Exploiting Food Embeddings for Ingredient Substitution

Chantal Pellegrini[a,*], Ege Özsoy[b,*], Monika Wintergerst[c,*] and Georg Groh[d]
Department of Informatics, Technical University of Munich, Boltzmannstr. 3, 85748 Garching, Germany

Keywords: Food Substitution, BERT, Word2Vec, Word Embeddings.

Abstract: Identifying ingredient substitutes for cooking recipes can be beneficial for various goals, such as nutrient optimization or avoiding allergens. Natural language processing (NLP) techniques can be valuable tools to make use of the vast cooking-related knowledge available online, and aid in finding ingredient alternatives. Despite previous approaches to identify ingredient substitutes, there is still a lack of research in this area regarding the most recent developments in the field of NLP. On top of that, a lack of standardized evaluation metrics makes comparing approaches difficult. In this paper, we present two models for ingredient embeddings, Food2Vec and FoodBERT. In addition, we combine both approaches with images, resulting in two multimodal representation models. FoodBERT is furthermore used for relation extraction. We conduct a ground truth based evaluation for all approaches, as well as a human evaluation. The comparison shows that FoodBERT, and especially the multimodal version, is best suited for substitute recommendations in dietary use cases.

1 INTRODUCTION

In the light of rising rates of non-communicable diseases such as type 2 diabetes, and with an unhealthy diet being one of the leading health risks globally, healthy eating has become increasingly important (World Health Organization, 2020). However, changing one's diet can be challenging as eating habits are hard to break. This is why the substitution of food items is a promising approach to foster healthy nutrition. Substituting ingredients in a cooking recipe for healthier alternatives can, for instance, help to optimize the meal's nutrient profile to meet individuals' dietary needs, while at the same time preserving the dish's culinary attributes. Thus, familiar meals can be enjoyed while approaching dietary aims in manageable steps. Apart from replacing ingredients for nutrient optimization, food substitutes can also be applied for other goals such as avoiding allergens or adapting dishes to dietary preferences.

A promising way to find food substitutes is to leverage the vast amounts of (mostly textual) cooking-related data online to draw conclusions about which food items are interchangeable. Thus, natural language processing (NLP) can be used to identify generally applicable ingredient substitutes. We refer to these replacements as context-free, meaning independent of the direct recipe context. The food items identified as fitting substitutes for an ingredient should, therefore, be suitable replacements in several cases, and lay the foundation for all substitution use cases mentioned above.

While there has been previous research in the NLP domain that explored this task, there are few generally applicable approaches that use neural embedding models to generate ingredient substitutes, despite these models' success on other NLP tasks. Notably, there is no approach to our knowledge that applies the most recent NLP-powerhouses, transformer-based models like BERT (Devlin et al., 2018), to the task of substitute generation.

In this paper, we propose and compare several approaches for context-free ingredient substitute generation. We train two models, word2vec (Mikolov et al., 2013) and BERT (Devlin et al., 2018), on recipe instructions from the Recipe1M+ dataset (Marin et al., 2019) to compute meaningful ingredient embeddings. For BERT, we start from a pre-trained checkpoint. We refer to these models as Food2Vec and FoodBERT. Additionally, we combine these text-only approaches with an image-based approach resulting in multimodal ingredient representations. We also use FoodBERT to perform relation extraction on recipe

[a] https://orcid.org/0000-0002-7555-0336
[b] https://orcid.org/0000-0003-2841-3838
[c] https://orcid.org/0000-0002-9244-5431
[d] https://orcid.org/0000-0002-5942-2297
*These authors contributed equally.

Pellegrini, C., Özsoy, E., Wintergerst, M. and Groh, G.
Exploiting Food Embeddings for Ingredient Substitution.
DOI: 10.5220/0010202000670077
In *Proceedings of the 14th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2021) - Volume 5: HEALTHINF*, pages 67-77
ISBN: 978-989-758-490-9

comment data to extract substitute pairs. Furthermore, we propose an approach for a ground truth based evaluation for all methods and conduct a human evaluation. Our code and the trained models are available on GitHub[1].

The remainder of the paper is organized as follows: Section 2 presents related work, and Section 3 provides the theoretical background. Section 4 introduces the used dataset. The approaches are explained in Section 5, and their results on ingredient substitution are presented and discussed in Section 6. Section 7 concludes the paper and proposes possible future work.

2 RELATED WORK

This section presents previous research in the area of NLP that has explored the task of substitute generation, as well as different approaches to evaluating such methods.

Some approaches have detected food substitutes in recipe websites' user comments, for instance by identifying segments with alteration suggestions (Druck and Pang, 2012), or through relation extraction (Wiegand et al., 2012a; Reiplinger et al., 2014). (Teng et al., 2012) used pattern extraction results to build an ingredient substitute network. Other researchers have focused on recipe texts and used statistical methods to find food substitutes (Shidochi et al., 2009; Boscarino et al., 2014; Yamanishi et al., 2015). (Achananuparp and Weber, 2016) identified substitutes from food diary data with food-context matrices. The authors refer to the distributional hypothesis, theorizing that ingredients that occur in similar contexts tend to be similar. Other researchers have followed this concept with neural embedding techniques. Some approaches employ vector arithmetic on word embeddings to find substitutes via ingredient analogies. They are designed for special use cases, such as "veganizing" meals (Lawo et al., 2020) or adapting recipes to different cuisines (Kazama et al., 2018). There is a lack of research on general-purpose ingredient embeddings for food substitution, as the general approaches so far have been rather exploratory and without in-depth evaluation (Hinds, 2016; Altosaar, 2017; Sauer et al., 2017). Additionally, there is no approach to our knowledge that employs transformer-based models like BERT (Devlin et al., 2018).

A big challenge for food substitution is the absence of a standard for comparing the results of different methods. Many approaches so far relied solely on human evaluation on a relatively small scale, either by cooking and testing recipes with substitutions (Kazama et al., 2018; Yamanishi et al., 2015) or through qualitative analysis of recipes or substitute pair examples (Shidochi et al., 2009; Sauer et al., 2017; Lawo et al., 2020). In contrast, (Achananuparp and Weber, 2016) had crowd workers rate 2,000 substitute pairs. The authors were then able to employ quantitative evaluation metrics. In an endeavor to create a gold standard for various food relations, (Wiegand et al., 2012b) found that two subjects' lists of substitute pairs had little overlap, indicating that finding a consensus on suitable substitutes is challenging. The authors used their gold standard in subsequent research to evaluate their approaches quantitatively (Wiegand et al., 2012a). (Reiplinger et al., 2014) created a gold standard for food relations by manually labeling sampled sentences from their domain-specific corpus. However, the above gold standards are both in German and in include terms specific to the German cuisine, such as "Maultaschen", a German kind of dumpling. Therefore, it is difficult to apply them to approaches in other languages, which in turn might require additional terms in their vocabulary to reflect the respective cuisines.

Given the lack of exploration of neural embedding models for ingredient substitution and the challenges concerning evaluation, we propose several learning-based approaches for substitute generation and conduct both a ground truth based and a human evaluation.

3 BACKGROUND

In the previous section, we observed that a concept that has been employed in many substitution approaches is the distributional hypothesis, which states that words occurring in similar contexts tend to be similar in meaning. It was first explored in linguistics (Harris, 1954), and later adopted in the area of computational linguistics (Rubenstein and Goodenough, 1965; Rieger, 1991). In recent years and with the surge of neural networks, neural approaches to capture this semantic similarity have been developed. Two of the most popular approaches are word2vec (Mikolov et al., 2013) and BERT (Devlin et al., 2018).

Word2vec (Mikolov et al., 2013) is an approach to compute continuous vector representations of words given a large corpus of training text. Training a word2vec model results in vectors representing the words in the corpus, with semantically similar words being closer to one another in the vector space. Each token is represented by exactly one vector, meaning

[1]https://github.com/ChantalMP/Exploiting-Food-Embeddings-for-Ingredient-Substitution

the embeddings are context-free.

BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018) is a transformer-based language model pre-trained on a large amount of textual data, which can be fine-tuned for more specific tasks. The training is based on two tasks, masked language modeling and next sentence prediction, and results in contextualized embeddings. The authors show that BERT has achieved state-of-the-art results on various NLP tasks.

One of these applications, R-BERT (Wu and He, 2019), employs BERT for relation extraction and is among the best methods for this task. The authors tagged two possible entities in a relation and had the model predict the relation between them.

In addition to fine-tuning the pre-trained BERT models, success was also achieved by pre-training BERT on domain-specific data for science or biology (Beltagy et al., 2019; Lee et al., 2020). This is especially useful if the target domain contains many words rarely encountered in common texts.

4 DATA

We use Recipe1M+ (Marin et al., 2019), a dataset containing one million recipes with information like title, ingredient list, instructions, and images. We specifically use 10,660,722 instruction sentences for training our models. We also extend this dataset by scraping recipe comments and normalize the content.

4.1 Comment Scraping

Recipe1M+ provides URLs to the original recipe pages for all recipes. We scrape accessible user comments for the most frequent web pages. A part of this data was already provided to us in a master's thesis by (Engstler, 2020). In total, our data comprises 1,525,545 comments.

4.2 Normalization

As Recipe1M+ (Marin et al., 2019) does not include a clean list of all ingredients, we use an ingredient dataset provided by Yummly (Yummly, 2015) as the starting point to create such a list. We clean this set by deleting terms in brackets and discarding ingredients with more than three words, as these generally include unneeded names of a brand or location, such as "Nakano Seasoned Rice Vinegar" or "Spice Islands Chili Powder". To match ingredients from the Yummly dataset with Recipe1M+, we lemmatize all nouns in the ingredient list. We do not alter other words like verbs as these are usually constant, and lemmatizing them can lead to wrong matchings. For instance, "baked" in "baked potatoes" would be normalized to "bake", thus prohibiting a differentiation between instruction and ingredient part. The same lemmatization is applied to the instruction and comment sentences. We also combine multi-word ingredients with an underscore ("ice cream" becomes "ice_cream") and discard ingredients that occur less than ten times in the whole Recipe1M+ dataset. Additionally, we manually delete non-food items from the ingredient list. After these steps, the ingredient list contains 4,372 ingredients.

5 METHODS FOR SUBSTITUTE GENERATION

In this section, we present several approaches that can be used for context-free ingredient substitute generation.

5.1 Pattern Extraction

One previously applied method for food substitute extraction (Wiegand et al., 2012a; Teng et al., 2012), which we use as a baseline for our work, is to use user comments from recipe sites since a good portion of them mention how to replace certain ingredients with others (Druck and Pang, 2012). We use *Spacy* (Honnibal and Montani, 2017) to extract these recommendations by applying the following hand-crafted patterns previously presented by (Engstler, 2020), where A and B reference ingredients:

- $\{replace \mid substitute\}$ A $\{with \mid for\}$ B
- A $\{instead\ of\}$ B

Instead of predicting every extracted pair as a substitute, only pairs of ingredient I and substitute S which occur at least t times will be predicted as substitutes. This way, precision can be traded off for recall.

5.2 Food2Vec

The context-free word representations created by word2vec (Mikolov et al., 2013) match our goal of context-free ingredient substitutes. Thus, we follow the idea first described in a blog post by Rob Hinds (Hinds, 2016) and adopt his experimental approach *food2vec* of training word2vec on recipe instructions. We train it on a bigger, pre-processed dataset and refine the process of substitute generation as explained below. Figure 1 shows an overview of our approach Food2Vec.

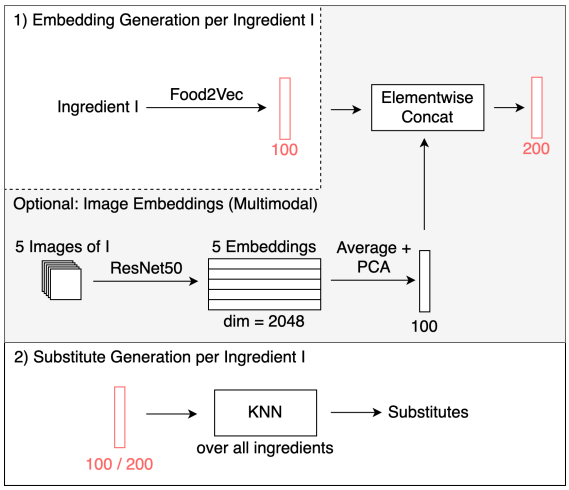


Figure 1: The Food2Vec approach is separated into two parts: The first part calculates text-based embeddings for all ingredients and optionally concatenates them with image-based embeddings. The second part uses these embeddings in conjunction with KNN to predict substitutes.

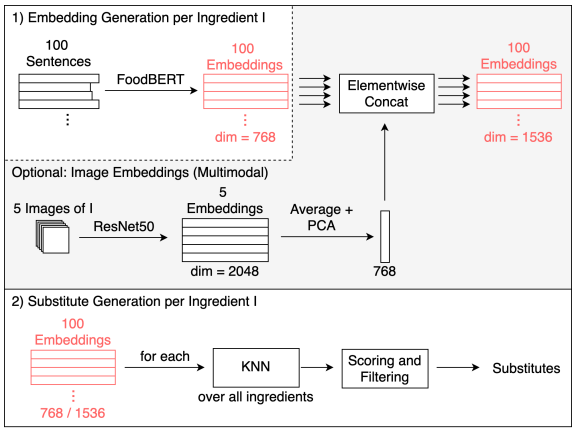


Figure 2: The FoodBERT approach is separated into two parts: The first part calculates text-based embeddings for up to 100 occurrences of every ingredient and optionally concatenates them with image-based embeddings. The second part employs these embeddings together with KNN and a further scoring and filtering step to predict substitutes.

5.2.1 Training

For creating Food2Vec, we trained the CBOW variant of word2vec on the normalized instructions of the Recipe1M+ (Marin et al., 2019) corpus. We use *Gensim* (Řehůřek and Sojka, 2010) for training with default settings, but set the min_count to 10 during training, as we only consider ingredients that occur at least ten times in Recipe1M+. This means that also non-ingredient words that occur less than 10 times will be ignored. These rare words usually do not provide substantial context and can be neglected. After completing the training, all our considered ingredients are represented by a 100-dimensional embedding.

5.2.2 Generating Substitutes

For generating substitutes, we search for the N nearest neighbors in the embedding space containing all ingredients. N functions as a threshold to balance the number of proposed substitutes and the model's precision. We filter the possible results by a handcrafted rule based on the assumption that for a given ingredient I, the substitute S should not be a specialization of that ingredient. For example, we do not want to substitute "chicken" with "chicken breast". To achieve this, we remove any potential substitute S that completely contains the original ingredient I. We refer to this approach as Food2Vec-Text.

5.3 FoodBERT

BERT achieves state-of-the-art results in many different NLP tasks (Devlin et al., 2018), making it a promising approach for generating substitutes. Arguably one of the main strengths of BERT is that it works with contextualized embeddings, meaning a word can have different embeddings depending on its context. This advantage can not be fully utilized for our task, as we make context-free predictions. Nonetheless, BERT's expressive power and language understanding can still be helpful. An overview of this approach is shown in Figure 2.

5.3.1 Modifying BERT for Recipes

The BERT version we use has about 29,000 tokens in its vocabulary but contains only about 3% of our ingredients. Even some common ingredients such as "onion" or "pasta" are not included. This means that those ingredients will be understood worse and split into multiple tokens, resulting in multiple model outputs for one ingredient. Average, min, or max pooling of multiple embeddings are possible ways to deal with this, but we eliminate it entirely by making sure no ingredient is split into multiple tokens. To achieve this, we extend the BERT vocabulary to include all of the 4,372 ingredients we mentioned in section 4. In total, we end up with 33,247 tokens in the vocabulary.

5.3.2 Training FoodBERT

We start with the pre-trained *bert-base-cased*, with extended vocabulary, and further train it on

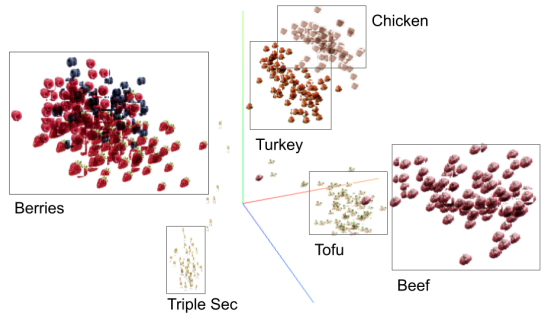


Figure 3: Embedding space visualization for ingredients: Raspberry, Blueberry, Strawberry, Tofu, Chicken, Turkey, Beef and Triple Sec. Each small image represents a contextualized embedding for one occurrence of the depicted ingredient. It can be seen that berries and meat-like products are each grouped together, while Triple Sec is far away from both groups.

Recipe1M+ (Marin et al., 2019) instructions for three epochs. We use *Hugging Face*'s *Transformers* library (Wolf et al., 2019) and the masked language modeling training script (Hugging Face, 2020), only adapting the BERTTokenizer to work with the extended vocabulary. This training took approximately three full days on an NVIDIA P-100 GPU.

5.3.3 Embedding Space Visualization

While BERT's hidden representation has a dimension of 768, it can still be projected into 3D space by using a dimensionality reduction algorithm such as principal component analysis (PCA). We visualize the embeddings for 100 occurrences of eight different ingredients. The visualization can be seen in Figure 3. It shows that the ingredients are separated by their type and use case. This is a valuable property, as the quality of the hidden representation of FoodBERT is directly correlated with the success of the approaches relying on it.

5.3.4 Generating Substitutes

We exploit the FoodBERT embedding space to generate substitutes. To this end, we first sample 100 random sentences for every ingredient (less if there are less than 100 occurrences) from the recipes in Recipe1M+. Afterward, a contextualized embedding with 768 dimensions is computed for every occurrence of every ingredient using FoodBERT. In total, we end up with ∼285,000 embeddings for all ingredients. For every embedding, we compute the 200 nearest neighbors using an approximate KNN (Bernhardsson, 2019). To calculate the substitutes for an ingredient I, we sum up how often every other ingredient appears in these 200 nearest neighbors for all the embeddings of the ingredient I. This amount of occurrences is assigned to every possible substitute as a score. Afterward, we sort the potential substitutes from the highest (most occurrences) to the lowest score and apply the same filtering as in Food2Vec. To get the final set of substitutes, we first require any potential substitute to at least have a score of 100. Second, we have a relative threshold that sets the minimum required score in relation to the highest score (HS) for the ingredient, e.g. $1/10$ of HS. This relative threshold is tuned to achieve different precision-recall trade-offs. We refer to this approach as FoodBERT-Text.

5.4 Multimodal Representations

We argue that a good indicator for two ingredients' interchangeability is their appearance. To utilize this additional knowledge, we propose using multimodal embeddings instead of pure text embeddings. Figure 1 and 2 show how we integrate the image embeddings in our methods.

5.4.1 Image Embeddings for Ingredients

For every ingredient, we download up to ten images from Google. Afterward, we perform a manual verification and cleanup to ensure we have exactly five correct images for every ingredient. For every image, we extract a 2,048 dimensional embedding from the last layer of a ResNet-50 (He et al., 2016), which is pre-trained on ImageNet (Deng et al., 2009). Then we average the embeddings of all five images to get one final image representation per ingredient.

5.4.2 Combining Image and Text Embeddings

As Food2Vec and FoodBERT embeddings have a dimension of 100 and 768 respectively, directly concatenating these image embeddings with the corresponding text embeddings would put more focus on the image embeddings. Therefore, we use PCA to decrease the image embedding size from 2,048 to 100 or 768 depending on the method, so that both embedding types have equal influence. We also normalize the image embeddings so that they have similar mean and variance as the text embeddings. Afterward, we concatenate the text-based and image-based ingredient representations to get one multimodal embedding of size 200 for Food2Vec or 1,536 for FoodBERT. This new ingredient representation is then used like the text-only embeddings described in the respective substitute generation sections of Food2Vec and FoodBERT. Empirically, we achieve better results when intersecting the predicted substitutes of our multimodal

approach with the predicted substitutes of the corresponding text-only approach. In this setting, an ingredient will only be proposed as a substitute if both the text-only and multimodal methods predict it. These intersected results are what we refer to as Food2Vec-Multimodal and FoodBERT-Multimodal.

5.5 Relation Extraction

Handcrafted patterns are not optimal when it comes to extracting the information from user comments. These patterns do not capture all substitute recommendations and also make mistakes as they are not context-specific. We, therefore, propose a learned and context-specific relation extraction method, which is based on FoodBERT.

5.5.1 Data Preparation

We extract pairs of mentioned ingredients for all the sentences in the user comments by matching the words to our ingredient list. For all pairs that occur together in one sentence, we mark the ingredients' beginning and end with unique characters:

"I used plain $ yogurt $ in place of the £ sour_cream £ and it is delicious."

This modified sentence is then added to our relation extraction dataset. To generate labels for training, 1,000 frequent ingredient pairs are extracted and labeled with two labels. The first label denotes if the first ingredient can be substituted by the second, the second label vice versa. We gather all 1.3 million sentences which include these frequent pairs. To further focus the dataset on relation extraction, we only use sentences which include one of the following expressions: "instead", "substitute", "in place of" or "replace". In this way, we bias our data towards comments which contain user substitute recommendations. After this step, we end up with approximately 170,000 sentences, which we use to train and evaluate our relation extraction model.

5.5.2 Modifying R-BERT

We base our implementation on R-BERT (Wu and He, 2019), which is one of the state-of-the-art relation extraction methods based on BERT. We differ in our implementation mainly in four places. First, unlike R-BERT, we do not have entities that span multiple tokens, so we do not need to average the representation over multiple embeddings. Instead, we directly use the embedding for an ingredient token. Second, we use $ and £ symbols, as underscore is reserved for multi-word ingredients. Third, we use FoodBERT instead of BERT in the backbone of the model. Finally, we adapt the classification layer to support our desired output format.

5.5.3 Model Architecture and Training

A marked sentence is given to FoodBERT, with the first token being the BERT specific classifier token. From the computed hidden state vector H, three parts are used further, namely the pooled output H_0, which represents the whole sentence, and embeddings corresponding to the first and second ingredient. The three representations are concatenated and fed into a fully connected layer which reduces the representation size to two. These two features correspond to the two labels described previously (Ingredient 1 can be substituted by 2 or vice versa). To get binary labels, a sigmoid is applied, and the output is thresholded at 0.5.

We train our network for ten epochs with a learning rate of 10^{-5} and AdamW (Loshchilov and Hutter, 2019) as the optimizer.

5.5.4 Substitute Generation

For generating a final list of substitutes, we use all of the comment data, also the part without labels, but still limit ourselves to sentences that contain substitute-relation indicating expressions. We only consider ingredient pairs that occur more than t times together. This threshold t can be tuned to achieve different precision-recall trade-offs. We then use our model to predict labels for all pairs and add a pair to our substitute list if more than 10% of the predicted labels are positive.

6 EXPERIMENTS

In this section, we present the effects of fine-tuning BERT for the food domain and the results for the task of substitute generation of all presented methods.

6.1 Evaluation Metrics

Our evaluation is twofold. First, we perform a human evaluation, where we label the correctness of generated substitute pairs. Additionally, we created a list of ground truth substitutes for a subset of ingredients, which we use for automatic evaluation. For creating this ground truth set as well as for the human evaluation, we consider an ingredient as substitute S for another ingredient I if we can think of several dishes where I can be substituted by S. The labeling was performed by two authors with amateur cooking experience.

6.1.1 Human Evaluation

We randomly sample 200 ingredient-substitute pairs (I, S) per method and label them regarding their correctness. We sample these pairs from two portions of our ingredient set. For 100 pairs, we only sample from substitute recommendations where the ingredient I is among the 1,000 most common ingredients in the recipe dataset. For the other 100 pairs, we sample from all recommendations. A substitute S can, in both cases, be from the whole ingredient set. Two authors labeled these pairs independently to get a more reliable evaluation, as we are dealing with an ambiguous task. The inter-rater reliability using Cohen's Kappa lies at 0.72, indicating a substantial reliability (Landis and Koch, 1977). Given the labels on these two sets, we compute the overall precision and precision on common ingredients for all methods. As we have two annotators, we compute the precision scores separately for both sets of annotations and then average them to get the final precision values.

6.1.2 Ground Truth based Evaluation

In comparison to human evaluation, an evaluation on a ground truth set can measure not only precision but also recall. The challenge in measuring recall is that it requires an extensive list of all possible substitutes for every ingredient. As this is unfeasible, we decided to create such a list only for a subset of 42 ingredients. Seven of these are not among the 1,000 most common ingredients. We started creating the ground truth set with a selection of 42 ingredients and some of their substitutes recommended in the Food Substitutions Bible (Joachim, 2010). We continuously extended the set by manually adding any correct predictions of our approaches if they were missing in the ground truth. In the end, we evaluated all approaches with the final ground truth set. This set contains on average 16.9 substitutes per ingredient and 708 unique substitute pairs in total.

6.2 Ablation Study on FoodBERT

Since this is to our knowledge the first attempt to fine-tune BERT for ingredient embeddings, we wanted to better understand the benefit of fine-tuning BERT for the food domain. Therefore, we performed an ablation study by replacing FoodBERT with the pre-trained BERT model *bert-base-cased* from *Hugging Face* (Wolf et al., 2019) for the two methods based on FoodBERT. The results of these experiments on the ground truth dataset are shown in Table 1.

It can be seen that fine-tuning on text from the food domain substantially improves performance. Directly using the basic BERT embeddings for substitute generation produces poor results with a precision of 0.151 compared to 0.806 and a lower recall. For relation extraction, the difference is smaller. BERT is able to learn the general task of relation extraction on this dataset, but the additional information about the food domain still increases the accuracy from 0.632 to 0.700 and the recall from 0.085 to 0.129. These results are not surprising, as the original BERT vocabulary only contains 137 ingredients from the 4,372 ingredients in our ingredient set. Moreover, several of these are ambiguous, like animals ("rabbit", "lamb") or body parts ("liver", "breast"), or are homonyms ("date").

Table 1: Precision and Recall of FoodBERT methods compared to the same methods using bert-base-cased.

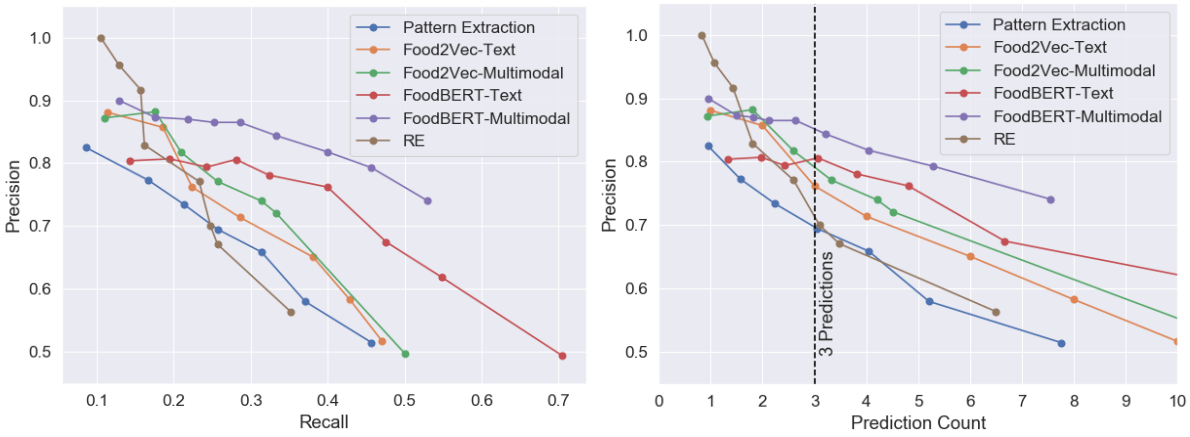
	Prec.	Recall
FoodBERT-Text	0.806	0.147
BERT	0.151	0.032
FoodBERT Relation Extraction	0.700	0.129
BERT Relation Extraction	0.632	0.085

6.3 Substitution Results

Here we present the quantitative results of our approaches. Some example substitute predictions can be found in the appendix in Table 5.

All presented approaches can be tuned for either higher precision or higher recall by setting a method-specific threshold value. Figure 4a shows the Precision-Recall curves of our approaches for different thresholds. Similarly, Figure 4b visualizes the precision in relation to the average number of predictions per ingredient. We can see in both cases that for a small recall or few predictions, Relation Extraction stands out with the highest precision, and FoodBERT-Text has the lowest precision together with Pattern Extraction. When looking at a higher recall or at least three predictions, FoodBERT shows better results than the rest, both on its own and in the multimodal setting. Relation Extraction's performance decreases sharply and becomes similar to Pattern Extraction, and the Food2Vec approaches are in between. Another observation from both curves is that combining text embeddings with image embeddings yields additional improvement.

For further evaluation, we consider that people have different food preferences and can have several medical conditions like allergies that forbid them from eating certain foods. In this context, it is vital to provide the user with some choice when recommending food substitutes. We therefore chose to evaluate a version of every approach in more detail,



(a) Precision - Recall Curve.

(b) Precision - Prediction Count Curve.

Figure 4: Precision in relation to recall or prediction count.

which at least predicts three substitutes on average for the ground truth ingredients. The detailed results for these versions can be found in Table 2.

The first thing to notice is that generally, learning-based methods perform better than the rule-based Pattern Extraction. We can further see the FoodBERT approaches, both the vanilla and the multimodal version, excel in our scenario of predicting at least three substitutes, with considerably better precision values of 0.806 and 0.844. The Food2Vec approaches follow with precision values around 0.77 and slightly worse recall. Relation Extraction and Pattern Extraction perform worst with a precision of around 0.70.

The overall low recall across all approaches can be explained by the average number of 16.9 substitutes per ingredient in our ground truth. The evaluated versions of the approaches make on average at least 3 predictions per ingredient, which makes it impossible to achieve a high recall, as the large number of ground truth substitutes per ingredient can not be predicted. Nevertheless, the differences in recall between the approaches still show that the multimodal approaches perform best with FoodBERT-Multimodal achieving the highest recall of 0.164. Pattern Extraction and Relation Extraction have the lowest recall, which once again indicates that they are not very suitable if several substitute predictions per ingredient are desired.

The human evaluation was performed using predictions from the same method versions, which make, on average, three predictions. The results are shown in Tables 3 and 4. In comparison to the ground truth based results in Table 2, Relation Extraction performs better, whereas Food2Vec approaches are in a worse position. In Table 3, we can see that approaches with fewer uncommon predictions achieve higher precision. At the same time, considering only common ingredients, meaning the 1,000 most frequent ones, leads to much fewer uncommon substitute predictions compared to the results of the overall case in Table 4. There, we see considerably more uncommon ingredients and substitute predictions, except for Pattern Extraction and Relation Extraction. Instead of a recipe corpus, these two approaches use user comments in which uncommon ingredients are rarely mentioned, thereby not surpassing the used threshold. Generally, suggesting fewer uncommon ingredients or, in other words, having limited creativity by being restricted to only the most common ingredients, can increase precision by ”playing it safe”. How much the creativity of the approach is worth against its precision can not be decided generally, as it depends strongly on the use case.

6.4 Discussion

Overall, the results are encouraging, as the approaches show good performance according to our evaluation. Nevertheless, accurately evaluating food substitutes remains a big challenge. Without a unified benchmark, it is hard to compare different approaches objectively. However, it is questionable whether the creation of an exhaustive substitute gold standard is even feasible, as culinary matters are subjective.

The ground truth dataset we created is biased because we used our own approaches to expand it. The unbiased alternative would have been to label all ingredient pairs (I, S), where I is from the 42 ingredients in the ground truth set, and S is from all 4,372 ingredients. This would have resulted in manually labeling 183,624 pairs, which was unfeasible. By labeling only pairs that any approach predicted, we obtain a correct precision value, but the recall is an upper-bound, as the ground truth might miss correct substitute pairs. Still, the relation between the recall values

Table 2: Precision, Recall, Top-5 Recall, average prediction count for the ground truth ingredients and total prediction count for all approaches. This shows the best versions of all approaches making at least 3 predictions per ingredient on average.

	Precision	Recall	Top-5 Recall	Avg Predictions	Total Predictions
Pattern Extraction	0.695	0.126	0.257	3.05	1238
Food2Vec-Text	0.762	0.136	0.224	3.0	13056
Food2Vec-Multimodal	0.771	0.153	0.257	3.33	5908
FoodBERT-Text	0.806	0.147	0.281	3.07	28385
FoodBERT-Multimodal	0.847	0.164	0.338	3.26	33941
Relation Extraction	0.700	0.129	0.248	3.10	1393

Table 3: Top1000: Precision of all approaches in human evaluation when considering only common ingredients for sampling. "S rare" is the approximate proportion of uncommon ingredients predicted as substitute S for pairs (I,S).

	Precision	S rare
Pattern Extraction	0.64	4%
Food2Vec-Text	0.51	41%
Food2Vec-Multimodal	0.63	39%
FoodBERT-Text	0.76	3%
FoodBERT-Multimodal	0.80	27%
Relation Extraction	0.79	8%

Table 4: Overall: Precision of all approaches in human evaluation when considering all ingredients for sampling. "I rare" is the approximate proportion of uncommon ingredients as ingredient I, "S rare" the proportion of uncommon ingredients predicted as substitute S for pairs (I,S).

	Prec.	I rare	S rare
Pattern Extraction	0.61	11%	4%
Food2Vec-Text	0.53	78%	86%
Food2Vec-Multimodal	0.66	47%	66%
FoodBERT-Text	0.60	90%	89%
FoodBERT-Multimodal	0.61	92%	91%
Relation Extraction	0.73	10%	8%

of different approaches is valid.

In our human evaluation, we randomly sampled ingredient-substitute pairs for each approach. This sampling may have been advantageous for one approach and less favorable in another method's case. However, using the same set of ingredients to evaluate every approach would carry the same risk of an unfair comparison, as some approaches might not have predicted a single substitute for certain ingredients.

Despite these challenges, we provide an overall impression of different approaches' strengths and weaknesses by considering several evaluation methods. For our envisioned use case of providing substitute recommendations while leaving room for personal choice, FoodBERT-Multimodal is the best approach, both according to the ground truth based evaluation and the Top1000 human evaluation. Of course, when considering a different use case, for instance, if one substitute suggestion per ingredient is sufficient, different methods can be preferable.

7 CONCLUSION AND FUTURE WORK

In this work, we presented several approaches for the task of context-free food substitute recommendation. We demonstrated that learning-based approaches perform very well on this task. The comparison of the approaches showed that in a dietary context, where it is beneficial to offer a variety of substitutes to choose from according to personal preferences, the exploitation of the FoodBERT embeddings and especially its multimodal enhancement perform best.

Our results suggest that FoodBERT's embedding space represents knowledge about food items much better than the embedding space of the original BERT model. This opens the possibility to apply our model to other food-related tasks like cuisine-prediction or estimating meals' healthiness. Another direction for future work could be the contextualized setting of food substitute recommendations, where the specific use case of an ingredient in a particular recipe is considered to find substitutes. Since BERT embeddings are contextualized, it is promising to apply it to this task. Also, since ImageNet is not food-focused, the multimodal model's understanding of food items could be improved by fine-tuning the image embedding model on food images.

While the models presented in this paper consider the usage context of ingredients in order to identify culinarily fitting alternatives, the selection process for substitutes can be extended to incorporate other factors as well, especially regarding healthy nutrition. With rules based on dietary guidelines, or even an individually tailored health metric considering personal nutrient requirements, the substitutes proposed by the models could be filtered or re-ranked, resulting in suggestions which are both culinarily adequate as well as conducive to health goals.

ACKNOWLEDGEMENTS

The preparation of this paper was supported by the enable Cluster and is catalogued by the enable Steering Committee as enable 065 (http://enable-cluster.de). This work was funded by a grant of the German Ministry for Education and Research (BMBF) FK 01EA1807A.

REFERENCES

Achananuparp, P. and Weber, I. (2016). Extracting food substitutes from food diary via distributional similarity. In *10th ACM Conference on Recommender Systems*.

Altosaar, J. (2017). food2vec - augmented cooking with machine intelligence. https://jaan.io/food2vec-augmented-cooking-machine-intelligence/. Accessed: 2020-09-05.

Beltagy, I., Lo, K., and Cohan, A. (2019). Scibert: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3606–3611.

Bernhardsson, E. (2019). Annoy. https://github.com/spotify/annoy. Accessed: 2020-07-04.

Boscarino, C., Nedović, V., Koenderink, N. J., and Top, J. L. (2014). Automatic extraction of ingredient's substitutes. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pages 559–564.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Druck, G. and Pang, B. (2012). Spice it up? mining refinements to online instructions from user generated content. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 545–553, Jeju Island, Korea. Association for Computational Linguistics.

Engstler, L. N. (2020). Ontology learning from text in the food domain. Unpublished master's thesis. Technical University of Munich, Munich, Germany.

Harris, Z. S. (1954). Distributional structure. *Word*, 10(2-3):146–162.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hinds, R. (2016). Unsupervised learning in scala using word2vec. https://automateddeveloper.blogspot.com/2016/10/unsupervised-learning-in-scala-using.html. Accessed: 2020-09-05.

Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Hugging Face (2020). Run language modeling. https://github.com/huggingface/transformers/blob/master/examples/language-modeling. Accessed: 2020-07-04.

Joachim, D. (2010). *The Food Substitutions Bible*. Robert Rose Inc., Toronto, Ontario, Canada, 2 edition.

Kazama, M., Sugimoto, M., Hosokawa, C., Matsushima, K., Varshney, L. R., and Ishikawa, Y. (2018). A neural network system for transformation of regional cuisine style. *Frontiers in ICT*, 5:14.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1).

Lawo, D., Böhm, L., and Esau, M. (2020). Supporting plant-based diets with ingredient2vec.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Marin, J., Biswas, A., Ofli, F., Hynes, N., Salvador, A., Aytar, Y., Weber, I., and Torralba, A. (2019). Recipe1m+: A dataset for learning cross-modal embeddings for cooking recipes and food images. *IEEE Trans. Pattern Anal. Mach. Intell.*

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Reiplinger, M., Wiegand, M., and Klakow, D. (2014). Relation extraction for the food domain without labeled training data–is distant supervision the best solution? In *International Conference on Natural Language Processing*, pages 345–357. Springer.

Rieger, B. B. (1991). *On distributed representation in word semantics*. International Computer Science Institute Berkeley, CA.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Sauer, C., Haigh, A., and Rachleff, J. (2017). Cooking up food embeddings.

Shidochi, Y., Takahashi, T., Ide, I., and Murase, H. (2009). Finding replaceable materials in cooking recipe texts considering characteristic cooking actions. In *Proceedings of the ACM multimedia 2009 workshop on*

Multimedia for cooking and eating activities, pages 9–14.

Teng, C.-Y., Lin, Y.-R., and Adamic, L. A. (2012). Recipe recommendation using ingredient networks. In *Proceedings of the 4th Annual ACM Web Science Conference*, pages 298–307.

Wiegand, M., Roth, B., and Klakow, D. (2012a). Web-based relation extraction for the food domain. In *International Conference on Application of Natural Language to Information Systems*, pages 222–227. Springer.

Wiegand, M., Roth, B., Lasarcyk, E., Köser, S., and Klakow, D. (2012b). A gold standard for relation extraction in the food domain. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 507–514.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., and Brew, J. (2019). Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

World Health Organization (2020). Healthy diet. https://www.who.int/news-room/fact-sheets/detail/healthy-diet. Accessed: 2020-09-05.

Wu, S. and He, Y. (2019). Enriching pre-trained language model with entity information for relation classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2361–2364.

Yamanishi, R., Shino, N., Nishihara, Y., Fukumoto, J., and Kaizaki, A. (2015). Alternative-ingredient recommendation based on co-occurrence relation on recipe database. *Procedia Computer Science*, 60:986–993.

Yummly (2015). What's cooking? https://www.kaggle.com/c/whats-cooking/overview. Accessed: 2020-07-04.

APPENDIX

Table 5: Example substitute predictions of all approaches.

potato	
Pattern Extraction	pasta, rice
Food2Vec-Text	beet, squash, turnip
Food2Vec-Multimodal	beet, brussel sprout, squash, turnip
FoodBERT-Text	beet, parsnip, yam
FoodBERT-Multimodal	parsnip, plantain
Relation Extraction	-
tuna	
Pattern Extraction	salmon, chicken
Food2Vec-Text	crab meat, pickle relish, salmon
Food2Vec-Multimodal	crab meat, pickle relish, salmon
FoodBERT-Text	crab, crab meat, fish, prawn, salmon, seafood, tilapia
FoodBERT-Multimodal	crab meat, prawn, salmon, swordfish, tofu
Relation Extraction	chicken, salmon
pork	
Pattern Extraction	beef, lamb, chicken
Food2Vec-Text	beef, chicken, lamb
Food2Vec-Multimodal	beef, chicken, lamb, veal
FoodBERT-Text	lamb, tenderloin
FoodBERT-Multimodal	beef, lamb, tenderloin
Relation Extraction	chicken, beef, fat, veal, shrimp, turkey, bacon, sausage, turkey bacon, chicken breast, turkey sausage