Subtask Mining from Search Query Logs for How-Knowledge Acceleration

Chung-Lun Kuo and Hsin-Hsi Chen
Department of Computer Science and Information Engineering, National Taiwan University
No. 1, Sec. 4, Roosevelt Road, Taipei, 10617 Taiwan
E-mail: lounera@gmail.com; hhchen@ntu.edu.tw

Abstract

How-knowledge is indispensable in daily life, but has relatively less quantity and poorer quality than what-knowledge in publicly available knowledge bases. This paper first extracts task-subtask pairs from wikiHow, then mines linguistic patterns from search query logs, and finally applies the mined patterns to extract subtasks to complete given how-to tasks. To evaluate the proposed methodology, we group tasks and the corresponding recommended subtasks into pairs, and evaluate the results automatically and manually. The automatic evaluation shows the accuracy of 0.4494. We also classify the mined patterns based on prepositions and find that the prepositions like *on*, *to*, and *with* have the better performance. The results can be used to accelerate how-knowledge base construction.

Keywords: Knowledge Acceleration; Query Log Mining; Subtask Mining.

1. Introduction

Nowadays wikipedia and large scale knowledge bases such as YAGO (Suchanek et al., 2007), freebase (Bollacker et al., 2008) and DBpedia (Lehmann et al., 2015) are available for supporting what-knowledge. Users can find facts about entities and know their relationships from such a kind of knowledge bases. How-knowledge such as *how to lose weight*, *how to rent a house*, and *how to fire lacks* are different from the above. Given a how-to task, we would like to find several alternative ways to complete the task. How-to websites like wikiHow[1] and eHow[2] help how to do things in daily life. However, the quantity of wikiHow and the quality of eHow cannot compete with those of wikipedia.

As we know, users often look for advices from the Web. They submit queries to search engines, click the results and browse the contents to meet their how-information needs. Users may have some idea about their tasks, e.g., *to lose weight by eating only vegetables*. Such a kind of queries is used to confirm the effectiveness of the subtask (e.g., *eating only vegetables*) and find its details. Users may further find more alternatives to complete the task by directly employing patterns such as "*how to*" + task description. Thus, potential how-knowledge is implicitly embedded in search query logs. It provides an opportunity to accelerate how-knowledge population by using search query logs. This paper aims at mining how-to patterns and applying them to extract subtasks to complete a given how-to task.

This paper is organized as follows. Section 2 survey the related work. Section 3 specifies how to extract task-subtask pairs from wikiHow, proposes an algorithm to mine linguistic patterns from search query logs, and presents recommendation for given how-to tasks. Section 4 shows and discusses the experimental results. Section 5 concludes the remarks.

[1] http://www.wikihow.com/Main-Page
[2] http://www.ehow.com/

2. Related Work

Search query logs keep users' search behaviors. Jones et al. (2008) define users' search structures. Users may submit one or more related queries to meet their information needs. Lucchese et al. (2011) observe that users' queries may cross more than one session divided by fixed time interval. They consider a task as a clustering unit. Queries in a same cluster are used to complete a specific task. Wang et al. (2013) propose latent structural SVM to identify queries in a same task. Hagen et al. (2013) introduce linked open data to measure query similarity. Kotov et al. (2011) annotate queries in a same task manually and automatically. Hua et al. (2013) utilize Probase to compute the semantic distance of queries.

Kokkalis et al. (2013) present a Genies workflow to develop action plans (tasks and to-do lists) by crowd wisdom. Schumacher et al. (2012) propose term-based and frame-based approaches to extract tasks, products, and task facets, and to organize them into a workflow. Addis and Borrajo (2011) present a suite of tools to extract knowledge from unstructured descriptions of plans in wikiHow for planning applications. Jung et al. (2010) propose an approach to automatically constructing a large-scale situation ontology from eHow and wikiHow. In this paper, we analyze data in wikiHow web site and extract pairs of tasks and subtasks. Different from the above approaches, we employ those pairs to mine linguistic patterns in search query logs, and use the patterns to find new subtasks. The new how-knowledge can be feedback to expand how-to web sites.

3. Methodology

Figure 1 describes the flow of mining patterns from search log and wikiHow. Figure 2 takes "*I want to lose weight*" as an example to explain how to utilize the mined patterns to recommend suitable how-to knowledge. The steps will be presented in detail in the following sections.

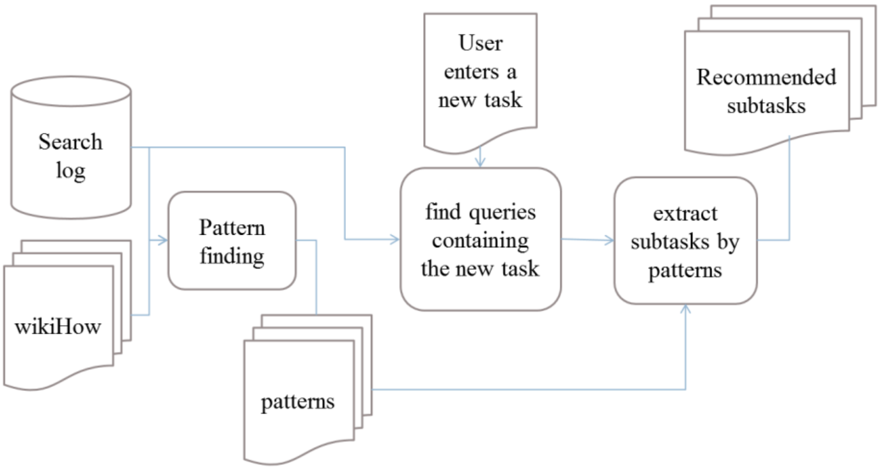

Figure 1: Flow of Subtask Mining and Applications

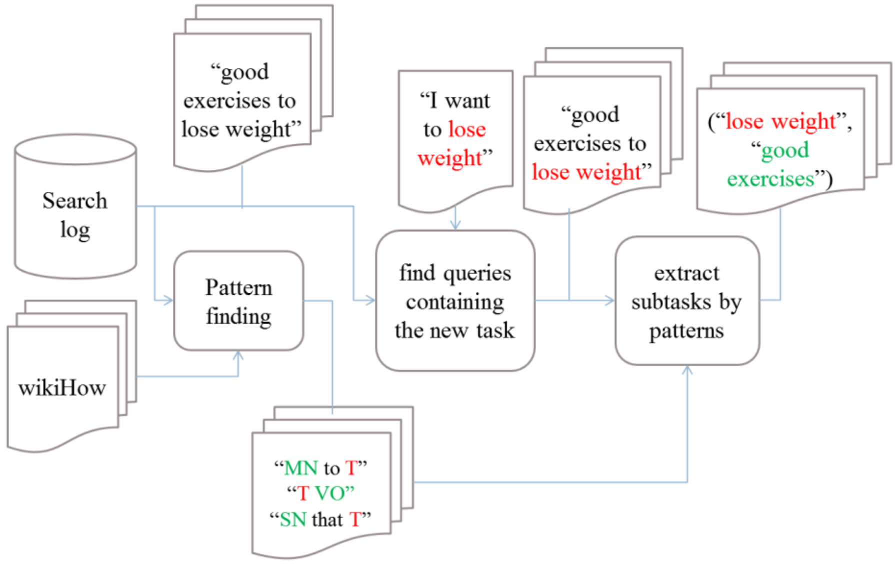

Figure 2: An Example of How-Knowledge Recommendation

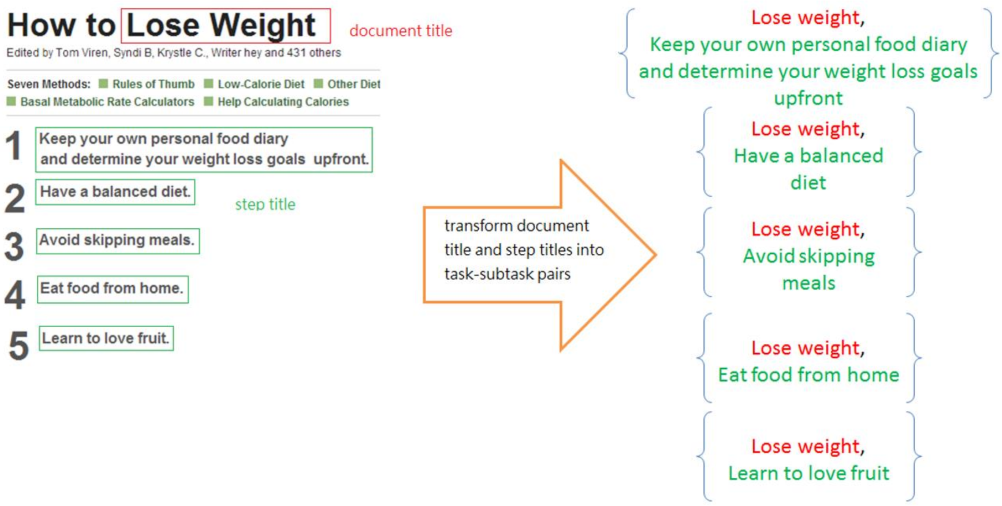

Figure 3: An Example of Task-Subtask Pair Extraction

3.1 Task-Subtask Pair Extraction

A wiki-like how-to website, *wikiHow*, provides solutions to how-tasks manually. Community editors complete articles together. A task includes document title, abstract, some steps, tips, warnings, and things needed to be prepared. A step contains step title and step description. We group a task and each of its subtasks as a pair for detecting and extracting task-subtask information. Figure 3 shows a document for "*how to lose weight*" and the corresponding step titles. *How to Lose Weight* is a title of a document which contains 5 step titles, including *Keep your own personal food diary and determine your weight loss goals upfront*, *Have a balanced diet*, *Avoid skipping meals*, *Eat food from home*, and *Learn to love fruit*. Thus, 5 task-subtask pairs are extracted.

We crawled 151,549 documents from wikiHow on Jan 6th, 2014, regard each document title as a task and the corresponding step titles as subtasks, and form 1,174,476 pairs. Document titles and step titles are tagged and parsed by Stanford dependency parser. The statistics of this dataset is as follows: 7.82 steps per document, 6.25 words per document title, and 9.53 words per step title.

3.2 Linguistic Pattern Mining

The Microsoft AdCenter logs (abbreviated as AdLogs) consisting of 101 million impressions and 7.82 million clicks during 84 days from Aug 10th to Nov 1st, 2007 is used for linguistic pattern mining and subtask extraction. Query terms in impressions are tagged and parsed by Stanford dependency parser. After removing duplicates, 27,922,224 queries remain.

Now we have two datasets: task-subtask pairs extracted from wikiHow and queries extracted from AdLogs. Next we try to find how users express a task-subtask pair (abbreviated as a *T*-*S* pair) in query *Q*. For each *T*-*S* pair, we first retrieve the *Q*s in AdLogs that contain task information *T*. To allow fuzzy matching, we only match the words with dependency DOBJ (direct object) or NPADVMOD (noun phrase adverbial modifier) in *T* with *Q*. For each relevant *Q*, we further check if it also contains subtask information *S*. In the checking, we only consider those words in *S* whose tags are not preposition, determiner, adjective and adverb.

We mine patterns from *Q*s containing both task and subtask information. Each *Q* is transformed into a pattern as follows.

(1) The words in *Q* containing task information is replaced with symbol *T*. The revised string is denoted by $Q^{(1)}$.
(2) The verbs and their direct objects in $Q^{(1)}$ are replaced with symbol *VO*. The revised string is denoted by $Q^{(2)}$.
(3) The compound nouns, or adjective + nouns in $Q^{(2)}$ are replaced with symbol *MN*. The string is denoted by $Q^{(3)}$.
(4) The remaining verbs and nouns are replaced with *SV* and *SN*, respectively. The final string forms a pattern.

Some examples in terms of *query* → *pattern* are shown as follows for reference.

(1) how you can build up your leg muscles to ride a horse → how you can *VO* to *T*
(2) eating only vegetables to lose weight → *VO* to *T*
(3) Chinese herbs to lose weight → *MN* to *T*
(4) how much fruit should i eat to lose weight → how much *SN* should i *SV* to *T*
(5) Walking and losing weight → *SV* and *T*

3.3 How-Knowledge Recommendation

A pattern is composed of words and symbols such as *T*, *VO*, *MN*, *SV*, and *SN*, which denote slots to be filled in. Given a how-to task, we substitute *T* in the mined patterns to form a set of queries, retrieve the relevant snippets from a reference dataset, e.g., AdLogs or the Web, and recommend the slot values in *VO*, *MN*, *SV*, or *SN* of the retrieved snippets as subtasks, i.e., how-knowledge to complete the task.

Assume the task is *lose weight* and pattern "*VO* to *T*" is adopted. The top-20 recommended subtasks are "*using hot and cold water*", "*eating only vegetables*", "*nicotine food*", "*using prayer*", "*find the incentive*", "*drinking vinegar*", "*eat fruit*", "*walking workout*", "*vinegar cocktail*", "*hear rate*", "*gluten free diet*", "*getting your preteen*", "*using fiber*", "*homemade all natural soup recipe*", "*calorie calculator*", "*using laxatives*", "*cutting out sodas*", "*Olive oil*", "*homemade all natural soup*", and "*exercising ways*".

4. Results and Discussion

In the experiments, four fifths and one fifth of documents in the crawled wikiHow dataset are used as training and testing, respectively. We conduct two types of evaluation. In the automatic evaluation, we concatenate "how to", a given task and each recommended subtask as a query and submit it to Google. If more than 5 of the top-10 returned snippets contain both task and subtask, the mined subtask is postulated to be correct. Table 1 lists number of test tasks for each category and their accuracies. The average accuracy is 0.4494. "Computers and Electronics" and "Travel" are the largest and the smallest categories. In the former category, tasks are more specific, e.g., "*Magnify Your iOS Device with Accessibility Zooming*." In the latter category, tasks are more generic, e.g., *Choose a Luxury Hotel*. The accuracies are similar, i.e., 0.4627 and 0.4596.

Categories	#Tasks	Accuracy
Education and Communications	692	0.4228
Relationships	315	0.4430
Food and Entertaining	1,220	0.4122
Sports and Fitness	342	0.4376
Computers and Electronics	2,011	0.4627
Finance and Business	477	0.4594
Travel	96	0.4596
Personal Care and Style	614	0.4803
Work World	182	0.4719
Hobbies and Crafts	1,254	0.4446
Family Life	221	0.4567
Youth	517	0.4524
Home and Garden	598	0.4803
Health	409	0.3842
Holidays and Traditions	166	0.4493
Cars & Other Vehicles	171	0.4304
Pets and Animals	330	0.4967
Philosophy and Religion	53	0.4294
Arts and Entertainment	397	0.4647

Table 1: Automatic Evaluation of Recommendation.

In the manual evaluation, we randomly sample 30 tasks and 50 recommended subtasks for each task. Total 1,500 subtasks are examined by human. The accuracy in the manual evaluation and the automatic evaluation for the 30 sampled tasks and the 1,500 recommended subtasks are 0.4980 and 0.4487, respectively. Table 2 lists the accuracies of these 30 tasks with manual and automatic evaluation.
Of the 30 sampled tasks, "*buy movie tickets early*", "*buy a car*", and "*write a cover letter for a recruitment consultant*", which achieve accuracies of 0.3000, 0.3200, and 0.2800, respectively, perform worse than the other tasks. In contrast, "*download music safely*" and "*play experimental music*", which have accuracies of 0.8200 and 0.7000, perform better.

Tasks	Manual	Automatic
sell your house quickly	0.4800	0.4000
play online games with your friends	0.4400	0.4400
get a job working for a rock band	0.4800	0.4600
make money online for free	0.5800	0.5600
play experimental music	**0.7000**	0.4200
build a martin house	0.4600	0.2800
send email using telnet	0.5600	0.3400
download music safely	**0.8200**	0.4200
gain weight when you have cancer	0.5200	0.3000
get money on animal crossing	0.4800	0.4000
get a boat loan	0.5400	0.6000
write a cover letter for a recruitment consultant	0.2800	0.4000
watch movies on a playstation 3	0.5200	0.5600
buy a car	0.3200	0.3600
have sex without falling in love	0.3800	0.4000
save money on food	0.5400	0.4800
pay bsnl telephone bills online	0.5400	0.3600
start a party planning business	0.5800	0.5000
rent a house in lake tahoe	0.4000	0.5600
find a job if you have a disability	0.3600	0.6200
download songs from spotify with apowersoft free online audio recorder	0.6400	0.4600
buy a house when bankrupt	0.5400	0.5200
lose weight fast	0.5600	0.5000
watch lost televison episodes on the internet	0.5000	0.8000
make a bank transfer payment	0.4400	0.3400
rent a car one way	0.4400	0.4000
earn money selling friendship bracelets	0.5200	0.4400
find an address on windows phone 7	0.5200	0.4400
buy movie tickets early	0.3000	0.3000
march in a military high school marching band	0.5000	0.4000

Table 2: Performance Differences between Manual and Automatic Evaluation.

Most of the queries in logs for "*buy movie tickets early*" are related to specific movie names like "*I want to buy movie tickets for Love Is Strange*". They seldom related to subtasks. Similarly, queries in logs for "*buy a car*" center on the place to buy a car, e.g., "*buy a car in Italy*". Location name here is not a subtask. Most of the queries for the task, "*write a cover letter for a recruitment consultant*", deal with invitation letters, resignation letter, etc.
We also refer to the steps for the 30 tasks in wikiHow. The results show (1) 20.66% of the recommended subtasks appear in wikiHow, (2) 28.98% of the steps in wikiHow appear in our recommendation set, and (3) 47.04% of the recommended subtasks not appearing in wikiHow are labelled as correct by human assessors.
Google search box provides Autocomplete function, which predicts relevant queries. For example, when we input "*how to lose weight*", 10 expanded strings, including "*how to lose weight fast*", "*how to lose weight in a health way*", "*how to lose weight in 10 days*", "*how to lose weight essay*", "*how to lose weight in a week*", "*how to lose weight healthily*", "*how to lose weight without exercise*", "*how to lose weight effectively*", "*how to lose weight without losing breast*", and "*how to lose weight heathy*", are suggested. Referring to the 10 recommended queries for each task, 7.28% of our recommended subtasks appear in Google recommended query set, and 42.26% of Google recommended queries overlap with our recommended subtask set.
Besides subtask evaluation, we also evaluate the quality of the mined patterns. We randomly sample 20 mined patterns, including "how to *T VO*", "*T VO*", "*T* with *MN*", "how to *T* with *MN*", "*T* with *SN*", "*SN* to *T*", "*MN* to *T*", "*VO* to *T*", "*MN* for *T*", "how to *T* for *MN*", "how do I *T* of *MN*", "*T* of *MN*", "*T* in *MN*", "*T* in *SN*", "*SN* that *T*", "how to *T* from *SN*", "*T* from *SN*", "how to *T* on *MN*", "*T* on *SN*", "*T* and *SN*", and 50 recommended subtasks for each pattern. Table 3 shows the performance.
To examine the performance of the mined patterns, we partition the 20 sampled patterns based on the function words (*FW*) used. Table 3 shows the partition and the accuracy. Because verb-object (*VO*) structure captures actions, "how to *T VO*" and "*T VO*" have better performance. The pattern "to-infinitive" expresses purpose (i.e., task), so performance is better. The

FW	Linguistic Pattern	ACC
NA	how to *T VO*, *T VO*	0.58
on	how to *T* on *MN*, *T* on *SN*	0.58
to	*SN* to *T*, *MN* to *T*, *VO* to *T*	0.58
with	*T* with *MN*, how to *T* with *MN*, *T* with *SN*	0.56
for	*MN* for *T*, how to *T* for *MN*	0.50
and	*T* and *SN*	0.50
in	*T* in *MN*, *T* in *SN*	0.45
from	how to *T* from *SN*, *T* from *SN*	0.44
that	*SN* that *T*	0.40
of	how do I *T* of *MN*, *T* of *MN*	0.32

Table 3: Analysis of Mined Linguistic Patterns

preposition *on* has the similar interpretation. The PP, *of MN*, often serves as modifier rather than subtask, e.g., *download songs of johny gaddar*. The accuracy is lower.

5. Conclusion and Future Work

In this paper, we analyze the semi-structured data from how-to websites, use NLP tools to extract keywords of tasks and subtasks, use these keywords to find queries for the same tasks and subtasks, and finally mine linguistic patterns from search query logs. In the experiments, we adopt training tasks and search query logs to extract patterns, while test tasks are treated as new tasks from users. By using the mined patterns, we recommend subtasks for those new tasks.

To evaluate the proposed methodology, we group tasks and the corresponding recommended subtasks into pairs, and evaluate the results automatically and manually. The automatic evaluation shows the accuracy of 0.4494. We also classify the mined patterns based on prepositions and find that the prepositions like *on*, *to*, and *with* have the better performance.

The mined subtasks can be regarded as seeds for how-knowledge editors. Issues to be tackled in the future include coverage of patterns and the boundary of subtasks. Situation information also occurs in query logs in addition to subtask information.

6. Acknowledgments

This research was partially supported by Ministry of Science and Technology, Taiwan, under grants MOST-102-2221-E-002-103-MY3 and MOST-104-2221-E-002-061-MY3.

7. Bibliographical References

Addis, A. and Borrajo, D. (2011). From Unstructured Web Knowledge to Plan Description. *Information Retrieval and Mining in Distributed Environments*, Studies in Computational Intelligence, 324, pp. 41–59.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of SIGMOD 2008*, pp. 1247–1250.

Jones, R. and Klinkner, K.L. (2008). Beyond the Session Timeout: Automatic Hierarchical Segmentation of Search Topics in Query Logs. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management* (*CIKM '08*), pp. 699-708.

Jung, Y., Ryu, J., Kim, K.M., and Myaeng, S.H. (2010). Automatic Construction of a Large-scale Situation Ontology by Mining How-to Instructions from the Web. *Journal of Web Semantics*, 8(2-3), pp. 110–124.

Hagen, M., Gomoll, J., Beyer, A. and Stein, B. (2013). From Search Session Detection to Search Mission Detection. In *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval* (*OAIR '13*), pp. 85-92.

Hua, W., Song, Y., Wang, H., and Zhou, X. (2013). Identifying Users' Topical Tasks in Web Search. In *Proceedings of the sixth ACM international conference on Web Search and Data Mining* (*WSDM '13*), pp. 93-102.

Kokkalis, N., Köhn, T., Huebner, J., Lee, M., Schulze, F. and Klemmer, S.R. (2013). TaskGenies: Automatically Providing Action Plans Helps People Complete Tasks. *ACM Transactions on Computer-Human Interaction*, 20(5), pp. 27:1–27:25.

Kotov, A., Bennett, P.N., White, R.W., Dumais, S.T., and Teevan, J. (2011). Modeling and Analysis of Cross-Session Search Tasks. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval* (*SIGIR '11*), pp. 5-14.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., and Bizer, C. (2015). DBpedia - A Large-Scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*, 6(2), pp. 167–195.

Lucchese, C., Orlando, S., Perego, R., Silvestri, F., and Tolomei, G. (2011). Identifying Task-based Sessions in Search Engine Query Logs. In *Proceedings of the fourth ACM International Conference on Web Search and Data Mining* (*WSDM '11*), pp. 277-286.

Schumacher, P., Minor, M., Walter, K., and Bergmann, R. (2012). Extraction of procedural knowledge from the web: a comparison of two workflow extraction approaches. In *Proceedings of the 21st International Conference Companion on World Wide Web*, pp. 739–747.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: A Core of Semantic Knowledge. In *Proceedings of WWW 2007*, pp. 697–706.

Wang, H., Song, Y., Chang, M.W., He, X., White, R.W. and Chu, W. (2013). Learning to Extract Cross-Session Search Tasks. In *Proceedings of the 22nd international conference on World Wide Web* (*WWW '13*), pp. 1353-1364.