Recipe Enrichment SMAD.ai
Project report for NLP Course, Winter 2022

Maciej Chrabaszcz
Warsaw University of Technology
maciej.chrabaszcz.stud@pw.edu.pl

supervisor: Anna Wróblewska
Warsaw University of Technology
anna.wroblewska1@pw.edu.pl

Aleksander Kozłowski
Warsaw University of Technology
aleksander.kozlowski.stud@pw.edu.pl

Abstract

The classification of dietary tags on ingredient lists is an important task in the food industry, as it allows for the accurate labelling and marketing of products to consumers with specific dietary needs or preferences. However, current methods for classifying these tags are often manual and time-consuming, leading to a research gap in the efficient and accurate classification of dietary tags.

To address this gap, the proposed work aims to develop a machine-learning model for the automatic classification of dietary tags on ingredient lists. This model will be trained on a large dataset of ingredient lists with annotated dietary tags, allowing for the accurate prediction of tags for new ingredient lists.

1 Introduction

In the food industry, the accurate labelling and marketing of products to consumers with specific dietary needs or preferences is crucial. However, previous methods for classifying dietary tags on ingredient lists, such as vegetarian, vegan, gluten-free, and nut-free, were often manual and time-consuming. This created a research gap in the efficient and accurate classification of these tags.

The scientific goal of this project was to develop a machine-learning model for the automatic classification of dietary tags on ingredient lists. While we did not have our own dataset of ingredient lists, we used a dataset which contained recipes from food.com which contained website tags and lists of ingredients. We then trained models on food ingredients and tags which we used to extract dietary tags. This allowed us to accurately predict dietary tags for new ingredient lists. To achieve this goal, we used natural language processing techniques to classify those tags.

The proposed project is pioneering in its use of natural language processing for the classification of dietary tags, and the results of this project have the potential to significantly improve the efficiency and accuracy of dietary tag classification in the food industry. This would have benefited both websites, by reducing the time and resources spent on manual classification, and consumers, by ensuring the accuracy of dietary tag labels on recipes.

The rest of this report will first provide a review of the state of the art in dietary tag classification and the research gap addressed by this project. It will then describe the methods and results of the proposed machine learning model, including the performance of the model on a test dataset. Finally, it will discuss the implications and potential future applications of the project results.

2 Related work

Britto et al. (2020) addressed this research gap by proposing a cooking recipe multi-label classification approach for food restriction identification. They trained their model using a dataset of over 200,000 cooking recipes with annotated dietary tags and demonstrated that their approach was able to accurately predict the presence of various dietary tags.

We found *foodcom-recipes-with-search-terms-and-tags* dataset on Kaggle which we used to create our models.

3 Approach & research methodology

The approach and research methodology for this project involved the development and training of a machine-learning model for the automatic classification of dietary tags on ingredient lists. To achieve this goal, we followed the following steps:

- Data collection: We collected a dataset of recipe ingredient lists and associated dietary tags from a website that provided such tags for a large number of recipes.
- Data preprocessing: We preprocessed the dataset by linking website tags with dietary tags.
- Model training: We fine-tuned pre-trained language models on the preprocessed dataset, using the ingredient lists as input and the associated dietary tags as labels.
- Model evaluation: We evaluated the performance of the model on a test dataset, measuring its ability to accurately predict the presence of various dietary tags.

We also tried to classify dietary tags using classical models in additional steps:

- Feature extraction: We used extracted ingredient names from the dataset and used the term frequency-inverse document frequency (TF-IDF) as their representation.
- Classical ML model training: We trained classical machine learning models (such as support vector machines, random forests and LightGBM) on the extracted features using the TF-IDF transformation.

To perform these tasks, we used a range of research tools, including Python libraries for natural language processing and machine learning (such as hugging face, scikit-learn, weights and biases etc.).

Overall, our approach and research methodology allowed us to effectively train and evaluate machine learning models for the classification of dietary tags on ingredient lists, using both advanced deep learning techniques and classical machine learning approaches.

4 Experiments and Results

Our dataset contained 0.5mln of recipes which is twice as big as previous work related to dietary tags classification.

4.1 Language models on ingredients text

In our experiments, we trained language models on a dataset of recipe ingredient lists and associated dietary tags. The goal of these experiments was to determine the performance of the models in accurately predicting the presence of various dietary tags.

We used a variety of evaluation metrics, including accuracy and F1 score, to measure the performance of the models on a validation dataset. The results of these experiments are shown in table 1

Dietary Tag	Accuracy	F1 Score
Dairy	0.84	0.11
Low-Calorie	0.81	0.27
Low-Carb	0.78	0.28
Low-Fat	0.89	0.13
Meat	0.88	0.77
Nuts	0.94	0.02
Plant-Based	0.83	0.16
Seafood	0.96	0.46
Sweet	0.91	0.50

Table 1: Results for fine-tuned BERT model on lists of ingredients as text.

Overall, our experiments and results demonstrated the potential of language models for classifying dietary tags on ingredient lists. However, further improvements may be necessary to achieve higher accuracy and F1 scores for particular tags.

4.2 Classicial ML on product names

In our experiments, we also trained a lightGBM model on the ingredient names of our recipe dataset, using the term frequency-inverse document frequency (TF-IDF) transformation to extract relevant features. The goal of these experiments was to determine the model's performance in accurately predicting the presence of various dietary tags.

We evaluated the lightGBM model's performance using various metrics, including accuracy and F1 score. The results of these experiments are shown in the table 2.

Overall, the lightGBM model achieved relatively high accuracy scores for most dietary tags, except for nuts and dairy, for which the scores were lower. The F1 scores for these tags were also lower, indicating that the model had difficulty accurately predicting their presence.

Despite these challenges, the lightGBM model showed promising performance in classifying dietary tags on ingredient lists. Further improvements may be necessary to achieve higher accuracy and F1 scores for specific tags.

Dietary Tag	Accuracy	F1 Score
Dairy	0.77	0.52
Low-Calorie	0.78	0.59
Low-Carb	0.76	0.61
Low-Fat	0.83	0.54
Meat	0.94	0.91
Nuts	0.83	0.39
Plant-Based	0.78	0.58
Seafood	0.99	0.93
Sweet	0.86	0.63

Table 2: Results for LightGBM on ingredients names and TF-IDF

5 Discussion

In our experiments, we trained language models and a lightGBM model on a dataset of recipe ingredient lists and associated dietary tags. The goal of these experiments was to determine the performance of the models in accurately predicting the presence of various dietary tags.

Overall, both approaches showed promising results for the classification of dietary tags. The language models achieved relatively high accuracy scores for most tags, except for dairy and nuts, for which the scores were lower. The F1 scores for these tags were also lower, indicating that the models had difficulty accurately predicting their presence.

Similarly, the lightGBM model achieved relatively high accuracy scores for most dietary tags, except nuts and dairy, for which the scores were lower. The F1 scores for these tags were also lower, indicating that the model had difficulty accurately predicting their presence.

Compared to the state-of-the-art results in the literature, our results are generally consistent with Britto et al. (2020), who used a multi-label classification approach to identify food restrictions in cooking recipes. Like our study, they found certain dietary tags more challenging to predict accurately than others.

Overall, our results suggest that both language models and the lightGBM model can potentially classify dietary tags on ingredient lists. Although in our setting, lightGBM, without a doubt, beat the pre-trained language model. Further improvements may be necessary to achieve higher accuracy, especially F1 scores for specific tags, and additional research is needed to explore the use of these models for this task.

6 Conclusions and future work

In this project, we aimed to classify dietary tags on ingredient lists in recipes. To accomplish this, we trained both language models and a lightGBM model on a dataset of recipe ingredient lists and associated dietary tags.

Our results showed that both approaches had a promising performance for classifying dietary tags. Further improvements may be necessary to achieve higher accuracy and F1 scores for specific titles, and additional research is needed to explore the use of these models for this task.

In future work, we plan to use pre-trained NER models for product name extraction and writing heuristics to eliminate common errors. The NER model will allow us to create lists of ingredients from recipes automatically. We also plan to add nutritional values to the output of our pipeline, which will use FoodOn API. Still, because this API returns many products without match scores, we will have to use Entity-linking, allowing us to extract proper product nutritional values from their database.

7 Work Division

Name	Work
Maciej Chrabaszcz	research, fine-tuning language models, website tags preprocessing, writing report
Aleksander Kozłowski	research, classical ML training, FoodOn API extraction, writing report

References

Bien, M., Gilski, M., Maciejewska, M., Taisner, W., Wisniewski, D., & Lawrynowicz, A. (2020). RecipeNLG: A Cooking Recipes Dataset for Semi-Structured Text Generation.

Ania, W., Agnieszka, K., Maciej, P., Dawid, W., Witold, S., Agnieszka, Ł. (2022). TASTEset – Recipe Dataset and Food Entities Recognition Benchmark. https://doi.org/10.48550/arxiv.2204.07775

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). Unsupervised Cross-lingual Representation Learning at Scale. https://doi.org/10.18653/v1/2020.acl-main.747

Britto, L., Pacífico, L., Oliveira, E., & Ludermir, T. (2020). A Cooking Recipe Multi-Label Classifica-

tion Approach for Food Restriction Identification. https://doi.org/10.5753/eniac.2020.12133

Nirav, D., Devansh, B., & Ganesh, B. (2020). A Named Entity Based Approach to Model Recipes. https://doi.org/10.48550/arxiv.2004.12184

Howard, Jeremy, and Sebastian Ruder. "Universal language model fine-tuning for text classification." arXiv preprint arXiv:1801.06146 (2018).

Li, Shuyang & Li, Yufei & Ni, Jianmo & McAuley, Julian. (2021). SHARE: a System for Hierarchical Assistive Recipe Editing.

FoodOn: A farm to fork ontology https://foodon.org/

https://www.kaggle.com/datasets/shuyangli94/foodcom-recipes-with-search-terms-and-tags

A Experimental setup

A.1 Language model

The language model which we selected was *bert-base-uncased*. Because of the amount of data (0.5mln examples) and our equipment limitations, we could only train the classification head without fine-tuning the whole BERT model. Our model was trained with a batch size of 64 for 50 000 steps. We used AdamW optimizer with a learning rate set to $1e-3$. We experimented with other models and learning rate schedules, but those changes gave no gain or even made model training unstable, which led to poor performance.

A.2 Classical ML

We saw that results from language models are unsatisfactory, and because of that, we wanted to test classical language models. Using them on pure texts of lists of ingredients also leads to poor performance and high demand in memory (TF-IDF returned sparse matrix with many columns ≈ 80000). Because of that, we tried to lower amount of unique words in the input data. We noticed that we were given lists of ingredients without the additional text in the dataset. We used it as our input to TF-IDF, which led to a massive reducement in the shape of a sparse matrix (≈ 4000 columns). With this trick, we were able to train models which didn't overfit on training data and showed promising results. We tested many classical models on this TF-IDF transformation (logistic regression, SVM, random forest, XGBoost, LightGBM etc.). Finally, we selected LightGBM as our primary model because of its performance and quick training time. Indeed, creating a more extensive dataset could improve these models, mainly because most dietary tags are highly unbalanced. With the help of NER, we could extract ingredients from recipes and classify new recipes for their dietary tags.

B Pre-trained NER model

At the beginning of our project, we found a model pre-trained on datasets from Nirav, D. (2020). This model showed promising results, which we want to address in our second project and could also help extract product lists from recipes.

C FoodOn API

We can query FoodOn API, but the results from this query are unsatisfaction. We will also focus on extracting the good product from the list of products returned by the API using Entity linking to give more nutritional values/features for models to predict dietary tags.