Enhancing Food Composition Databases: Predicting Missing Values via Knowledge Graph Embeddings

Marko Možina
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, Slovenia
marko.mozina1@gmail.com

Slavko Žitnik
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, Slovenia
slavko.zitnik@fri.uni-lj.si

Barbara Koroušić Seljak
Computer Systems Department
Jožef Stefan Institute
Ljubljana, Slovenia
barbara.korousic@ijs.si

Tome Eftimov
Computer Systems Department
Jožef Stefan Institute
Ljubljana, Slovenia
tome.eftimov@ijs.si

ABSTRACT

Food composition databases (FCDBs) have presented an integral part of food and nutritional research, dietary assessment, and related (e.g., health, environmental) fields. However, as with other scientific disciplines, the domain of nutrition and food composition is no exception to the problem of missing data. This can significantly reduce the accuracy and reliability of analyses based on food composition, as it introduces an element of ambiguity and can, therefore, limit their usage. To address this issue, researchers have explored various methods for imputing missing data. The easiest and most common approach to this problem is to calculate the mean or median from available data in the same FCDB or to borrow values from other FCDBs. However, such simple methods may produce notable errors. In this paper, we investigate the use of knowledge graph embedding models for borrowing and imputing missing values in FCDB. We used the ComplEx model from the Ampligraph library and results are very promising. By employing the approach described in our paper, the model can capture the underlying structure and relationships in the data, providing accurate imputations even when there are missing values. Ultimately, the use of the proposed technique could lead to more accurate and reliable analyses in the field of nutritional research and dietary monitoring.

CCS CONCEPTS

• **Computing methodologies** → **Knowledge representation and reasoning**; **Supervised learning**; **Machine learning approaches**.

KEYWORDS

food composition database, nutrient values, missing data, data exploration, missing value imputation, graph machine learning, ampligraph, knowledge graph embeddings

1 INTRODUCTION

Food Composition Data (FCD) refers to detailed sets of information that provide valuable insights into the nutritional components of food, including nutrient values, energy content, and values of other elements like bio-actives, toxins etc. [5]. These data sets, accompanied by metadata such as classifiers and descriptors, are organized within Food Composition Databases (FCDBs). FCDBs serve as the primary sources for Food and Nutrition Science, as well as for various public health domains, the food industry, and clinical practices.

The quality of FCD within FCDBs varies due to the diverse sources and methods employed to obtain the data [8, 14]. To identify and categorize the data, codes and references are used, giving priority to specific data types and sources. The preferred approach involves acquiring original analytic values from published literature or laboratory reports. Alternatively, estimated values derived from similar foods, data calculated from recipes based on ingredient nutrient contents, or borrowed values from other databases can be used [2].

FCDBs differ not only in quality but also in quantity of data and accompanying metadata, leading to certain limitations in their usage. Incompatibility between databases, limited coverage of food items and nutrients, errors in database use, and restrictions in measuring food intake are among these limitations. However, the most significant limitation is the incomplete coverage of foods and nutrients, resulting in missing data within FCDBs. Addressing missing data is crucial for maintaining the integrity of the database, and different methods are employed, including ignoring the missing data, imputing plausible estimated values [10], or utilizing model-based techniques for calculation [9, 12, 13]. Borrowing data from other databases or calculating mean/median values from similar foods within the same FCDB are common approaches to resolving missing data. However, these methods can be inaccurate due to the inherent compositional variations in food samples, necessitating the development of improved techniques for imputation and calculation of missing FCD.

Our contribution: In this paper, we propose the utilization of knowledge graph embeddings [3] for imputing missing values in FCDBs. Several diverse approaches have been employed in attempts to address this issue. However, the advancements in Graph Machine Learning (GML) and its growing prominence have inspired us to apply it to FCD. As we shall observe, by embedding relational food data as a knowledge graph, we can uncover fundamental connections between different foods that may remain undetectable through other non-graph-based alternative methods.

Outline: The rest of this paper is organized as follows: In Section 2 we present the related work on missing value imputation in FCDBs. Section 3 describes the data used in our experiment and provides an overview of knowledge graph embeddings and the metrics used to evaluate their performance. In Section 4, we present the results of our experiment. We begin by exploring the data to identify any potential patterns, and then proceed to construct our models and test their link prediction capability. Finally, in Section 5, we discuss directions for future work.

Reproducibility: The code will be available after the review process.

2 RELATED WORK

Addressing missing data in FCDBs has been an ongoing issue, prompting research into potential solutions. Statistical approaches, such as Null Hypothesis Testing, have also been explored to handle missing data in FCDBs [12]. Evaluations of statistical methods for missing values in FCDBs, conducted by Ispirova et al. [13], have compared Non-Negative Matrix Factorization (NMF), Multiple Imputations by Chained Equations (MICE), Nonparametric Missing Values Imputation using Random Forest and K-Nearest Neighbors against mean or median value imputation [10]. Additionally, the issue of missing data in FCDBs has been investigated by utilizing autoencoders, a deep learning algorithm, for imputing missing values [9]. Autoencoders possess the capability to approximate values by acquiring a higher-level understanding of the input data. Nevertheless, all the aforementioned studies face the challenge of requiring a complete training dataset to develop an effective predictive model. This limitation has served as a driving force to convert the FCDB into a graph structure (use all available data) and investigate the potential of knowledge graph-based techniques for predicting missing values.

3 METHODS AND MATERIALS

In this section, we commence by providing an explanation of the data used in our experiment, its original purpose, organization, and the structure of a dataset created specifically for our study. Then follows a description of the methods used in our research. Initially, we provide a brief overview of knowledge graph embeddings and then explain the evaluation metrics that we used to measure the performance of our models.

3.1 Food composition data

At present, numerous international bodies, organizations, and projects are actively engaged in the field of food composition. One of them is the OPKP (*Odprta platforma za klinično prehrano*, Open Platform for Clinical Nutrition) [16], which is a Slovenian platform for dietary assessment complying with the CEN Food standard. It was primarily designed for patients, clinical dietitians, and other healthcare providers at the Pediatric Clinic and the Oncology Institute. It serves as assistance in assessing patients' dietary habits, creating dietary plans, and designing menus throughout the treatment process. In an extensive database (food lexicon), users of the platform can search for food and dishes that they have consumed or intend to consume and verify their compositional values. The project receives funding from both the Ministry of Higher Education, Science and Technology and the European Regional Development Fund, and it is overseen by the Jožef Stefan Institute (IJS). OPKP comprises a wide range of foods from multiple countries, with a primary focus on foods from Slovenia. For our research, we specifically utilized the data from the Slovenian food database [15]. However, it is designed to comply with any other FCDB following the CEN Food standard and EuroFIR thesauri.

Table 1: Format of our dataset. The values are presented in grams per 100 g of the food item

	Water	Fat	Protein	Sugar	...
Chicken egg	75.95	8.94	12.940	0.70	...
Sheep milk	82.70	6.26	5.270	4.70	...
Chicken thigh	69.24	12.61	17.006	0.00	...
Cherry	89.50	0.29	0.720	12.80	...
:	:	:	:	:	...

To make use of the data effectively, an initial step involved extracting the necessary information, specifically nutrient values for each food item. Subsequently, the dataset was constructed as a relational database, with each row representing a food item and each column representing a nutrient. Overall, nearly 1,000 food items and approximately 300 distinct nutrients were extracted. However, the data suffered from significant incompleteness and disorganization. To ensure a complete and homogeneous database, without missing values and multi-unit nutrients, a process of data cleaning and value conversion was conducted as a preparatory step for further analysis. From the multitude of nutrients, 25 were chosen based on criteria such as minimal missing values, significance in Food and Nutrition Science, and maximal diversity. In the end, our dataset consisted of 351 distinct food items, spanning somewhat diverse food groups. The structure of the final dataset is illustrated in Table 1.

3.2 Supervised learning on knowledge graphs

In this section, we present the details needed to understand the methodologies used in our experiments.

For our experiment on imputing missing values in FCDBs, we decided to use Ampligraph [6]. Ampligraph is a Python library containing a suite of neural machine learning models for relational learning, a branch of machine learning that deals with supervised learning on knowledge graphs. A knowledge graph is a structured representation of knowledge that captures relationships between entities, concepts, or facts. It organizes information in the form of nodes (entities) and edges (relationships) to create a graph-like structure. Formally, a knowledge graph is a subset of the cross product $N \times L \times N$, where N represents a set of nodes and L represents a set of labels. Each member of this set is known as a triple. To accommodate popular deep machine learning models that require numerical inputs, symbolic or discrete structures within a knowledge graph need to be converted into numerical representations. This can be achieved by assigning a vector representation to each node, enabling calculation of similarity between nodes based on the difference between their corresponding vectors. These vectors, associated with each node, are also referred to as knowledge graph

embeddings. The goal of knowledge graph embeddings is to encode the structural and semantic information of the knowledge graph in a way that facilitates various downstream tasks, such as link prediction [18], entity classification, or relation extraction.

The mean reciprocal rank (MRR) score is a statistic measure commonly used in information retrieval and ranking tasks [11]. It evaluates any process that generates a list of possible responses to queries, ordered by the probability of correctness. MRR score is calculated by taking the average of the reciprocal ranks across a set of queries or instances, as seen in Equation 1:

$$MRR = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{R_i} \tag{1}$$

The reciprocal rank is the inverse of the rank R_i at which the correct answer or relevant item is found. A higher MRR value indicates better performance, as it signifies that the correct answers are consistently ranked higher among the options provided by the model.

The Hits@n score is another widely used evaluation metric in information retrieval that assesses the quality of predictions made by a model. It checks whether the desired entities/relationships or ground truth are present among the top-n ranked predictions. A general formula to calculate Hits@N is:

$$\text{Hits@}n = \frac{\text{number of hits at } n}{\text{number of total queries or test cases}} \tag{2}$$

A higher Hits@n score indicates better performance, as it means a larger proportion of the ground truth items are among the top-n predictions.

4 EXPERIMENT AND RESULTS

In this section, we illustrate findings of our research. We start by analyzing the selected food composition data to identify potential similarities among food items. Following that, we evaluate the link prediction capability of the knowledge graph derived from our FCD.

4.1 Exploratory data analysis

To identify potential similarities among food items, our initial task was to define what it means for two foods to be considered similar. Each food item in FCDBs can be represented as a vector, with each component representing a specific nutrient value. By calculating the cosine similarity between these vectors, one can determine the degree of similarity or dissimilarity between food items based on their nutritional composition. Cosine similarity is particularly suitable for this task because it measures the cosine of the angle between two vectors, disregarding their magnitudes. This property is desirable when comparing food items since it focuses on the relative proportions of nutrients rather than their absolute values.

To start off, we constructed a similarity graph, connecting only those foods that exceeded a specified threshold of similarity. Additionally, we colored foods from the same food group with the same color. The observation of Figure 1 reveals that while there are some highly similar food pairs, their number remains low. This outcome is desirable because we want as diverse dataset as possible. Notably, certain foods from distinct food groups are also connected, indicating either the presence of erroneous data or that those foods belong to different yet related food groups. Further examination revealed that the latter scenario was predominantly true, and also uncovered instances where some food items had been erroneously assigned to incorrect food groups.

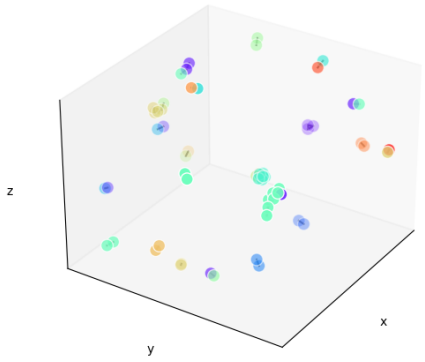

Figure 1: 3D drawing of a similarity graph with threshold of 0.0005. Some of the food groups that can be observed are meats (turquoise green and orange), fruits (light blue), and vegetables (dark blue).

Next, we applied three widely recognized dimensionality reduction algorithms (i.e., MDS [19], PCA [7], and t-SNE [22]) to visualize the data in a two-dimensional space, with the expectation of identifying any distinct clusters. While MDS and PCA did not produce any noticeable outcomes, t-SNE, on the contrary, delivered the desired results. This can be partially explained, as FCDBs typically involve complex and high-dimensional data. Therefore linear dimensionality reduction techniques like PCA may not capture the intricate relationships present in the data, as they primarily focus on capturing global structure. MDS, while capable of preserving pairwise distances, may struggle to handle the nonlinear relationships that exist within the database. On the other hand, t-SNE excels in capturing both local and global structures by focusing on preserving the neighborhood relationships between data points. It performs well in revealing clusters, making it particularly suitable for exploring and visualizing complex and non-linear relationships within food composition databases.

As depicted in Figure 2 below, we can clearly observe distinct clusters within our data. By conducting a closer analysis and associating the appropriate food names with each node, we can map each cluster to a naturally occurring food group. These four groups are:

- Vegetables (dark blue, top left)
- Fruits (light blue, top right)
- Meats (red and orange, middle)
- Cheeses (light green, bottom right)

The final cluster, located on the right side, appears to lack homogeneity and compactness. This can be attributed to its composition, as it consists of multiple smaller-sized food groups. As a result, these groups are not tightly mapped together nor significantly distanced apart. Among the food groups identified within this cluster are grains, pasta, rice, nuts, and more. It is worth noting that due to the large number of food groups in our data and some incorrectly

assigned groups, it may appear that certain foods in this cluster should be categorized differently. For instance, within this cluster, we may observe light blue and red points, which primarily represent nuts and grain products rather than meats or vegetables. Additionally, there is a cluster shaped like a banana at the bottom of the figure, primarily consisting of dairy products. Therefore, it is unsurprising that it is closely mapped to cheeses compared to other foods.

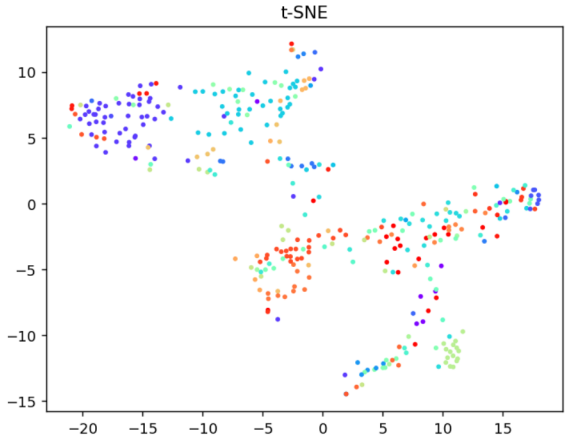

Figure 2: Visualization of the data using t-SNE, showing detectable clusters. The dark blue cluster at the top left represents vegetables, while the light blue cluster to their right represents fruits. Below them, the orange and red cluster denote meats, and the light green cluster in the bottom right corresponds to cheeses.

To determine whether our clustering results aligned with a more scientific approach, we conducted KMedoids clustering [17] and calculated the average silhouette score [20]. As depicted in Figure 3 below, the KMedoids procedure yielded similar clusters to those we had previously discovered. Specifically, vegetables, fruits, and meats were assigned to their respective distinct clusters. The orange cluster, although encompassing several smaller food groups, predominantly comprised nutritionally related foods such as grains and pasta. Thus, it is unsurprising that these foods were allocated to the same cluster rather than any other. Similarly, cheeses and other dairy products (black) were naturally grouped together and remained so even when using more medoids, whereas other clusters dissolved. We computed the silhouette score for a range of k values, from 2 to 10, and found that the score was highest for k = 5, confirming our earlier conclusion. However, there was one exception for k = 2, where the associated score was slightly better. Upon analyzing the formed clusters, we observed that the first cluster primarily consisted of fruits and vegetables, while the second cluster included all other food items. Although this grouping can be partially explained, the higher score alone did not provide sufficient evidence to support the existence of only two distinct food groups in our dataset.

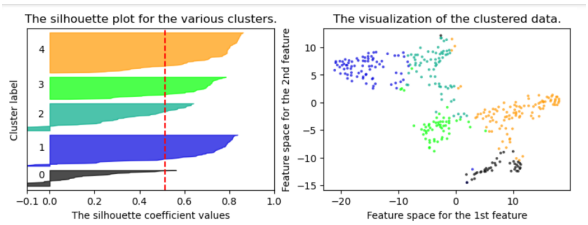

Figure 3: KMedoids clustering with the silhouette plot.

4.2 Knowledge graph construction

After confirming the diversity of our data, ensuring that we have a range of food items that are not overly similar, we proceeded to construct a knowledge graph. The diversity of our data was crucial, particularly because of the relatively small size of our dataset. Without adequate representation of all food groups, our link prediction model might not effectively train on certain categories. Consequently, the accuracy of predicting missing values for those specific foods would likely be compromised.

Since all models in Ampligraph [1] require a knowledge graph as input, our first step was to transform our relational database into a graph format, specifically a list of 3-tuples. As numerical values of nutrients cannot be directly represented as relationships in a knowledge graph due to the finite number of relationships allowed, we addressed this issue by discretizing the nutrient values. By calculating the minimum and maximum values for each column, we defined a finite number of classes that corresponded to equally sized intervals of nutrient values. Consequently, each food item would be associated with a nutrient through a relationship interval if the interval included the actual nutrient value. For example, the protein content of an egg, as shown in Table 1, would be represented as a 3-tuple:

$$(Egg,\ contains\ between\ x\ and\ y,\ Protein)$$

Here x and y represent the bounds of the interval that contains the value 12.94. By discretizing the data, we inevitably sacrifice some information. Our model will only know the interval that encompasses the actual value, rather than the exact value itself. Increasing the number of classes in our column division allows us to retain more information about the data. However, this also leads to a larger number of possible relationships, which can result in certain issues, like creating a more complex knowledge graph. When generating embeddings for such a graph, a large number of relationship types can increase the model's complexity, making it harder to learn embeddings that are meaningful and easily interpretable. Additionally, complex models are more susceptible to overfitting, where the embeddings become too specific to the training data and fail to generalize well. Hence, striking a balance between minimizing information loss and obtaining better results becomes crucial. It is important to emphasize that the division of the range between the minimum and maximum values of each nutrient (column) into equal intervals has only been done for illustrative purposes of the methodology. The size of these intervals can be further determined by domain experts and may vary.

Table 2: Evaluation metrics for our 3 models

	n = 4	n = 10	n = 25
MRR	0.60	0.53	0.52
Hits@10	0.79	0.69	0.64
Hits@3	0.64	0.56	0.54
Hits@1	0.50	0.45	0.46

4.3 Imputing missing values via link prediction

Considering our scenario, the number of entities is not substantial enough to accommodate too many distinct relationships. We decided to build three different knowledge graphs with 4, 10, and 25 classes (relations) per column and compare their performances. For generating knowledge graph embeddings [23], we utilized the ComplEx model [21] with default values for hyperparameters from the Ampligraph library. The ComplEx model represents entities and relationships as complex-valued vectors in a latent space. By employing complex-valued tensor factorization, the model can capture both symmetric and antisymmetric patterns in the data. Because this model has shown promising performance in knowledge graph completion tasks [4], which involve predicting missing relationships, it was a suitable choice for our experiment.

To train and evaluate our models, we utilized the built-in functions provided by Ampligraph and specifically designed for this purpose. Following the standard practice in machine learning, we initially split our dataset into training and test datasets, with sizes of 8475 and 300 respectively. It is worth noting that, unlike in other datasets, our data points relate to two entities that are linked together by some relationship. Therefore, it was necessary to ensure that all entities were represented in both the training and test sets by at least one triple. Hence, a random sampling approach for the test set was not possible. To solve this problem, Ampligraph offers a built-in function `train_test_split_no_unseen`, which takes care of splitting the data. We used the `evaluate_performance` function to compute ranks, which enabled us to determine two types of evaluation metrics: MRR score and Hits@n score. The results of the evaluation are presented in Table 2.

Based on the scores, it can be observed that our assumption regarding more complex models yielding worse results appears to be correct. The model with only four classes demonstrated the best performance across all metrics. Although the results may seem pessimistic, it is important to consider that judging the models solely based on scores without accounting for how other state-of-the-art models would perform on our limited dataset can be misleading. It is worth noting that an MMR score between 0.5 and 0.6 may still be considered good in some cases. Regardless, this motivated us to improve our models.

Although FCDBs contain food items and their nutrient values, it is not uncommon to extract additional information from them. One such metadata is knowing the food group to which a food item belongs. To improve the models, we incorporated information about food groups into our knowledge graph. This was accomplished by adding 351 new 3-tuples, thereby increasing the size of our training set, of the following form:

([Food item], belongs-to, [Food group])

Table 3: Evaluation metrics for our 3 improved models

	n = 4	n = 10	n = 25
MRR	0.81	0.77	0.67
Hits@10	0.94	0.91	0.82
Hits@3	0.87	0.79	0.71
Hits@1	0.74	0.71	0.58

Table 4: Prediction of unseen data

Statement	Rank	Prob.
Camembert containts between 20 and 30 Fat	1	0.99
Camembert containts between 50 and 60 Fat	57	0.90
Camembert containts between 0 and 10 Fat	62	0.71
Duck meat belongs to poultry	18	0.99
Duck meat belongs to Sea fish	162	0.95
Duck meat belongs to Fruits	520	0.22

Having done that, we proceeded to conduct the same analysis as with our previous models, and the outcomes were remarkable (see Table 3). It is evident that both MRR and Hits@n significantly improved across all three models. Once again, the model with the fewest classes emerged as the top performer, but this time with exceptionally high scores, which are universally regarded as superb.

Although scores such as MRR and Hits@n give us a general idea on how well a model is performing, it is hard to interpret what that means for imputing missing values. Therefore, it is beneficial to see how our models perform on unseen data, specifically by determining the probability of a previously unseen 3-tuple being true. To accomplish this, we selected a few relations that were not present in the training set. For making predictions, we utilized the improved model with 10 classes per nutrient. Initially, we chose three relations pertaining to the nutrient value of fat in Camembert cheese, out of which only one was true. The results are presented in Table 4. It is evident that the model predicted the correct relation with a very high probability. Although it also assigned a relatively high probability of 0.9 to the incorrect statement, this doesn't concern us significantly because, when imputing missing values, our objective is to select the best prediction based on the lowest rank. Furthermore, we evaluated the model's performance in predicting food groups. While this aspect may not usually be associated with missing values in FCDBs, it can still provide valuable insights into specific food items. In this test, the model was tasked with identifying the correct food group for duck meat. As before, the true statement emerged as the clear winner. Additionally, the model correctly recognized that duck meat is much more likely to belong to the sea fish category than to fruits, further validating the effectiveness of knowledge graph embeddings.

It is common for real-world data to exhibit varying levels of noise and variations. Our dataset was no exception, as observed during the exploration step. Hence, to conclude the evaluation process, there was one final step that was necessary to be undertaken. To demonstrate the robustness of our models against minor perturbations or outliers, it was essential to evaluate the stability of our

Table 5: Results of the robustness analysis

Iteration	MRR	Hits@10	Hits@3	Hits@1
First	0.71	0.82	0.72	0.65
Second	0.70	0.81	0.72	0.63
Third	0.70	0.82	0.72	0.64
Fourth	0.72	0.82	0.76	0.66
Fifth	0.72	0.86	0.73	0.66

models when exposed to variations in the input data. To accomplish this, we carefully selected 12 of the most significant macro and micro nutrients from our overall collection of 25 nutrients. Subsequently, we constructed our test set to fulfill two specific conditions:

- Every one of the 10×12 relations was represented at least once, provided it was present in the data.
- The test set was designed to be as diverse as possible, ensuring that we included the minimal number of foods that satisfied the first condition.

By following these rules, we obtained a test set consisting of 90 relations and a training set comprising 9,048 relations. As before, we decided to use our improved model with 10 classes. The evaluation was conducted five times, ensuring that each test set differed as much as possible from the ones in previous iterations while considering the two aforementioned conditions. The results are presented in Table 5. As it is evident, our model performed consistently across all five iterations, thus demonstrating its robustness.

5 CONCLUSION

Food Composition Databases are an important information resource used in various domains, including food and nutritional science, food industry for food production and consumption, as well as public health. As the presence of many (even tens of percents) missing data significantly restricts their usability, many solutions to this problem have been proposed. As one of them, we explored the use of graph embedding models for missing value imputation in FCDBs. By embedding the nodes in a low-dimensional space, these models can capture the underlying structure and relationships in the data, providing accurate imputations even when there are missing values. However, the small size of our database limited our experiment, so further research is needed to explore the effectiveness of these models for different types of food databases and missing value patterns. Ultimately, the use of these techniques shows promise and could lead to more accurate and reliable analyses in the field of nutritional research and dietary assessment. For future work, we are planning to present a comparison of the performances of the proposed model against baselines.

ACKNOWLEDGMENTS

This work has been supported the Slovenian Research Agency [research core funding programme P2-0098; the European Union's Horizon 2020 research and innovation programme [grant agreement 863059] (FNS-Cloud, Food Nutrition Security) and [grant agreement 101005259] (COMFOCUS), and the project funded by ECSEL Joint Undertaking (JU) [grant agreement 876038] (InSecTT).

REFERENCES

[1] Accenture. [n. d.]. AmpliGraph. https://github.com/Accenture/AmpliGraph. [Accessed: May 26, 2023].
[2] Yusentha Balakrishna, Samuel Manda, Henry Mwambi, and Averalda van Graan. 2022. Statistical Methods for the Analysis of Food Composition Databases: A Review. *Nutrients* 14, 11 (2022), 2193.
[3] Xiaojun Chen, Shengbin Jia, and Yang Xiang. 2020. A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications* 141 (2020), 112948.
[4] Zhe Chen, Yuehan Wang, Bin Zhao, Jing Cheng, Xin Zhao, and Zongtao Duan. 2020. Knowledge graph completion: A review. *Ieee Access* 8 (2020), 192435–192456.
[5] SM Church. 2006. The history of food composition databases. *Nutrition Bulletin* 31, 1 (2006), 15–20.
[6] Luca Costabello, Sumit Pai, Chan Le Van, Rory McGrath, Nicholas McCarthy, and Pedro Tabacof. [n. d.]. AmpliGraph: a Library for Representation Learning on Knowledge Graphs, March 2019. *URL https://doi. org/10.5281/zenodo* 2595043 ([n. d.]).
[7] Andreas Daffertshofer, Claudine JC Lamoth, Onno G Meijer, and Peter J Beek. 2004. PCA in studying coordination and variability: a tutorial. *Clinical biomechanics* 19, 4 (2004), 415–428.
[8] Paul M Finglas, Rachel Berry, and Siân Astley. 2014. Assessing and improving the quality of food composition databases for nutrition and health applications in Europe: the contribution of EuroFIR. *Advances in Nutrition* 5, 5 (2014), 608S–614S.
[9] Ivana Gjorshoska, Tome Eftimov, and Dimitar Trajanov. 2022. Missing value imputation in food composition data with denoising autoencoders. *Journal of Food Composition and Analysis* 112 (2022), 104638.
[10] Heather Greenfield and David AT Southgate. 2003. *Food composition data: production, management, and use.* Food & Agriculture Org.
[11] Charles Tapley Hoyt, Max Berrendorf, Mikhail Galkin, Volker Tresp, and Benjamin M Gyori. 2022. A unified framework for rank-based evaluation metrics for link prediction in knowledge graphs. *arXiv preprint arXiv:2203.07544* (2022).
[12] Gordana Ispirova, Tome Eftimov, Peter Korošec, and Barbara Koroušić Seljak. 2019. MIGHT: Statistical methodology for missing-data imputation in food composition databases. *Applied Sciences* 9, 19 (2019), 4111.
[13] Gordana Ispirova, Tome Eftimov, and Barbara Koroušić Seljak. 2020. Evaluating missing value imputation methods for food composition databases. *Food and Chemical Toxicology* 141 (2020), 111368.
[14] Maria Kapsokefalou, Mark Roe, Aida Turrini, Helena S Costa, Emilio Martinez-Victoria, Luisa Marletta, Rachel Berry, and Paul Finglas. 2019. Food composition at present: new challenges. *Nutrients* 11, 8 (2019), 1714.
[15] Mojca Korošec, Terezija Golob, Jasna Bertoncelj, Vekoslava Stibilj, and Barbara Koroušić Seljak. 2013. The Slovenian food composition database. *Food chemistry* 140, 3 (2013), 495–499.
[16] Peter Novak, Franc Novak, and Barbara Koroušić Seljak. 2013. Enhancement of Web Application Design of the Open Platform for Clinical Nutrition. In *Human Factors in Computing and Informatics: First International Conference, SouthCHI 2013, Maribor, Slovenia, July 1-3, 2013. Proceedings*. Springer, 791–802.
[17] Hae-Sang Park and Chi-Hyuck Jun. 2009. A simple and fast algorithm for K-medoids clustering. *Expert systems with applications* 36, 2 (2009), 3336–3341.
[18] Andrea Rossi, Denilson Barbosa, Donatella Firmani, Antonio Matinata, and Paolo Merialdo. 2021. Knowledge graph embedding for link prediction: A comparative analysis. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 15, 2 (2021), 1–49.
[19] Shadman Sakib, Md Abu Bakr Siddique, and Md Abdur Rahman. 2020. Performance evaluation of t-SNE and MDS dimensionality reduction techniques with KNN, ENN and SVM classifiers. In *2020 IEEE Region 10 Symposium (TENSYMP)*. IEEE, 5–8.
[20] Scikit-learn. 2015. Selecting the Number of Clusters with Silhouette Analysis on Kmeans Clustering—Scikit-Learn 0.17 Documentation. (2015).
[21] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International conference on machine learning*. PMLR, 2071–2080.
[22] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).
[23] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743.