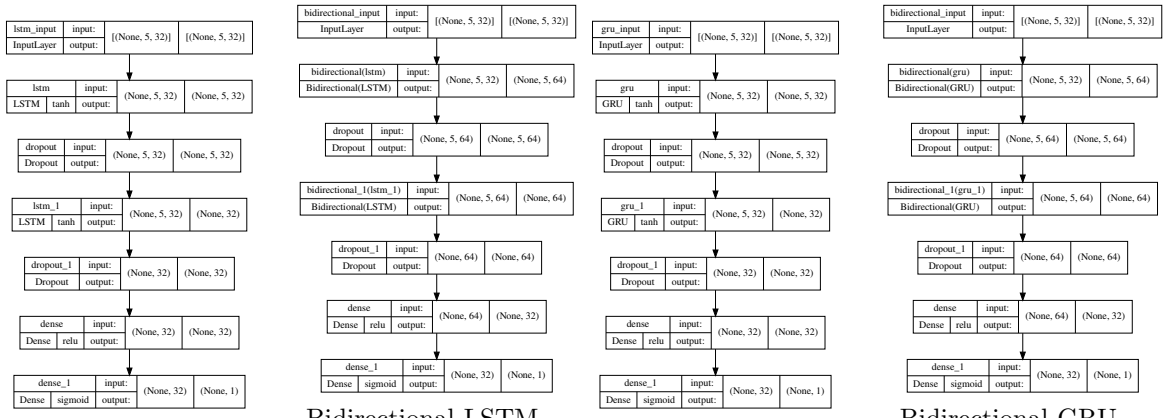


Figure 5: Model Architecture with ReLU

No. Units \ Model	LSTM	GRU	Bi-LSTM	Bi-GRU	Max
No ReLU	0.301	0.302	0.300	**0.304**	0.304
32 Unit	0.300	0.302	0.298	0.302	0.302
64 Unit	0.300	0.301	0.300	0.302	0.302
Max	0.301	0.302	0.300	0.304	0.304

Table 9: IOU score on different models using ReLU layer

Based on the result shown in Table 9, we could conclude that gradient vanishing did not occur here. We thought that it was because the LSTM, GRU had already handled the problem of gradient vanishing. Another possibility is that maybe ReLU could not reduce the gradient vanishing.

9.2 Re-Normalize Data

After having tried a lot of things that we could do with the model, we reviewed the dataset again and realized that the mean of all features is very low, like what we has plotted of the distribution in Figure 6 and distribution of values in Figure 1. This might cause the problem that the model could not have learnt anything. Therefore, we decided to standardize the data by Z-score method, which has the formula:

$$z_i = \frac{x_i - \overline{x}}{\sigma}$$

.

However, the result we obtained was actually not as good as we had expected. The best result of the models we showed in the Table 10 did not improve at all. We figured out that our assumption was wrong because if we re-normalize the data, it will break the dependency of features in each row, which is the sum of features in row is always equal to 1.

9.3 Using Accuracy Metric

At first, we did not use AUC as a metric for training. Instead of that, we used accuracy because we thought that it was enough to evaluate the model. Nevertheless,

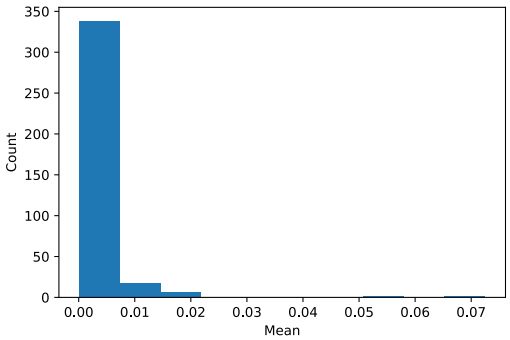


Figure 6: Distribution of mean values all class

Models	Precision		Recall		F1-Score		IOU
	0	1	0	1	0	1	
LSTM	0.84	0.35	0.60	0.65	0.70	0.46	0.298
GRU	0.84	0.35	0.59	0.66	0.70	0.46	0.298
LSTM	0.83	0.36	0.64	0.62	0.72	0.46	0.297
GRU	0.84	0.36	0.61	0.65	0.71	0.46	0.300

Table 10: Performance of best sequence models with Normalize Data

the result is actually bad. This was one of the reasons that we changed the metric to AUC. The best result that we obtained by Accuracy metric was shown in Table 9.

Models	Precision		Recall		F1-Score		IOU
	0	1	0	1	0	1	
LSTM	0.85	0.31	0.42	0.78	0.56	0.44	**0.284**
GRU	0.87	0.30	0.33	0.85	0.48	0.44	0.281

Table 11: Performance of best sequence models with Accuracy metric

10 References

[1] Le, Yen Hollenhorst, Steven Triplett, Jay. (2005). Business Perspectives of Adopting Sustainable Tourism Practices: A Study of Tourism Companies in Vietnam.

[2] Phuong, H. (2017, April 11). Vietnamese cuisine in impression of foreigners

[3] Thuan, Chi, Trung. (2019). Factors affecting international tourists' satisfaction of street food in Ho Chi Minh City.

[4] Henderson, J.C. (2019). Street Food and Tourism: A Southeast Asian Perspective. In: Park, E., Kim, S., Yeoman, I. (eds) Food Tourism in Asia. Perspectives on Asian Tourism. Springer, Singapore.

[5] Ed Burns. (2019). What is artificial intelligence (AI)?

[6] Arslan, B., Gönültaş, S., Gökmen, E. et al. Does YouTube include high-quality resources for training on laparoscopic and robotic radical prostatectomy?. World J Urol 38, 1195–1199 (2020).

[7] Sarker, M. M. K., Rashwan, H. A., Talavera, E., Banu, S. F., Radeva, P., Puig, D. (2019). MACNet: Multi-scale Atrous Convolution Networks for Food Places Classification in Egocentric Photo-Streams. Computer Vision – ECCV 2018 Workshops, 423–433. doi:10.1007/978-3-030-11021-5$_2$6.

[8] Hochreiter, S. and Schmidhuber, J., 1997. Long Short-Term Memory. Neural Computation, 98, pp.1735-1780. doi: 10.1162/neco.1997.9.8.1735

[9] Kitamura, K., Yamasaki, T., Aizawa, K. (2008). Food log by analyzing food images. Proceeding of the 16th ACM International Conference on Multimedia - MM '08. doi:10.1145/1459359.1459548

[10] Yu Cheng, Quanfu Fan, Sharath Pankanti IBM T.J. Watson Research Center, Alok Choudhary EECS Department, Northwestern University

[11] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[12] Neuhofer, B., Magnus, B. Celuch, K. The impact of artificial intelligence on event experiences: a scenario technique approach. Electron Markets 31, 601–617 (2021).

[13] Sidiropoulos, P., Mezaris, V., Kompatsiaris, I., Meinedo, H., Bugalho, M. and Trancoso, I., 2011. Temporal Video Segmentation to Scenes Using High-Level Audiovisual Features. IEEE Transactions on Circuits and Systems for Video Technology, 21(8), pp.1163-1177. doi:10.1109/TCSVT.2011.2138830.

[14] Bansal, S., Khandelwal, S., Gupta, S. and Goyal, D., 2013. Kitchen activity recognition based on scene context. 2013 IEEE International Conference on Image Processing, doi: 10.1109/ICIP.2013.6738714

[15] Friedman, J., 2002. Stochastic gradient boosting. Computational Statistics amp; Data Analysis, 38(4), pp.367-378. doi: 10.1016/S0167-9473(01)00065-2

[16] Ali, Jehad Khan, Rehanullah Ahmad, Nasir Maqsood, Imran. (2012). Random Forests and Decision Trees. International Journal of Computer Science Issues(IJCSI). 9.

[17] Chen, Tianqi Guestrin, Carlos. (2016). XGBoost: A Scalable Tree Boosting System. 785-794. 10.1145/2939672.2939785.

[18] Padilla, Rafael Netto, Sergio da Silva, Eduardo. (2020). A Survey on Performance Metrics for Object-Detection Algorithms. 10.1109/IWSSIP48289.2020.

[19] D. Kingma and J. Ba, Adam: A method for stochastic optimization, 2014. 10.48550/arXiv.1412.6980.

[20] Yu, Y., Si, X., Hu, C. and Zhang, J., 2019. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. Neural Computation, 31(7), pp.1235-1270.

[21] Dey, R. and Salem, F., 2017. Gate-variants of Gated Recurrent Unit (GRU) neural networks. 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)