Web-based Relation Extraction for the Food Domain

Michael Wiegand, Benjamin Roth, and Dietrich Klakow

Spoken Language Systems, Saarland University, D-66123 Germany
{michael.wiegand|benjamin.roth|dietrich.klakow}@lsv.uni-saarland.de

Abstract. In this paper, we examine methods to extract different domain-specific relations from the food domain. We employ different extraction methods ranging from surface patterns to co-occurrence measures applied on different parts of a document. We show that the effectiveness of a particular method depends very much on the relation type considered and that there is no single method that works equally well for every relation type. As we need to process a large amount of unlabeled data our methods only require a low level of linguistic processing. This has also the advantage that these methods can provide responses in real time.

1 Introduction

There has been only little research on natural language processing in the food domain even though there is a high commercial potential in automatically extracting relations involving food items. For example, such knowledge could be beneficial for virtual customer advice in a supermarket. The advisor might suggest products available in the shop that would potentially complement the items a customer has already in their shopping cart. Additionally, food items required for preparing a specific dish or typically consumed at a social occasion could be recommended. The advisor could also suggest an appropriate substitute for a product a customer would like to purchase if that product is out of stock.

In this paper, we explore different methods, such as simple manually designed surface patterns or statistical co-occurrence measures applied on different parts of a document. Since these methods only require a low level of linguistic processing, they have the advantage that they can provide responses in real time. We show that these individual methods have varying strength depending on which particular food relation is considered.

Our system has to solve the following task: It is given a *partially instantiated relation*, such as *Ingredient-of(FOOD-ITEM=?, pancake)*. The system has to produce a ranked list of possible values that are valid arguments of the unspecified argument position. In the current example, this would correspond to listing ingredients that are necessary in order to prepare *pancakes*, such as *eggs*, *flour*, *sugar* and *milk*. The entities that are to be retrieved are always food items. Moreover, we only consider binary relations. The relation types we examine (such as *Ingredient-of*) are domain specific and to the best of our knowledge have not been addressed in any previous work.

2 Data and Resources

For our experiments we use a crawl of *chefkoch.de*[1] as a domain-specific dataset. *chefkoch.de* is the largest web portal for food-related issues in the German language. We obtained the crawl by using *Heritrix* [1]. The plain text from the crawled set of web pages is extracted by using *Boilerpipe* [2]. The final domain-specific corpus consists of 418,558 webpages (3GB plain text). In order to have an efficient data access we index the corpus with *Lucene*.[2]

3 The Different Relations

In this section, we will briefly describe the four relation types we address in this paper. Due to the limited space of this paper, we just provide English translations of our German data in order to ensure general accessibility.

- ***Suits-to(FOOD-ITEM, EVENT)*** describes a relation about food items that are typically consumed at some particular cultural or social event. Examples are <*roast goose*, *Christmas*> or <*popcorn*, *cinema visit*>.
- ***Served-with(FOOD-ITEM, FOOD-ITEM)*** describes food items that are typically consumed together. Examples are <*fish fingers*, *mashed potatoes*>, <*baguette*, *ratatouille*> or <*wine*, *cheese*>.
- ***Substituted-by(FOOD-ITEM, FOOD-ITEM)*** lists pairs of food items that are almost identical to each other in that they are commonly consumed or served in the same situations. Examples are <*butter*, *margarine*>, <*anchovies*, *sardines*> or <*Sauvignon Blanc*, *Chardonnay*>.
- ***Ingredient-of(FOOD-ITEM, DISH)*** denotes some ingredient of a particular dish. Examples are <*chickpea*, *falafel*> or <*rice*, *paella*>.

4 Method

4.1 Surface Patterns (PATT)

For this work, we only considered manually compiled patterns. Our objective was to have some very few generally applicable and, if possible, precise patterns. As a help for building such patterns, we looked at mentions of typical relation instances in our corpus, e.g. <*butter, margarine*> for *Substituted-by* or <*mince meat, meat balls*> for *Ingredient-of.*

The formulation of such patterns is difficult due to the variety of contexts in which a relation can be expressed. This was further confirmed by computing lexical cues automatically with the help of statistical co-occurrence measures, such as the *point-wise mutual information*, which were run on automatically extracted sentences containing mentions of our typical relation instances. The output of that process did not reveal any additional significant patterns.

[1] www.chefkoch.de

[2] lucene.apache.org/core

Our final patterns exclusively use lexical items immediately before, between or after the argument slots of the relations. Table 1 illustrates some of these patterns. The level of representation used for our patterns (i.e. word level) is very shallow. However, these patterns are precise and can be easily used as a query for a search engine. Other levels of representation, e.g. syntactic information, would be much more difficult to incorporate. Moreover, in our initial exploratory experiments, we could not find many frequently occurring patterns using these representations to help us find relation instances that could not be extracted by our simple patterns. Additionally, since our domain-specific data comprise informal user generated natural language, the linguistic processing tools, such as syntactic parsers, i.e. tools that are primarily built with the help of formal newswire text corpora, are severely affected by a domain mismatch.

The extraction method PATT comprises the following steps: Recall from the task description in Section 1 that we always look for a list of values for an unspecified argument in a partially instantiated relation (PIR) and that the unspecified argument is always a food item. Given a PIR, such as *Substituted-by(butter, FOOD-ITEM=?)*, we partially instantiate each of the pertaining patterns (Table 1) with the given argument (e.g. *FOOD-ITEM instead of FOOD-ITEM* becomes *FOOD-ITEM instead of butter*) and then check for any possible food item (e.g. *margarine*) whether there exists a match in our corpus (e.g. *margarine instead of butter*). The output of this extraction process is a ranked list of those food items for which a match could be found with any of those patterns. We rank by the frequency of matches. Food items are obtained using GermaNet [3]. We collected all those lexical items that are contained within the synsets that are hyponyms of *Nahrung* (English: *food*).

Relation Type	**#Patterns**	**Examples**
Suits-to	6	FOOD-ITEM at EVENT; FOOD-ITEM on the occasion of EVENT; FOOD-ITEM for EVENT
Served-with	8	FOOD-ITEM and FOOD-ITEM; FOOD-ITEM served with FOOD-ITEM; FOOD-ITEM for FOOD-ITEM
Substituted-by	8	FOOD-ITEM or FOOD-ITEM; FOOD-ITEM (FOOD-ITEM); FOOD-ITEM instead of FOOD-ITEM
Ingredient-of	8	DISH made of FOOD-ITEM; DISH containing FOOD-ITEM

Table 1. Illustration of the manually designed surface patterns.

4.2 Statistical Co-occurrence (CO-OC)

The downside of the manual surface patterns is that they are rather sparse as they only fire if the exact lexical sequence is found in our corpus. As a less constrained method, we therefore also consider statistical co-occurrence. The rationale behind this approach is that if a pair of two specific arguments co-occurs significantly often (at a certain distance), such as *roast goose* and *Christmas*, then there is a likely relationship between these two linguistic entities.

As a co-occurrence measure, we consider the *normalized Google distance (NGD)* [4] which is a popular measure for such tasks. The extraction procedure of CO-OC is similar to PATT with the difference that we do not rank food items by the frequency of matches in a set of patterns but the correlation score with the given entity. For instance, given the PIR *Suits-to(FOOD-ITEM=?, Christmas)*, we compute the scores for each food item from our (food) vocabulary and *Christmas* and sort all these food items according to the correlation scores.

We believe that this approach will be beneficial for relations where the formulation of surface patterns is difficult – this is typically the case when entities involved in such a relation are realized within a larger distance to each other.

4.3 Relation between Title and Body of a Webpage (TITLE)

Rather than computing statistical co-occurrence at a certain distance, we also consider the co-occurrence of entities between title and body of a webpage. We argue that entities mentioned in the title represent a predominant topic and that a co-occurrence with an entity appearing in the body of a webpage may imply that the entity has a special relevance to that topic and denote some relation. The co-occurrence of two entities in the body is more likely to be co-incidental. None of those entities needs to be a predominant topic. If our experiments prove that the co-occurrence of entities occurring in the title and body of a webpage is indicative of a special relation type, we would have found an extraction method for a relation type that (similar to CO-OC) bypasses difficult/ambiguous surface realizations that present a significant obstacle for detection methods that explicitly model those surface realizations, such as PATT (Section 4.1).

The extraction procedure of this method selects those documents that contain the given argument of a PIR (e.g. *lasagna* in *Ingredient-of(FOOD-ITEM=?, lasagna)*) in the title and ranks food items that co-occur in the document body of those documents according to their frequency. We do not apply any co-occurrence measure since the number of co-occurrences that we observed with this method is considerably smaller than we observed with CO-OC. This makes the usages of those measures less effective.

5 Experiments

We already stated in Section 1 that the unspecified argument value of a partially instantiated relation (PIR) is always of type FOOD-ITEM. This is because these PIRs simulate a typical situation for a virtual customer advisor, e.g. such an advisor is more likely to be asked what food items are suitable for a given event, i.e. *Suits-to(FOOD-ITEM=?, EVENT)*, rather than the opposite PIR, i.e. *Suits-to(FOOD-ITEM, EVENT=?)*. The PIRs we use are presented in Table 2.[3] For

[3] Since the two relation types *Served-with* and *Substituted-by* are reflexive, the argument positions of the PIRs do not matter.

each relation, we manually annotated a certain number of PIRs as our gold standard (see also Table 2). The gold standard including its annotation guidelines are presented in detail in [5]. Since our automatically generated output are ranked lists of food items, we use *precision at 10 (P@10)* and *mean reciprocal rank (MRR)* as evaluation measures.

Partially Instantiated Relations (PIRs)	**#PIRs**
Suits-to(FOOD-ITEM=?, EVENT)	40
Served-with(FOOD-ITEM, FOOD-ITEM=?)	58
Substituted-by(FOOD-ITEM, FOOD-ITEM=?)	67
Ingredient-of(FOOD-ITEM=?, DISH)	49

Table 2. Statistics of partially instantiated relations in gold standard.

Table 3 compares the different individual methods on all of our four relation types. (Note that for CO-OC, we consider the best window size for each respective relation type.) It shows that the performance of a particular method varies greatly with respect to the relation type on which it has been applied. For *Suits-to*, the methods producing some reasonable output are CO-OC and TITLE. For *Served-with*, PATT and CO-OC are effective. For *Substituted-by*, the clear winner is PATT. For *Ingredient-of*, TITLE performs best. This relation type is difficult to model with PATT. It is also interesting to see that TITLE is much better than CO-OC. From our manual inspection of relation instances extracted with CO-OC we found that this method returns relation instances of any relation type that exclusively involve entities of type FOOD-TYPE (i.e. *Served-with*, *Substituted-by* and *Ingredient-of*).[4] TITLE, on the other hand, produces a much more unambiguous output. It very reliably encodes the relation type *Ingredient-of*. It therefore comes as no surprise that TITLE, in return, performs poorly on *Served-with* and *Substituted-by*.

	Suits-to		**Served-with**		**Substituted-by**		**Ingredient-of**	
Method	**P@10**	**MRR**	**P@10**	**MRR**	**P@10**	**MRR**	**P@10**	**MRR**
PATT	0.023	0.133	**0.343**	**0.617**	**0.303**	**0.764**	0.076	0.331
CO-OC	**0.340**	**0.656**	0.310	0.584	0.172	0.553	0.335	0.581
TITLE	0.300	0.645	0.171	0.233	0.049	0.184	**0.776**	**0.733**

Table 3. Comparison of the different individual methods.

Table 4 illustrates some automatically generated output using the best configuration for each relation type. Even though not all retrieved entries match

[4] Note that the entity type DISH in *Ingredient-of* is a subset of FOOD-ITEM.

with our gold standard, most of them are (at least) plausible candidates. Note that for our gold standard we aimed for high precision rather than completeness.

Suits-to(?, picnic)	Served-with(mince meat, ?)	Substituted-by(beef roulades, ?)	Ingredient-of(?, falafel)
sandwiches*	onions	goulash*	chickpea*
fingerfood	leek	marinated beef*	cooking oil*
noodle salad*	zucchini*	roast*	water
meat balls*	bell pepper	roast beef*	coriander*
potato salad*	noodle casserole	braised meat*	onions*
melons*	feta cheese	cutlet*	flour*
fruit salad*	spinach	rabbit*	salt*
small sausages	rice*	rolling roast*	garlic*
sparkling wine	tomatoes	rolled pork	peas*
baguette*	sweet corn	game	shortening*

Table 4. The 10 most highly ranked food items for some automatically extracted relations; *: denotes match with the gold standard.

6 Conclusion

In this paper, we examined methods for relation extraction in the food domain. We have shown that different relation types require different extraction methods. Since our methods only require a low level of linguistic processing, they may serve for applications that have to provide responses in real time.

Acknowledgements

The work presented in this paper was performed in the context of the Software-Cluster project EMERGENT and was funded by the German Federal Ministry of Education and Research (BMBF) under grant no. “01IC10S01”.

References

1. Mohr, G., Stack, M., Ranitovic, I., Avery, D., Kimpton, M.: An Introduction to Heritrix, an open source archival quality web crawler. In: Proc. of IWAW. (2004)
2. Kohlschütter, C., Fankhauser, P., Nejdl, W.: Boilerplate Detection using Shallow Text Features. In: Proc. of WSDM. (2010)
3. Hamp, B., Feldweg, H.: GermaNet - a Lexical-Semantic Net for German. In: Proc. of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications. (1997)
4. Cilibrasi, R., Vitanyi, P.: The Google Similarity Distance. IEEE Transactions on Knowledge and Data Engineering **19**(3) (2007) 370 – 383
5. Wiegand, M., Roth, B., Lasarcyk, E., Köser, S., Klakow, D.: A Gold Standard for Relation Extraction in the Food Domain. In: Proc. of the LREC. (2012)