Recipe Pos Tagger

Mehmet Özgen*, İbrahim Ardıç†, Gönenç Ercan‡ and Pınar Duygulu Şahin§
Department of Computer Engineering, Hacettepe University
Ankara
Email: *mehmet.ozgen@hacettepe.edu.tr, †ibrahim.ardic@hacettepe.edu.tr, ‡gonenc.ercan@hacettepe.edu.tr, §pduygulu@hacettepe.edu.tr

***Abstract*—The instruction list is a key for a success of doing any task correctly. The validation of correction of the task by examined only the result of the performed instructions, actions. But on the other hand, when performing the actions according to the instructions under the supervision of domain expert, there is high probability to detect disruptions and errors without waiting the result of the task. In this context, mapping instructions to actions is getting high importance when the domain is one of safety-critical systems, physiotherapy exercises, food recipes, chemical experiments, etc. In this work, any written instructions in a specific domain are learned using natural language processing techniques (word2vec, CRF++) and will generate a model to use this information for revealing hidden objects in intermediate states during cooking.**

***Keywords*—Information Retrieval, Natural Language Processing, weaky-labelled data, text analysis, big data**

I. Introduction

Computers have become increasingly more ambitious and popular for people, even for learning new things, by learning new things from textual sources, along with improvements in artificial intelligence and machine learning. It is aimed to automatically model the steps of a certain job by using text data which is easily accessible from the Internet and to create methods and tools that can direct people. Because there are many recipe documents and videos available on the Internet, the study area is selected as the recipe area. The data set for recipes was created from recipes culled from allrecipes.com. The 1830 recipes were scanned and stored as text documents, each separated into its contents and recipes. [1]

A. Subsection Heading Here

Subsection text here.

1) Subsubsection Heading Here: Subsubsection text here.

II. Related Works

Algorithmic processing of training texts and / or commands expressed in natural language has long been an interesting question (Harnad, 1990). In this problem, the artificial intelligence agent (such as a robot) is trying to learn the operations corresponding to symbols that are expressed in a field. The system is expected to automatically interact with definitions in the physical world as symbols to interact with objects and objects. We obtained through www.allrecipes.com

Chen and Mooney (2011) have developed a system that can act according to the expressions expressed in natural language in the system they developed. Based on the reinforced learning methods, the system can translate the sentences into natural language by using the information in the location (such as the objects on the wall) and the state flow that is modeled. As can be seen from these studies in the literature, reinforcement learning techniques are used in problems where the current situation can be modeled more clearly. A slightly different problem is Kushman et al. (2014) developed a similar method for solving problems expressed in mathematical texts. According to this method, the system is mapping the problem described in the text to a set of equations and producing the result. To learn the method, we use a training set marked at different levels. Equations that can be created in the system can be given or can be learned only when the last answer is given. In the method they developed, the solution is defined as diagrams, and a probability model that maps the information given in the text to the diagram is developed. Of course, it is necessary that the structure of the questionnaire is similar and that all of them can be transferred to the equation diagram.

Recipes can also be thought of as a set of steps that must be followed. From this point of view, there are common aspects to the problems described above. The most important difference is that it is difficult to convert the media knowledge to a reward function to provide reinforcement learning. The biggest reason for this is that it is difficult to model the correct sequence of operations that can be done while cooking. To overcome this problem in the processing of recipes, three different basic approaches are striking. The first is knowledge-based methods that are dependent on previously created knowledge vocabulary, the second is those using supervised learning methods, and the third is unsupervised learning and extracting a specific model from the data in the compilation. The first approach involves all the recipes, contents, tools and processes, and methods that can show the relationship between them. In the second approach, it is necessary to generate the data with the equivalent of the text for the training compilation. In the third category, transductive learning is performed according to the solutions given, ie it is not possible to classify the given samples and to process new ones.

A. Information Based Methods

Walter et al. (2011) describes a method of generating flow diagrams from recipe texts, extracting recipe statements as material, processing and finishing conditions. Based on this information, a flow diagram is created. This method requires pre-tagged sentences and a dictionary that holds all of these tags.

Another problem that may be related is the creation of an information dictionary. Gaiilard et al. (2012) foods and their relationships with a Wiki-style method. Its approach basically consists of 6 hierarchies; diet and mealtime according to different features categorize the food. With this approach it is possible to adapt a new recipe or an existing recipe.

Although food-related ontology is increasingly detailed, these methods often fail to adapt to dynamic language and variety. Thus, when ontology methods are used alone, successful results are achieved for some data sets, but success is often low. Moreover, in order to adapt these methods to a different language, ontologies should be structured in accordance with other languages.

B. Supervised Learning Methods

One of the most important features of tutorial learning techniques is the target description model. Basically, the information that this model needs to be able to show is where the processes described in the steps are done on which materials and where the output of this step is used. Some of these associations show in the tree structure (Jermsurawong and Habash, 2015), while some researchers show it in a non-cyclic manner (Malmoud et al., 2014). It is possible that this notation can both express all the recipes and learn it automatically. Jermsurawong and Habash (2015) defined a representation in the tree data structure that stores food items and their relationships over the description steps. This demonstration can show which step of the recipes is related to which steps and materials.

Malmoud et al. (2014) sees text in recipes as a Markov Decision Process problem by expanding semantic role labeling. It holds two things between the process and the material. The purpose of this relationship is to demonstrate the processes within the training that cause these situations to occur. Mori et al. (2014) created a labeling tool by showing recipes as non-cyclic charts in their initial work. With this tool, the recipe created a graph that indicates the skis over the text. The given Japanese recipes first extract the word segmentation, locate the words in the clan's tasks, label the named entities, and finally construct the predicate-argument structure. In order to resolve the uncertainties in the contents, Erica Greene (2016) tagged a total of 187,000 sentences for labeling and edited the tags of the words in the table of contents using the Conditional Random Fields method.

C. Unsupervised Learning Methods

In 2015, Kiddon et al. (2015) has developed a method of learning without a teacher for the processing of recipes. Unlike previous works, this method learns the linkages of the line by using different parameters according to the variables in the system and using the expectation maximization method. Since this model is constructed according to the definition in the drawing, it is not possible to learn the features of an extragalactic recipe. In this respect, it is a transduction method, all the recipes to be processed have to be obtained at the model learning time. Moreover, the generated data model has a high number of implicit variables because it defines both the verbs, the contents, and the content of the contents in relation to the steps as one of the parameters of the model. When such high numbered models are learned, they can be found in a high number of local maxima, which prevents the optimum model from being found. It is intended to remove them as to which action is associated with the material, removing the flow of the recipe as a whole. These methods are inspired by semantic spaces used in text mining and meaning spaces that show word meaning relations. By using these material spaces, it is possible to determine which materials can be replaced instead of which materials. Nedovic (2013) used a similar method to define a method of learning materials for different types of food. Latent Dirichlet Allocation (LDA) and Deep Belief Networks (DBN) are used in the method. It has been seen that the output of the system can group materials according to the foods in different kitchens. A similar study has been proposed by Achananuparp and Weber (2016) to produce safer food recommendations in meals. In this method, Singular Value Decomposition (SVD) method, which is widely used for extracting word meaning relations, is used.

TABLE I
TAGS

I	can	chopped	green	chile	peppers
ADJ	NN	VB	NN	NN	NN

III. Experimental Setup

As seen in the literature, there are many methods aiming at revealing the relationship between the description texts and the contents of the description texts, the sequence of the process flow and the situations that occur during the actions and showing them with the models by many different methods. The texts should be easily perceivable by the computer by passing 1830 recipes we obtained through www.alrecipes.com through certain operations.

Using NLTK libraries, each of the recipes was first converted into a clean text by separating individual recipes and separating them into separated words, punctuation, and meaningless words (names, categories, contents, descriptions, and comments).

A. Labelling

Using the NLTK library, labeling was done according to the culled state (VB, NN, ADJ, etc.) that each of the belts passed. As a result of the labeling made, "I can chopped green chile peppers" in the table of contents is labeled as below in TABLE 1.

As can be seen in Table 1, it does not allow me to know that the word "life" is a unit of measure, "1" actually tells the quantity, and that the words "green" and "choped" are actually interpretations of a "papers" word.

The challenge of parsing the recipe is to be able to distinguish content components from component cues. Erica Greene (2016) has trained 171,244 sets (labeled as UNIT, QUANTITY, COMMENT and OTHER) with the very specific set of

TABLE II
TAGS

I	can	chopped	green	chile	peppers
QT	UNIT	CMMT	CMMT	NAME	NAME

CRF ++ (Conditional Random Fields) method she has created to solve this problem and has been able to label the contents portion probabilistically with a newly given description. Let us have the sentence ”1 teaspoon sugar”. The model is using 171,224 tagged data to learn a model that can predict the tag sequence for any sentence we have given to it, even though we have never seen this component count before. It approaches this by modeling the conditional probability of a set of labels.

p (UNIT UNIT UNIT— ”1 teaspoon sugar”)

p (QUANTITY UNIT UNIT— ”1 teaspoon sugar”)

p (UNIT QUANTITY UNIT— ”1 teaspoon sugar”)

p (UNIT UNIT QUANTITY ”1 teaspoon sugar”)

p (UNIT QUANTITY QUANTITY ”1 teaspoon sugar”)

p (QUANTITY QUANTITY QUANTITY — ”1 teaspoon sugar”)

p (UNIT QUANTITY NAME— ”1 teaspoon sugar”)

...

As mentioned above, it calculates all the probabilities that can be labeled ”1 teaspoon sugar”. The beauty of the linear-chain CRF model makes some conditional independence assumptions that allow us to use dynamic programming to efficiently search the area of ??all possible label sequences. As a result, we have re-tagged our data with Erica Greene (2016), and the result is shown in TABLE 2. The results are shown in Table 2, which shows the best label sequence at a time that is linear with the number of second- .

It was observed that 450 of 171.224 labeled samples were separated and tested and 76 percentage correctly tagged.

B. Calculation of Proximity

Kiddon et al. (2015), some words are not in the table of contents, but more than one is meant. (Eg mixture, them) Kiddon et al. (2015) defines this as a hidden object. He has created a probabilistic model to make hidden objects clear. In order to reveal these hidden objects, we present a relationship between the words ’NAME’ in this action and its contents. In fact, each word is represented as a vector and we calculate the cosine similarity between them.

Using the Word2Vec library that displays 3.5 billion kelimins created by a working group led by Tomas Mikolov (Google) as a 300-dimensional vector vocabulary, the words that pass through the intellectual part and are labeled ’NAME’ have been converted into a 300-dimensional vector.

In the recipe sentence set, phrases without any word labeled ’NAME’ are removed. When the words labeled as ’VB’ are extracted, all the words are converted into 300 dimensional vectors and the averages are taken. Because the other words around us can also provide us with information about what materials the hidden object contains. If you only go through the actions, you will be made a comment by looking at the similarity of only two calves. However, the sentence can sometimes contain words that characterize the words in the contents. Let’s take a look at ”Cook the mixture until caramelized”. It is labeled as ”cook-VB”. When we look at the similarity of the cosine with the contents, it is seen that almost all of them resemble to each other. However, when the complex average of cümlena is taken, ”caramelized” kelimeside is included. And the result is more similar to ”Onion” and ”sugar”. Taking the average of the blame for this reason actually brought us closer to the right conclusion.

IV. CONCLUSION

The conclusion goes here.

ACKNOWLEDGMENT

The authors would like to thank...

REFERENCES

[1] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.