

A Rich Recipe Representation as Plan to Support Expressive Multi-Modal Queries on Recipe Content and Preparation Process

Vishal Pallagani Priyadharsini Ramamurthy* Vedant Khandelwal Revathy Venkataramanan

Kausik Lakkaraju

Sathyanarayanan N. Aakur*

Biplav Srivastava

AI Institute, University of South Carolina and *Oklahoma State University
{vishalp@mailbox.,vedant@mailbox.,revathy@email.,kausik@email.,biplav.s@}sc.edu
{pramamu@ostateemail.,saakurn@}okstate.edu

Abstract

Food is not only a basic human necessity but also a key factor driving a society's health and economic well-being. As a result, the cooking domain is a popular use-case to demonstrate decision-support (AI) capabilities in service of benefits like precision health with tools ranging from information retrieval interfaces to task-oriented chatbots. An AI here should understand concepts in the food domain (e.g., recipes, ingredients), be tolerant to failures encountered while cooking (e.g., browning of butter), handle allergy-based substitutions, and work with multiple data modalities (e.g. text and images). However, the recipes today are handled as textual documents which makes it difficult for machines to read, reason and handle ambiguity. This demands a need for better representation of the recipes, overcoming the ambiguity and sparseness that exists in the current textual documents. In this paper, we discuss the construction of a machine-understandable rich recipe representation (R3), in the form of plans, from the recipes available in natural language. R3 is infused with additional knowledge such as information about allergens and images of ingredients, possible failures and tips for each atomic cooking step. To show the benefits of R3, we also present TREAT, a tool for recipe retrieval which uses R3 to perform multi-modal reasoning on the recipe's content (plan objects - ingredients and cooking tools), food preparation process (plan actions and time), and media type (image, text). R3 leads to improved retrieval efficiency and new capabilities that were hitherto not possible in textual representation.

Introduction

Food is an innate psychological need for human life, along with water, warmth and rest. In addition to being a basic human necessity, food is also a key factor driving a society's health and economic well-being. However, maintaining a healthy diet is essential for good health and nutrition as it prevents many chronic non-communicable diseases. The importance of diet and food has attracted increasing attention, leading to the coining of a new term called as *precision nutrition* (Chatelan, Bochud, and Frohlich 2019; Wang and Hu 2018) - which seeks to better the health of a

person through precise dietary intake based on unique characteristics of an individual such as DNA, race, gender, and lifestyle habits. The process of cooking is a key enabler for precision nutrition, which has led to extensive research in building and deploying decision-support artificial intelligence (AI) tools - varying from information retrieval (IR) using interfaces to task-oriented chatbots (Min et al. 2019; Jiang and Min 2020).

Recent advancements in learning-based approaches have seen many AI researchers channeling their effort in building large datasets and learning vector representations for recipes in the latent space. The learnt vector representations are used for downstream tasks (Li and Zaki 2020; Lien, Zamani, and Croft 2020) such as cooking (ingredient/recipe/cuisine) recommendation (Song, Yang, and Xu 2022; Tian et al. 2022; Kim et al. 2021), food conversation chatbots (Frummet, El-sweiler, and Ludwig 2022), and cooking activity recognition (Ramadan and El-Jaroudi 2022). The learning-based approaches are heavily dependent on the data they are trained on which, for the cooking domain, is the text-based recipes. However, text is a difficult representation for machines, using natural language processing (NLP), to read, reason and handle ambiguity. While learning an embedding - the state-of-the-art NLP method to pre-process documents - for the cooking domain, there is a need for representation that captures the cooking actions explicitly. Additionally, the recipe embeddings being learned only from the cooking instructions often lack constraints and attributes such as possible allergens, failures, and relation between multi-modal data points.

In this paper, inspired by PDDL plans, we consider recipes as plans that are composed of a sequence of steps. A step corresponds to a single food preparation action indicated by a verb and instantiated by world objects that are subject and object of the action (parameters). The actions proposed in R3 have additional knowledge such as allergen information, possible failures that can happen during preparation and their workarounds. As a preliminary implementation, we have created 25 egg-based recipes in R3 manually from original recipes taken from the RecipeQA dataset (Yagcioglu et al. 2018). We further created a tool - TREAT, to demonstrate the advantage of using R3 over natural language from the dataset for the task of information retrieval. We demonstrate how TREAT can perform multi-

modal constrained queries from the user for recipe retrieval. Constrained queries consist of restrictions specified by the user whilst obtaining information. Two kinds of constraints can be performed by TREAT, namely, outcome (e.g., *I want a recipe without leaf allergy.*) and process (e.g., *Suggest me a recipe that is not fried.*) constraints.

We make following contributions in the paper: (a) introduce a machine-readable rich recipe representation for recipes inspired by the structure of PDDL plans called R3, (b) implement the TREAT multi-modal recipe retrieval system which responds to user queries by reasoning on recipe content and process using R3, and (c) evaluate the impact of R3 by comparing recipes in R3 over text representation for the task of (recipe) information retrieval. In the remainder of the paper, we give a background of related work, followed by our approach describing the construction of R3 and its use in the TREAT system. We demonstrate the effectiveness of using R3 by performing a comparative study and conclude with pointers to future work.

Related Work

In this section, we describe existing work on representing recipes and plans to motivate the need and benefits of our approach.

Representing Recipes

Building representations for recipes using various learning-based techniques has been active research area in the recent past (Li and Zaki 2020). However, there are inherently many limitations of learnt recipe embeddings from text. The closest effort to ours in obtaining a new representation for recipes has been to manually convert them to a flowgraph (Mori et al. 2014). However, R3 differs from flowgraphs, as the former is created with the goal of offering flexibility to be used by both learning and planning based approaches. An automated planner cannot work with the representation from a flowgraph, but can do so on R3 enabling additional functionalities such as generating multiple similar plans (i.e., similar recipes) or alternative plans in case of failures (e.g., missing ingredient or cooking tool).

Existing datasets in the cooking domain mostly constitute of text and images (Lien, Zamani, and Croft 2020; Leroux and London 2002; Yagcioglu et al. 2018). There also have been efforts in building a cooking workflow dataset (Pan et al. 2020). Researchers also have looked at representing recipes in the form of a knowledge graph (Shirai et al. 2021) but as the graph grows in size, deploying and performing real-time decision support becomes a challenge (Hausmann et al. 2019). R3 has been developed to foster the continual search for finding a better representation of procedural texts and to demonstrate its viability, we adopted the task of recipe retrieval as it is an accepted baseline to validate a new representation (Lien, Zamani, and Croft 2020; Chen and Ngo 2016; Zhu et al. 2019). Roither et al (Roither, Kurz, and Sonnleitner 2022) have worked on classifying ingredients into allergen categories. In our work, through our representations, we enable searching recipes without an allergen. This enhances user convenience and ease-of-use.

Plan Representation

Planning is the problem of finding a path to lead an agent from an initial state to a goal state, given a set of legal actions to transition between states. The field of planning has seen many representations. In classical planning, which assumes complete information about the world, there was STRIPS (Fikes and Nilsson 1971), Action Description Language (ADL) (Pednault 1994) and SAS+ (Bäckström 1995) before Planning Domain Description Language (PDDL) (McDermott et al. 1998; Fox and Long 2003) standardized the notations.

In PDDL, a planning environment is described in terms of objects in the world, predicates that describe relations that hold between these objects, and actions that bring change to the world by manipulating relations. To facilitate automated solving, PDDL envisages two files, a domain description file which specifies information independent of a problem like predicates and actions, and a problem description file which specifies the initial and goal states. PDDL provides a relaxed specification of the output plan – it is a series of time steps, each of which can have one or more instantiated actions with concurrency semantics. Specification of the correctness of plans was formalized much later when tools like VAL were created (Howey and Long 2003) but they assume access to domain and problem files.

There is a long history in planning of viewing plans independently from how they may have been generated or evolve over time e.g., by an automated planner or by a human. In (Myers and Lee 1999), the need for diverse planning was motivated and grounded in a meta-theory allowing the plans to be modified and compared. In (Srivastava, Vanhatalo, and Koehler 2005), the authors describe a platform to store, search and manage plans, representing Information Technology workflows, as they evolve over time.

In this work, we adopt the plan representation of PDDL for recipes but do not assume how recipes may have been created. We thus allow not only efficient management of recipe data (storage and retrieval) but also enable future automation in recipe creation (by reuse and composition of recipes to create new recipes), and food preparation (by monitoring the execution of agents) by allowing reasoning with both the data and control flow inherent in the plan-based recipe representation consisting of text and images.

Approach

In this section, we first describe our plan-based recipe representation (R3) and then describe a system, TREAT, that uses it to support information retrieval.

Rich Recipe Representation

We define a recipe (**R**) as a tuple consisting of a required set of ingredients (**IG**), preparation time, cooking time, number of people served, and is said to be complete on performing a series of instructions (**I**). Each ingredient is a set consisting of the name, quantity (**Q**) required for the recipe, allergen information (**A**), alternative replacements for the ingredient, quality characteristic which signifies on state of the ingredient (e.g. grated cheese or cheese slice are two different qual-

Recipe: Egg-drop Chicken Noodle Soup	
Original Instruction	Instruction in R3
Stir together the cornstarch and 2 tablespoons water in a small bowl. Mix it well. Whisk into broth and boil for 1 minute or until thickened.	<pre>{ "original_text": "Stir together the cornstarch and 2 tablespoons water in a small bowl. Mix it well. Whisk into broth and boil for 1 minute or until thickened.", "input_condition": ["have_cornstarch", "have_water", "have_cooked_broth_mixture", "have_bowl"], "task": [{ "action_name": "Stir cornstarch and 2 tbsp water.", "activity": ["stir"], "objects": ["cornstarch", "water"], "output_quality": ["Use a whisker to mix well."], "background_knowledge": { "tool": ["spatula"], "failure": ["cornstarch is not mixed well", "overused cornstarch", "no cornstarch"] } }, { "action_name": "Boil this mixture for 1 minute", "activity": ["boil"], "objects": ["cornstarch water mixture"], "output_quality": ["until thickened"], "background_knowledge": { "tool": ["mixing bowl", "whisker"], "failure": ["overcooked broth"] } }], "output_condition": ["have_broth_with_cornstarch"], "modality": { "image": ["/images/egg-drop-chicken-noodle-soup/egg-drop-chicken-noodle-soup_5_0.jpg", "/images/egg-drop-chicken-noodle-soup/egg-drop-chicken-noodle-soup_5_1.jpg"], "video": "" } }</pre>

Figure 1: Difference between textual representation and R3 for a single instruction



Figure 2: Data flow between TREAT and R3

ity characteristics for cheese) and corresponding image. **Q** is a tuple consisting of measurement and corresponding units. **A** captures the allergen information - ID to index all possible allergens, category of allergen to which the ingredient belongs, reference to the source from where this information is captured, and additional details for the allergen in the Knowledge Graph.

An instruction is broken down into atomic actions. **I** is a tuple consisting of the original recipe text from which the representation is being built, the input condition and output condition which define the requirements and changes that happen in the cooking environment once an instruction is performed, tasks (**T**) which captures the atomic actions and modality (**M**) which has visual information regarding

the instruction. The set **T** also contains information regarding the objects used and the activities being performed on them, output quality of the action, and background knowledge (**BK**). **BK** is a 2-tuple consisting of tools and failures associated with the atomic action. Figure 3 shows the skeletal structure of the R3. In an initial effort, we have manually created R3 for 25 egg-based recipes obtained from the RecipeQA dataset. We show the difference between the original representation and R3 for a single instruction of *Egg-drop Chicken Noodle Soup* in Figure 1. We now proceed to explain how we created a proof-of-concept corpus of recipes in R3 and how this representation is vital in building a prototypical recipe retrieval system - TREAT.

```

{
  "recipe-ids": {
    "0": {
      "recipe_name": "",
      "ingredients": [
        {
          "name": "",
          "quantity": {
            "measure": "",
            "unit": ""
          },
          "allergies": {
            "id": "",
            "category": "",
            "ref": "",
            "details": ""
          },
          "alternative": "",
          "quality_characteristic": "",
          "image": ""
        }
      ],
      "prep_time": "",
      "cook_time": "",
      "serves": "",
      "instructions": [
        {
          "original_text": "",
          "input_condition": [""],
          "task": [
            {
              "action_name": "",
              "activity": "",
              "objects": "",
              "output_quality": [""],
              "background_knowledge": {
                "tool": [""],
                "failure": [""],
              }
            }
          ],
          "output_condition": [""],
          "modality": {
            "image": [""],
            "video": ""
          }
        }
      ]
    }
  }
}

```

Figure 3: Skeletal structure of R3.

Populating R3

Most of the attributes in R3 are analogous to the data points present in RecipeQA. However, there are additional attributes which are modeled using manual curation of relevant data from Creative Commons. Figure 4 gives an overview of how each attribute is populated in R3, out of which, *allergies* needs more explanation.

For a given recipe, it is the ingredient(s) to which a consumer is allergic to and this would also help in making intelligent AI models. For instance, a current AI system might classify *pizza* as a dish containing *gluten* allergy, but that pizza might be completely made up of gluten-free ingredients. Thus, in R3 we focus on the possibility of allergies that can be faced by a consumer based on the ingredients. We have created an allergen lexicon with 17 different classes following the guidelines provided by the Institute of Agriculture and Natural Resources¹.

However, all the ingredients that belong to a particular allergen class are not present in the lexicon. Thus, we performed a comparative study of multiple word embedding methods (Word2Vec (Mikolov et al. 2013)/Concept-

¹<https://ianr.unl.edu/>

net Numberbatch (Speer, Chin, and Havasi 2017)/GloVe (Pennington, Socher, and Manning 2014)/Transformers (Vaswani et al. 2017)) to find similar ingredients to one’s already present in the allergen lexicon. Figure 5 summarizes the performance of the mentioned models. We see that Transformer outperforms the others in finding similar ingredients. For example, (Transformer) embeddings help us establish the similarity between an unseen ingredient such as *yolk* or *egg whites* with egg which is present in the allergen lexicon. This helps us in deriving the allergy information to unseen ingredients present in R3.

TREAT for Recipe Retrieval

R3 can be used in building decision-support tools in the food domain, out of which information retrieval is considered in this paper. We build a simple web-based tool called TREAT, which makes use of R3 to understand and answer user’s constrained queries - which can be either text, image or a combination of both. Figure 2 shows the overall data flow happening between R3 and the TREAT system based on the user query.

A user can query TREAT using multiple modalities (text, image, text + image) in order to perform constrained queries, which are of two types:

- **Process Constraints:** where the user puts restrictions on the cooking process, e.g., *Give me a recipe with less than 5 steps and is completed in 20 minutes* or *Suggest me a recipe which is not deep-fried*. For now, TREAT is equipped with the functionality to filter recipes only by length and time.
- **Outcome Constraints:** where the limitations set by the

R3 Attributes	Information Source
Recipe Name	Recipe QA dataset
Ingredients (IG)	Recipe QA dataset
Name Quantity (Q) Alternative Quality Characteristic Allergies (A) Image	Allergen Lexicon Creative Commons
Preparation Time, Cook Time, Serves	Recipe QA dataset
Instructions (I)	Recipe QA dataset
Original Text Input Condition Output Condition Modality (M)	Handcrafted Predicates Recipe QA dataset
Task (T)	Recipe QA dataset
Action Name Activity Objects Output Quality Background Knowledge (BK)	Human Curation

Figure 4: Information source for different attributes in R3.

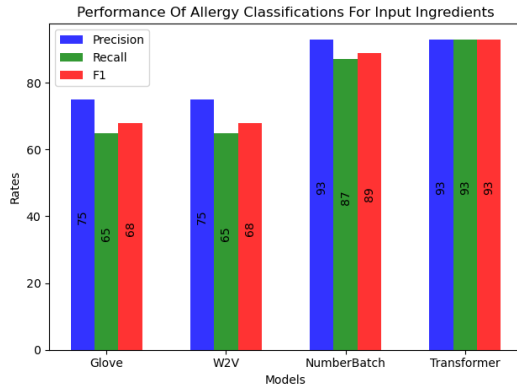


Figure 5: Comparison of different word embedding models for ingredient similarity.

user bring a change about in the general course of the recipe. For example, if the user asks for the recipe of a cake but sets a constraint that he is allergic to gluten, then the system is expected to retrieve gluten-free cake recipes. TREAT can answer user’s queries with limitations set on allergens, ingredients, and cuisine.

When the input is a string, the similarity metric employed to match with attributes in R3 is Levenshtein distance, where



Figure 6: User Interface of TREAT, showing a query where the user is searching for recipes without maize allergen



Figure 7: An image query using TREAT. The query is an image of bacon asking for recipes containing it.

the threshold for matching is a hyperparameter which is set to 0.7 by default. For image as an input, which can be of an ingredient or final cooked recipe, similarity is computed using the Scale Invariant Feature Transform (SIFT) algorithm (Lowe 2004). The algorithm locates local features in the images called key points and transforms them to vectors called descriptors. The similarity between two images is obtained by computing the euclidean distance between these descriptors. The similarity scores in both approaches are normalized and set to the same default value.

Figure 6 shows a demo query posed to TREAT where the user is seen performing a query with allergen constraint on the recipes - *Give me a recipe without maize allergen?* and the TREAT systems retrieves all 6 relevant recipes. Figure 7 shows an image query where an image of bacon is given as input to our system, it retrieves 3 recipes with bacon as an ingredient.

In this section, we presented the R3 representation and a tool which uses it for recipe retrieval. In the next section, we present a preliminary evaluation of TREAT to assess R3’s impact.

Experiments and Results

We want to establish that the representation is helpful in driving better recipes for users needs. We perform an evaluation of the TREAT system using 50 unique queries, consisting of both process and outcome constraints. The queries are generated based on two random functions - first selecting the parameters (such as allergen category, image, recipe name, ingredient) and the other selecting the value for those parameters (such as egg, maize category, scotch-eggs, parsley). This systematic process of generating queries avoids the risk of cherry-picking, and allows us to test our system on various edge cases.

Further, a manual process is used to obtain the ground truth values for each of the queries present. The ground truth is compared with the retrieved output from the system to calculate the evaluation metrics, which are **Coverage of Ground Truth (CVG)** and **Intersection Over Union (IOU)**.

- Coverage of Ground Truth (CVG): is defined as the intersection of ground truth and retrieved recipes over the ground truth recipe - Equation 1.

$$CVG = \frac{C(R_r \cap R_{gt})}{C_{gt}} \quad (1)$$

In the above equation, R_r, R_{gt} refers to the set of retrieved recipe and ground truth recipes for a given query, respec-

Query Type	Based on	CVG	IOU
Process Constraint	Length	1.00	1.00
	Ingredient	1.00	1.00
Output Constraint	Allergen	0.9	0.94
	Name	1.00	1.00
	Image	0.88	0.78

Table 1: Results of TREAT System

tively. The C represents length of the set of recipes and ϵR (set of real numbers).

- Intersection over Union (IOU): This is intersection of ground truth and retrieved recipe over the union of both - Equation 2.

$$IOU = \frac{C(R_r \cap R_{gt})}{C(R_r \cup R_{gt})} \quad (2)$$

In the above equation, R_r, R_{gt} refers to the set of retrieved recipe and ground truth recipes for a given query, respectively. The C represents length of the set of recipes and ϵR (set of real numbers).

Performance of the TREAT system

Table 1 shows the performance of the TREAT system on the test set of queries. Results relevant to the user’s search query are obtained using the similarity function mentioned in the Approach. We can see that TREAT is able to achieve a performance of 100% on queries pertaining to process constraints, owing the effectiveness of elaborate representation in R3. It can also be seen that the system is able to perform well on the output constraints.

Role of R3 in Improving Performance

Table 2 shows the a comparative study of the queries supported by R3 and the RecipeQA dataset (original) from which it is built. R3 acts as an enabler for the user to perform 3 additional queries, namely, allergen, image, and length based queries in addition to the queries supported by the original dataset.

Query Type	Original	Proposed
Allergen Based	No	Yes
Ingredient Based	Yes	Yes
Text data-modality	Yes	Yes
Image data-modality	No	Yes
Length Based	No	Yes
Name Based	Yes	Yes

Table 2: Different types of query supported by original and proposed data representation

We further evaluate the performance of the two types of representations using the test set of queries. Table 3 shows that the TREAT system can perform around 40% better than the original textual representation in enabling the user find answers for the queries.

Representation Type	CVG	IOU
Original	0.54	0.61
Proposed	0.96	0.94

Table 3: Performance Comparison of Original and Proposed representations

Support for Allergy Based Search

R3 is bound to achieve better performance than the original procedural text as it captures information in a machine-readable and understandable manner, similar to a Planning domain. Thus, this enables easier search for user queries. However, R3 can help answer a wider set of queries, out of which the queries related to allergens are of utmost importance. There has been extensive effort in building the allergy lexicon and use it to populate R3. Thus, we wanted to evaluate the performance of the TREAT system solely on allergy-constrained queries as well and also see how R3 would fare against the original recipe data.

We divide our allergy based queries into two categories - (1) where the user would *explicitly* mention the allergen in his query and (2) where the user just asks for a query without a particular allergy causing ingredient, thereby, having an *implicit* allergen information in the query. An example of an explicit and implicit allergen query can be *Give me a recipe without maize allergen* and *Give me a recipe without parsley* respectively. The original recipe representation, as Table 3 is incapable of performing explicit allergy based queries, but can perform implicit allergy queries to some extent, i.e., when the query consists of matching ingredients. Table 4 shows how R3 is able to outperform any kind of allergy based queries in comparison to the original dataset.

Query	Representation Type	CVG	IOU
Explicit allergen	Original	0.00	0.00
	Proposed	1.00	1.00
Implicit allergen	Original	0.16	0.24
	Proposed	0.9	0.94

Table 4: Allergen based query results

Conclusion and Future Work

In this work, we have curated a PDDL plan inspired representation for recipes with additional knowledge, i.e. R3 for building intelligent decision-support tools in the food domain. In order to validate the effectiveness of R3, we build a tool for recipe retrieval which can perform multi-modal reasoning. After experimental evaluation, we arrive at a conclusion that R3 leads to an improved performance in the task of information retrieval when compared to the original textual representation of recipes. In this initial phase of research, we have created R3 for 25 recipes using a semi-automated approach. However, the datasets in food domain have around a million recipes. Thus, scaling R3 using fully-automated approaches is our current ongoing effort. Having a large-scale R3 data would enable us to experiment and evaluate R3 on various fronts and the learnings might help the research community in exploring representations for other formats of procedural texts.

Beyond efficient information retrieval, a plan based representation could enable future automation in recipe creation by reusing existing recipes, reasoning on their content and composing, as necessary, to create new recipes based on

characteristics desired by the user. It can also help automation in food preparation by enabling monitoring the execution of agents by allowing reasoning with both the data and control flow inherent in the plan-based recipe representation consisting of text and images. These are avenues for future work.

References

- [Bäckström 1995] Bäckström, C. 1995. Expressive equivalence of planning formalisms. *Artificial Intelligence* 76(1):17–34. Planning and Scheduling.
- [Chatelan, Bochud, and Frohlich 2019] Chatelan, A.; Bochud, M.; and Frohlich, K. L. 2019. Precision nutrition: hype or hope for public health interventions to reduce obesity? *International journal of epidemiology* 48(2):332–342.
- [Chen and Ngo 2016] Chen, J., and Ngo, C.-W. 2016. Deep-based ingredient recognition for cooking recipe retrieval. In *Proceedings of the 24th ACM international conference on Multimedia*, 32–41.
- [Fikes and Nilsson 1971] Fikes, R. E., and Nilsson, N. J. 1971. Strips: A new approach to the application of theorem proving to problem solving. In *Proceedings of the 2nd International Joint Conference on Artificial Intelligence, IJCAI’71*, 608–620. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- [Fox and Long 2003] Fox, M., and Long, D. 2003. Pddl2.1: An extension to pddl for expressing temporal planning domains. *Journal of Artificial Intelligence Research* 20:61–124.
- [Frummet, Elswailer, and Ludwig 2022] Frummet, A.; Elswailer, D.; and Ludwig, B. 2022. “what can i cook with these ingredients?”-understanding cooking-related information needs in conversational search. *ACM Transactions on Information Systems (TOIS)* 40(4):1–32.
- [Hausmann et al. 2019] Hausmann, S.; Seneviratne, O.; Chen, Y.; Ne’eman, Y.; Codella, J.; Chen, C.-H.; McGuinness, D. L.; and Zaki, M. J. 2019. Foodkg: a semantics-driven knowledge graph for food recommendation. In *International Semantic Web Conference*, 146–162. Springer.
- [Howey and Long 2003] Howey, R., and Long, D. 2003. Val’s progress: The automatic validation tool for pddl2.1 used in the international planning competition. In *ICAPS 2003 workshop on “The Competition: Impact, Organization, Evaluation, Benchmarks”*, Trento, Italy.
- [Jiang and Min 2020] Jiang, S., and Min, W. 2020. Food computing for multimedia. In *Proceedings of the 28th ACM International Conference on Multimedia*, 4782–4784.
- [Kim et al. 2021] Kim, K.; Park, D.; Spranger, M.; Maruyama, K.; and Kang, J. 2021. Recipebowl: A cooking recommender for ingredients and recipes using set transformer. *IEEE Access* 9:143623–143633.
- [Leroux and London 2002] Leroux, K., and London, D. 2002. Flavour-changing neutral currents and leptophobic z gauge bosons. *Physics Letters B* 526(1-2):97–103.
- [Li and Zaki 2020] Li, D., and Zaki, M. J. 2020. Recipitor: An effective pretrained model for recipe representation learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1719–1727.
- [Lien, Zamani, and Croft 2020] Lien, Y.-C.; Zamani, H.; and Croft, W. B. 2020. Recipe retrieval with visual query of ingredients. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1565–1568.
- [Lowe 2004] Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision* 60(2):91–110.
- [McDermott et al. 1998] McDermott, D.; Ghallab, M.; Knoblock, C.; Wilkins, D.; Barrett, A.; Christianson, D.; Friedman, M.; Kwok, C.; Golden, K.; Penberthy, S.; Smith, D.; Sun, Y.; and Weld, D. 1998. PDDL - the planning domain definition language. Technical report, Technical Report.
- [Mikolov et al. 2013] Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- [Min et al. 2019] Min, W.; Jiang, S.; Liu, L.; Rui, Y.; and Jain, R. 2019. A survey on food computing. *ACM Computing Surveys (CSUR)* 52(5):1–36.
- [Mori et al. 2014] Mori, S.; Maeta, H.; Yamakata, Y.; and Sasada, T. 2014. Flow graph corpus from recipe texts. In *LREC*, 2370–2377.
- [Myers and Lee 1999] Myers, K. L., and Lee, T. J. 1999. Generating qualitatively different plans through metatheoretic biases. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence, AAAI ’99/IAAI ’99*, 570–576. USA: American Association for Artificial Intelligence.
- [Pan et al. 2020] Pan, L.-M.; Chen, J.; Wu, J.; Liu, S.; Ngo, C.-W.; Kan, M.-Y.; Jiang, Y.; and Chua, T.-S. 2020. Multimodal cooking workflow construction for food recipes. In *Proceedings of the 28th ACM International Conference on Multimedia*, 1132–1141.
- [Pednault 1994] Pednault, E. P. D. 1994. ADL and the State-Transition Model of Action. *Journal of Logic and Computation* 4(5):467–512.
- [Pennington, Socher, and Manning 2014] Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.
- [Ramadan and El-Jaroudi 2022] Ramadan, M., and El-Jaroudi, A. 2022. Action detection and classification in kitchen activities videos using graph decoding. *The Visual Computer* 1–14.
- [Roither, Kurz, and Sonnleitner 2022] Roither, A.; Kurz, M.; and Sonnleitner, E. 2022. The chef’s choice: System for

allergen and style classification in recipes. *Applied Sciences* 12(5):2590.

[Shirai et al. 2021] Shirai, S. S.; Seneviratne, O.; Gordon, M. E.; Chen, C.-H.; and McGuinness, D. L. 2021. Identifying ingredient substitutions using a knowledge graph of food. *Frontiers in Artificial Intelligence* 111.

[Song, Yang, and Xu 2022] Song, Y.; Yang, X.; and Xu, C. 2022. Self-supervised calorie-aware heterogeneous graph networks for food recommendation. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*.

[Speer, Chin, and Havasi 2017] Speer, R.; Chin, J.; and Havasi, C. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

[Srivastava, Vanhatalo, and Koehler 2005] Srivastava, B.; Vanhatalo, J.; and Koehler, J. 2005. Managing the life cycle of plans. In *Proceedings of the 17th Conference on Innovative Applications of Artificial Intelligence - Volume 3, IAAI'05*, 1569–1575. AAAI Press.

[Tian et al. 2022] Tian, Y.; Zhang, C.; Metoyer, R.; and Chawla, N. V. 2022. Recipe recommendation with hierarchical graph attention network. *Frontiers in big Data* 4:778417.

[Vaswani et al. 2017] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems* 30.

[Wang and Hu 2018] Wang, D. D., and Hu, F. B. 2018. Precision nutrition for prevention and management of type 2 diabetes. *The lancet Diabetes & endocrinology* 6(5):416–426.

[Yagcioglu et al. 2018] Yagcioglu, S.; Erdem, A.; Erdem, E.; and Ikizler-Cinbis, N. 2018. Recipeqa: A challenge dataset for multimodal comprehension of cooking recipes.

[Zhu et al. 2019] Zhu, B.; Ngo, C.-W.; Chen, J.; and Hao, Y. 2019. R2gan: Cross-modal recipe retrieval with generative adversarial network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11477–11486.