Automatic Extraction of Human Activity Knowledge from Method-Describing Web Articles

Jihee Ryu ZZIHEE5@KAIST.AC.KR
Yuchul Jung ENTHUSIA77@KAIST.AC.KR
Kyung-min Kim KIMDARWIN@KAIST.AC.KR
Sung Hyon Myaeng MYAENG@KAIST.AC.KR
Korea Advanced Institute of Science and Technology, 335 Gwahang-ro, Yuseong-gu, Daejeon, South Korea

Abstract

Knowledge on daily human activities in various domains is invaluable for many customized user services that can benefit from context-awareness or activity predictions. Past approaches to constructing a knowledge base of this kind have been domain-specific and not scalable. A recent attempt to extract activities of daily living (ADL) from Web resources deal with activities and objects involved in achieving them but not the sequence of actions in an activity. This paper describes an approach to automatically extracting human activity knowledge from Web articles that describe methods for performing tasks in a variety of domains. The target knowledge base is comprised of activity goals, actions, and ingredients, which are extracted with syntactic pattern-based and probabilistic machine learning based methods. The result is evaluated for accuracy and coverage against some baselines.

1. Introduction

Human activities often define or constrain the user context. As such, the importance of understanding and expecting what people would do for their goals and tasks is becoming more and more important for context-aware personal services. For instance, when a driver comes across a situation with a flat tire on a highway alone, s/he would have to deal with it by changing it by herself/himself, or call for some help. In this situation, an intelligent system with enough knowledge on various human activities of daily living (ADL) can give her/him some recommendations for alternative next activities or services.

However, capturing and storing knowledge on human activities reflecting the real world is a humongous task due to the diversity of activities and the dynamic nature of daily lives of people. Although there have been some attempts to build a knowledge base containing such knowledge (Singh et al., 2002; Gupta and Kochenderfer, 2004; Liu and Singh, 2004; Perkowitz et al., 2004; Morgan and Singh, 2006), they suffer from one of the following limitations: they are either bound to a narrowly defined area of human activities, often requiring many participants and employing a simple activity model with related objects, or diffused with other types of general knowledge, making it difficult to glean the necessary knowledge and apply for a given task.

Our approach to constructing a knowledge base of daily human activities is to utilize activity-related knowledge available in the form of text on the Web. We target at descriptions of how-to knowledge in eHow.com, which contains the most extensive collection of step-by-step instructions on the Web. The freely accessible site currently holds more than one million articles including text and video instructions contributed by individuals, which collectively cover almost every domain of daily lives and grows continuously with new contributed articles as Wikipedia does. We consider extracting activity knowledge from this vast resource is an important corner stone for building an activity ontology.

This paper focuses on the single Web site with a hope that the result can be a basis for further extension using other less structured Web resources. While an effort to process any Web pages whose styles and goals for the creation vary widely is worthwhile, the quality of the result is not likely to be high enough to be used for any practical purposes. Instead, we focus on method-describing articles whose overall format is reasonably uniform. Once a vast amount of instructional knowledge is obtained with high accuracy, it can be a basis for processing less structured Web resources to extend the content further. The syntactic patterns and the probabilistic machine learning based method can be also extended to cover more general natural language sentences containing how-to knowledge.

Appearing in *Proceedings of the 1st Workshop on Automated Knowledge Base Construction*, Grenoble, France, 2010.

The main thrust of this paper is to show how task-oriented human activity knowledge can be mined from the how-to articles and represented in the form of goals, actions, and ingredients that are manipulated by the actions. The resulting knowledge base containing a huge number of <goal, actions, ingredients> triples would be essential for understanding user situations based on detected actions and ingredients. We first collected the entire set of text instructions from the Web site and applied some natural language processing (NLP) techniques to extract actions expressed in a verb form and associated contextual ingredient items from the goal and subsequent action sequences expressed in natural language in an article.

We propose two methods of extracting actions and ingredients that form activity knowledge base. The pattern-based approach lends itself to the extraction problem because regularities exist in the articles containing step-by-step instructions. A relatively small set of hand-crafted pattern-based rules can be applied to extract actions that start with imperative verbs and their ingredients. While applying the rules alone tends to be effective in extracting triples with high precision, it is not sufficient to cover all the variations of the contributors' writing styles. Secondly, we employ a probabilistic machine learning technique, considering the extraction task as a problem of classifying words in a sentence into three categories: verb (action) component, ingredient component, and other. Since individual words need to be labeled with one of the three, the conditional random field (CRF) model is a natural fit for the task.

We first describe some related work in section 2 to provide a context in which our method can be understood. We then analyze the characteristics of the how-to data so as to define our activity knowledge structure and set the stage for the description of the details of the proposed methods of activity mining in section 4. The evaluation of our approach is shown in section 5 with some discussions, followed by the conclusion section.

2. Related Work

There have been past attempts to construct a knowledge base about human daily lives. A monumental effort to collect shallow knowledge about commonsense including daily living is the Open Mind Common Sense (OMCS) project (Singh et al., 2002). More than 729,000 raw sentences representing commonsense knowledge were collected from the general public who entered them through a template-based Web interface. Based on this work, 30,000 temporal relation instances were extracted among commonsense events to form EventNet (Espinosa and Lieberman, 2005). These relations were utilized as a basis for an intelligent household system. However, the knowledge is skewed toward household activities.

Shah and Gupta (2005) tried to utilize Open Mind Indoor Common Sense (OMICS) which is composed of task steps to populate an initial robot knowledge base with default household task plans. They extracted action-object pairs from the data by simply applying a part-of-speech (POS) tagger and extracting the first verb and noun phrase as the action and its object. This is possible because the input data are composed of simple instructions, like "collect clothes", "move to washing machine", and "place clothes in washing machine".

Perkowitz et al. (2004) proposed a method for mining human activity models in terms of a sequence of objects involved in an activity and their probabilities. For example, they modeled "making tea" with a set of involved objects, like teapot, faucet, water, milk, etc. From the definitions of activities obtained in external resources, such as, how-to instructions, recipes, and training manuals, they attempted to extract objects by identifying noun phrases and their hyponyms under 'object' or 'substance' categories in WordNet. They estimated the probabilities of associations by the ratio between the hit count of an activity label occurring together with the associated object and that of the activity label alone.

There also have been attempts for attribute extractions within the recent information extraction community. They tried to acquire attributes, possibly along with corresponding values, from Web documents. Tokunaga et al. (2005) employed lexico-syntactic patterns for unstructured text in a small collection of Web documents. Ravi and Pasca (2008) and Reisinger and Pasca (2009) employed weakly supervised approaches for unstructured Web documents to deal with a large number of classes and extract attributes that are not restricted to any pre-defined pattern types (e.g., X-of Y patterns). While related, they do not deal with activities in general nor actions and ingredients in particular.

Compared to the past approaches, we attempt to extract human activity knowledge automatically from how-to instructions on the Web with a larger scale and a finer representation. We define an activity in terms of a sequence of actions and associated ingredients such as objects and its goal. In other words, an activity always has a goal and its ingredients. For instance, the goal of "making a tea" has an action sequence including "fill a teapot" and an associated ingredient "teapot".

3. Data Characteristics

While other Web resources can be used for the intended purpose, the focus of our current work is on eHow because it has the widest coverage. The site is an online community dedicated to providing visitors the ability to research, share, and discuss instructional solutions that help complete daily practicable tasks. It covers a wide variety of topics organized into a hierarchy of categories, such as *automotive, college, real estate, health, and weddings*. The content is created by both professional experts and casual users. Each eHow article contains practical knowledge of everyday to help people discuss,

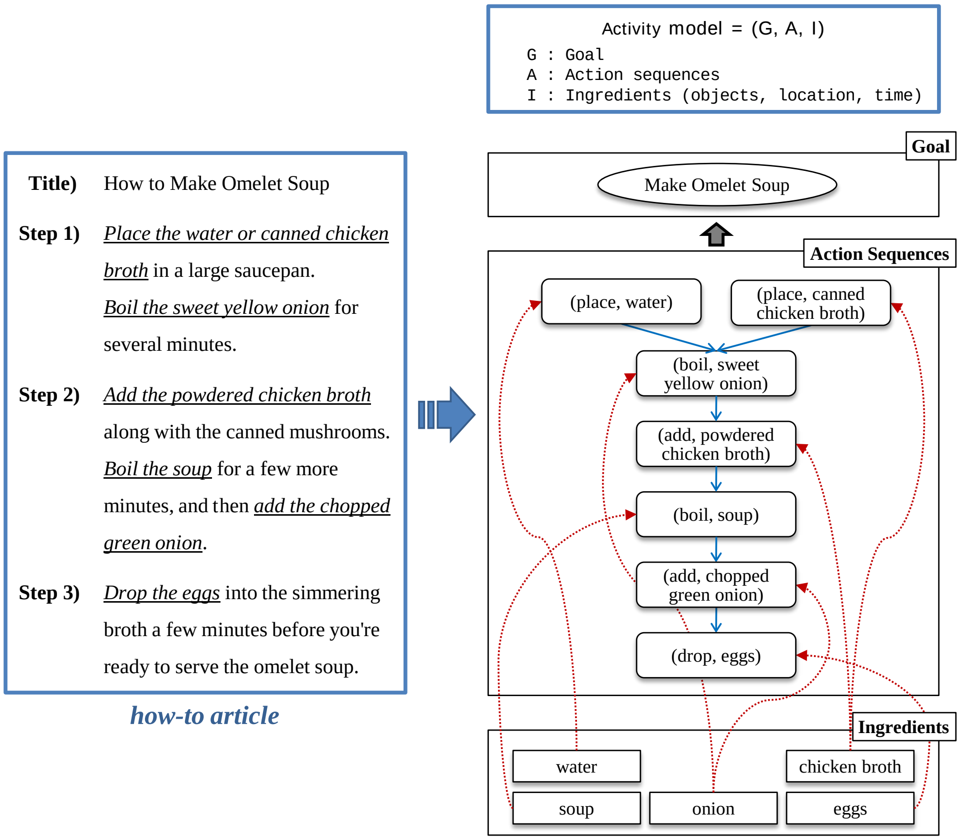

Figure 1. Our activity knowledge model

plan, and complete things like "how to handle health insurance if you lose your job", "how to eat well for less", "how to format a hard drive with windows XP", etc. The site has over 10-year history of running it as a way of sharing how-to knowledge on the Web.

It currently contains more than 1.5 million articles[1] and 150,000 high-quality videos, and the numbers are still growing. Owing to its evolving nature and the growing number of interactions among users, the content is expected to be more accurate, highly achievable and easier to understand in the future, making them a solid basis for knowledge of humans' general activities. As such, our effort to mine human activities out of this resource is by no means a one-time effort; the methodology and its feasibility to automatically construct activity knowledge are of utmost importance. As improvements are made to the content and the methodologies, the quality of the resulting knowledge base will be enhanced, too.

A unique characteristic of how-to articles is that many of the sentences in the instructions are imperative starting with a base form of a verb. We found that about 56% of all the instruction step sentences in the entire data in eHow have this form. Another observation is that the instructions usually start with a practically doable action, and that the details follow the first sentence.

Other remaining sentences include an explanation of the status like "you are almost done" or background knowledge like "caffeine can interfere with sleep up to 12 hours after it is consumed". Nonetheless, some non-imperative sentences can contain instructional actions as in "you should take your private transport." In addition, some ingredients on an instruction may have a form of noun phrase modified by a prepositional phrase, but we ignore prepositional phrases as they are not as important as the noun phase part in searching for action steps.

4. Structure of Activity Knowledge

Based on the above observations of data characteristics, we define our activity knowledge structure comprising three components: a goal, an action sequence for goal achievement, and ingredients (objects, location, and time) that represent context variables as in Figure 1. A goal is a desired state, and its action sequence contains actual steps to achieve it. A sequence of actions is associated with a goal, and the goals are inter-related. While the prepositional phrase "for a few minutes" is vital information as an instructional step, "boil" and "soup"

[1]The number of eHow contents is appeared at http://www.ehow.com/about_us/about_us.aspx.

extracted as an action and an ingredient, respectively, can be sufficient for detecting the goal or activity of the user. The issue of using such contextual or auxiliary information for ingredients is discussed in the work of Jung et al. (2010).

5. Activity Knowledge Extraction

Our activity mining process takes eHow articles as an input and produces action component sequence necessary to achieve a goal. The input document goes through a set of NLP preprocessing modules for automatic processing of given text data, including sentence detection, parsing, and generating typed dependencies. With the processed document text, we apply extraction methods for capturing action components. Each action component is composed of a verb form and associated ingredients.

We now describe the preprocessing steps and two methods for activity knowledge extraction: syntactic pattern-based and CRF-based methods.

5.1 Preprocessing

To apply an automatic extraction algorithm, we first apply a set of NLP tools as shown in Figure 2. As the first step, we detect sentences from a how-to article. The detected sentences are parsed with Stanford Parser[2] after an artificial subject is added to an imperative sentence as the parser usually makes a mistake of treating the first word as the subject of sentence. From the parsing results, meaningless adverbial phrases and determiners are pruned to group them into clusters of the same parse tree patterns.

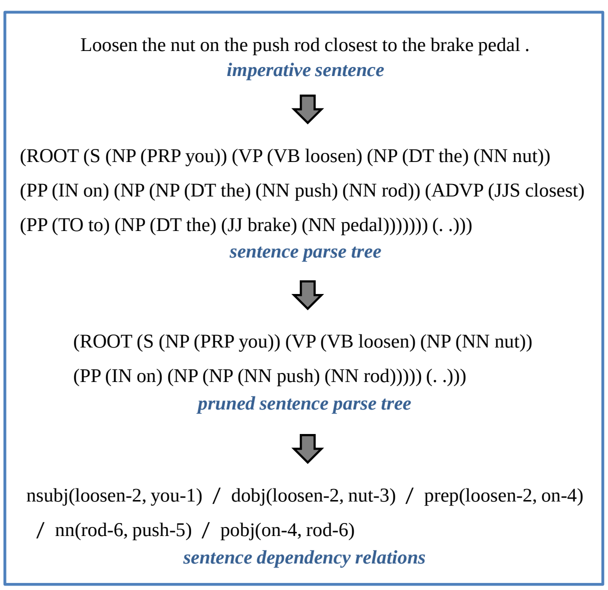

Figure 2. Preprocessing steps

[2]http://nlp.stanford.edu/software/lex-parser.shtml

After a preliminary analysis of the results, we observed that using typed dependency patterns was better in getting much higher coverage rate than using parse tree patterns (from 7% using the parse tree patterns to 56% using typed dependency patterns in our sample). We automatically derived typed dependencies (de Marneffe and Manning, 2008) from pruned sentence parse trees. Note that a sentence parse tree can be converted into the corresponding typed dependencies by using an inherent 'printTree' function of 'TreePrint' class in the Stanford Parser library.

5.2 Syntactic Pattern-based Method

For the syntactic pattern-based method, a set of prominent pattern rules must be generated from the instruction sentences. We developed an unsupervised learning method based on simple heuristics that an action verb and its ingredient item(s) occur as a predicate in the form of a verb phrase connecting a verb and its object(s) (e.g., "find a car").

We first selected 2,400 articles, 100 from each domain, as the training data. The number of sentences was 20,366 that covered 0.2% of the entire set of eHow articles. As shown in Figure 3, a skeletal pattern is generated from a set of sentence dependency relations by replacing every word into a variable. Having converted sentences into skeletal syntactic patterns, they were sorted to collapse the same patterns into unique ones so that we could count the frequency of each unique pattern. Since it is too time-consuming to tag all the generated skeletal syntactic patterns, we selected most frequently occurring top 256 patterns that occur at least three times, among more than 14,000. An expert annotator manually tagged the verb and the ingredient for the patterns.

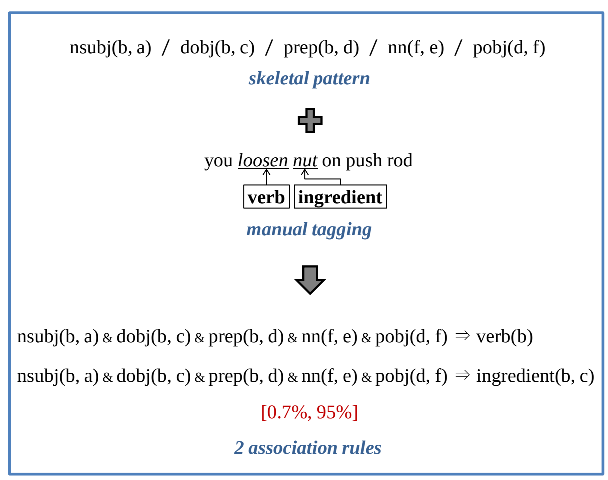

Figure 3. Rule generation process

From those tagged skeletal patterns we generate two association rules respectively by recognizing a VP and its components based on tagged information. A dependency pattern with verb and ingredient annotations can be easily transformed into two association rules using unsupervised association rule mining method (Cios et al., 2007). Each transformed association rule has two values: support and confidence. The former measure how many examples from the data set were found to include both the right-hand side and left-hand side of a rule, indicating the strength of the rule. The latter, on the other hand, measures the number of times the right-hand side is found given all the examples including the left-hand side. An association rule is considered interesting if it satisfies minimum values of support and confidence, which were specified by the experts who examined the data. The set of 256 patterns was initially selected based on the support values. We then tested each of the patterns for confidence and took only 184 of them having the confidence value higher than 85% as the final set of syntactic patterns.

5.3 CRF-based Method

Although the pattern-based approach achieves high-level accuracy, its coverage is limited to the association rules constructed semi-automatically from the small fraction of the sentences and possible patterns. Thus, there are a large number of sentences that fell through the association rules yet contain legitimate actions. To catch them, we adopt a more lenient method – Conditional Random Field (CRF) based open information extraction (OIE) (Banko et al., 2007) to deal with the sentences not covered by the association rules.

CRF is an undirected graph model trained to maximize the conditional probability of a finite set of labels Y given a set of input observations X. It allows some transitions to vote more strongly than others in computing state sequence probabilities. It is a general and expressive modeling technique, considering a whole sequence rather than per-state normalization. Furthermore, CRF can incorporate domain knowledge easily without increasing the state space or designing a special purpose transition structures. By making a first-order Markov assumption about the dependencies among the output variables Y, and arranging variables sequentially in a linear chain, finding a relevant action (verb and its ingredient) can be treated as a sequence labeling problem. We used the CRF implementation provided by MALLET[3].

Assumptions and considerations in the CRF model are as follows. Let $O = \{ o_1, o_2, \ldots, o_t \}$ be an observed input data sequence, such as a sequence of words in text. Furthermore, let S be a set of states of a finite state machine (FSM), each of which is associated with a label, $l \in L$ and $S = \{ s_1, s_2, \ldots, s_t \}$ be a sequence of states. By Hammersley-Clifford theorem, CRF defines the conditional probability of a state sequence given an input sequence.

[3]http://mallet.cs.umass.edu

$$P_{CRF}(s|o) = \frac{1}{Z_0} exp\left(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(s_{t-1}, s_t, o, t)\right)$$

where Z_0 is a normalization factor over all state sequences, $f_k(s_{t-1}, s_t, o, t)$ is an arbitrary feature function over its arguments, and λ_k is a learned weight for each feature function. Higher λ_k weights make their corresponding FSM translations more likely. CRF defines the conditional probability of a label sequence based on total probability over the state sequences, $P_{CRF}(l|o) = \sum_{s:l(s)=l} P_{CRF}(s|o)$, where $l(s)$ is the sequence of labels corresponding to the labels of the state in sequences.

A CRF model was learned from the 920 sentences generating 220 patterns having higher than 75% confidence value. The threshold and resulting patterns were determined empirically after checking the accuracy and coverage used for the evaluation. The CRF model was able to cover more sentences with reasonable accuracy. Figure 4 shows an example of action tracking model as sequence labeling, where the verb "remove" and ingredient "timing belt" are identified.

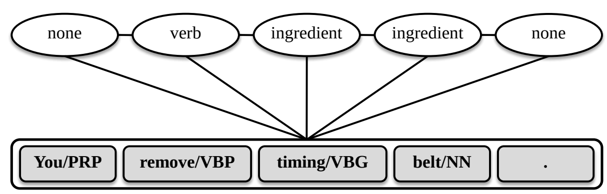

Figure 4. A graphic model for CRF

6. Evaluation

The purpose of our evaluation is to measure performance of the proposed automatic extraction methods. Our focus is how well the proposed methods extract actions and ingredients from the how-to articles. We measure accuracy and coverage of our approach and compared them with those of the two baseline approaches. The test set containing 2,400 eHow articles, 100 for each of the 24 domains, was constructed by employing ten graduate and undergraduate students who have a good command of English. To achieve maximum efficiency and accuracy in the annotation task, we provided a Web interface[4] for them to use. Each article was given to two annotators who made independent decisions on which parts are verbs or ingredients. When there is a partial agreement, only the common parts were taken as a gold standard.

We compared the two extraction methods against two baselines that correspond to the two previous approaches in Shah and Gupta's work (2005) and Perkowitz et al.'s

[4]http://www.zzihee.net/research/activity_mining/

work (2004). The first baseline is exactly the same as the method in Shah and Gupta's work (2005), which just applies part-of-speech (POS) tagging and captures the first verb and noun phrase as action and object components, respectively.

The other baseline was formed based on the object extraction method in Perkowitz et al.'s work (2004) where the primary goal was to produce a set of objects with their probabilities for a given activity. Since it extracts objects but not not actions, we employed the verb extraction method in the first baseline. We attempt to capture the first verb and all noun phrases that are a hyponym of 'object' or 'substance' in the WordNet hierarchy.

For determining the correctness of action extraction, we measured a cosine similarity between an annotated part[5] and an extracted one as follows.

$$Similarity(Part_A, Part_B) = \frac{n(Part_A \cap Part_B)}{\sqrt{n(Part_A) \cdot n(Part_B)}}$$

where $\mathrm{n}(\mathrm{Part}_\mathrm{A})$ is the number of words in the Part_A. We considered an extracted result is correct when the similarity is higher than 0.7, which is determined empirically. We observed that when the similarity value is lower than 0.7, two parts were not always considered semantically equivalent.

Table 1 shows performance differences in terms of accuracy and coverage. The baselines 1 and 2 work pretty well because the data set contain quite a number of simple imperative sentences although they are inferior to the proposed method. Since the baseline 2 captures more objects, its coverage is a little bit higher than that of the baseline 1 at the expense of a much lower accuracy. The syntactic pattern-based method shows the highest average accuracy among all the tested methods while its coverage is much lower because the number of patterns is quite small. The CRF-based method gave reasonable performance with its coverage closed to 95% and yet accuracy is also high. While the accuracy drop is not big from the pattern-based method, the coverage is considerably higher. Compared to baseline 1, accuracy increased at the expense of a drop in coverage. We consider accuracy is more important than coverage because the current knowledge source is not complete anyway and continuously expanded. When the resulting knowledge base is utilized for service recommendation, for example, it is more important not to suggest anything than making a non-sense recommendation.

[5] A part is usually a verb phrase or noun phrase. When there are more than one pair of a verb phrase and a noun phrase, however, a part may include more than one phrase.

Table 1. Performance comparison among extraction methods

Method	Average Accuracy	Average Coverage
Baseline 1	0.7866	0.9821
Baseline 2	0.5432	0.9897
Syntactic Pattern-based	0.9130	0.5660
CRF-based	0.8192	0.9499
Pattern-based & CRF-based	0.8261	0.9501

Finally we combined the pattern-based and CRF-based methods to take advantage of the strengths of both. The pattern-based method was applied first to select the action and ingredient pairs that satisfy the conservative rules and then the CRF-based method next for the remaining sentences. The performance was improved slightly as shown on the last row in Table 1.

7. Analysis

The result of the CRF-based method in Table 1 shows the best performance among all the experiments with different features. Among the features, the typed dependency tags of each word and POS tags were deemed most important. Table 2 shows the result when either of the two feature types were used. While the difference is not big, POS tag features appear to be more important than dependency features. When they were used together, the performance increased slightly. This is because most of verb forms occur as the first verb of the sentence and dependency tags helps detect associated ingredients at the position of direct object of the verb.

Table 2. Performance comparison among different conditions

Method	Average Accuracy	Average Coverage
CRF-based (dependency)	0.7827	0.9443
CRF-based (POS)	0.8001	0.9488
CRF-based (POS, dependency)	0.8192	0.9499

We found that dependency tags and POS tags were less useful when they were used blindly than when additional information is used. For POS tags, it was critical to add the ordering information from the beginning of the sentence. Instead of VB or NN, for example, VB1 (the first VB in the sentence) or NN2 (the second NN in the sentence) were employed to result in more than 20% increase in accuracy. The ordering information was also important for the dependency features.

8. Discussion

While the evaluation results show promising performance of the proposed approach in extracting human activities across various domains, our work has some limitations. The first one is loss of essential information about ingredients, caused by our decision not to deal with prepositional phrases whose attachment may be ambiguous. On the contrary to our initial assumption that prepositional phrases would give only supplementary information without which the extracted activity knowledge would still be useful, we found that some nouns are so general that without additional modifiers in the form of a prepositional phrase, they may not be sufficiently useful in identifying human contexts. For instance, if we extract only “check for” and “dates” from “check for dates on the inside of the wheels” ignoring the prepositional phrase part, “dates” alone would be too general to identify an associated activity.

Second, we did not consider the verb-noun Multi-Word Expression (MWE) (Diab and Krishna, 2009) defined as a multi-word unit that refers to a single concept such as “kick the bucket”. We encountered this kind of expressions in our data set, but did not treat them as a single concept that can be expressed with a single verb. Currently, such an idiomatic expression is divided into a pair of a verb and an ingredient, which is different from the intended meaning.

Still another problem is related to verb scoping, which is also a well-known linguistic issue. When two verbs are conjoined, it is not clear whether the ingredient part should be associated with both or just the latter one. With “put and cut apple” and “stand and cut apple”, for example, the system needs to decide whether the ingredient part, “apple” should be associated with both of the verbs or just the second one. It is ambiguous when the first verb is used as both transitive and intransitive. Our heuristic solution is to associate the ingredient with both of the verbs only when they are exclusively transitive.

Finally we need to enrich the representation of actions to be used for real world applications. This is because our action model consisting of a verb form and an ingredient item may not be sufficient to represent the original action. Moreover, there are additional ingredients that should be included with the model, such as, time and location that are not an object of a verb.

9. Conclusion

In this paper, we introduce our effort to automatically extract human activity knowledge from how-to articles on the Web by means of a linguistically based syntactic pattern matching method and a probabilistic machine learning method based on the CRF model. Representing human activities as a triple consisting of a goal, a sequence of actions, and associated ingredients, we focused on automatically extracting action sequences and the ingredients from how-to instructions.

The experimental result is promising in that the final performance was better than the baselines. More meaningfully, we managed to learn high-precision patterns in a semi-supervised way and extract more than a half of the knowledge (actions and ingredients) very accurately. In order to increase robustness and coverage, we developed a CRF-based method. With additional human efforts for annotations, the accuracy of the patterns as well as the coverage would increase. With higher accuracy from the pattern-based method, the errors made by the CRF-based method would be compensated in the combined method. The sensitivity of performance to the amount of annotated data is left for future research.

Given the limitations of the current effort as addressed in the Discussion Section, we will attempt to improve the performance by remedying them. Although we have applied the current result to a toy problem, our future work is to enhance the activity mining work to the extent that the resulting knowledge base can be used for real-life applications like those in (Langheinrich et al., 2000; Cai and Xue, 2006; Luther et al., 2006; Maekawa et al, 2009) that require awareness of users’ activity context.

Acknowledgments

This research was partially supported by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2010-C1090-1011-0008) and partially supported by the IT R&D program of MKE/KEIT under grant KI001877 (Locational/Societal Relation-Aware Social Media Service Technology).

References

Banko, M., Cafarella M. J., Soderland, S., Broadhead, M., and Etzioni, O. Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, pp. 2670–2676, 2007. San Francisco, CA, Morgan Kaufmann Publishers Inc.

Cai, G. and Xue, Y. Activity-oriented Context-aware Adaptation Assisting Mobile Geo-spatial Activities. In *Proceedings of the 2006 International Conference on Intelligent User Interfaces (IUI 2006)*. pp. 354–356, New York, NY, 2006. ACM.

Cios, K., Swiniarski, R., Pedrycz, W., and Kurgan L. Unsupervised Learning: Association Rules. In *Data Mining: A Knowledge Discovery Approach*, chapter 10, pp. 289–306. Springer-Verlag New York, Inc., Secaucus, NJ, 2007.

de Marneffe, M. and Manning, C. D. Stanford typed dependencies manual, September 2008.

Diab, M. T. and Krishna, M. Unsupervised Classification of Verb Noun Multi-Word Expression Tokens. In Gelbukh, A. (ed.) *Proceedings of 10th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2009)*, pp. 98–110, Berlin, Heidelberg, 2009. Springer.

Espinosa, J., and Lieberman, H. EventNet: Inferring Temporal Relations Between Commonsense Events. In Gelbukh, A., de Albornoz, A., and Terashima, H. (eds.) *Proceedings of the 4th Mexican International Conference on Artificial Intelligence (MICAI 2005)*, pp. 61–69, Berlin, Heidelberg, 2005. Springer.

Gupta, R. and Kochenderfer, M. J. Common Sense Data Acquisition for Indoor Mobile Robots. In *Proceedings of the 19th National Conference on Artificial intelligence (AAAI 2004)*, pp. 605–610, 2004. AAAI Press / The MIT Press.

Jung, Y., Ryu, J., Kim, K., and Myaeng, S. Automatic Construction of a Large-Scale Situation Ontology by Mining How-to Instructions from the Web. *Journal of Web Semantics,* 8(2):10.1016/j.websem.2010.04.006, 2010.

Langheinrich, M., Mattern, F., Römer, K., and Vogt, H. First Steps Towards an Event-Based Infrastructure for Smart Things. In *Ubiquitous Computing Workshop (PACT 2000)*, 2000.

Liu, H., and Singh, P. ConceptNet: A Practical Commonsense Reasoning Toolkit. *BT Technology Journal*, 22(4):211–226, 2004.

Luther, M., Fukazawa, Y., Souville, B., Fujii, K., Naganuma, T., Wagner, M., and Kurakake, S. Classification-based Situational Reasoning for Task-oriented Mobile Service Recommendation. In *Proceedings of the 17th European Conference on Artificial Intelligence (ECAI 2006) Workshop on Contexts and Ontologies*, pp. 15–19, 2006.

Maekawa, T., Yanagisawa. Y., Sakurai, Y., Kishino, Y., Kamei, K., and Okadome, T. Web Searching for Daily Living. In *Proceedings of the 32nd international ACM Special Interest Group on Information Retrieval conference (SIGIR 2009)*. pp. 27–34, New York, NY, 2009. ACM.

Morgan, B. and Singh, P. Elaborating Sensor Data using Temporal and Spatial Commonsense Reasoning. In *Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006)*, pp. 187–190, Washington, DC, 2006. IEEE Computer Society.

Perkowitz, M., Philipose, M., Fishkin, K., and Patterson, D. J. Mining Models of Human Activities from the Web. In *Proceedings of the 13th International World Wide Web Conference (WWW 2004)*, pp. 573–582, New York, NY, 2004. ACM.

Ravi, S. and Pasca, M. Using Structured Text for Large-Scale Attribute Extraction. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM 2008)*, pp. 1183-1192, New York, NY, 2008. ACM.

Reisinger, J. and Pasca, M. Low-Cost Supervision for Multiple-Source Attribute Extraction. In Gelbukh, A. (ed.) *Proceedings of 10th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2009)*, pp. 382–393, Berlin, Heidelberg, 2009. Springer.

Shah, C., and Gupta, R. Building Plans for Household Tasks from Distributed Knowledge. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005) Workshop on Modeling Natural Action Selection*, 2005.

Singh, P., Lin, T., Mueller, E. T., Lim, G., Perkins, T., and Zhu, W. L. Open Mind Common Sense: Knowledge Acquisition from the General Public. In Meersman, R. and Tari, Z. (eds.), *Proceedings of CoopIS/DOA/ ODBASE 2002*, pp. 1223–1237, Berlin, Heidelberg, 2002. Springer.

Tokunaga, K., Kazama, J. and Torisawa, K. Automatic Discovery of Attribute Words from Web Documents. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP 2005)*, pp. 106-118, 2005.