FoodIE: A Rule-based Named-entity Recognition Method for Food Information Extraction

Gorjan Popovski[1], Stefan Kochev[1], Barbara Koroušić Seljak[2] and Tome Eftimov[2]
[1]*Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Rugjer Boshkovikj 16, 1000 Skopje, Macedonia*
[2]*Computer Systems Department, Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia*

Keywords: Information Extraction, Rule-based Named-entity Recognition, Food-entity Recognition.

Abstract: The application of Natural Language Processing (NLP) methods and resources to biomedical textual data has received growing attention over the past years. Previously organized biomedical NLP-shared tasks (such as, for example, BioNLP Shared Tasks) are related to extracting different biomedical entities (like genes, phenotypes, drugs, diseases, chemical entities) and finding relations between them. However, to the best of our knowledge there are limited NLP methods that can be used for information extraction of entities related to food concepts. For this reason, to extract food entities from unstructured textual data, we propose a rule-based named-entity recognition method for food information extraction, called FoodIE. It is comprised of a small number of rules based on computational linguistics and semantic information that describe the food entities. Experimental results from the evaluation performed using two different datasets showed that very promising results can be achieved. The proposed method achieved 97% precision, 94% recall, and 96% F_1 score.

1 INTRODUCTION

Nowadays, a large amount of textual information is available in digital form and published in public web repositories (e.g., online news, scientific publications, social media). The textual information is presented as unstructured data, meaning that the data has no predefined data model. Working with textual data is a challenge because of its variability - the same concepts can be mentioned in different ways regarding the fact how people express themselves and use different writing styles.

Information Extraction (IE) is a task of automatically extracting information from unstructured data and, in most cases, is concerned with the processing of human language text by means of natural language processing (NLP) (Aggarwal and Zhai, 2012). The idea behind IE is to provide a structured representation of extracted information obtained from analyzed text. The information to be extracted is defined by users, and consists of predefined concepts of interest and related entities, as well as relationships between entities and events.

One of the classic IE tasks is named-entity recognition (NER), which addresses the problem of identification and classification of predefined concepts (Nadeau and Sekine, 2007). It aims to determine and identify words or phrases in text into predefined labels (classes) that describe concepts of interest in a given domain. Various NER methods exist: *terminological-driven*, *rule-based*, *corpus-based*, *methods based on active learning (AL)*, and *methods based on deep neural networks (DNNs)*.

In this paper, we focus on IE of food entities. To the best of our knowledge, not a large amount of research focusing on food entities has been done. However, nowadays, the knowledge about extracted food entities and their relations with other biomedical entities (like genes, drugs, diseases, etc.) is important for improving public health.

The main contributions of this paper are:

- A rule-based NER method for IE of food entities.
- Evaluation of the proposed method, which provides promising results on unstructured data, without a need for an annotated corpus.

In the remainder of the paper, we first present an overview of the related work. Then, we present the proposed rule-based NER method for IE of food entities. Next, the data used for evaluation is explained, followed by the results and discussion. Finally, the conclusions of the paper and a discussion for future work are presented.

Popovski, G., Kochev, S., Seljak, B. and Eftimov, T.
FoodIE: A Rule-based Named-entity Recognition Method for Food Information Extraction.
DOI: 10.5220/0007686309150922
In *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2019)*, pages 915-922
ISBN: 978-989-758-351-3

2 RELATED WORK

IE from biomedical literature is a very important task with the goal of improving public health. Because NER methods which have the best performances are usually corpus-based NER methods, there is a need for an annotated corpus from biomedical literature that includes the entities of interest. For this purpose, different annotated corpora are produced by shared tasks, where the main aim is to challenge and encourage research teams on NLP problems.

In comparison with the extensive work done for biomedical tasks, in the food science domain the situation is different. Several studies have been conducted, but with different goals. For example, in (Xia et al., 2013) authors presented an approach to identify rice protein resistant to *Xanthomonas oryzae pv. oryzae*, which is an approach to enhance gene prioritization by combining text mining technologies with a sequence-based approach. Co-occurrence methods were also used to identify ingredients mentioned in food labels and extracting food-chemical and food-disease relationship (do Nascimento et al., 2013; Jensen et al., 2014).

A ML approach to Japanese recipe text processing was proposed in (Mori et al., 2012), where one task, which was evaluated, was food-named entity recognition. This approach used the r-FG corpus, which is composed solely from Japanese food recipes. Another similar approach for generating graph structures from food recipes was proposed in (Chen, 2017), where authors manually annotated a recipe corpus that is then used for training a ML model.

The UCREL Semantic Analysis System (USAS) is a framework for automatic semantic analysis of text, which distinguishes between 21 major categories, one of which is “food and farming” (Rayson et al., 2004), being heavily utilized in our rule-based system - FoodIE. The USAS can provide additional information about the food entity, but the limitation is that it works on a token level. For example, if in the text two words (i.e. tokens), like “grilled chicken”, denote one food entity that needs to be extracted and analyzed, the semantic tagger would actually parse the words “grilled” and “chicken” as separate entities and obtain separate semantic tags.

In (Eftimov et al., 2017), a rule-based NER used for IE from evidence-based dietary recommendation, called drNER, is presented, where among other entities, food entities were also of interest.

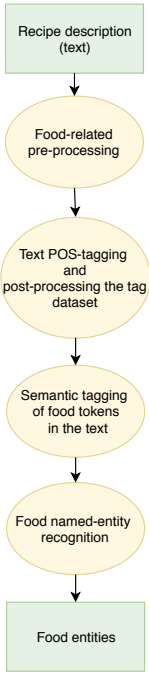

Figure 1: The flowchart of the Foodie methodology.

3 FOODIE: A RULE-BASED FOOD-NAMED ENTITY RECOGNITION

To enable food-named entity recognition, in this paper, we propose a rule-based approach, called FoodIE. It works with unstructured data (more specifically, with a recipe that includes textual data in form of instructions on how to prepare the dish) and consists of four steps:

- Food-related text pre-processing
- Text POS-tagging and post-processing of the tag dataset
- Semantic tagging of food tokens in the text
- Food-named entity recognition

The flowchart of the methodology is presented in Figure 1. Further, we are going to explain each part in more detail.

3.1 Food-related Text Pre-processing

The pre-processing step takes into account the discrepancies that exist between the outputs of the taggers we are utilizing, *coreNLP tagger* from the R programming language (Arnold and Tilton, 2016) and the *UCREL Semantic Analysis System (USAS)* (Rayson et al., 2004). It is also used to remove any characters that are unknown to the taggers.

Firstly, quotation marks should be removed from the raw text, for the simple reason that they are treated differently by both used NLP libraries, causing a discrepancy.

Secondly, every white space sequence (including tabulation, newlines, etc.) is converted into a single white space to provide a consistent structure to the text.

Additionally, ASCII transliteration is performed, which means characters that are equivalent to ASCII characters are transliterated. An example of such characters is [è, ö, à], which are transliterated to [e, o, a], respectively.

Finally, fractions should be converted into real numbers. Usually, when a food-related text is written (e.g., recipe), fractions are used when discussing quantities. However, they are usually written in plain ASCII format and in a manner which is confusing to NLP taggers. For example, "2.5" is usually written as "2 1/2" in such texts. This does not bode well with *coreNLP* and the *USAS semantic tagger*. Thus, in the pre-processing step, all fractions are converted into the standard mathematical decimal notation for real numbers.

3.2 Text POS-tagging and Post-processing of the Tag Set

To obtain the morphological information from a textual data, we use UCREL Semantic Analysis System (USAS) and coreNLP.

The USAS semantic tagger provides word tokens associated with their POS tags, lemmas, and semantic tags. The semantic tags show semantic fields that group together word senses that are related at some level of generality with the same contextual concept. The groups include not only synonyms and antonyms but also hypernyms and hyponyms. More details about semantic tags can be found in (Rayson et al., 2004; Alexander and Anderson, 2012).

Furthermore, the same is done using the coreNLP library, which includes all of the above except semantic tags.

For example, the sentence "Heat the beef soup until it boils" is processed by both libraries. The results from the coreNLP library for the above mentioned example sentence are presented in Table 1, while the results from USAS are presented in Table 2. Observing the results presented in the tables, it is obvious that there is a discrepancy between the POS tags for the token "Heat".

As is evident, both the USAS semantic tagger and the coreNLP library, do not provide perfect tags (e.g., sometimes verbs are misclassified as nouns, as is the case with the first token in the example given in Table 1). For this reason, the tags returned by both taggers are post-processed and modified using the following linguistic rules:

Table 1: Tags obtained from coreNLP for one recipe sentence.

Token ID	Token	Lemma	POS tag
1	Heat	heat	NN
2	the	the	DT
3	beef	beef	NN
4	soup	soup	NN
5	until	until	IN
6	it	it	PRP
7	boils	boil	VBZ
8	.	.	.

- If at least one of the taggers classify a token as a verb, mark it as a verb.
- If there exists a discrepancy between the tags for a specific token, prioritize the tag given by the USAS semantic tagger.
- If a past participle form or a past simple form of a verb precedes and is adjacent to a noun, and it is classified as a verb, change the tag from verb to adjective.

Finally, we keep two versions of the modified tag set, one in each format. These modified tags in the coreNLP format and USAS format are presented in Table 3 and Table 4, respectively.

3.3 Semantic Tagging of Food Tokens in Text

To define phrases in the text related to food entities, we first need to find tokens that are related to food entities. For this purpose, the USAS semantic tagger is utilized. Using it, a specific rule is defined to determine the food tokens in the text. Food tokens are predominantly nouns or adjectives, so we account for this as to improve the false positive rate, i.e. allowing a token to be categorized as a food token if and only if it is either a noun or an adjective. The decision rule combines three conditions using the following Boolean expression $((Condition_1\ OR\ Condition_2)\ AND\ Condition_3)$. If the expression is true, then the token is classified as food token. For clarity, let us assume that t is a token and s_t is the semantic tag that is assigned to it using the USAS semantic tagger. Each condition is constructed using the following rules:

- $Condition_1$:
  - Food tag F(1|2|3|4), or

Table 2: Tags obtained from USAS for one recipe sentence.

Token ID	Token	Lemma	POS tag	Semantic tag 1	Semantic tag2
1	Heat	heat	VV0	O4.6+	AJ.03.c.02 [Heat]; AJ.03.c.02 [Heat]; AJ.03.c.02.a [Heating/making hot/warm];
2	the	the	AT	Z5	ZC [Grammatical Item];
3	beef	beef	NN1	F1	AG.01.d.03 [Beef]; AE.14.m.03 [Subfamily Bovinae (bovines)]; AE.14.m.03 [Subfamily Bovinae (bovines)];
4	soup	soup	NN1	F1	AG.01.n.02 [Soup/pottage]; AA.04.g.04 [Wave]; AA.11.h [Cloud];
5	until	until	CS	Z5	ZC [Grammatical Item];
6	it	it	PPH1	Z8	ZF [Pronoun];
7	boils	boil	VVZ	O4.6+ E3-	AJ.03.c.02.b [Action of boiling]; AJ.03.c.02.b [Action of boiling]; AJ.03.c.02.b [Action of boiling];
8	.	PUNC	YSTP	PUNC	NULL

Table 3: Modified tags from coreNLP for one recipe sentence.

Token ID	Token	Lemma	POS tag
1	Heat	heat	VB
2	the	the	DT
3	beef	beef	NN
4	soup	soup	NN
5	until	until	IN
6	it	it	PRP
7	boils	boil	VBZ
8	.	.	.

  - Living tag L(2|3), or
  - Substance tag (liquid and solid) O1.(1|2).
- $Condition_2$:
  - Body part tag B1, and
  - Not Linear order tag N4, and
  - Not Location and direction tag M6, and
  - Not Texture tag O4.5.
- $Condition_3$:
  - Not General Object tag O2, and
  - Not Quantities tag N5, and
  - Not Clothing tag B5, and
  - Not Equipment for food preparation tag AG.01.t.08, and
  - Not Container for food, place for storing food tag AG.01.u, and
  - Not Clothing tag AH.02.

More formally, using Boolean algebra, we can write these rules as:

$Condition_1$:

$s_t \in \{\text{F1},\text{F2},\text{F3},\text{F4}\} \vee s_t \in \{\text{L2},\text{L3}\} \vee s_t \in \{\text{O1.1},\text{O1.2}\}$

$Condition_2$:

$s_t = \text{B1} \wedge s_t \neq \text{N4} \wedge s_t \neq \text{M6} \wedge s_t \neq \text{O4.5}$

$Condition_3$:

$s_t \neq \text{O2} \wedge s_t \neq \text{N5} \wedge s_t \neq \text{B5} \wedge s_t \neq \text{AG.01.t.08} \wedge s_t \neq \text{AG.01.u} \wedge s_t \neq \text{AH.02}.$

$Rule_1$:

$(Condition_1 \vee Condition_2) \wedge Condition_3$

Additionally, we define one rule to determine object tokens. Determining the object tokens will further help us in the definition of food entities, mainly to avoid false positives. The rule consists of

- General Object tag O2, or
- Clothing tag B5, and
- Not Body Part tag B1, and
- Not Living tag L(2|3), and
- Not a food token as defined by the aforementioned first rule.

Using Boolean algebra, this rule is represented as

$Rule_2$:
$(s_t = \text{O2} \vee s_t = \text{B5}) \wedge s_t \neq \text{B1} \wedge s_t \neq \text{L2} \wedge s_t \neq \text{L3} \wedge \neg \text{Rule}_1$.

If this condition is met, the token is tagged as general object.

The single rule for defining color noun is consisted of

- Color tag O4.3.

The rule for defining a color noun is then formally defined as

$Rule_3$:

$s_t = \text{O4.3}.$

These tags are useful when food entities ending on a color, such as “egg whites” or “hash browns”, appear in the text, which indeed are to be treated as food entities.

At the end, one additional rule is constructed for defining what is explicitly disallowed to be the main token in a food entity, and is defined as

- Equipment for food preparation AG.01.t.08, and

Table 4: Modified tags from USAS for one recipe sentence.

Token ID	Token	Lemma	POS tag	Semantic Tag 1	Semantic tag 2
1	Heat	heat	VV0	O4.6+	AJ.03.c.02 [Heat]; AJ.03.c.02 [Heat]; AJ.03.c.02.a [Heating/making hot/warm];
2	the	the	AT	Z5	ZC [Grammatical Item];
3	beef	beef	NN1	F1	AG.01.d.03 [Beef]; AE.14.m.03 [Subfamily Bovinae (bovines)]; AE.14.m.03 [Subfamily Bovinae (bovines)];
4	soup	soup	NN1	F1	AG.01.n.02 [Soup/pottage]; AA.04.g.04 [Wave]; AA.11.h [Cloud];
5	until	until	CS	Z5	ZC [Grammatical Item];
6	it	it	PPH1	Z8	ZF [Pronoun];
7	boils	boil	VVZ	O4.6+ E3-	AJ.03.c.02.b [Action of boiling]; AJ.03.c.02.b [Action of boiling]; AJ.03.c.02.b [Action of boiling];
8	.	PUNC	YSTP	PUNC	NULL

- Container for food, place for storing food AG.01.u, and
- Clothing tag AH.02, and
- Temperature tag O4.6, and
- Measurement tag N3.

  This rule can be represented as

$Rule_4$:

$s_t = \text{AG.01.t.08} \wedge s_t = \text{AG.01.u} \wedge s_t = \text{AH.02} \wedge s_t = \text{O4.6} \wedge s_t = \text{N3}$.

This rule is utilized when isolating entities that could be potential false positives. An example of this would be “oil temperature” or “cake pan”. Additionally, there are some manually added resources in this disallowed category, which frequently occur in the texts.

3.4 Food-named Entity Recognition

To obtain food chunks, we used the modified tag set from the USAS semantic tagger obtained in Subsection 3.2 in combination with the food tokens obtained in Subsection 3.3. The process of food-named entity recognition consists of three steps.

Firstly, we iterate through every food token which we extracted previously from the text, and for each token we define a set of rules that constitute a food entity.

Adjacent to the left of the food token we allow chaining of adjectives (JJ), nouns (NN), proper nouns (NP), genitive tag (GE), unknown tags (Z99) and general tokens tagged as food, but explicitly omit general objects. The purpose of including the unknown POS tag (Z99) is to catch tokens that do not concisely fall into one of the tags in the standard POS tag set, yet still are of importance to the semantics of the food entity. Such an example would be “Colby-Jack cheese”, whose POS tags are Z99 and NN, respectively.

Adjacent to the right the logic is the same, differing only by allowing general object to be part of the food entity and tokens that have been been tagged as a color noun by the rule engine. We also keep track not to use a token twice.

Then, to determine if it truly is a food entity chunk or just a chunk related to food but not a food entity in and of itself, we check the last token of the chunk. The whole chunk is discarded if the last token is:

- A noun (starts with NN) and a general non-food object, or
- in the disallowed category as defined by the rule engine, or
- in the disallowed category as defined by the resources.

Some examples where this would be a false positive are “muffin liner”, “casserole dish” or “egg timer”. If this check passes and the last token is not a general object, we mark each token in the new food chunk with an index unique to the whole chunk and continue iterating through the remaining food tokens.

After the first step, we now must concatenate all relevant information for each food entity. For each indexed food entity, we join all the instances into one entry, thus creating a vector where each token is its own entry, except for the food entities which are represented as one entry. If initially we had a vector of tokens such as [Chop, the, hot, Italian, sausage, into, pieces, .] the output would be [Chop, the, hot Italian sausage, into, pieces, .]. This also applies to other relevant information we might want to track, such as lemmas, POS tags, sentence indexes or even individual token indexes.

For additional robustness, we perform a check to assure that each food chunk we have isolated indeed contains a food token, and that the token is marked under some food chunk. For this we only mark a chunk as a food entity if it contains at least one word that has previously been tagged as a food token and has been indexed as part of the respective chunk as well.

4 EVALUATION

The evaluation was performed manually, since there is no pre-existing method to evaluate such a text corpus. To avoid any kind of bias when evaluating food-related text, one person was tasked with manually

performing food chunk extraction from each individual text, while another person cross referenced those manually obtained chunks with the ones obtained from FoodIE. Using this method, a figure for true positives (TPs), false negatives (FNs) and false positives (FPs) was procured, while it was decided that the category true negative was not applicable to the nature of the problem and its evaluation. Additionally, it was decided that a "partial (inconclusive)" category was necessary, as some of the food chunks were incomplete, but nevertheless caught, thus including significant information. This category encompasses all the extracted food chunks which were caught, but missed at least one token. An example would be "bell pepper", where FoodIE would only catch "pepper".

We would like to compare the results using the model presented in (Chen, 2017), but we were unable to obtain the requested model and corpus. We provide a small example of comparing FoodIE with drNER (Eftimov et al., 2017), in order to show that they provide food entities on different level, so a fair comparison cannot be made.

While the evaluation was being done, we kept track of all the False Negative instances and have constructed a resource set that will improve the performance of FoodIE in future implementations.

4.1 Data

Firstly, a total of 200 recipes were processed and evaluated. The original 100 recipes, which were analyzed and upon which the rule engine was built, were taken into consideration, as well as 100 new recipes which had not been analyzed beforehand. The recipes were taken from two separate user-based sites, Allrecipes (https://www.allrecipes.com/) and MyRecipes (https://www.myrecipes.com/), where there is no standardized format for the recipe description. This was chosen as such to ensure that the linguistic constructs utilized in each written piece varied and had no pattern behind them. The texts were chosen from a variety of topics, as to provide further diversity.

Secondly, we selected 1,000 independently obtained recipes from Allrecipes (Groves, 2013), which is the largest food-focused social network, where everyone plays part in helping cooks discover and share home cooking. We selected the Allrecipes because there is no limitation as to who can post recipes, so we have variability in how users express themselves. The recipes were selected from five recipe categories: Appetizers and snacks, Breakfast and Lunch, Dessert, Dinner, and Drinks. From each recipe category 200 recipes were included in the evaluation set.

The evaluation datasets, including the obtained results, are publicly available at http://cs.ijs.si/repository/FoodIE/FoodIE_datasets.zip.

4.2 Results and Discussion

The results for TPs, FPs, and FNs of evaluating the FoodIE using the dataset of 200 recipes are presented in Table 5. The group "Partial (Inconclusive)" was left out of these evaluations, as some would argue they should be counted as TPs, while other that they should be included in the FNs. Some examples included here are: "empty passion fruit juice", "cinnamon" and "soda", where the actual food entity chunks would be "passion fruit juice", "cinnamon sticks" and "club soda", respectively. These are mostly due to the dual nature of words, meaning that a word that is a synonym of both a noun and a verb or an adjective and a verb, occur. For such words, the tagger sometimes incorrectly classifies the tokens. In these examples, "empty" is tagged as an adjective, where in context it, in fact, is a verb. The same explanation holds for the other two examples. For these reasons, when the evaluation metrics were calculated, this category was simply omitted. Moreover, even if they are grouped with either TPs or FNs, this does not significantly affect the results.

Regarding the FN category (type II error), there were some specific patterns that produced the most instances. One very simple type of a FN instance is where the author of the text refers to a specific food using the brand name, such as "allspice" or "Jägermeister". These are difficult to catch if there is no additional information following the brand name. However, if the user includes the general classification of the branded food, FoodIE will catch it. An example of this would be by simply writing "Jägermeister liqueur". Another instance of a type II error is when the POS taggers give incorrect tags, as was the case with some "Partial (Inconclusive)" instances. An example of this is when the tagger misses chunks such as "mint leaves" and "sweet glazes", where both "leaves" and "glazes" are incorrectly classified as verbs when in this context they should be tagged as nouns. Another example would be when the semantic tagger incorrectly classifies some token within the given context, such as "date" being classified as a noun meaning day of year, as opposed to it being a certain fruit. Furthermore, there exist FNs which are simply due to the rarity of the food, such as "kefir", "couscous" or "stevia", the last one being of immense importance to people suffering from diabetes, as it is a safe sugar substitute. Another category of type II errors is due to the fact that some foods are often referred by their colloquial name, such as "half-

and-half" and "spring greens". The final category of this type of error is where there exist spelling variations for a single food, such as "eggnog", "egg nog", "egg-nog". These are very difficult, if not impossible, to correctly predict since grammatical and morphological styles vary with each user, which extend as far as including simply improper use of the English language. This is a separate problem in and of itself, i.e. spellchecking and spelling correction.

The second type of error to discuss is the FP category (type I error), which is often due to the existence of objects that are not foods, but are closely related to food entities. These include instances such as "dollop" or "milk frother", where the first example has a meaning very closely related to food, thus making it difficult to distinguish using the semantic tags. The second chunk is simply an instrument related to food and cooking, while being rare enough such that the semantic tagger does not classify it properly as an object.

Table 5: Predictions (200 recipes).

True Positive (TP)	3063
False Positive (FP)	75
False Negative (FN)	185
Partial (Inconclusive)	97

Using the results reported in Table 5, the evaluation metrics for F_1 score, precision, and recall, are presented in Table 6.

Table 6: Evaluation metrics (200 recipes).

F_1 Score	Precision	Recall
0.9593	0.9761	0.9430

The results from evaluation the FoodIE on the dataset with 1000 recipes are reported in tables 7 and 8.

Table 7: Predictions (1000 recipes).

True Positive (TP)	11461
False Positive (FP)	258
False Negative (FN)	684
Partial (Inconclusive)	359

Comparing the results obtained from the evaluations (tables 6 and 8), we can conclude that FoodIE behaves consistently. Evaluating the dataset with 200 recipes, which consists of 100 recipes that were analyzed to build the rule engine and 100 new recipes that were not analyzed beforehand, we obtained 0.9761 precision, 0.9430 recall, and 0.9593 F_1 score. Furthermore, by evaluating it on a dataset that consists of 1000 new recipes, it obtained 0.9780 for precision, 0.9437 for recall, and 0.9605 for F_1 score. Comparing these results provides that FoodIE gives very promising and consistent results.

Table 8: Evaluation metrics (1000 recipes).

F_1 Score	Precision	Recall
0.9605	0.9780	0.9437

We also provided the TPs, FPs, FNs, and Partial predictions, together with the evaluation metrics for each recipe category separately (Table 9). Using them, we can see that Dinner category provides most FNs (223), while the Breakfast/lunch category provides the least FNs (82). Regarding the FNs, the Breakfast/lunch category provides the most FPs (108), while the Drinks category provides the least FPs (31). Looking at the results, it is evident that FoodIE retains the aforementioned consistency, even when comparing the evaluation metrics from each category between themselves.

Table 9: Predictions and evaluation metrics for each recipe category.

Recipe category	TP	FP	FN	Partial	F_1 Score	Precision	Recall
Appetizers/snacks	2147	27	162	45	0.9578	0.9876	0.9298
Breakfast/lunch	2443	33	82	108	0.9770	0.9876	0.9675
Desserts	2612	87	127	124	0.9607	0.9678	0.9536
Dinner	3176	47	223	51	0.9592	0.9854	0.9344
Drinks	1083	64	90	31	0.9336	0.9442	0.9233

In Table 10, we present the results obtained for 10 sentences (i.e evidence-based dietary recommendations) previously used in (Eftimov et al., 2016; Eftimov et al., 2017), in order to present the difference between FoodIE and drNER. Semicolon was used to split separate food entities. Using the table, we can see that drNER and FoodIE provide results on a different level. For example, let us consider the sixth recommendation. drNER extracted only one food entity, which is "Milk, cheese, yogurt and other dairy products", while FoodIE extracted four separate food entities, i.e. "Milk", "cheese", "yogurt", and "other dairy products". From this, it follows that FoodIE provides more precise results, which means it can also be used as a post-processing tool for drNER in order to extract the food entities on a individual level.

The performance of the rule-based system FoodIE heavily depends on the taggers used, so the improvement of the qualities of the POS-tagging and semantic tagging methods will also improve the evaluation metrics for FoodIE.

5 CONCLUSIONS

To extract food entities from unstructured textual data, we propose a rule-based named-entity recognition method for food information extraction, called FoodIE. It is a rule engine, where the rules are

Table 10: Food entities extracted by drNER and FoodIE.

	Recommendation	drNER	FoodIE
1.	Good sources of magnesium are: fruits or vegetables, nuts, peas and beans, soy products, whole grains and milk.	fruits or vegetables, nuts, peas and beans; soy products; whole grains and milk	fruits; vegetables; nuts; peas; beans; whole grains; milk
2.	The RDAs for Mg are 300 mg for young women and 350 mg for young men.	-	-
3.	Increase potassium by ordering a salad, extra steamed or roasted vegetables, bean-based dishes fruit salads, and low-fat milk instead of soda.	salad; extra steamed or roasted vegetables; fruit salads; low-fat milk	salad; roasted vegetables; bean-based dishes; fruit salads; low-fat milk; soda
4.	Babies need protein about 10 g a day.	-	-
5.	1 teaspoon of table salt contains 2300 mg of sodium.	table salt	table salt
6.	Milk, cheese, yogurt and other dairy products are good sources of calcium and protein, plus many other vitamins and minerals.	Milk, cheese, yogurt and other dairy products	Milk; cheese; yogurt; other dairy products
7.	Breast milk provides sufficient zinc, 2 mg/day for the first 4-6 months of life.	Breast milk	milk
8.	If you're trying to get more omega-3, you might choose salmon, tuna or eggs enriched with omega-3.	salmon, tuna; eggs	salmon; tuna; eggs
9.	If you need to get more fiber, look to beans, vegetables, nuts and legumes.	beans, vegetables, nuts, and legumes	beans; vegetables; nuts; legumes
10.	Excellent sources of alpha-linolenic acid, ALA, include flaxseeds and walnuts.	flaxseeds and walnuts	alpha-linolenic acid; flaxseeds; walnuts

based on computational linguistics and semantic information that describe the food entities. Evaluation showed that FoodIE behaves consistently using different independent evaluation datasets and very promising results have been achieved.

To the best of our knowledge, there is a limited number of NLP tools that can be used for IE of food entities. Moreover, there is a lack of annotated corpora that can be used to train corpus-based NER methods. Motivated by the evaluation results obtained, we are planning to use it in order to build an annotated corpus that can be further used for extracting food entities together with their relations to other biomedical entities. By performing this, we can easily follow the new knowledge that comes rapidly with each day with new scientifically published papers aimed at improving public health.

ACKNOWLEDGEMENTS

This work was supported by the Slovenian Research Agency Program P2-0098 and ERA Chair ISO-FOOD for isotope techniques in food quality, safety and traceability [grant agreement no. 621329].

REFERENCES

Aggarwal, C. C. and Zhai, C. (2012). *Mining text data*. Springer Science & Business Media.

Alexander, M. and Anderson, J. (2012). The hansard corpus, 1803-2003.

Arnold, T. and Tilton, L. (2016). *coreNLP: Wrappers Around Stanford CoreNLP Tools*. R package version 0.4-2.

Chen, Y. (2017). *A Statistical Machine Learning Approach to Generating Graph Structures from Food Recipes*. PhD thesis.

do Nascimento, A. B., Fiates, G. M. R., dos Anjos, A., and Teixeira, E. (2013). Analysis of ingredient lists of commercially available gluten-free and gluten-containing food products using the text mining technique. *International journal of food sciences and nutrition*, 64(2):217–222.

Eftimov, T., Koroušić Seljak, B., and Korošec, P. (2017). A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. *PloS One*, 12(6):e0179488.

Eftimov, T., Koroušić Seljak, B., and Korošec, P. (2016). Grammar and dictionary based named-entity linking for knowledge extraction of evidence-based dietary recommendations. In *KDIR*, pages 150–157.

Groves, S. (2013). How allrecipes. com became the worlds largest food/recipe site. roi of social media (blog).

Jensen, K., Panagiotou, G., and Kouskoumvekaki, I. (2014). Integrated text mining and chemoinformatics analysis associates diet to health benefit at molecular level. *PLoS computational biology*, 10(1):e1003432.

Mori, S., Sasada, T., Yamakata, Y., and Yoshino, K. (2012). A machine learning approach to recipe text processing. In *Proceedings of the 1st Cooking with Computer Workshop*, pages 29–34.

Nadeau, D. and Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Rayson, P., Archer, D., Piao, S., and McEnery, A. (2004). The ucrel semantic analysis system.

Xia, J., Zhang, X., Yuan, D., Chen, L., Webster, J., and Fang, A. C. (2013). Gene prioritization of resistant rice gene against xanthomas oryzae pv. oryzae by using text mining technologies. *BioMed research international*, 2013.