Capturing Food Knowledge with Semantics and Embeddings

Anonymous authors

Abstract

Poor quality eating patterns contribute significantly to the incidence of preventable chronic diseases. The proliferation of recipes and other food information sources on the Web presents an opportunity for discovering and organizing diet related knowledge into a knowledge graph (KG), which can in turn be used to generate food recommendations tailored to an individual’s dietary habits and preferences. In this paper, we present our work on building a food knowledge graph using semantics oriented knowledge ingestion. We further augment the knowledge graph by incorporating inferences we have made on the similarity between food ingredients. These similarities are derived from embeddings generated from online recipe data. In the true spirit of linked data, we have linked to many of the existing concepts in other related ontologies, as well as community maintained resources such as DBpedia. The resulting knowledge graph is capable of answering questions related to the composition of dishes, nutritional content of food items, and potential substitutions.

1. Introduction

Chronic diseases such as cardiovascular disease, high blood pressure, type 2 diabetes, some cancers, and poor bone health are linked to poor dietary habits [van Dam et al., 2002, Meyer et al., 2000, DeSalvo et al., 2016]. Although much progress has been made in the development and implementation of evidence-based nutrition recommendations in the past few decades [Musso et al., 2003, Ley et al., 2014, U.S. Department of Health and Human Services, 2015], that knowledge has not translated into day-to-day dietary practices. One of the barriers to putting recommended dietary guidelines into practice is that the personalization of the guidelines (e.g., with respect to cultural and lifestyle differences) is largely left to individuals. Much more than just watching one’s caloric, fat, salt and sugar intake, guidelines also advise individuals to eat a variety of nutrient-dense foods. Thus, the number of nutritional parameters that need to be considered can become overwhelming.

A natural solution to this problem is to provide an intelligent and automated method for recommending foods. [Trattner and Elsweiler, 2017] provides a comprehensive review of the state-of-the-art in food recommender systems. They highlight a recent but growing focus on not only recommending likeable foods, but healthful foods. They also mention that despite its importance, food item recommendation, in comparison to other domains, is relatively under-researched. Among the several works they reviewed, only [El-Dosuky et al., 2013] involved the use of semantics, motivating the need for methodologies for constructing
a food-focused knowledge graph. Knowledge graphs have an important role in organizing the information we encounter on a day-to-day basis and making it more broadly available to both humans and machines. They have been used for a variety of tasks, including question answering, relationship prediction, and searching for similar items. While machine learning algorithms can effectively answer questions, they are notorious for producing answers that are hard to explain, especially automatically. Knowledge graphs make it possible to produce automatic explanations of how answers were derived. Interoperability is another important aspect of knowledge graphs, as it enables understanding and reuse among various users. However, the elusiveness of standards or best practices in this area poses a substantial challenge for knowledge engineers who want to maximize their discovery and reuse, as dictated by the FAIR (Findable, Accessible, Interoperable, Reusable) principles [Wilkinson et al., 2016].

In this paper, we discuss our methodology for extracting publicly available data about food, and constructing a knowledge graph that can be consumed by both humans and machines for providing useful food recommendations, that can in turn promote healthier life styles.

1.1 Use Case

Our use case is designed to assist people in personalizing their dietary goals by providing them with information to improve the alignment between their eating behaviors and general dietary recommendations. For example, consider the American Diabetes Association’s (2017) recommendation that “Carbohydrate intake from whole grains, vegetables, fruits, legumes, and dairy products, with an emphasis on foods higher in fiber and lower in glycemic load, should be advised over other sources, especially those containing sugars”. Unfortunately, translating this into healthful yet palatable food choices can be a daunting task for many individuals. Our goal is to assist them in exploring how different modifications to their meals can affect their alignment with guidelines.

Some of the competency questions (i.e., the questions that help capture the scope, content, and the form of evaluation of the knowledge that is modeled) include questions such as: “What are the ingredients and the total calorie count of a piece of a chocolate cake according to the USDA nutritional data?” The answer may include butter, eggs, sugar, flour, milk, and cocoa powder for the ingredients, and a calorie count of 424. For a diabetic who is trying to abide by the ADA guidelines, a question like, “How can I increase the fiber content of this cake?” may be a natural follow-up question to ask. Similarly, a person suffering from lactose-intolerance may ask “What can I substitute for butter in chocolate cake?”. Answering questions like this is not possible from sources such as DBpedia alone, because the information from those sources is not complete. For example, the dbo:ingredients for the resource dbr:Chocolate_cake contains only dbr:Cocoa_powder and dbr:Chocolate. Our augmented knowledge graph contains additional information from online recipe sites, along with the corresponding nutrient information from USDA, that has more relevant informa-

1. DBpedia [Auer et al., 2007] has structured content from the information created in the Wikipedia project.
tion than what is available on DBpedia. Therefore, to answer this question, we can use the semantic structure of our knowledge graph to suggest that whole wheat flour be used instead of white flour, or that soy or almond milk be used instead of cow’s milk, or that margarine be used instead of butter.

To address questions like those posed above, we present a methodology that can be used to extract publicly available data on food and construct a semantically meaningful knowledge graph that can power applications to help consumers understand their foods and discover substitutions.

2. Related Work

Ontologies representing food are a well-studied topic. The Food Ontology is a universal “farm to fork” food vocabulary [Griffiths et al., 2016] that covers the provenance of food contained within the ontology. The Personalized Information Platform for Health and Life Services (PIPS) ontology describes a food ontology from a nutritional and health care viewpoint [Cantais et al., 2005]. Similarly, the Healthy Life Style (HeLiS) Ontology includes a subportion focused on food that contains ‘BasicFood’ and ‘Recipe’ [Dragoni et al., 2018]. The Food Product Ontology [Kolchin and Zamula, 2013] is designed for business purposes. It includes concepts such as price and brand, which is more suitable for food suppliers than a recipe lookup. The Cooking Ontology [Batista et al., 2006] comprises four main classes–actions, foods, recipes, and utensils–with supplementary class units, measures, and equivalencies, and the ontology is integrated into a dialogue system to answer the questions. However, they currently do not support a version in English, and have not mapped to comparable classes in other ontologies, which is essential for reuse. Similarly, the BBC Food Ontology [BBC Food Ontology] only constructs the important concepts and needs to cooperate with other existing ontologies to work better. Due to the lack of a formal ontology in the field of nutritional studies, [Vitali et al., 2018] constructed an extensible nutritional ontology called Ontology for Nutritional Studies (ONS), which integrates some carefully selected pre-existing ontologies to health and nutritional information. An information retrieval system that incorporates knowledge from domains of food, health, and nutrition, to recommend food health information based on the users conditions and preferences is described in [Helmy et al., 2015]. It is clear that different food ontologies focus on different aspects of food, such as chemical compositions, or food sources/packaging, but our focus is on recommendation in the context of personalized health, i.e., suggesting similar or alternative foods that are more healthy.

Whilst heavily opinion-based, research into concrete metrics for food pairings and similarity has borne fruit. Ingredient relationships via analysis of shared flavor compounds is a topic that has gained prominence in recent years. For example, [Ahn et al., 2011] discuss trends in recipes across different cultures; they noted surprising disparity between Western and Eastern dishes: the former’s ingredients tend to have more overlapping flavors than expected, and the latter’s tend to have fewer.

The topic of knowledge graph construction is well-studied, with extensive work invested in all stages of the process. Entity resolution, the process of detecting and reconciling multiple records that refer to the same concept, is discussed in detail in [Brizan and Tansel, 2006]. This includes multiple techniques that we are presently applying, such as fuzzy
string matching [Chaudhuri et al., 2003] and string cosine similarity [Mihalcea et al., 2006]. Fundamentals of *link mining*, which is the process of producing useful information from the linkages between entities in a graph, are discussed in [Getoor and Diehl, 2005]. Although this survey does not focus specifically on knowledge graphs, it is highly relevant to our work as it focuses on *object ranking* and *object classification*, which is relevant to food recommendation. Using the structured nature of ontologies for named-entity recognition is explored in [Giuliano and Gliozzo, 2008]; they use supervised machine learning approach for instance-based ontology population based on lexical substitution techniques. Defining the relatedness between pairs of concepts to derive appropriate substitution rules is discussed in [Sethi and Shekar, 2018]. An automated synonym-substitution method, which is constructed based on the lexical information of concepts and the hierarchical structure of an ontology is discussed in [Taboada et al., 2017].

Investigation into the usage of clustering and similarity retrieval, in the vein of the *word2vec* model ([Mikolov et al., 2013]), has produced several interesting approaches. For example, *food2vec* [Altosaar, 2017] searches for relationships between ingredients based on their co-occurrence in recipes. [Tansey et al., 2016] apply several stages of *word2vec* and *paragraph2vec* and clustering to work up from names and nutrients to meals and diets. In this paper we explore the use of such vector embedding models to find similar ingredients and recipes. More importantly, we show that using information from our food knowledge graph helps us derive better embeddings than using raw text.

### 3. Data Acquisition

We investigated various publicly available data sources on food as we constructed our knowledge graph. There exists a vast amount of data related to food, recipes, and nutrients on the Web. Tens of thousands of distinct recipes, thousands of ingredients, and dozens of nutritional data points per ingredient need to be considered. For this reason, we first briefly examine the data sources to offer an insight into the process we followed in acquiring the data for our knowledge graph.

Online recipe sites allow users to browse and share recipes. Some display content from specific commercial sources; others permit users to upload their own recipes. Each website has specific convention for how data is presented. However, even sites that have made no effort to accommodate automated data collection can still be scraped reliably.

*Schema.org* is a collaborative community project with a mission to create, maintain, and promote schemata for structured data on the Internet, on web pages, in email messages, and so on [Guha et al., 2016]. It provides a standard way to present structured data, with specifications for a wide variety of concepts, including recipes. Some recipe websites include this machine readable data (as XML or JSON) alongside the traditional human readable content. There is one notable drawback to the Schema.org representation of a recipe: the *name*, *unit*, *quantity*, and *comments* are combined into a single element. This means that additional parsing is required to separate the different concepts. However, the format tends to be a stable between different recipes on the same site. We further discuss this process in our description of the knowledge graph construction.
3.1 Data Sources

The “backbone” of our knowledge graph is made up of links derived directly from the input data collected from public sources such as the food and nutrition data from the United States Department of Agriculture (USDA) (https://ndb.nal.usda.gov/ndb/), restaurant data from MenuStat (http://menustat.org), and recipes from online websites. USDA [Ahuja et al., 2012] provides an API and downloadable datasets containing food items and their corresponding nutrients, whereas MenuStat provides crowd-sourced interactive restaurant datasets that include the dishes served at various chain restaurants. Whereas a large numbers of recipes can be found online, but they have to be scraped and cleaned from the unstructured textual sources to identify ingredients, units, quantities, comments and text. For example when considering recipe data, that includes: i) Recipes (e.g., green bean salad, spaghetti bolognese, crock pot creole casserole), ii) Recipe tags (e.g., salad, BBQ), iii) Ingredient names (e.g., green bean, white wine vinegar, nusalt, olive oil, onion, slice bacon), and iv) Ingredient usage (e.g., 1 cup, 2 oz).

3.2 Challenges

Gathering and integrating data from many sources, either through web scraping or bulk loading, leads to several problems with consistency, accuracy, and completeness:

- **Invalid data** - some textual data contains characters that are illegal in an RDF based knowledge graph, requiring escaping. Escaping itself can pose problems for entity recognition and resolution; it must be applied consistently in all stages of the process.

- **Incomplete data** - recipes may lack quantities for ingredients, or provide non-standard units of measure (e.g., “to taste”, “as needed”, “a few shakes”). Nutrient data might be incomplete, with only some nutrients tabulated.

- **Ambiguous entities** - many ingredients are difficult to tie to a specific food item. This has several root causes, such as local spellings and spelling errors, local names and synonyms; and use of different languages. This can lead to a large number of equivalent names - corn masa, masa harina, corn flour, for example.

- **Extraneous information** - ingredients are occasionally listed with overly-complicated units (e.g., 1/3 of a 375g can of beans) or unnecessary information (e.g., black beans from the store).

4. Knowledge Graph Construction

A knowledge graph includes resources with attributes and entities; relationships between such resources, and annotations to express metadata about the resources. In this section, we illustrate the methodology we followed in constructing our food knowledge graph.

4.1 Food Ontology

We construct an ontology to represent the instances of the food knowledge graph, as shown in Figure 1, which captures the attributes and the relationships between concepts such as Food, Nutrient, Dish, Restaurant, and the Source that captures the provenance information.
This ontology was constructed after an investigation of data sources introduced in Section 3.1.

![Figure 1: Food Ontology](image)

4.2 Data Ingestion

Having described how we source our data, we now turn to the construction of the knowledge graph’s records for recipes, foods, and nutrients. Our complete knowledge graph contains several key components: i) Recipes and their ingredients, ii) Nutritional data for individual food items, and iii) Mappings to DBpedia concepts.

We created linkages to augment the knowledge graph; the recipe ingredients are linked to knowledge about their nutritional content, and the nutrient entries are linked to further information.

4.2.1 Nutrients

From recipes, we can produce a network of foods and their ingredients. However, without information about the nutritional content of each ingredient, we cannot make meaningful health-related suggestions. We use the USDA public nutrition dataset for this information. The data from USDA exists in a tabular form, describing several dozen nutritional statistics, such as calories, macro-nutrients (protein, carbohydrates, fats), and micro-nutrients (vitamins and minerals). Nutrients are provided per 100 grams of the food item. Two non-mass measurements of the food are also provided, along with the number of grams found in each measure.

We make use of the Semantic Data Dictionary approach [Rashid et al., 2017] that provides a toolkit to produce RDF triples from non-triple data sources. This turns our
tabular data into something that can be integrated into our base of existing knowledge. A small subset of the data converted to the concepts and linkages in the knowledge graph can be seen in Table 1.

<table>
<thead>
<tr>
<th>id</th>
<th>description</th>
<th>water</th>
<th>energy</th>
<th>protein</th>
<th>lipid</th>
<th>carbohydrate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1001</td>
<td>Butter, With Salt</td>
<td>15.87</td>
<td>717</td>
<td>0.85</td>
<td>81.11</td>
<td>0.06</td>
</tr>
<tr>
<td>1002</td>
<td>Butter, Whipped, With Salt</td>
<td>15.87</td>
<td>717</td>
<td>0.85</td>
<td>81.11</td>
<td>0.06</td>
</tr>
<tr>
<td>1003</td>
<td>Butter Oil, Anhydrous</td>
<td>0.24</td>
<td>876</td>
<td>0.28</td>
<td>99.48</td>
<td>0</td>
</tr>
<tr>
<td>1004</td>
<td>Cheese, Blue</td>
<td>42.41</td>
<td>353</td>
<td>21.4</td>
<td>28.74</td>
<td>2.34</td>
</tr>
<tr>
<td>1005</td>
<td>Cheese, Brick</td>
<td>41.11</td>
<td>371</td>
<td>23.24</td>
<td>29.68</td>
<td>2.79</td>
</tr>
<tr>
<td>1006</td>
<td>Cheese, Brie</td>
<td>48.42</td>
<td>334</td>
<td>20.75</td>
<td>27.68</td>
<td>0.45</td>
</tr>
<tr>
<td>1007</td>
<td>Cheese, Camembert</td>
<td>51.8</td>
<td>300</td>
<td>19.8</td>
<td>24.26</td>
<td>0.46</td>
</tr>
<tr>
<td>1008</td>
<td>Cheese, Caraway</td>
<td>39.28</td>
<td>376</td>
<td>25.18</td>
<td>29.2</td>
<td>3.06</td>
</tr>
<tr>
<td>1009</td>
<td>Cheese, Cheddar</td>
<td>37.1</td>
<td>406</td>
<td>24.04</td>
<td>33.82</td>
<td>1.33</td>
</tr>
<tr>
<td>1010</td>
<td>Cheese, Cheshire</td>
<td>37.05</td>
<td>387</td>
<td>23.37</td>
<td>30.6</td>
<td>4.78</td>
</tr>
</tbody>
</table>

Table 1: Example USDA data with 10 food items and 5 nutrients.

Given this data, we can define the shape of the resulting knowledge graph via semantic relationships, as can be seen in Table 2. The Column represents the column in the raw data. The Attribute/Entity column represents what rdf:type this food item is. The Unit column refers to the unit of measure for that nutrient from community accepted terminologies such as DBpedia and the Units Ontology. The Label column gives a textual description for the data item that can be used in text mining, and auto-completion tasks in applications that use our food knowledge graph.

Notice the interlinking to other ontologies in the Attribute/Entity and the Unit columns. The various prefixes\(^4\) in the annotations in Tables 2 and 3 point to the following ontologies:

- **chebi**: Chemical Entities of Biological Interest Ontology (http://bio2rdf.org/chebi/)
- **dbr**: DBpedia Resource Ontology (http://dbpedia.org/resource/)
- **sio**: Semanticscience Integrated Ontology (http://semanticscience.org/resource/)
- **envo**: Environment Ontology (http://purl.obolibrary.org/obo/envo.owl#)
- **foodon**: Food Ontology (http://purl.obolibrary.org/obo/foodon.owl#)
- **schema**: Schema.org mappings (https://schema.org/)
- **uo**: Units Ontology (http://purl.obolibrary.org/obo/uo.owl#)

After the semantic structure of the data has been encoded, we then turn to the actual food items. Supposing the rows in the dataset have values such as ‘Butter, With Salt’, we list those values in the code column as can be seen in Table 3. Manual annotation of these concepts (e.g., ‘Cheese, Blue’) to external sources of information (e.g., the DBpedia concept for blue cheese, which in turn is linked to the corresponding Wikipedia page), is tedious and time-consuming. Automatic identification of these links is a non-trivial process, due to variations in nomenclature of different food items, but is still of great interest.

The format of the USDA food labels does provide some exploitable structure for automated processing: names are made up of comma-separated words, where each word provides

\(^4\) These prefixes can be dereferenced using http://prefix.cc or http://www.ontobee.org.
<table>
<thead>
<tr>
<th>Column</th>
<th>Attribute/Entity</th>
<th>Unit</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>chebi:33290, dbr:Food</td>
<td>USDA Id for the food</td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>sio:StatusDescriptor</td>
<td>Short description</td>
<td></td>
</tr>
<tr>
<td>water</td>
<td>envo:00002006, chebi:15377, dbr:Water</td>
<td>dbr:Gram, uo:0000021</td>
<td>Water (g)</td>
</tr>
<tr>
<td>energy</td>
<td>foodon:03510045</td>
<td>dbr:Kcal</td>
<td>Energy (Kcal)</td>
</tr>
<tr>
<td>protein</td>
<td>dbr:Protein</td>
<td>dbr:Gram, uo:0000021</td>
<td>Protein (g)</td>
</tr>
<tr>
<td>lipid</td>
<td>dbr:Lipid</td>
<td>dbr:Gram, uo:0000021</td>
<td>Lipid Total (g)</td>
</tr>
<tr>
<td>carbohydrate</td>
<td>dbr:Carbohydrate, schema:carbohydrateContent</td>
<td>dbr:Gram, uo:0000021</td>
<td>Carbohydrate (g)</td>
</tr>
<tr>
<td>fiber</td>
<td>dbr:Dietary_fiber</td>
<td>dbr:Gram, uo:0000021</td>
<td>Fiber (g)</td>
</tr>
<tr>
<td>sugar</td>
<td>dbr:Sugar</td>
<td>dbr:Gram, uo:0000021</td>
<td>Sugar Total (g)</td>
</tr>
<tr>
<td>calcium</td>
<td>dbr:Calcium</td>
<td>uo:0000022</td>
<td>Calcium (mg)</td>
</tr>
</tbody>
</table>

Table 2: Semantic structural representation of a subset of the USDA data.

a more specific category for the food. The previous example, ‘Cheese, Blue’, begins with the general category of ‘Cheese’, then further specifies its type. From this, we can attempt to not only directly identify the food item in external resources, but also search for categories into which the item falls.

We also examined using the DBpedia spotlight service\(^5\) [Mendes et al., 2011] to link food items to their corresponding DBpedia concepts. In the case of ‘Butter, With Salt’, the corresponding DBpedia concept is http://dbpedia.org/resource/Butter, i.e., dbr:Butter. By linking to the DBpedia resource, we are augmenting our knowledge graph with additional information; for example, dbr:Butter has additional properties such as ‘is dbo:ingredient’ of’ many well known food items such as dbr:Cake and dbr:Butterscotch, as well as lesser known food items such as dbr:Fank (a Hungarian sweet dish) and dbr:Fruit_ha_(pudding) (a dessert from the UK).

<table>
<thead>
<tr>
<th>Column</th>
<th>Code</th>
<th>Label</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>Butter, With Salt</td>
<td>Butter, With Salt</td>
<td>dbr:Butter</td>
</tr>
<tr>
<td>description</td>
<td>Butter, Whipped, With Salt</td>
<td>Butter, Whipped, With Salt</td>
<td>dbr:Butter</td>
</tr>
<tr>
<td>description</td>
<td>Butter Oil, Anhydrous</td>
<td>Butter Oil, Anhydrous</td>
<td>dbr:Butter</td>
</tr>
<tr>
<td>description</td>
<td>Cheese, Blue</td>
<td>Cheese, Blue</td>
<td>dbr:Blue_cheese</td>
</tr>
<tr>
<td>description</td>
<td>Cheese, Brick</td>
<td>Cheese, Brick</td>
<td>dbr:Cheese</td>
</tr>
<tr>
<td>description</td>
<td>Cheese, Brie</td>
<td>Cheese, Brie</td>
<td>dbr:Brie</td>
</tr>
<tr>
<td>description</td>
<td>Cheese, Camembert</td>
<td>Cheese, Camembert</td>
<td>dbr:Camembert</td>
</tr>
<tr>
<td>description</td>
<td>Cheese, Caraway</td>
<td>Cheese, Caraway</td>
<td>dbr:Cheese</td>
</tr>
<tr>
<td>description</td>
<td>Cheese, Cheddar</td>
<td>Cheese, Cheddar</td>
<td>dbr:Cheddar_cheese</td>
</tr>
<tr>
<td>description</td>
<td>Cheese, Cheshire</td>
<td>Cheese, Cheshire</td>
<td>dbr:Cheshire_cheese</td>
</tr>
</tbody>
</table>

Table 3: Semantic linkages of the content in the USDA data set to DBpedia resources.

---

After these annotations are completed, the semantic data dictionary conversion script is run to convert the tabular USDA data into quads (triples with additional provenance information) that are included in the knowledge graph. A small piece of the high level structure of the resulting graph can be seen in Figure 2.

Figure 2: An example of USDA data, pruned to display a handful of features. The prefixes usda-kb and ss refers to custom namespaces within our knowledge graph.

4.2.2 Recipes

Each recipe describes the ingredients needed to produce a dish. Each recipe receives a unique identifier, which is accompanied by its name, any provided tags, and a set of ingredients. Each ingredient points to its name, unit, and quantity. An example of the resulting structure is provided in Figure 3.

Individual ingredient records frequently appear in the form of (quantity, unit, name), such as 2 cups flour or 1 1/2 lb cabbage, chopped. Due to the lack of context, parsing these phrases with natural language toolkits is not feasible in all cases, and naive parsing methods fail due to minor quirks. To effectively parse such records, we utilize the following steps: 1) Parenthesized statements, such as (freshly picked) or (or chicken), are stripped. These provide additional cues to the reader, but are not strictly necessarily to understand components that make up the recipe. Similarly, any text following the first comma is dropped. 2) Numerical values, such as 1/2 or 2.5, are removed from the start of the string and saved as the quantity. 3) A list of units is compared against the first word in the string; if one matches, it is removed and stored as the unit. 4) The remaining text is
tokenized with the Natural Language Toolkit [Bird et al., 2009]. Non-color adjectives are eliminated (consider green bell peppers or red onion). Verbs and adverbs are also eliminated. Text following a conjunction is removed. Finally, NLTK’s lemmatizer is used to eliminate plurals. The resulting text is then saved as the name. Examples of inputs and results are provided in Table 4. High-quality name recognition significantly improves the quality of later results.

<table>
<thead>
<tr>
<th>Input</th>
<th>Quantity</th>
<th>Unit</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 cup milk</td>
<td>1</td>
<td>cup</td>
<td>milk</td>
</tr>
<tr>
<td>1 tablespoon parsley, chopped</td>
<td>1</td>
<td>tablespoon</td>
<td>parsley</td>
</tr>
<tr>
<td>5 tablespoons cold water</td>
<td>5</td>
<td>tablespoons</td>
<td>water</td>
</tr>
<tr>
<td>6 tablespoons red currant jelly</td>
<td>6</td>
<td>tablespoons</td>
<td>red currant jelly</td>
</tr>
<tr>
<td>1 cup butter, softened</td>
<td>1</td>
<td>cup</td>
<td>butter</td>
</tr>
<tr>
<td>2 tablespoons finely chopped parsley</td>
<td>2</td>
<td>tablespoons</td>
<td>parsley</td>
</tr>
</tbody>
</table>

Table 4: Examples of processed ingredient data

5. Knowledge Graph Augmentation

After importing food and recipe data, the next task is the creation of linkages between the resulting “islands” of data (recipes and nutrient data), which is non-trivial. There is also a need to continuously update the knowledge graph as new data becomes available and some of the existing data becomes obsolete. Therefore, we augment the knowledge base through entity resolution and relationship discovery.

5.1 Entity Resolution

Names are the only common factor between our two datasets. For this reason, we chose to resolve entities based on cosine similarity, which measures the degree of alignment between the vectors represented by two collections of words. We found it beneficial to limit the domain of concepts to match against, both for the sake of performance (matching is linearly expensive with respect to the number of entities) and to maximize accuracy (more spurious entities to match against cause more false positives). We initially did this by limiting DBpedia resources to those tagged with the Food type; however, we found that this included many spurious resources (non-food items) and excluded many important resources (staple ingredients, such as onions and tomatoes).

Therefore, we constructed a set of resources to match against based on several heuristics designed to recognize resources that could reasonably be considered to be ingredients. All DBpedia resources marked as ingredientOf were included, as was anything with a carbohydrate value. Whilst not perfect, this gave us a reasonably sized set of entities to match against. As an example, ‘Crumbled blue cheese’ and ‘Cheese, Blue’ have a combined vocabulary of crumbled, blue, cheese; therefore, the names represent vectors of $<1,1,1>$ and $<0,1,1>$, respectively, which can be used to compute the cosine similarity between them. Each ingredient name is compared against all names in the nutrient data set, and the best match is selected.
As mentioned earlier, we also used the DBpedia spotlight service\(^6\) [Mendes et al., 2011] to annotate the food items to their corresponding DBpedia concepts. We compare our results against those from the annotations given solely by DBpedia Spotlight [Mendes et al., 2011]. We sampled 500 recipes, from which 873 unique ingredients were collected. These were then mapped to DBpedia concepts as follows. Spotlight was fed 100 ingredients at a time, each delimited by a newline. The first identified concept from each line was recorded as the DBpedia mapping for that ingredient. We then fed each ingredient name into our resolver, which found the closest match among the set of DBpedia resources we consider as food concepts.

Both Spotlight and our approach agreed on 357 ingredients, but disagreed on 326. In addition, our method identified 150 ingredients that Spotlight was not able to identify, whereas there were 22 ingredients that Spotlight identified that were missed by our approach. The results are summarized in Table 5.

<table>
<thead>
<tr>
<th>Result</th>
<th>Concept count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our Method and Spotlight in agreement</td>
<td>357</td>
</tr>
<tr>
<td>Our Method and Spotlight in disagreement</td>
<td>326</td>
</tr>
<tr>
<td>Our Method only</td>
<td>150</td>
</tr>
<tr>
<td>Spotlight only</td>
<td>22</td>
</tr>
<tr>
<td>Neither method</td>
<td>18</td>
</tr>
</tbody>
</table>

Table 5: Results from Spotlight vs. Our Method

As seen in the numbers, whilst agreement occurred for roughly half of the sampled ingredients, a large number of names were either disagreed upon or were not identified by Spotlight at all. We found that many items that Spotlight failed to recognize were poorly formed or obscure. Conversely, entities that we could not identify tended to be well-formed, but were not captured in our sample of DBpedia resources and thus missed. Examples of the four categories of outcomes are provided in Table 6 to illustrate the aggregate results in Table 5.

<table>
<thead>
<tr>
<th>Result</th>
<th>Input</th>
<th>Our Method</th>
<th>Spotlight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agree</td>
<td>beef chuck</td>
<td>Beef</td>
<td>Beef</td>
</tr>
<tr>
<td>Agree</td>
<td>cut green bean</td>
<td>Green_bean</td>
<td>Green_bean</td>
</tr>
<tr>
<td>Disagree</td>
<td>chunky salsa</td>
<td>Salsa_(sauce)</td>
<td>Salsa_music</td>
</tr>
<tr>
<td>Disagree</td>
<td>soy cheese</td>
<td>Cheese</td>
<td>Cheese_analogue</td>
</tr>
<tr>
<td>Spotlight Only</td>
<td>cognac</td>
<td>N/A</td>
<td>Cognac</td>
</tr>
<tr>
<td>Spotlight Only</td>
<td>linguine</td>
<td>N/A</td>
<td>Linguine</td>
</tr>
<tr>
<td>Our Method only</td>
<td>ripe tomato</td>
<td>Tomato</td>
<td>N/A</td>
</tr>
<tr>
<td>Our Method only</td>
<td>charmagaj paste</td>
<td>Shrimp_paste</td>
<td>N/A</td>
</tr>
<tr>
<td>Our Method only</td>
<td>frozen</td>
<td>Ice</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 6: Selected examples of various mapping outcomes

---

5.2 Ingredient Embeddings for Relationship Discovery

In addition to the previously described relationships between ingredients, new relationships can be discovered by examining the co-occurrence of ingredients in various recipes, and these relationships can assist reasoning tasks performed over the knowledge graph. Word-embedding techniques have become a well-established approach for learning semantic relationships between words in text documents. This can be extended to the food domain, as in [Altosaar, 2017], where ingredients are analogous to words and recipes are analogous to documents. Additionally, [Teng et al., 2012] measure the co-occurrence of ingredients in recipes using point-wise mutual information and examine its usefulness in predicting the ratings of recipes, but find the correlation to be weak.

Such embeddings may be useful for augmenting the knowledge graph in two ways. First, new relationships between concepts can be discovered and incorporated into the graph. Second, concepts in the knowledge graph that were distant (e.g., multiple hops along the graph) may have a stronger relationship than previously realized. Of course, this doesn’t reveal exactly how these entities may be related, but the relation is important nonetheless and may aide in further exploration.

In order to learn efficient vector-representations of ingredients, we utilize the popular word-embedding technique FastText [Bojanowski et al., 2016], an extension of Word2Vec [Mikolov et al., 2013] that learns efficient (i.e., fast and of low dimensionality) distributed vector-representations of words. It breaks words into their constituent n-grams, learns vector representations of those n-grams, and then represents each word as the sum of the n-gram vector. We use the Gensim v3.4.0\(^7\) implementation of FastText to train our embedding model. We set the vector size to 300, train over 100 epochs, use an n-gram size of 4, and exclude ingredients which occur in fewer than 5 recipes. We use the Continuous Bag-Of-Words (CBOW) model and employ negative sampling as our training algorithm.

The data set used for computing ingredient similarity consists of 8958 recipes and 4671 ingredients scraped from an online recipe website. The ingredient names are mapped to DBpedia entries to help eliminate redundant entities and effectively reduce the size of the embedding vocabulary to 813 ingredients. For example, ingredients broccoli, bagfrozen up broccoli, head broccoli, and green broccoli cut all map to the same broccoli concept in DBpedia.

One can leverage existing relationships in the knowledge graph to assess the quality or feasibility of the similarity matching. However, it is beneficial to perform manual inspection of the new relationships using specific criteria for assessing the validity of the similarity. We propose 3 general criteria below:

- Criteria 1: if A is part of B, or B is part of A (e.g., hamburger contains beef (meat)).
- Criteria 2: if A and B share the same category type or hierarchical ancestor (e.g., apple and pear are both of the family rosaceae).
- Criteria 3: if B can be used as a substitute for A (e.g., margarine is used as a substitute for butter).

\(^7\) Gensim: https://radimrehurek.com/gensim/
In Table 7 we show an example of the ingredient *butter* and the top 10 most similar ingredients as learned by our ingredient embeddings. The ingredients listed in **bold** are those that meet at least one of the aforementioned validity criteria.

<table>
<thead>
<tr>
<th>Ingredient</th>
<th>Similarity</th>
<th>A ∈ B OR B ∈ A</th>
<th>Same cat.</th>
<th>Sub</th>
</tr>
</thead>
<tbody>
<tr>
<td>Margarine</td>
<td>0.5556</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Buttermilk</td>
<td>0.4580</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Milk</td>
<td>0.4531</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Bourbon_whiskey</td>
<td>0.4077</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Cream</td>
<td>0.3885</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Nutmeg</td>
<td>0.3734</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Shortening</td>
<td>0.3701</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>White_chocolate</td>
<td>0.3646</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Oleoresin</td>
<td>0.3475</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Hazelnut</td>
<td>0.3444</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 7: Examples of ingredients that are similar to the ingredient *butter*, along with the cosine similarity score. Each ingredient is evaluated against the three evaluation criteria and those that meet at least one criteria are indicated in **bold**.

Once these similar ingredients are discovered, they are fed back into the knowledge graph with assertions containing the relationships `similar-to` and `similarity-score`. The sub-graph containing the relationships of the similar food items and their associated scores as discovered from the ingredient similarity in Table 7 is depicted in Figure 4. With these assertions in place, we can answer the types of questions posed in Section 1.1, for e.g., “What can I substitute for butter in chocolate cake?” To answer this question, our augmented knowledge graph shows that margarine can be used instead of butter, which has the highest associated similarity score (0.5556) to support that assertion.

As a more quantitative evaluation of the quality of ingredient embeddings, we used a subset of 4967 recipes that have a geographic/region-based tag from one of four choices, namely, Italian, Cajun, Indian, Mexican. Using the FastText approach we find embeddings for the ingredients using the raw text names, and contrast those with the embeddings obtained after we resolve the ingredients by mapping them to the corresponding Dbpedia entities in our food KG.

To evaluate the quality of the embeddings, we first compute the probability of an ingredient being used in a recipe from each of the four regions. Thus, each ingredient is associated with a 4-dimensional probability vector. For example, if “turmeric” has a probability of 0.05 for Italian, 0.15 each for Cajun and Mexican, and 0.65 for Indian, then its
probability vector is $(0.05, 0.15, 0.65, 0.15)$ in the order of Italian, Cajun, Indian, Mexican. After obtaining the embeddings, we retrieve the top 5, 10, 20 and 50 nearest neighbors of a given query ingredient, and compute the average cosine similarity between the probability vectors of the query ingredient versus its top-K neighbors. The average of these values over all the test queries is shown in Table 8. Out of the total of 4966 unique ingredients, we were able to resolve them to 793 Dbpedia entities. Therefore, we chose these 793 ingredients as our test set to evaluate the performance of ingredient similarity retrieval. We can see that mapping the raw ingredients to the Dbpedia concepts clearly improves the quality of the embeddings due to better name resolution. For example, among the top-5 most similar ingredients, the Dbpedia resolved embeddings have an average cosine similarity of 0.619 versus 0.534 when using text only.

<table>
<thead>
<tr>
<th>Ingredients</th>
<th>Top 5</th>
<th>Top 10</th>
<th>Top 20</th>
<th>Top 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw Ingredients</td>
<td>0.534</td>
<td>0.517</td>
<td>0.506</td>
<td>0.495</td>
</tr>
<tr>
<td>Dbpedia Resolved Ingredients</td>
<td>0.619</td>
<td>0.608</td>
<td>0.599</td>
<td>0.588</td>
</tr>
</tbody>
</table>

Table 8: Quality of Ingredient Embeddings in Similarity Retrieval Task.

### 5.3 Knowledge Graph Enhanced Recipe Embeddings

To further test the efficacy of our food knowledge graph, we also evaluated the quality of the recipe embeddings, obtained using Doc2Vec [Le and Mikolov, 2014], where the set of all ingredients in a recipe is treated as a document, and each ingredient is therefore a word. We used the same 4967 recipes with geographic/region-based tags from four regions, namely, Italian, Cajun, Indian, Mexican.

<table>
<thead>
<tr>
<th>Ingredients</th>
<th>Top 5</th>
<th>Top 10</th>
<th>Top 20</th>
<th>Top 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw Ingredients</td>
<td>0.335</td>
<td>0.332</td>
<td>0.326</td>
<td>0.329</td>
</tr>
<tr>
<td>USDA Resolved Ingredients</td>
<td>0.345</td>
<td>0.344</td>
<td>0.345</td>
<td>0.346</td>
</tr>
<tr>
<td>Dbpedia Resolved Ingredients</td>
<td>0.351</td>
<td>0.351</td>
<td>0.346</td>
<td>0.342</td>
</tr>
<tr>
<td>Split Raw Ingredients</td>
<td>0.444</td>
<td>0.438</td>
<td>0.431</td>
<td>0.418</td>
</tr>
<tr>
<td>Split USDA Resolved Ingredients</td>
<td>0.498</td>
<td>0.487</td>
<td>0.473</td>
<td>0.454</td>
</tr>
<tr>
<td>Split Dbpedia Resolved Ingredients</td>
<td>0.437</td>
<td>0.432</td>
<td>0.425</td>
<td>0.415</td>
</tr>
</tbody>
</table>

Table 9: Quality of Recipe Embeddings in Similarity Retrieval Task.

We evaluate the recipe embeddings by retrieving for each query recipe its the top 5, 10, 20, and 50 nearest neighbors in the embedding space, and evaluating the fraction of neighbors having the same region tag as the query recipe. Table 9 shows the results of this evaluation, averaged over all 4967 recipes serving as queries. The first row, marked ‘Raw Ingredients’, is for embeddings obtained using only the text from all the recipes. The ‘USDA Resolved Ingredients’ row refers to embeddings obtained after mapping each of the ingredients to the corresponding USDA entities in the food KG. Likewise, ‘Dbpedia Resolved Ingredients’ is when we obtain embeddings after mapping to corresponding DBpedia concepts. Also, since many of the ingredients are compound names, which can be difficult to resolve, we also split them on whitespace and used these split names as ingredients. The recipe embeddings using the split raw ingredients, and USDA and Dbpedia resolved ingredients appear in the bottom three rows, respectively. As we can observe, using resolved
ingredients from the food KG results in better embeddings compared to using text only ingredients, with even more improvements obtained when we split the ingredient names. The best results are obtained for USDA resolved ingredients with split names. In the top 5 neighbors across recipes, on average 49.8% of the neighbors had the same regional cuisine tag as the query recipe, whereas using raw ingredients (split) the result was 44.4%. As another example, for the top 50 neighbors, using USDA food KG links improved the embeddings by 3.6%, with 45.4% of the neighbors having the same regional tag for USDA versus 41.8% for raw ingredients. These results clearly show the value of the food KG in creating better embeddings for the recipes, via better resolution of the ingredients.

6. Conclusions and Future Work

It is evident that information on food, while readily available on the web, is not able to be used by an individual for general healthy behavior choices. To address the issue of aggregating all the pertinent information on food in a manner that is consumable by an individual specific to their health and taste preferences, we have created an integrated knowledge model for food, which can be used to suggest healthier food and restaurant menu item alternatives. We model structured sources in terms of a target ontology, and augment the knowledge graph with other unstructured sources. Furthermore, we support better linkages using instance-based and hybrid ontology mappings between the ontologies.

More specifically, we extracted the relevant data on food from authoritative sources such as the USDA, as well as online recipe sources. We apply a semantics based extract-transform-load procedure to structure the food knowledge using our ontology as well as community accepted terminologies, and link to relevant DBpedia resources to support further exploration and augmentation of our food knowledge graph. The linkages to DBpedia resources are done using techniques involving lexical similarity and fuzzy matching to find the corresponding concepts in DBpedia for food names that are not exact one-to-one matches. We utilized embedding techniques to retrieve similar food items to augment our knowledge graph with relationships such as similar-to and similarity-score. These semantically describe similar ingredients that can be used when recommending food substitutions. This enabled us to group and link concepts to resolve discrepancies arising from food synonyms (e.g., aubergine versus eggplant), as well as foods that can be substitutes (e.g., butter versus margarine). We also conducted some tests with respect to DBpedia Spotlight to gain an insight as to how our entity resolution techniques perform compared to theirs. Finally, we demonstrate the value of our food knowledge graph via better ingredient and recipe embeddings than those obtained using only textual information, for similarity retrieval.

In the future we would like to develop novel embedding models to produce more meaningful representations. We can further leverage the food knowledge graph and relationships between ingredients and recipes to improve the embeddings. In conclusion, we have presented a reusable methodology that integrates information on food into a knowledge graph. Our preliminary Food Knowledge Graph is available for download at https://xxx.github.io/FoodKG. Using this constructed knowledge graph, we can answer complex questions related to recipes, ingredients, nutrition and food substitutions that can power applications that target healthy life style behaviors.
References


