Towards action comprehension for searching: mining actionable intents in query entities

Xin Kang, Yunong Wu, and Fuji Ren

Tokushima University

Author Note

Xin Kang, Yunong Wu, and Fuji Ren are with the Faculty of Engineering, Tokushima University.

This research has been partially supported by the Ministry of Education, Science, Sports and Culture of Japan, Grant-in-Aid for Scientific Research(A), 15H01712.

Correspondence concerning this article should be addressed to Xin Kang, Faculty of Engineering, Tokushima University 2-1, Minamijyousanjima-cho, Tokushima 770-8506 Japan.

Contact: kang-xin@is.tokushima-u.ac.jp

Abstract

Understanding search engine users' intents has been a popular study in information retrieval, which directly affects the quality of retrieved information. One of the fundamental problems in this field is to find a connection between the entity in a query and the potential intents of the users, the latter of which would further reveal important information for facilitating the users' future actions. In this paper, we present a novel research for mining the actionable intents for search users, by generating a ranked list of the potentially most informative actions based on a massive pool of action samples. We compare different search strategies and their combinations for retrieving the action pool and develop three criteria for measuring the informativeness of the selected action samples, i.e. the significance of an action sample within the pool, the representativeness of an action sample for the other candidate samples, and the diverseness of an action sample with respect to the selected actions. Our experiment based on the Action Mining (AM) query entity dataset from Actionable Knowledge Graph (AKG) task at NTCIR-13 suggests that the proposed approach is effective in generating an informative and early-satisfying ranking of potential actions for search users.

Keywords: Action mining, actionable knowledge, actionable intent ranking

Towards action comprehension for searching: mining actionable intents in query entities

Introduction

Predicting the search users' intents based on their queries is similar to solving a Guess My Number problem. You have to firstly know the potential numbers for guessing and then reason for the true number. Previous researches (Broder, 2002; Rose & Levinson, 2004) suggested that search users' intents are closely related to their ultimate actions, such as shopping and finding services. Therefore, a first step of finding the users' actionable intents could reveal vital information for the search engines to retrieve the real-needed documents.

The previous studies of search query sessions suggested that entity-centric queries have been very popular among the massive log of queries (Guo, Xu, Cheng, & Li, 2009; Yin & Shah, 2010) yet have become a major source of ambiguity for understanding the users' real needs (Spirin, He, Develin, Karahalios, & Boucher, 2014; Uyar & Aliyu, 2015). To fulfill the users' diverse intents in queries, researchers have proposed several instinctive approaches, such as diversity searching (Agrawal, Gollapudi, Halverson, & Ieong, 2009; J. Li, Wu, Zhang, Song, & Wang, 2017; Santos, Macdonald, & Ounis, 2010; B. Zhang et al., 2005), query recommendation (Hassan Awadallah, White, Pantel, Dumais, & Wang, 2014; Song & He, 2010; Z. Zhang & Nasraoui, 2006), and knowledge graph (Auer et al., 2007; Bollacker, Evans, Paritosh, Sturge, & Taylor, 2008; Carlson et al., 2010; Hoffart, Suchanek, Berberich, & Weikum, 2013; Qian, Sakai, Ye, Zheng, & Li, 2013; Singhal, 2012; Suchanek, Kasneci, & Weikum, 2007). However, identifying the users' potential intents in query entities is still impossible in these approaches.

In this paper, we propose a novel research for extracting users' actionable intents from query entities. Specifically, our method generates a ranked list of intentional action samples from an open domain, which any people, organization, or other subject may take at any time. Given an entity like *language acquisition*, our method will find highly-related actions like "*learn* semantics" and recommended actions like "*find* the school that's right for you". To make the actionable intents semantically complete, we decompose the actions into two parts, i.e. a verb part which describes the actionable activities like *learn* and *find* and a modification part which indicates the target objectives like "semantics" and "the school that's right for you".

We employ a pooling-based method to generate the users' actionable intents given the query entities. For a query entity e, the corresponding pool $\mathcal{D}$ of action samples is constructed by massive online search results retrieved by one or more searching strategies given the query entity e. To generate a ranked list $\mathcal{A}$ of actionable intents from pool $\mathcal{D}$, we iteratively take the potentially most informative action sample a out of $\mathcal{D}$ and append it to the end of $\mathcal{A}$, each at a time. The mining strategy is based on three criteria, each of which assigns a real-valued priority score to the candidate samples as follows.

1. An action a is considered to be significant if it has been observed many times through $\mathcal{D}$ and describes a specific actionable intent for entity e, such as "*learn* semantics" for entity *language acquisition*.

2. An action a is considered to be representative if there are many actions $a' \in \mathcal{D}$ which describe similar actionable intents as a. For example, action "*master* a semantic distincion" is similar to "*learn* semantics".

3. An action a is considered to be diverse if it is semantically different from the already selected actions $a' \in \mathcal{A}$. For example, action "*learn* semantics" is semantically unlike "*find* the school that's right for you".

An action sample which has the largest sum of the significance, representativeness, and diverseness priority scores is considered to be the most informative in an iteration of mining and is taken as the next actionable intent into $\mathcal{A}$. We report the normalized Expected Reciprocal Rank (nERR) and the normalized Discounted Cumulative Gain (nDCG) evaluations of our actionable intent mining results, based on the Action Mining (AM) query entity dataset from the Actionable Knowledge Graph (AKG) task at NTCIR-13. Experimental results suggest that our approach is effective in generating satisfying and informative ranking of actionable intents given a query entity.

Related Work

To satisfy users' diverse intents, many researchers focused on the diversity search approach to retrieve very different results for a given query (Agrawal et al., 2009; J. Li et al., 2017; Santos et al., 2010; B. Zhang et al., 2005). Although these retrieved documents diversify in the aspects of information coverage (J. Li et al., 2017; B. Zhang et al., 2005) and ranking position score (J. Li et al., 2017; Zuccon & Azzopardi, 2010), their contents were not guaranteed to cover the various intents of users. The query recommendation study (Song & He, 2010; Z. Zhang & Nasraoui, 2006) instead provided refined queries according to the expert users' query sessions based on a large-scale query-log mining. Recently, the subtask recommendation study (Hassan Awadallah et al., 2014) identified the complex search tasks through a supervised learning process and recommended subtasks based on a random walking along a task-association graph. Although the recommended subtasks could assist users in exploring the complex tasks, there was no evidence that they reflected the users' potential intents.

In existing studies, users' intents were considered functionally independent and were separated into several broad categories according to the ultimate goal of the users. For example, intents were classified into *informational*, *navigational*, *transactional* (Broder, 2002) based on the user survey and query-log analysis results. The study suggested that a significant part of queries (36% in user survey and 30% in query-log analysis) ended with other transactional actions, such as shopping, finding web-mediated services, downloading data of interest. Rose and Levinson (2004) further divided these categories into 12 subcategories by manually analyzing query samples from the AltaVista search engine[1] through a propose-classify-refine cycle. Human experts' descriptions of these subcategories indicated that query intents were closely related to the ultimate actions in the users' minds, such as to *obtain* a resource or to get *adviced*.

The anatomic studies of search query suggested that named entities were very popular in queries. For example, the studies of Bing queries found that 71% queries contained named entities (Guo et al., 2009) and that at least 20-30% queries were

[1] http://www.altavista.com

simply named entities (Yin & Shah, 2010). Recent QIC studies (Broder, 2002; Jansen, Booth, & Spink, 2007, 2008; Rose & Levinson, 2004) further revealed the close relation between users' actions and query entities. However, the pre-defined intent categories in QIC were too general to explain the diverse user actions (Lin, Pantel, Gamon, Kannan, & Fuxman, 2012). By observing actions and entities frequently in the query sessions, Lin et al. (2012) proposed a Bayesian model to explore the relationship between entities and their actionable contexts in query-logs. Only 50 actions were heuristically "translated" from the actionable context clusters due to complex Bayesian inference, and various actions which should reflect the users' real intents were ruled out in the translation. Reinanda, Meij, and de Rijke (2015) further studied the user intents in query entities by progressively proposing three tasks for identifying the entity context clusters as user intents, ranking the intents based on their importance to the entities, and recommending the intents as alternative queries to users. However, there was no evidence that such intents reflected the ultimate actions in users' minds.

To respond the users' various intents in query entities, knowledge graphs (Auer et al., 2007; Bollacker et al., 2008; Carlson et al., 2010; Hoffart et al., 2013; Qian et al., 2013; Singhal, 2012; Suchanek et al., 2007) were developed for specifying the semantic relations between various entities. For example, YAGO (Suchanek et al., 2007) and YAGO2 (Hoffart et al., 2013) extracted 9.8 million entities and 447 million semantic relations from semi-structured data like Wikipedia, GeoNames, and WordNet. They specified the relational facts like "Albert Einstein won the Nobel Prize" for entities Albert Einstein and Nobel Prize in a triple (AlbertEinstein, HasWonPrize, NobelPrize). DBpedia (Auer et al., 2007) was similar to YAGO, which extracted 4.6 million entities and 538 million semantic relations from Wikipedia. Because the semi-structured data required manual editing, the extracted semantic relations were found sparse and limited for the real-world inference. With the relational learning approach (Nickel, Murphy, Tresp, & Gabrilovich, 2016) based on raw Internet texts, even larger knowledge graphs (Carlson et al., 2010; Qian et al., 2013; Singhal, 2012) were constructed. The Google Knowledge Graph (Singhal, 2012) was reported to have included 570 million entities and 18 billion semantic relations, and the Satori graph (Qian et al., 2013) was reported to have consisted of 1.1 billion entities and 21 billion semantic relations. In all these knowledge graphs, the number of distinct predicates was very small compared to that of entities and semantic relations. For example, YAGO2 and DBpedia consisted of only 114 and 1,367 predicates respectively, and even the Google Knowledge Graph consisted of 35,000 predicates, which has restricted them in mining the users' actionable intents, leaving many actions unrecognized.

To the best of our knowledge, there has been no previous study of mining actionable intents for query entities from an open-domain. The most related research was the Query Intent Classification (QIC) (Arguello, Diaz, Callan, & Crespo, 2009; Broder, 2002; Celikyilmaz, Hakkani-Tür, & Tur, 2011; Hu, Wang, Lochovsky, Sun, & Chen, 2009; Jiang, Leung, & Ng, 2016; X. Li, Wang, & Acero, 2008; Qian et al., 2013; Ren et al., 2015; Sadikov, Madhavan, Wang, & Halevy, 2010; C.-J. Wang & Chen, 2014), which aimed at classifying queries into several pre-defined categories. These included the categories of Informational, Navigational, and Transactional based on the intent taxonomy work of Broder (2002) and the intention-related categories, such as Product, Job, Travel, Personal Name proposed by X. Li et al. (2008) and Hu et al. (2009). Although these categories were carefully designed so that a search engine could rearrange its retrievements in a better order, they were still too general compared to the specific user intents and actions. Query-logs have been intensively studied in many QIC studies (Jiang et al., 2016; Qian et al., 2013; Ren et al., 2015; Sadikov et al., 2010; C.-J. Wang & Chen, 2014). Jiang et al. (2016) classified query intents based on the Yahoo query-log, C.-J. Wang and Chen (2014) employed the MSN search query-log for classifying user intents, and Qian et al. (2013) captured query intents from the Sogou query-log. Although query-logs contained abundant user-behavior information, they were mixed with large amount of noise and deprived of the real-time trends. Besides, the expanded query features, such as search snippets, titles, click-through, and Wikipedia facets were employed in many QIC studies (Hu et al., 2009; X. Li et al., 2008; Sadikov et al., 2010). In this paper, we employ the vector representation of query tokens in an unified high-dimensional space and infer the informativeness of the actions based on this enriched representations.

Actionable Intents Mining

Problem Description

The problem of actionable intent mining can be formally defined as follows. Given a query entity e, the action mining algorithm needs to generate actionable intents in the form of a ranked list of actions $\mathcal{A}$, for the user who submits the query.

Definition 0.1 *An **entity** e is the textual representation of an existing thing in the real world, and a **query entity** is an entity that is submitted in the search queries.*

For example, *Acoustic guitar* is an entity that represents a kind of music instrument, and *Angola* is an entity which represents a country located in southern Africa. These entities are found in the Action Mining query entity dataset (Blanco, Joho, Jatowt, Yu, & Yamamoto, 2017) at NTCIR-13 and are associated with different actionable intents of search users.

Definition 0.2 *The **actionable intents** for a query entity e are a ranked list of actions $\mathcal{A} = (a_1, a_2, \dots)$ which are potentially informative in directing search user actions with respect to the query entity e. An action $a = (v, o)$ consists of a verb part v which represents the actionable activities and an optional modification part o which indicates the target objective of v or specifies the movement of v.*

For example, (*see* Santosky v. Kramer, 455 U.S. 745 (1982), ..., *choose* to be involved in the child's life, ...) are the actionable intents for the query entity *adoption*. In this case "Santosky v. Kramer, 455 U.S. 745 (1982)" is the target objectives of verb *see* and "to be involved in the child's life" specifies the movement of verb *choose*.

Definition 0.3 *An **action pool** $\mathcal{D}$ for a query entity e is the combination of massive informative actions in the form of a loosely-ranked list, which are contextually or topically related to e in the web texts.*

The candidate actions a for a query entity e are extracted from various online search results which have been combined in an action pool $\mathcal{D}$. The contextual constraint in the above definition requires that an action a is observed in the same sentence with the query entity e, while the topical constraint requires that an action a is observed in a web text which is under the topic of e.

Action Pool Development

To develop an action pool $\mathcal{D}$ for a query entity e, we propose different search strategies based on various external resources. The first external resource we consider is the social networking service (SNS) Twitter, for the huge amount of active users and the massive influx of real-time information. In fact, previous studies (Hollerit, Kröll, & Strohmaier, 2013; J. Wang, Cong, Zhao, & Li, 2015; Zhao et al., 2014) have proved that Twitter was a very important yet effective resource for exploring the user needs. We propose three search strategies for retrieving the related tweets for a query entity e, as follows. The (Q) strategy retrieves at most 1,500 pages of entity-related tweets, with the Twitter Search API. These tweets contain e as a key word and the closely related actions in context. For example, the action *send* is contextually related to the entity "language acquisition" in tweet "I'm about to *send* my future kids to SJKC for sec language acquisition". The (S) strategy retrieves random tweets which include the entity in content, with the Twitter Streaming API. By analysing the retrieved texts, we find that e could be mentioned in various ways in these tweets, with a large amount of noise. This is probably because that the (S) strategy lacks a mechanism to assess the Twitter users, thus has retrieved from many advertising accounts. In general this has weakened the contextual relatedness in the retrieved tweets. For example, tweet "*Travelling* with a private tutor enriches second-language acquisition . . . " is contextually related to entity "language acquisition" but with an obvious advertising noise. As a compensation, the (U) strategy first queries the Twitter User Search API to retrieve at most 1,000 Twitter users whose profiles are closely related to the query entity e and then queries the Twitter User Time API to retrieve these users' latest tweets. These tweets may not include e in content, but the authors' interests and concerns are closely related to the entity. For example, "Any other crazy English sentences to *add*? ..." from a Twitter author who is interested in "language acquisition" is also topically related to the entity. This allows us to discover the entity-related action samples from a general perspective of view.

The second external resource we consider is the broad Internet, for which we develop the (G) strategy based on Google search. Compared to the SNS messages on Twitter, texts on the broad Internet are more general and are associated to the query entity in a wider variety of contextual and topical relations. For example, the (G) strategy retrieves "Study.com ... can help you *find* the school that's right for you" which contains an action that is topically related to "language acquisition". In another example "The key to any bodybuilding routine is *providing* your muscles with enough energy ..." contains an action that is contextually related to "Bodybuilding". We also find that action samples retrieved by the (G) strategy are seldom related to the authors themselves but often provide beneficial information such as suggestions and instructions to the readers.

Table 1 shows the example of retrieved tweets based on different search strategies. The extracted action samples constitute distinct pools of $\mathcal{D}^{(\mathrm{Q})}$, $\mathcal{D}^{(\mathrm{S})}$, $\mathcal{D}^{(\mathrm{U})}$, and $\mathcal{D}^{(\mathrm{G})}$, respectively.

Informative Action mining

For a query entity e, we generate a ranked list of actionable intents $\mathcal{A}$ by iteratively selecting the most informative samples from an action pool $\mathcal{D}$. Since our action mining method is general for any distinct or combined action pools, we do not specify searching strategy in the upper scripts for either the action pool $\mathcal{D}$ or the actions a and $\mathcal{A}$. And because an action $a = (v, o)$ consists of two parts, i.e. a verb part v which represents the activity and a modification part o which indicates the target objective or specifies the movement, our action mining method accordingly consists of two procedures, i.e. to generate the related verbs v in a ranked verb list $\mathcal{V}$ and for each

Entity	St.	Document
Body-building	Q	#Health #Supplement Top Tips To Break Your Weight Loss Plateau `http://some.site.com` `http://some.site.com` #Bodybuilding
	S	#fit #bodybuilding #healthyliving Getting Started With The Raw Foods Diet `http://some.site.com` `http://some.site.com`
	U	It's time to maximize your strength gains! #Bodybuildingcom `http://some.site.com` `http://some.site.com`
	G	The key to any bodybuilding routine is providing your muscles with enough energy to complete your training programme
Language acquisition	Q	I'm about to send my future kids to SJKC for sec language acquisition
	S	Travelling with a private tutor enriches second-language acquisition `http://some.site.com` #inauguration `http://some.site.com`
	U	Any other crazy English sentences to add? `http://some.site.com` `http://some.site.com`
	G	Study.com has thousands of articles about every imaginable degree, area of study and career path that can help you find the school that's right for you.

Table 1
Examples of retrieved documents from different search strategies.

verb to generate the corresponding modifications o in a ranked modification list $\mathcal{O}$. The actionable intents $\mathcal{A} = (\mathcal{V}, \mathcal{O})$ is concatenated thereafter.

For generating a ranked list of verbs $\mathcal{V}$ given entity e, our action mining method firstly extracts all the candidate verbs from the action pool $\mathcal{D}$ into a verb set $\mathcal{U}$, then iteratively takes the most informative verb samples $v \in \mathcal{U}$, each in a mining iteration, into the target verb list $\mathcal{V}$. We consider the informativeness of a candidate verb v to the search users from three aspects, i.e. the significance of v, the representativeness of v, and the diverseness of v with respect to the already selected verbs $v' \in \mathcal{V}$. Each aspects corresponds to a real-valued priority score, and the verb with the largest sum of three priority scores is considered as the most informative verb in the mining iteration and is taken from $\mathcal{U}$ as the next actionable verb into $\mathcal{V}$. Algorithm 1 depicts the procedure for generating a ranked list of verbs $\mathcal{V}$ given a query entity e.

Definition 0.4 *The **significance** criterion evaluates the observation frequency of a candidate verb v through an action pool $\mathcal{D}$, by integrating the contributions of usage frequency and use-case rarity.*

Specifically, the significance of a candidate verb $\mathrm{s}(v)$ is given by

$$\mathrm{s}(v) = \mathrm{uf}(v) \times \mathrm{ucr}(v), \tag{1}$$

in which $\mathrm{uf}(v)$ evaluates a pool-level usage frequency of verb v

$$\mathrm{uf}(v) = \frac{f_v}{\sum_{v' \in \mathcal{U}} f_{v'}}, \tag{2}$$

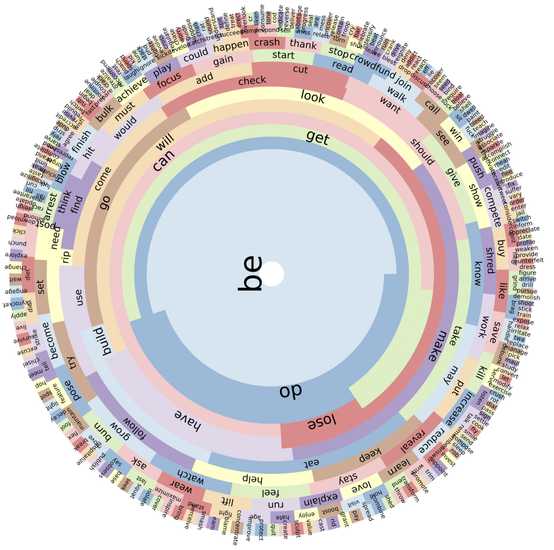
with f_v counting the observation of v through $\mathcal{D}$, and $\mathrm{ucr}(v)$ evaluates an use-case rarity of verb v in the candidate actions

$$\mathrm{ucr}(v) = \log \frac{|\mathcal{D}|}{|\{d|v \in d, d \in \mathcal{D}\}|}, \tag{3}$$

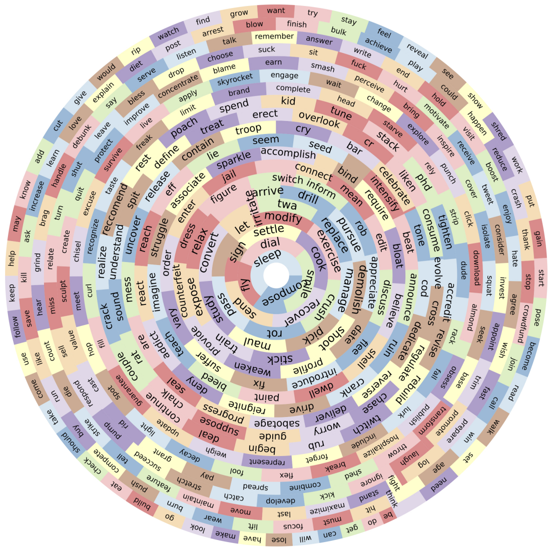
with d indicating an action sample of one or more verbs in $\mathcal{D}$ and $|\mathcal{D}|$ counting the number of action samples in the pool. The significance criterion is based on the TF-IDF statistics in information retrieval for identifying the importance of a verb within an action pool.

Intiutively, the usage frequency highlights the candidate verbs of a high observation frequency among the pool, in order to select actions that are popular among people's minds. On the other hand the use-case rarity is sensitive to the usage scenarios of the candidate verbs and renders high scores for the candidate verbs with unique or special usage scenarios.

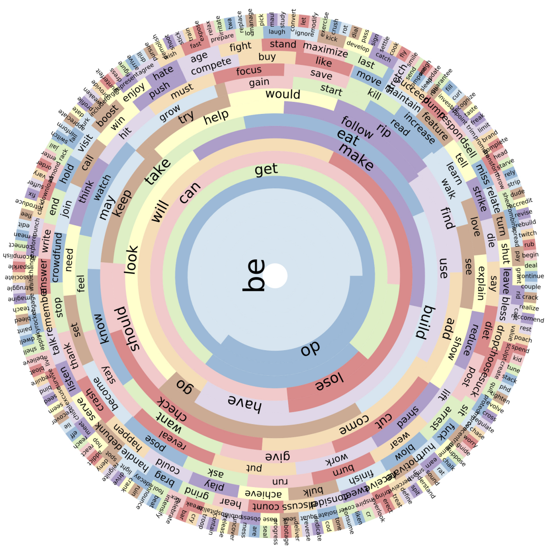
The visualization of usage frequency, use-case rarity, and significance for the candidate verbs of entity "Bodybuilding" are shown in Fig. 1a to 1c, in which verbs with larger evaluation scores are wider and located more closely to the diagram center. The usage frequency reveals the frequently used verbs that are informative for the given entity, such as *lose*, *build*, *eat*, but also favors some verbs such as *be*, *do*, *get*, which are commonly used in English yet not so interesting for the search users. On the other hand, different cases are found in the top ranked verbs by the use-case rarity, such as *sleep*, *fly*, *recover*, which reveals the potentially informative actions under very specific



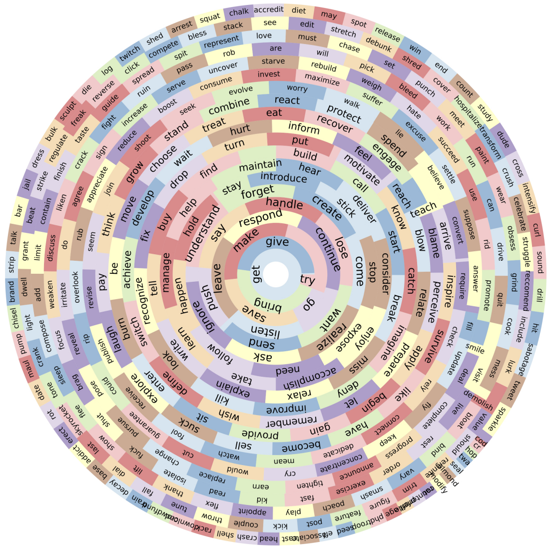
(a) Verb Usage Frequency



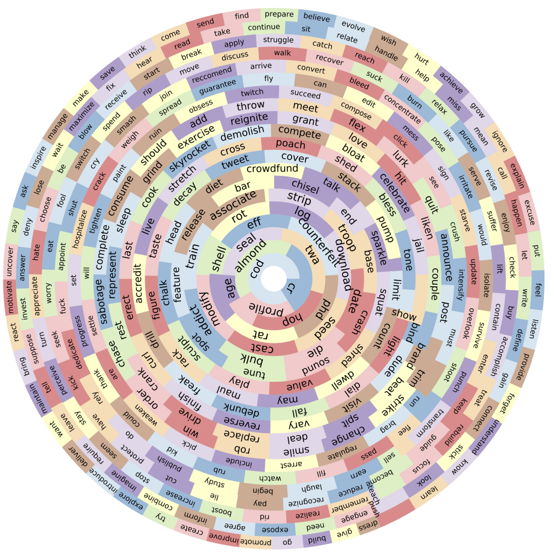
(b) Verb Use-Case Rarity



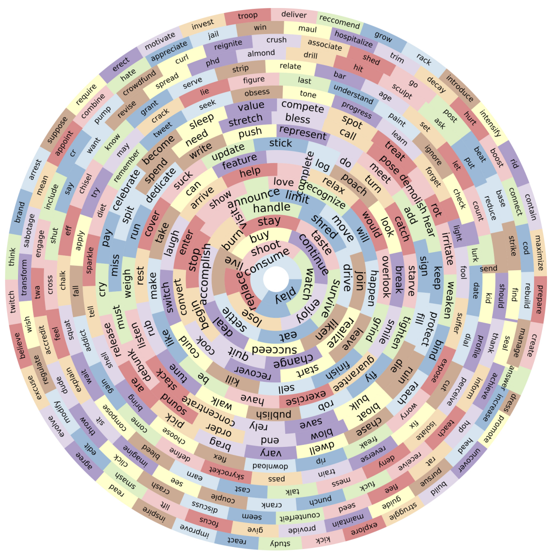
(c) Verb Significance



(d) Verb Representativeness



(e) Verb Diverseness



(f) Verb Informativeness

Figure 1. Quantitative visualization of the verb candidates for entity “Bodybuilding”.

scenarios. Verbs of high usage frequency such as *be* and *do* are significantly restrained by the use-case rarity, but some random verbs such as *compose*, *dial*, *send* are inevitably raised since the criterion is insensitive to verbs of very low usage frequency. Finally, the significance criterion integrates the contributions of both and raises popular verbs that are informative for specific action scenarios.

The significance criterion reveals the verb informativeness based on their observation frequency but ignores their semantic meanings. For example, a high usage frequency of a verb does not necessarily indicate a general semantic meaning in the verb, the latter of which would refine the generalization for verb ranking. And a high use-case rarity for the verbs does not necessarily discriminate their semantic meanings, the latter of which would refine the diversity for verb ranking. To reveal the semantic generalization and diversity of the candidate verbs for ranking, we propose the significance criterion and the diverseness criterion as below.

Definition 0.5 *The **representativeness** criteria evaluates the average resemblance of a candidate verb v to the other candidate verbs $v' \in \mathcal{U}$ in a learned semantic space, in which the semantic meanings of verbs are represented by high-dimensional vectors.*

Specifically, the representativeness of a candidate verb $\mathrm{r}(v)$ is given by

$$\mathrm{r}(v) = \frac{1}{|\mathcal{U}|} \sum_{v' \in \mathcal{U}} \mathrm{sim}(v, v'), \tag{4}$$

in which $\mathrm{sim}(v, v')$ evaluates the semantic resemblance between verbs v and v' through a cosine similarity of their semantic vectors in $\vec{v}$ and $\vec{v'}$

$$\mathrm{sim}(v, v') = \frac{\vec{v} \cdot \vec{v'}}{||\vec{v}|| \times ||\vec{v'}||}. \tag{5}$$

The semantic vectors of verbs are pre-trained on 4.7 million articles from English Wikipedia (Feb 2015) based on a word2vec model (Idio, 2015). Each word is embedded in a 1,000 dimensional real value vector, which is sufficiently large to evaluate the semantic similarity of words (Kang, Wu, & Ren, 2016).

Definition 0.6 *The **diverseness** criterion evaluates the semantic distinction between a candidate verb v and the whole set of already selected verbs in $\mathcal{V}$, which is defined as the minimum distinction between v and all $v' \in \mathcal{V}$.*

Specifically, the diverseness of a candidate verb $\mathrm{d}(v)$ is given by

$$\mathrm{d}(v) = \min_{v' \in \mathcal{V}} \overline{\mathrm{sim}}(v, v'), \tag{6}$$

in which $\overline{\mathrm{sim}}(v, v')$ evaluates the semantic distinction between v and an already selected verb v' through the opposite of their semantic resemblance

$$\overline{\mathrm{sim}}(v, v') = -\mathrm{sim}(v, v'). \tag{7}$$

Intuitively, the minimum of semantic semantic distinctions between a candidate verb v and all selected verbs $v' \in \mathcal{V}$ quantifies the diverseness of the candidate verb v to the selected verb list $\mathcal{V}$.

The visualization of verb representativeness and diverseness[2] for entity "Bodybuilding" are shown in Fig. 1d and 1e, respectively. The representativeness criterion reveals a concentration of the semantic meanings among candidate verbs, in which case the informative and semantically general verbs, such as *save*, *respond*, *listen*, are highly ranked. Although some general and frequently observed verbs such as *give*, *bring*, *try* are inevitably ranked at the front, the other criteria such as inverse use-case frequency and diverseness would strictly restrain their ranking scores. For example, *give* is given the second smallest diverseness score in Fig. 1e. On the other hand, the diverseness criterion has a significant effect in restraining the ranking for candidate verbs which are semantically similar to the already selected ones, so that many variously informative verbs could be selected. For example, *dress* is given the smallest diverseness score in Fig. 1e after *wear* has been selected.

The algorithm for generating a ranked list of informative verbs $\mathcal{V}$ from an action

[2] Since diverseness scores are with the range of $[-1, 0]$, we have to shift them into $[0, 1]$ by plusing 1 to make the width of bar meaningful in visualization.

pool $\mathcal{D}$ is depicted in Algorithm 1. Before the iterative mining, a set $\mathcal{U}$ is initialized with all candidate verbs from $\mathcal{D}$, and a ranked list $\mathcal{A}$ is initialized to be empty $\varnothing$. To generate n informative verbs, the algorithm iteratively selects verb samples $v \in \mathcal{U}$ each at a time, by considering the weighted sum of the significance priority score $\mathrm{s}(v)$, the representativeness priority score $\mathrm{r}(v)$, and the diverseness priority score $\mathrm{d}(v)$ in its mining strategy. The values of weight hyper-parameters λ_1 and λ_2 are by default 1.0 and could be selected by evaluating the generated actions through model selection. The selected verb v is then removed from $\mathcal{U}$ and appended to the end of $\mathcal{V}$. At the end of the algorithm, a ranked list of informative verbs in $\mathcal{V}$ is returned.

Algorithm 1 Algorithm for mining a ranked list of verbs.

```
Initialize 𝒰 = {v|v ∈ 𝒟}
Initialize 𝒱 = ∅
for i = 1 → n do
    for v ∈ 𝒰 do
        score(v) = λ1 s(v) + λ2 r(v) + d(v)
    end for
    v = arg max_{v∈𝒰} score(v)
    Move v from 𝒰 to 𝒱
end for
return 𝒱
```

In the second procedure, our method generates a ranked list of modifiers given an entity e and a selected verb $v \in \mathcal{V}$. Fig. 2 depicts the procedure of extracting a candidate modifier for verb *Started* in a retrieved tweet for entity "Bodybuilding", in which the candidate modifier o consists all the syntactic descendants of v in the parsing tree.

Our algorithm firstly extracts all the candidate modifiers of v from the action pool $\mathcal{D}$ to construct a modifier set $\mathcal{N}$. Then it iteratively takes the most informative modifier $o \in \mathcal{N}$, each in a mining iteration, into the target modifier list $\mathcal{O}$. We consider the informativeness of a candidate modifier o for verb v from two aspects, i.e. the representativeness of o and the diverseness of o with respect to the already selected modifiers $o' \in \mathcal{O}$, both of which correspond to the real-valued priority scores. The modifier with the largest sum of two priority scores is then considered to be the most informative and is taken from $\mathcal{N}$ as the next informative modifier into $\mathcal{O}$. Algorithm 2

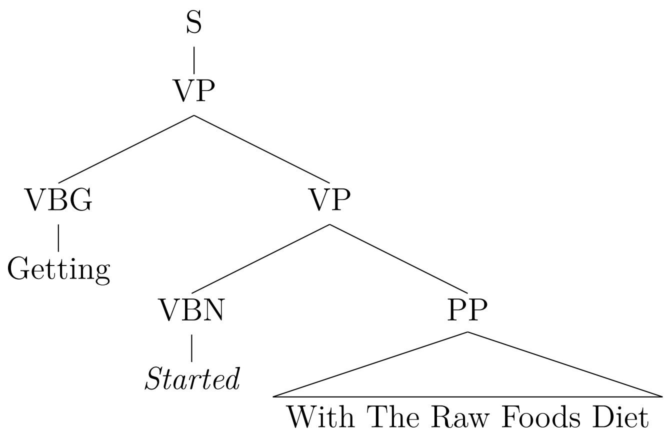


Figure 2. Extracting modifier “With The Raw Foods Diet” for verb *Started*.

depicts the procedure for generating a ranked lit of modifiers $\mathcal{O}$ for verb v given an entity e.

The **representativeness** of modifier follows Def. 0.5, which evaluates the average semantic resemblance of a candidate modifier o to the other candidates $o' \in \mathcal{N}$. Given a verb *learn* for example, the modifier “word meanings” is semantically similar to “semantics” but not to “Business English”. We take the mean of semantic vector representations of the component words in o and o' as their semantic representations

$$\vec{o} = \frac{1}{|o|} \sum_{w \in o} \vec{w}, \tag{8}$$

$$\vec{o'} = \frac{1}{|o'|} \sum_{w \in o'} \vec{w}, \tag{9}$$

and employ the cosine similarity $\mathrm{sim}(o, o')$ to evaluate a semantic resemblance of o and o' as in Eq. 5. Finally, the representativeness of a candidate modifier o is given by

$$r(o) = \frac{1}{|\mathcal{N}|} \sum_{o' \in \mathcal{N}} \mathrm{sim}(o, o'). \tag{10}$$

The **diverseness** of modifier follows Def. 0.6, which evaluates the biggest semantic distinction of a candidate modifier o to the already selected modifiers in $\mathcal{O}$. Given a verb *learn* for example, the candidate modifier “Business English” is semantically more distinct to an already selected modifier “semantics” than another candidate modifier “word meanings”. As in Eq. 7, the semantic distinction of modifiers

o and o' is given by the opposite of their semantic resemblance $\overline{\mathrm{sim}}(o, o')$. And the diverseness of a candidate modifier o to all the selected modifiers $o' \in \mathcal{O}$ is given by

$$\mathrm{d}(o) = \min_{o' \in \mathcal{O}} \overline{\mathrm{sim}}(o, o'). \tag{11}$$

The algorithm for generating a ranked list of informative modifiers $\mathcal{O}$ from an action pool $\mathcal{D}$ for verb v is depicted in Algorithm 2. Before the iterative mining, a set $\mathcal{N}$ is initialized with all candidate modifiers of v in the action pool $\mathcal{D}$, and a ranked list $\mathcal{O}$ is initialized to be empty $\varnothing$. To generate m informative modifiers, the algorithm iteratively selects samples $o \in \mathcal{N}$ each at a time, by considering the weighted sum of the representativeness priority score $\mathrm{r}(o)$ and the diverseness priority score $\mathrm{d}(o)$ in its mining strategy. The value of weight hyper-parameter λ is by default 1.0 and could be further selected by evaluating the generated modifiers through model selection. The selected modifier o is then removed from $\mathcal{N}$ and appended to the end of $\mathcal{O}$. At the end of the algorithm, a ranked list of informative modifiers in $\mathcal{O}$ is returned.

Algorithm 2 Algorithm for mining a ranked list of modifiers.

```
Initialize N = {o|o ∈ D(v)}
Initialize O = ∅
for i = 1 → m do
    for o ∈ N do
        score(o) = λr(o) + d(o)
    end for
    o = arg max_{o∈N} score(o)
    Move o from N to O
end for
return O
```

Experiment

We report our experiment of mining actionable intents in query entities, based on the Action Mining (AM) entity dataset of the Actionable Knowledge Graph (AKG) task in NTCIR-13. The dataset consists of a Dry Run Set of 100 distinct query entities for developing the action mining algorithm and for model selection and a Formal Run Set of 300 different query entities for evaluating the generated actions. Totally 68 types of entities are found in the Action Mining entity dataset, from which we list 15 common types and the corresponding entities in Fig. 3.

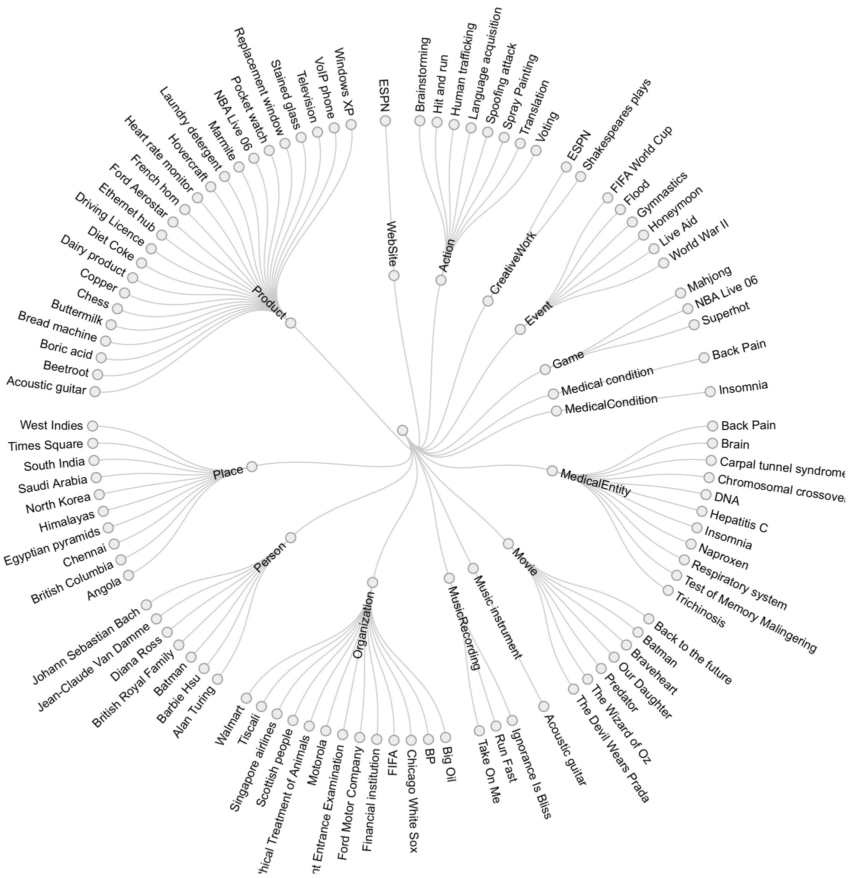


Figure 3. The 15 most common entity types and corresponding entities in the Action Mining Dry Run Set.

With the Dry Run Set, we firstly evaluate the informativeness of actions generated from different search strategies, i.e. (Q) querying the Twitter Search API, (S) querying the Twitter Streaming API, (U) querying the Twitter User Search API and User Timeline API, and (G) querying Google. Each search strategy renders a different pool, based on which we run the informative action mining algorithm to generate the ranked list of actions $\mathcal{A}^{(\mathrm{Q})}$, $\mathcal{A}^{(\mathrm{S})}$, $\mathcal{A}^{(\mathrm{U})}$, and $\mathcal{A}^{(\mathrm{G})}$, respectively.

We randomly choose 20 Dry Run entities and for each entity pick the top 10 actions. 7 people are asked to manually evaluate the informativeness of these actions with numerical scores, in which 0.0 indicates the not informative actions, 1.0 indicates the sort of informative actions, and 2.0 indicates the highly informative actions. We calculate the mean of these scores for each search strategy and conclude from the results that strategy (G) renders the most informative actions with a mean score of 1.0486, the (Q) and (U) strategies render less informative actions with the mean scores of 0.9416 and 0.8741 respectively, and the (S) strategy renders the least informative actions with a mean of 0.5112. These results are consistent with our quality analysis of the action pools in Table 1. Because the (S) strategy has retrieved many irrelevant actions, most of which are from advertisements, we decide to exclude it from the final action mining experiment.

Besides, our case study indicates that different search strategies could retrieve documents of different characteristics. To fulfill the users' diverse needs, we further combine several action pools together to enrich the action mining results. With a similar model selection on the Dry Run Set, we find that combining search strategies into (GQ), (QG), and (QUG) could generate actions with higher mean informativeness scores than the other combined and single strategies. We use the order of strategy names in combination to distinguish the priority for picking actions from each pool. Take the actions $\mathcal{A}^{(\mathrm{GQ})}$ for combination strategy (GQ) for example. Verbs from $\mathcal{A}^{(\mathrm{G})}$ are picked with a higher priority than those from $\mathcal{A}^{(\mathrm{D})}$, while given a common verb, the modifiers from $\mathcal{A}^{(\mathrm{G})}$ are picked with a higher priority than those from $\mathcal{A}^{(\mathrm{D})}$. In our Formal Run experiment, we follow these empirical results and generate informative actions based on these combined Action Mining strategies.

We report four groups of action mining results, in which the first three groups TUA1-0, TUA1-1, TUA1-2 correspond to our submissions to the AM task at NTCIR-13, based on the combined AM strategies (GQ), (QG), (QUG). The last group TUA1-3 corresponds to our additional experiment based on the (QG) strategy, with the hyper-parameters of $\lambda_1 = 0.05$, $\lambda_2 = 0.6$, and $\lambda = 0.6$ decided through a model selection. Specifically, by decreasing λ_1 our action mining algorithm could effectively restrain the general verbs like "get" and "do", and by decreasing λ_2 and λ our algorithm could increase the diversity in selected actions.

We follow the AM task evaluation metric (Blanco et al., 2017), in which the informativeness of generated actions is divided into four levels, with L0 if there is no information of a to e, L1 if a can be informative for e, L2 if a has been or will be definitely taken by e, and L3 if some people, organizations or other subjects definitely have taken or will take a for e. With these labels manually annotated to each pair (a, e) by the CrowdFlower workers, the quality of informative action ranking is then evaluated based on the normalized Expected Reciprocal Rank (nERR) and the normalized Discounted Cumulative Gain (nDCG) metrics as illustrated below.

Given an entity e, the nERR@k score evaluates an early-satisfaction of the top k ranked actions to search users' aim by

$$\text{nERR@}k = \frac{\text{ERR@}k}{\text{idealERR@}k}, \tag{12}$$

$$\text{ERR@}k = \sum_{i=1}^{k} \frac{1}{i} p(e, a_i) \prod_{j=1}^{i-1} (1 - p(e, a_j)), \tag{13}$$

in which $p(e, a_i)$ is the probability of the ith ranked action a_i matches the user's aim in e, and $p(e, a_i) \prod_{j=1}^{i-1}(1 - p(e, a_j))$ measures the probability that user's aim is not matched until the ith action in the list. Since ERR@k score is sensitive to the position of the firstly matched action, if a front-ranked action matches the user's aim then ERR@k score is large. idealERR@k corresponds to ERR@k of an ideal ranked list, which normalizes the nERR@k score. Therefore, a large nERR@k in range of $[0, 1]$ suggests an early satisfaction of the generated actions in $\mathcal{A}$ to the users' aim.

The nDCG@k score evaluates the ranking quality of informativeness in the top k ranked actions for an entity by

$$\text{nDCG@}k = \frac{\text{DCG@}k}{\text{idealDCG@}k}, \tag{14}$$

$$\text{DCG@}k = \sum_{i=1}^{k} \frac{2^{\text{rel}_i} - 1}{\log_2(i+1)}, \tag{15}$$

in which rel_i is the informativeness evaluation from human experts for the ith action. DCG@k measures the ranking quality for the top k actions, i.e. DCG@k is large only if the informativeness of generated actions is large and the more informative actions are ranked in front of the less informative ones. nDCG@k in Eq. 14 normalizes the DCG@k

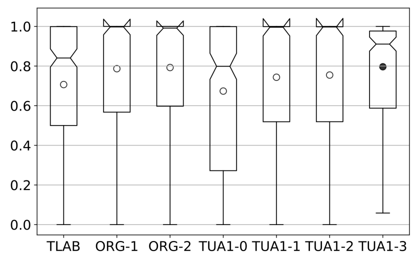
score into $[0, 1]$, where idelDCG@k corresponds to DCG@k of an ideal ranking.

Because our action mining approach is divided into a verb mining procedure and a modifier mining procedure, we firstly report our quantitative evaluations of ranked verb lists in $\mathcal{V}$ and compare them with the results from the other participation groups in the AM task, i.e. the TLAB group and the two ORG groups. Specifically, TLAB (Rahman & Takasu, 2017) retrieved actions from multiple data sources[3] with a probabilistic model which maximized the conditional probabiilty of an action given the entity instance and the entity type. ORG (Blanco et al., 2017) retrieved two groups of actions that differ in their parameterizations from the Yahoo Webscope (L4-L9) dataset and the first Quora answers dataset, based on a probabilistic graphical model in which the presence of an action was inferred from the part-of-speech of action verbs, the language model for actions, and the similarity between entities. The nERR@10, nERR@20, nDCG@10, and nDCG@20 metric scores from all participation groups are shown in Fig. 4, with the minimum, first quartile, median, third quartile, and maximum shown in box plots. We also scatter the average scores with hollow circles and the largest average score among all groups with a solid circle.

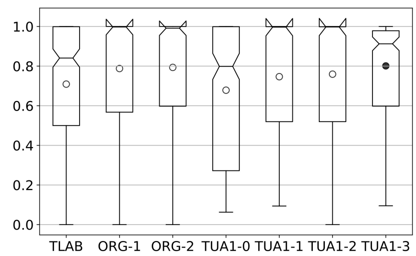
In Fig. 4 TUA1-3 achieves the best nERR@10 and nERR@20 scores, with the average of 0.7964 and 0.8011 respectively. Even the lowest nERR@10 and nERR@20 scores of the TUA1-3 group, i.e. 0.0584 and 0.0952, are still higher than those of the other groups. The results suggest that our ranked lists of verbs are the most informative among all the participation groups and that the users' aims get properly matched even for the entities which are essentially difficult to match.

The nDCG@10 and nDCG@20 scores in Fig. 4 suggest that the verb list from ORG-1 group are more properly ranked than the other groups. However, we find that by combining search strategies and selecting the weight hyper-parameters λ_1 and λ_2 in Algorithm 1, our approach also achieves steady improvements. For those essentially difficult entities TUA1-3 group outperforms the others with significantly higher nDCG@10 and nDCG@20 scores of 0.2164 and 0.5756 respectively. The results suggest

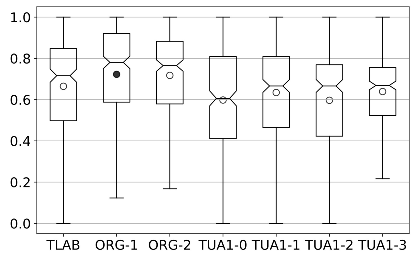
[3] The TALAB experiment data consists of the Reuters and Leipzig corpora, Wikipedia, Trip Advisor and Amazon user review, movie review, and Medline dataset



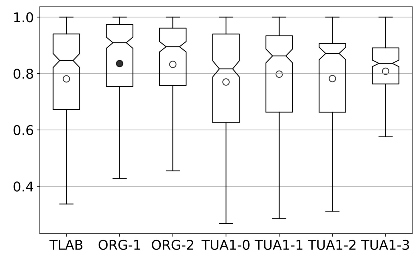
(a) nERR@10



(b) nERR@20



(c) nDCG@10



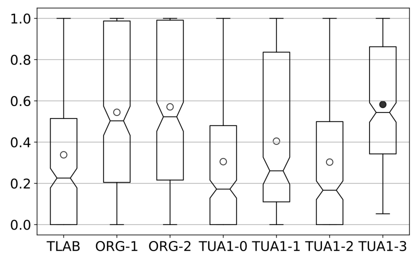
(d) nDCG@20

Figure 4. Quantitative evaluations for the ranked verb lists.

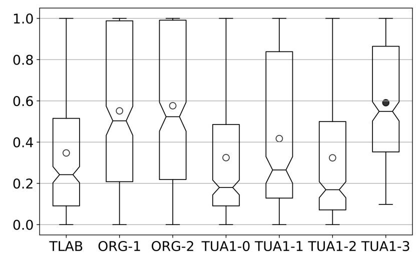
that our approach could generate properly ranked lists of verbs for any query entity and that for the essentially difficult entities our ranking is stabler than the other groups.

We secondly report the nERR@10, nERR@20, nDCG@10, and nDCG@20 scores of ranked list of actions, i.e. verb-modifier pairs, in $\mathcal{A}$. The minimum, first quartile, median, third quartile, and maximum of the evaluation scores for all participation group are shown in Fig. 5, with the average scores for each group scattered as hollow circles and the largest average scores among all participation groups scattered as a solid circle.

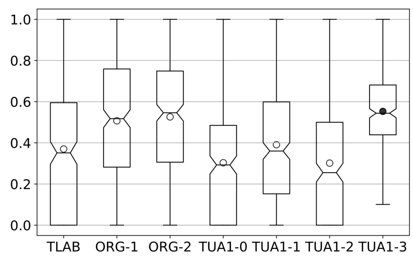
In Fig. 5 TUA1-3 group achieves the best nERR@10 and nERR@20 scores, with the averages of 0.5825 and 0.5910 respectively. Compared with TUA1-1 which mines actions from similar pools, we find that adjusting the weight hyper-parameters λ_1, λ_2 in Algorithm 1 and λ in Algorithm 2 could significantly improve the matching of generated actions to users' intents. The result also suggests that putting more weights on the diverseness priority, i.e. decreasing λ, helps selecting more informative actions. Besides, we observe that even the lowest nERR@10 and nERR@20 scores of our TUA1-3 group, i.e. 0.0526 and 0.0981 are still higher than those of the other participation groups, which implies a stabler performance of our approach.



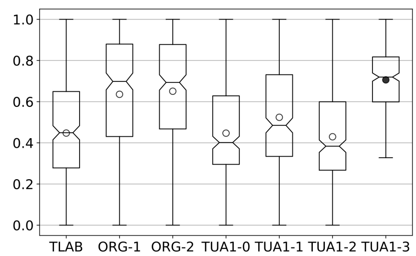
(a) nERR@10



(b) nERR@20



(c) nDCG@10

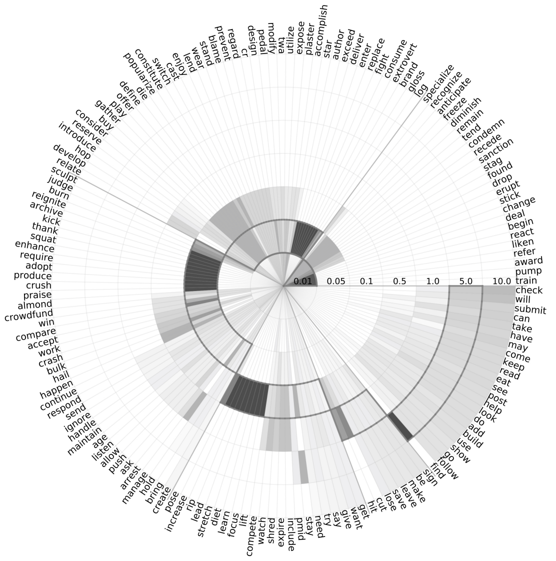


(d) nDCG@20

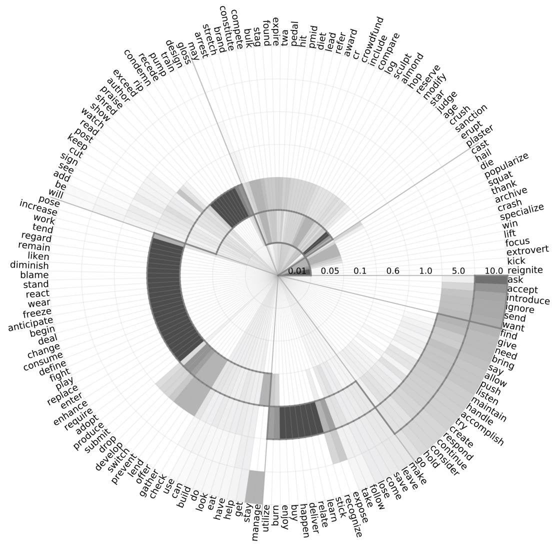
Figure 5. Quantitative evaluations for the ranked action lists.

TUA1-3 group outperforms the other groups in the nDCG@10 and nDCCG@20 scores in Fig. 5, with the averages of 0.5528 and 0.7059 respectively. The results suggest that our approach is effective for generating properly ranked list of actions and that the diverseness priority contributes significantly to the proper ranking of informative actions. Given essentially difficult entities, the lowest nDCG@10 and nDCG@20 scores of TUA1-3 are significantly better than those of the other groups, which suggests that our approach is the stablest for action ranking.

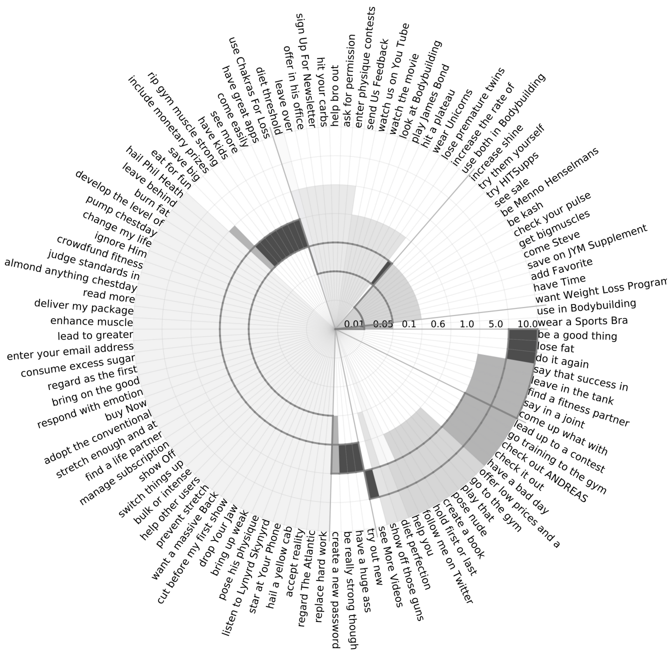
We demonstrate the association of retrieved verbs and actions with different hyper-parameter settings for the entity “Bodybuilding” in Fig. 6. Specifically, hyper-parameters λ_1 and λ_2 control the significance and representativeness scores in Algorithm 1 for verb selection and λ controls the representativeness score in Algorithm 2 for modifier selection. For each parameter combination, we retrieve informative verbs and actions and associate them with the values in λ_1, λ_2, and λ, respectively. The radii are segmented into seven parts according to the parameter value settings, and the circumferences are annotated with the retrieved items. The association of retrieved items with parameter values are encoded into grayscale colors, in which strong associations are encoded in dark gray while weak associations are encoded in light gray. We cluster the retrieved items of the same median parameter value into a group, with a dark boundary denoting the median of parameter values.



(a) Retrieved verbs w.r.t. different λ_1's



(b) Retrieved verbs w.r.t. different λ_2's



(c) Retrieved actions w.r.t. different λ's

Figure 6. Visualization of retrieved actions for entity "Bodybuilding" under different hyper-parameter settings in λ_1, λ_2, and λ.

Fig. 6a demonstrates that for informative verb mining, λ_1 has a significant impact on the repression of common verbs, such as *go*, *come*, *take*. A reasonably small λ_1 will push the results towards the semantically interesting ones, such as *fight*, *wear*, *burn*.

Fig. 6b demonstrates that a reasonably small λ_2 could balance the generality and informativeness and encourage the informative verbs, such as *consume*, *drop*, *build*, while a large λ_2 will attract the semantically more general but less interesting verbs, such as *give*, *bring*, *try.* Fig. 6c demonstrate that a reasonably small λ could balance the generality and diversity and retrieve the informative actions, such as *eat* for fun, *consume* excess suger, *listen* to Lynyrd Skynyrd, while a large λ will only restrain the diverseness of retrieved objectives and encourage the semantically general objectives, such as "a good thing", "it again", "up what with".

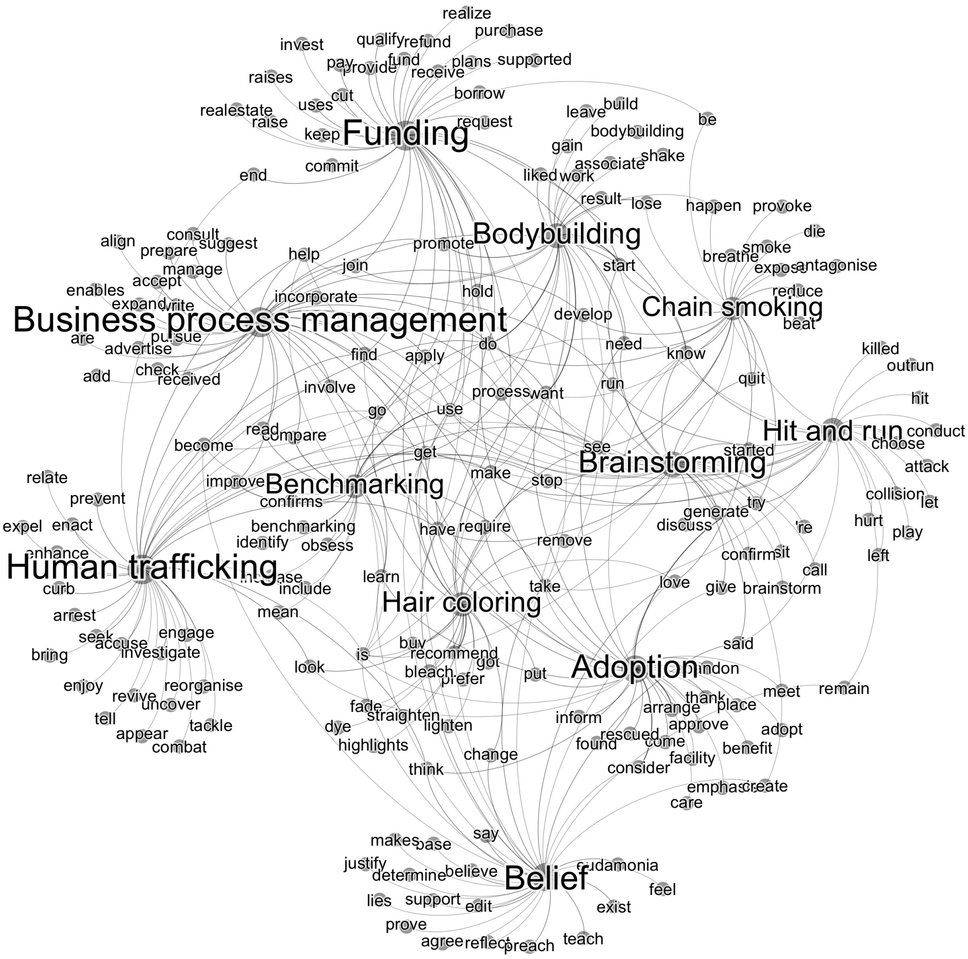


Figure 7. Verb-entity relation diagram for the TUA1-3 action mining results.

We report a case study of the generated verbs and actions. Fig. 7 plots 188 informative verbs generated for 11 query entities from the Formal Run dataset. Among these verb-entity relations, we find exclusively informative verbs for query entities, such as *rescued* for Adoption, *expose* for Chain smoking, *bleach* for Hair coloring, *accuse*, *arrest*, and *investigate* for Human trafficking. Most of these exclusively informative verbs are labeled with L3 by the CrowdFlower workers, which implies that such verbs are also highly reliable for inferring the search uses' real intents. We also find a few nonexclusive verbs that are informative to multiple entities, such as *know* for Adoption, Bodybuilding, Brainstorming, Chain smoking, Funding, Hit and run and *use* for Benchmarking, Bodybuilding, Brainstorming, Business process management, Funding, Hair coloring, Hit and run. Analysing these nonexclusive verbs could reveal the general needs of search users and expose the underlying relationships among different entities. The exclusive verbs might turn into nonexclusive as the number of entities and verbs grows. However, evaluating a degree of exclusiveness for the informative verbs would still reveal the users' real intents in particular query entities and expose the general needs among different entities.

Table 2 shows the action examples generated for the Formal Run entities. These include advising actions such as "*recommend* good home hair colour" for Hair coloring and "*need* a good workout schedule . . . " for Bodybuilding, as well as many highly recommendable actions, such as "*stop* smoking cigarette" for Chain smoking and "*borrow* money" for Funding. Besides, non-exclusive verbs together with the properly generated modifiers such as "*use* the playdoh technique" and "*use* Journey Mapping . . . " for Brainstorming and "*use* process or project management tools" for Business Process management are found considerably informative in our approach. For those essentially difficult entities annotated by the CrowdFlower works, such as Belief, our approach still generates actions of "*challenge* one's Belief" and "*change* their mind" which are considerably meaningful and informative for the search users.

Conclusion

Understanding users' potential intents in very short queries is an important study in information retrieval. Most of the current studies focus on organizing search queries into a few pre-defined intent categories, leaving out many potentially informative actions. In this paper, we present a novel research for mining a ranked list of such informative actions for a query entity and explore a variety of search strategies for generating a pool of candidate actions. We propose three criteria, i.e. significance, representativeness, and diverseness, for evaluating the informativeness of candidate actions and propose two action mining algorithms for iteratively generating the ranked list of action verbs and verb modifiers respectively based on these criteria. A thorough experiment is performed based on the AM query entity dataset, with the quantitative evaluations of generated actions according to the normalized Expected Reciprocal Rank (nERR) metric and the normalized Discounted Cumulative Gain (nDCG) metric. Experiment results suggest that our approach could generate both early-satisfying actionable intents to match the search users' aims, i.e. the best nERR scores, and informative actions according to their informativeness, i.e. the promising nDCG scores. Besides, our approach is proved to be very stable even for the essentially most difficult entities. An analysis of the generated actions further indicates that emphasizing the diverseness priority in our action mining algorithm could steadily improve the nERR and nDCG results.

An interesting finding in our study is that the (S) search strategy which queries the Twitter Streaming API does not render proper results as the other search strategies. This is probably because that the (S) strategy has retrieved too much noise from the real-time Twitter stream, which increased the difficulty of selecting proper actions for our action mining algorithm. As such sensitivity has inevitably limited the diverseness in our action mining results, our future work will focus on the reasoning of more accurate relationships between entities and actions from the general action pools.

Entity	Action
Adoption	want to adopt daughter found a wonderful couple with parent profiles put up for adoption
Benchmarking	compare procedures compare to peer companies improve productivity living standard
Bodybuilding	need a good workout schedule ... lose fat skinnybody `http://some.site.com` get start in bodybuilding
Brainstorming	discuss issues use the playdoh technique use Journey Mapping in their own way
Business process management	consult business development use process or project management tools promote online business
Chain smoking	expose to second hand smoke stop smoking cigarette know that get cancer
Funding	apply for our funding borrow money cut federal funding for the arts : `http://some.site.com`
Hair coloring	recommend good home hair colour remove hair dye from skin buy hair color
Hit and run	stop at the scene of an accident run over run away
Human trafficking	stop a cross-border human trafficking gang help fight against human trafficking stop the horrors of slavery and human trafficking
Belief	challenge one's Belief change their mind contemplate credulity

Table 2
Examples of the informative actions from the TUA1-3 results.

References

Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In *Proceedings of the second acm international conference on web search and data mining* (pp. 5–14).

Arguello, J., Diaz, F., Callan, J., & Crespo, J.-F. (2009). Sources of evidence for vertical selection. In *Proceedings of the 32nd international acm sigir conference on research and development in information retrieval* (pp. 315–322).

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. *The semantic web*, 722–735.

Blanco, R., Joho, H., Jatowt, A., Yu, H., & Yamamoto, S. (2017). Overview of ntcir-13 actionable knowledge graph (akg) task. In *Proceedings of the ntcir-13 conference.*

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 acm sigmod international conference on management of data* (pp. 1247–1250).

Broder, A. (2002). A taxonomy of web search. In *Acm sigir forum* (Vol. 36, pp. 3–10).

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E. R., & Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *Aaai* (Vol. 5, p. 3).

Celikyilmaz, A., Hakkani-Tür, D., & Tur, G. (2011). Leveraging web query logs to learn user intent via bayesian latent variable model. In *Proceedings of the 28th international conference on machine learning.*

Guo, J., Xu, G., Cheng, X., & Li, H. (2009). Named entity recognition in query. In *Proceedings of the 32nd international acm sigir conference on research and development in information retrieval* (pp. 267–274).

Hassan Awadallah, A., White, R. W., Pantel, P., Dumais, S. T., & Wang, Y.-M. (2014). Supporting complex search tasks. In *Proceedings of the 23rd acm international conference on conference on information and knowledge management* (pp. 829–838).

Hoffart, J., Suchanek, F. M., Berberich, K., & Weikum, G. (2013). Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artificial Intelligence*, *194*, 28–61.

Hollerit, B., Kröll, M., & Strohmaier, M. (2013). Towards linking buyers and sellers: detecting commercial intent on twitter. In *Proceedings of the 22nd international conference on world wide web* (pp. 629–632).

Hu, J., Wang, G., Lochovsky, F., Sun, J.-t., & Chen, Z. (2009). Understanding user's query intent with wikipedia. In *Proceedings of the 18th international conference on world wide web* (pp. 471–480).

Idio. (2015). Enwiki word2vec model 1000 dimensions. Retrieved from `https://github.com/idio/wiki2vec`

Jansen, B. J., Booth, D. L., & Spink, A. (2007). Determining the user intent of web search engine queries. In *Proceedings of the 16th international conference on world wide web* (pp. 1149–1150).

Jansen, B. J., Booth, D. L., & Spink, A. (2008). Determining the informational, navigational, and transactional intent of web queries. *Information Processing & Management*, *44*(3), 1251–1266.

Jiang, D., Leung, K. W.-T., & Ng, W. (2016). Query intent mining with multiple dimensions of web search data. *World Wide Web*, *19*(3), 475–497.

Kang, X., Wu, Y., & Ren, F. (2016). Kgo at the ntcir-12 temporalia task: Exploring temporal information in search queries. In *Ntcir.*

Li, J., Wu, Y., Zhang, P., Song, D., & Wang, B. (2017). Learning to diversify web search results with a document repulsion model. *Information Sciences*, *411*, 136–150.

Li, X., Wang, Y.-Y., & Acero, A. (2008). Learning query intent from regularized click graphs. In *Proceedings of the 31st annual international acm sigir conference on research and development in information retrieval* (pp. 339–346).

Lin, T., Pantel, P., Gamon, M., Kannan, A., & Fuxman, A. (2012). Active objects: Actions for entity-centric search. In *Proceedings of the 21st international conference on world wide web* (pp. 589–598).

Nickel, M., Murphy, K., Tresp, V., & Gabrilovich, E. (2016). A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, *104*(1), 11–33.

Qian, Y., Sakai, T., Ye, J., Zheng, Q., & Li, C. (2013). Dynamic query intent mining from a search log stream. In *Proceedings of the 22nd acm international conference on information & knowledge management* (pp. 1205–1208).

Rahman, M. M., & Takasu, A. (2017). Tlab at the ntcir-13 akg task. In *Proceedings of the ntcir-13 conference.*

Reinanda, R., Meij, E., & de Rijke, M. (2015). Mining, ranking and recommending entity aspects. In *Proceedings of the 38th international acm sigir conference on research and development in information retrieval* (pp. 263–272).

Ren, P., Chen, Z., Ma, J., Wang, S., Zhang, Z., & Ren, Z. (2015). Mining and ranking users' intents behind queries. *Information Retrieval Journal*, *18*(6), 504–529.

Rose, D. E., & Levinson, D. (2004). Understanding user goals in web search. In *Proceedings of the 13th international conference on world wide web* (pp. 13–19).

Sadikov, E., Madhavan, J., Wang, L., & Halevy, A. (2010). Clustering query refinements by user intent. In *Proceedings of the 19th international conference on world wide web* (pp. 841–850).

Santos, R. L., Macdonald, C., & Ounis, I. (2010). Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th international conference on world wide web* (pp. 881–890).

Singhal, A. (2012). Introducing the knowledge graph: things, not strings. *Official google blog*.

Song, Y., & He, L.-w. (2010). Optimal rare query suggestion with implicit user feedback. In *Proceedings of the 19th international conference on world wide web* (pp. 901–910).

Spirin, N. V., He, J., Develin, M., Karahalios, K. G., & Boucher, M. (2014). People search within an online social network: Large scale analysis of facebook graph search query logs. In *Proceedings of the 23rd acm international conference on conference on information and knowledge management* (pp. 1009–1018).

Suchanek, F. M., Kasneci, G., & Weikum, G. (2007). Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on world wide web* (pp. 697–706).

Uyar, A., & Aliyu, F. M. (2015). Evaluating search features of google knowledge graph and bing satori: entity types, list searches and query interfaces. *Online Information Review*, *39*(2), 197–213.

Wang, C.-J., & Chen, H.-H. (2014). Intent mining in search query logs for automatic search script generation. *Knowledge and information systems*, *39*(3), 513–542.

Wang, J., Cong, G., Zhao, W. X., & Li, X. (2015). Mining user intents in twitter: A semi-supervised approach to inferring intent categories for tweets. In *Aaai* (pp. 318–324).

Yin, X., & Shah, S. (2010). Building taxonomy of web search intents for name entity queries. In *Proceedings of the 19th international conference on world wide web* (pp. 1001–1010).

Zhang, B., Li, H., Liu, Y., Ji, L., Xi, W., Fan, W., … Ma, W.-Y. (2005). Improving web search results using affinity graph. In *Proceedings of the 28th annual international acm sigir conference on research and development in information retrieval* (pp. 504–511).

Zhang, Z., & Nasraoui, O. (2006). Mining search engine query logs for query recommendation. In *Proceedings of the 15th international conference on world wide web* (pp. 1039–1040).

Zhao, X. W., Guo, Y., He, Y., Jiang, H., Wu, Y., & Li, X. (2014). We know what you want to buy: a demographic-based system for product recommendation on microblogs. In *Proceedings of the 20th acm sigkdd international conference on knowledge discovery and data mining* (pp. 1935–1944).

Zuccon, G., & Azzopardi, L. (2010). Using the quantum probability ranking principle to rank interdependent documents. In *Ecir* (Vol. 10, pp. 357–369).