Overview of the NTCIR-12 MobileClick-2 Task

Makoto P. Kato
Kyoto University
kato@dl.kuis.kyoto-u.ac.jp

Tetsuya Sakai
Waseda University
tetsuyasakai@acm.org

Takehiro Yamamoto
Kyoto University
tyamamot@dl.kuis.kyoto-u.ac.jp

Virgil Pavlu
Northeastern University
vip@ccs.neu.edu

Hajime Morita
Kyoto University
morita@nlp.ist.i.kyoto-u.ac.jp

Sumio Fujita
Yahoo Japan Corporation
sufujita@yahoo-corp.jp

ABSTRACT

This is an overview of the NTCIR-12 MobileClick-2 task (a sequel to 1CLICK in NTCIR-9 and NTCIR-10). In the MobileClick task, systems are expected to output a concise summary of information relevant to a given query and to provide immediate and direct information access for mobile users. We designed two types of MobileClick subtasks, namely, iUnit ranking and summarization subtasks, in which twelve research teams participated and submitted 66 runs. We describe the subtasks, test collection, and evaluation methods and then report official results for NTCIR-12 MobileClick.

1. INTRODUCTION

Current web search engines usually return a ranked list of URLs in response to a query. After inputting a query and clicking on the search button, the user often has to visit several web pages and locate relevant parts within those pages. While these actions require significant effort and attention, especially for mobile users, they could be avoided if a system returned a concise summary of relevant information to the query [22].

The NTCIR-12 MobileClick task (and its predecessors, 1CLICK tasks organized in NTCIR-9 [23] and NTCIR-10 [6]) aims to directly return a summary of relevant information and immediately satisfy the user without requiring a lot of interaction with the device. Unlike the 1CLICK tasks, we expect the output to be a two-layered summary where the first layer contains the most important information and an outline of additional relevant information, while the second layer contains detailed information that can be accessed by clicking on links in the first layer. As shown in Figure 1, for query "NTCIR-11", a MobileClick system presents general information about NTCIR-11 and a list of core tasks in the first layer. When the "MobileClick" link is clicked by the user, the system shows text in the second layer that explains the topic of that link.

Textual output of the MobileClick task is evaluated based on *information units* (*iUnits*) rather than document relevance. The performance of a submitted system is scored higher if it generates summaries including more important iUnits. In addition, we require systems to minimize the amount of text the user has to read or, equivalently, the time she has to spend in order to obtain relevant information. Although these evaluation principles were also taken into account in the 1CLICK tasks, they are extended to two-layered summaries where users can read a summary in multiple ways. We assume a user model that reads different parts of the summary by probabilistically clicking on links and compute an evaluation metric based on the importance of iUnits read as well as the time spent to obtain them.

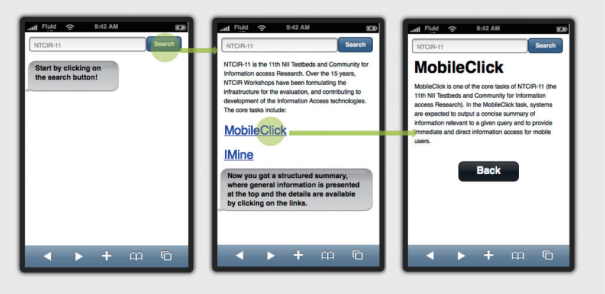


Figure 1: An application of the MobileClick task. A concise two-layered summary can fit a small screen of the mobile device, and can satisfy diverse information needs.

MobileClick-2 attracted twelve research teams from eight countries. Table 1 provides a list of NTCIR-12 MobileClick participants with the number of iUnit ranking and summarization submissions. The total number of submissions was 66.

One of the biggest changes from the previous round of MobileClick was the evaluation system: we finished all the evaluation processes before releasing test data, and have returned evaluation results right after run submissions at our website[1]. This might enable participants to improve their systems based on returned results. In addition, the reproducibility was highly improved since there is no need to conduct additional assessments for new submissions. Another new trial in MobileClick-2 was *leader board*, by which participants can see evaluation results of the others. We expected more participants and higher performances by enhancing the visibility of state-of-the-arts performances achieved so far.

The remainder of this paper is structured as follows. Section 2 describes the details of the iUnit ranking and summarization subtasks. Section 3 introduces a test collection consisting of queries, iUnits, and a document collection. Section 4 describes our evaluation methodology. Section 5 reports on the official evaluation results for both subtasks. Finally, Section 6 concludes this paper.

2. SUBTASKS

MobileClick-2 comprises iUnit ranking and summarization subtasks. This section explains the two types of subtasks, and their input, output, and evaluation methodology.

2.1 iUnit Ranking Subtask

[1] http://www.mobileclick.org/

Table 1: NTCIR-12 MobileClick-2 participants and the number of iUnit ranking and summarization submissions.

Team ID	Team name	iUnit ranking		iUnit summarization		Total
		English	Japanese	English	Japanese	
ALICA [13]	University of Alicante	1	0	1	0	2
cuis [10]	The Chinese University of Hong Kong	7	0	3	0	10
IISR [4]	National Central University	4	0	0	0	4
IRIT [2]	Toulouse Institute of Computer Science Research University of Paul Sabatier Toulouse France	4	0	3	0	7
JUNLP [3]	JADAVPUR UNIVERSITY	1	0	1	0	2
NUTKS [28]	Nagaoka University of Technology	0	2	0	0	2
ORG	MobileClick Organizers	2	2	3	3	10
RISAR [15]	RMIT University	2	0	1	0	3
rsrch [27]	Rakuten, Inc	0	2	0	1	3
TITEC [8]	Tokyo Institute of Technology	1	0	1	0	2
UHYG [5]	University of Hyogo	2	2	2	2	8
YJST [16]	Yahoo Japan Corporation	1	4	1	7	13
		25	12	16	13	66

The iUnit ranking subtask is a task where systems are expected to rank a set of information pieces (iUnits) based on their importance for a given query. This subtask was devised to enable *componentized* evaluation, where we could separately evaluate the performance of estimating important iUnits and summarizing iUnits into a two-layered summary.

We provided a set of queries, a set of iUnits, and documents from which the iUnits were extracted. Note that the set of iUnits included irrelevant iUnits, which participants should rank below the other iUnits. We then asked participants to submit, for each query, a list of iUnits that are ordered by their estimated importance. More concretely, we accept a tab-delimited-values (TSV) file as an iUnit ranking run, where the first line must be a simple system description, and each of the other lines must represents a single iUnit. Therefore, a run file should look like the one shown below:

Listing 1: Example of an iUnit ranking run

```
This is an example run file
qid    uid    score
qid    uid    score
....
```

where "qid" is a query ID, "uid" is a iUnit ID, and "score" is estimated importance of the iUnit. In many ways, the iUnit ranking runs are similar with TREC ad-hoc runs in that they are essentially a ranked list of the objects retrieved. Note that we did not use "score" values for evaluation, and used the order of iUnits in run files.

2.2 iUnit Summarization Subtask

The iUnit summarization subtask is defined as follows: Given a query, a set of iUnits, and a set of intents, generate a structured textual output. In MobileClick, more precisely, the output must consist of two layers. The first layer is a list of iUnits and links to the second layer, while the second layer consists of lists of iUnits. Each link must be one of the provided intents and be associated with one of the iUnit lists in the second layer. Each list of iUnits in the first and second layers can include at most X characters so that it fits ordinary mobile screen size. The length of links is counted, while symbols and white spaces are excluded. In MobileClick-2, X is set to 420 for English and 280 for Japanese.

Each run must be a XML file that satisfies a DTD shown below:

Listing 2: DTD for an iUnit summarization run

```
<!ELEMENT results (sysdesc, result*)>
<!ELEMENT sysdesc (#PCDATA)>
<!ELEMENT result (first, second*)>
<!ELEMENT first (iunit | link)*>
<!ELEMENT second (iunit)*>
<!ELEMENT iunit EMPTY>
<!ELEMENT link EMPTY>
<!ATTLIST result qid NMTOKEN #REQUIRED>
<!ATTLIST iunit uid NMTOKEN #REQUIRED>
<!ATTLIST link iid NMTOKEN #REQUIRED>
<!ATTLIST second iid NMTOKEN #REQUIRED>
```

where

- The XML file includes a [results] element as the root element;
- The [results] element contains exactly one [sysdesc] element;
- The [results] element also contains [result] elements, each of which corresponds a two-layered summary and has a [qid] attribute;
- A [result] element contains a [first] element and [second] elements;
- The [first] element contains [iunit] and [link] elements;
- A [second] element has an attribute [iid], and contains [iunit] elements.
- An [iunit] element has an attribute [uid] (iUnit ID); and
- A [link] element has an attribute [iid] (intent ID), which identifies a [second] element to be linked.

Note that the same [iunit] element may appear multiple times, e.g. an iUnit may appear in the [first] element and two [second] elements.

An XML file example that satisfies the DTD is shown below:

Listing 3: Example of an iUnit summarization run

```
<?xml version="1.0" encoding="UTF-8" ?>
<results>
  <sysdesc>
  Organizer Baseline
  </sysdesc>
  <result qid="MC-E-0001">
    <first>
      <iunit uid="MC-E-0001-U001" />
      <iunit uid="MC-E-0001-U003" />
      <link iid="MC-E-0001-I006" />
      <iunit uid="MC-E-0001-U004" />
      <link iid="MC-E-0001-I002" />
    </first>
    <second iid="MC-E-0001-I006">
      <iunit uid="MC-E-0001-U011" />
      <iunit uid="MC-E-0001-U019" />
    </second>
    <second iid="MC-E-0001-I002">
      <iunit uid="MC-E-0001-U029" />
      <iunit uid="MC-E-0001-U021" />
    </second>
  </result>
</results>
```

3. TEST COLLECTION

The NTCIR-12 MobileClick test collection includes queries, iUnits, intents, and a document collection. We describe the details of those components in the following subsections.

3.1 Queries

The NTCIR-12 MobileClick test collection includes 100 English and 100 Japanese queries. Unlike the MobileClick-1 task, we selected more ambiguous/underspecified, or short queries like the 1CLICK tasks held in the past NTCIR. This is because we opt to focus on queries that are often utilized in mobile devices, and to tackle the problem of diverse intents in searchers.

We used a Wider Planet toolbar log from April to July 2014 for obtaining real-users' queries, and translated them into English and Japanese. We selected frequent queries that belong to either *CELEBRITY*, *LOCAL*, and *DEFINITION* categories. Questions posted on Yahoo! Japan Chiebukuro[2] were used to generate *QA* queries. Those query categories were also employed in the 1CLICK tasks, since they are frequently used by mobile users [11]. The definition of those categories is shown below (numbers in the brackets indicate the number of queries in the category):

CELEBRITY (20) names of celebrities such as artists, actors, politicians, and athletes.

LOCAL (20) landmarks and facilities (*e.g.* "tokyo sky tree"), or entities with geographical constraints (*e.g.* "banks Kyoto").

DEFINITION (40) ambiguous terms that are often input to know their definition.

QA (20) natural language questions

3.2 Documents

To provide participants with a set of iUnits for each query, we downloaded 500 top-ranked documents that were returned by Bing search engine[3] in response to each query, from which we extracted iUnits as explained in the next subsection. This crawling was conducted from May 29 to June 1, 2016. As we failed to access some of the documents, the number of downloaded documents per query is fewer than 500. The average number of documents for English queries is 418 and that for Japanese queries is 442.

NTCIR participants can obtain this document collection after their registration, and utilize them to estimate the importance of each iUnit and intent probability, *etc.*

3.3 iUnits

Like the 1CLICK tasks held in the past NTCIR, we used iUnits as a unit of information in the MobileClick task. iUnits are defined as *relevant*, *atomic*, and *dependent* pieces of information, where

- *Relevant* means that an iUnit provides useful factual information to the user;
- *Atomic* means that an iUnit cannot be broken down into multiple iUnits without loss of the original semantics; and
- *Dependent* means that an iUnit can depend on other iUnits to be relevant.

Please refer to the 1CLICK-2 overview paper for the details of the definition [6]. Although iUnits can depend on other iUnits to be relevant according to our definition, we excluded depending iUnits in this round for simplicity.

As this work requires careful assessment lasting for a long time and consideration on the three requirements of iUnits, we decided not to use crowd-sourcing mainly due to low controllability and high education cost. We hired assessors for extracting iUnits by hand and kept the quality of extracted iUnits by giving timely feedback on their results. Assessors were asked to extract as many iUnits as possible within an hour, by using *iUnit Extractor*[4], a Firefox plugin we developed. The screenshot of the tool for iUnit extraction is shown in Figure 2. Each assessor worked on different sets of queries.

The total number of iUnits is 2,317 (23.8 iUnits per query) for English queries and 4,169 (41.7 iUnits per query) for Japanese queries. Examples of iUnits for English queries are shown in Table 2.

3.4 Intents

We introduce the notion of *intents* to the MobileClick-2 task, which have been utilized in the NTCIR INTENT and IMine tasks [12, 20, 25]. An intent can be defined as either a specific interpretation of an ambiguous query ("Mac OS" and "car brand" for "jaguar"), or an aspect of a faceted query ("windows 8" and "windows 10" for "windows"). In this round, intents were taken into account in evaluating the importance of iUnits, and were used as candidates of links to the second layer in the iUnit summarization subtask.

In the NTCIR INTENT and IMine tasks, the organizers clustered subtopics to form intents, while we constructed intents by clustering iUnits as follows:

(1) Cluster iUnits by using a clustering interface,

(2) Give each cluster a label representing iUnits included in the cluster, and

(3) Let each label of a cluster represents an intent.

[2] http://chiebukuro.yahoo.co.jp/

[3] https://datamarket.azure.com/dataset/bing/search

[4] https://addons.mozilla.org/ja/firefox/addon/iunit-extractor/

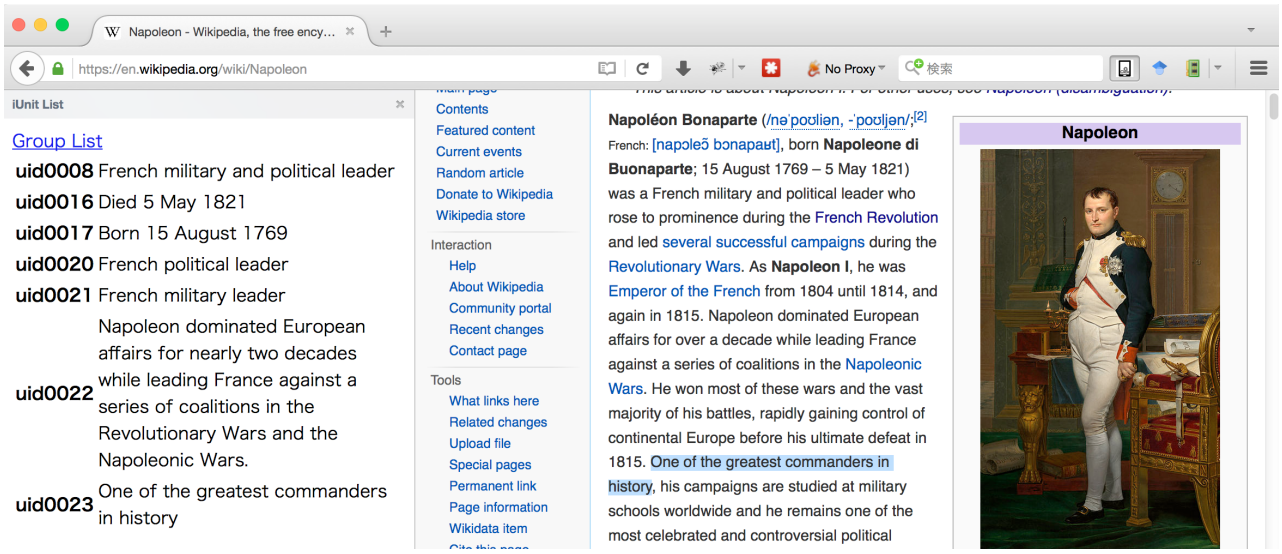


Figure 2: iUnit Extractor, a Firefox plugin for extracting iUnits from Web pages. Assessors can select a part of sentences and save its text, position, and URL by pressing Shift+Ctrl+x.

Table 2: Examples of iUnits for NTCIR-12 MobileClick English queries. Query MC2-E-0007 is "napoleon".

Query ID	iUnit ID	iUnit
MC2-E-0007	MC2-E-0007-0001	born on the island of Corsica
MC2-E-0007	MC2-E-0007-0002	defeated at the Battle of Waterloo
MC2-E-0007	MC2-E-0007-0003	established legal equality and religious toleration
MC2-E-0007	MC2-E-0007-0004	an innovator
MC2-E-0007	MC2-E-0007-0005	absent during Peninsular War
MC2-E-0007	MC2-E-0007-0006	cut off European trade with Britain
MC2-E-0007	MC2-E-0007-0007	general of the Army of Italy
MC2-E-0007	MC2-E-0007-0008	one of the most controversial political figures
MC2-E-0007	MC2-E-0007-0009	won at the Battle of Wagram
MC2-E-0007	MC2-E-0007-0010	baptised as a Catholic

We hired assessors for the manual iUnit clustering, in which two iUnits were grouped together if

(1) They are information about the same interpretation of an ambiguous query or the same aspect of a faceted query, and

(2) They are likely to be interesting for the same user.

The criteria used in the label selection are listed below:

(1) The label of a cluster should be descriptive enough for users to grasp the iUnits included in the cluster, and

(2) The label of a cluster should be often used as a query or anchor text for the included iUnits.

These clustering and labeling tasks were conducted on *Clusty*[5], a Web system for clustering. The screenshot of this system is shown in Figure 3.

As a result, we obtained 4.48 intents per query on average in the English subtasks, while we obtained 4.37 intents per query on average in the Japanese subtasks.

[5] https://github.com/mpkato/clusty

Subsequently, we let 10 crowd sourcing workers vote whether each intent is important or not. This voting was carried out to estimate the intent probability, which is the probability of intents of users who input a particular query, as was conducted in the NTCIR INTENT and IMine tasks. The assessors were asked to vote for multiple intents if they believed that they were interested in the intent when they had a chance to search by the query. We normalized the number of votes for each intent by the total number of votes for a query, and let $P(i|q)$ denote the normalized one for i, which we call intent probability of intent i of query q. More precisely, $P(i|q) = n_{i,q}/n_{\cdot,q}$ where $n_{i,q}$ is the number of votes intent i received, and $n_{\cdot,q}$ is the total number of votes for query q.

3.5 iUnit Importance

The importance of each iUnit was evaluated in terms of each intent, and *global importance* was derived from the per-intent importance and intent probability.

We asked two assessors to assess each iUnit in terms of each intent, and evaluate the importance at a five-point scale: 0 (unimportant), 1, 2 (somewhat important), 3, and 4 (highly important). The assessors were instructed to evaluate the importance by assuming that they were interested in a given intent. We defined the importance of an iUnit in terms of an intent as follows: *an iUnit is more*

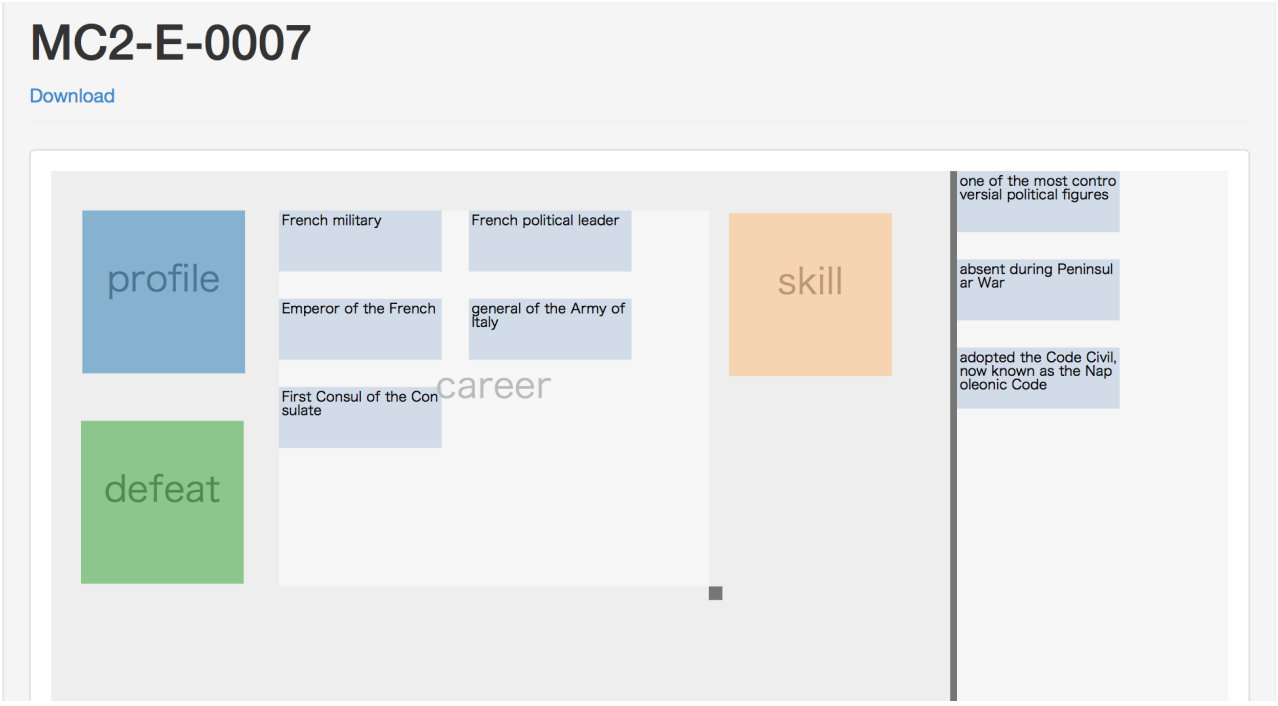


Figure 3: Clusty, a Web system for clustering. Assessors can drag iUnits (blue rectangles) shown at the right pane, and drop them at one of the clusters shown at the left pane. In the screenshot, only "career" cluster was expanded to display all the iUnits in it.

important if it is more necessary for more users who are interested in the intent. For example, given intent "Mac OS" in response to query "jaguar", iUnit "car company in UK" is *unimportant*, while it is *highly important* given intent "car brand".

We used the average of the per-intent importance scores given by multiple assessors in our evaluation. The inter-assessor agreement was moderate: 0.556 in terms of *quadratic-weighted kappa* [24].

In the iUnit ranking subtask, we used the global importance of each iUnit for evaluation. Letting $P(i|q)$ be the intent probability of query q, the global importance of iUnit u is defined as follows:

$$G(u) = \sum_{i \in I_q} P(i|q) g_i(u), \tag{1}$$

where I_q is a set of intents for query q, and $g_i(u)$ denotes the per-intent importance of iUnit u in terms of intent i.

4. EVALUATION MEASURES

This section describes evaluation methodology used in the NTCIR-12 MobileClick tasks.

4.1 iUnit Ranking Subtask

Runs submitted by participants include a ranked list of iUnit IDs for each query, which can be handled in the same way as ad-hoc retrieval runs. Therefore, we employed standard evaluation metrics for ad-hoc retrieval in this subtask.

One of the evaluation metrics used in the iUnit ranking subtask was *normalized discounted cumulative gain* (nDCG). Discounted cumulative gain (DCG) is defined as follows:

$$\mathrm{nDCG@}K = \sum_{r=1}^{K} \frac{G(u_r)}{\log_2(r+1)}, \tag{2}$$

where K is a cutoff parameter, and u_r is the r-th iUnit in a submitted ranked list. The normalized version of DCG (nDCG) is therefore defined as follows:

$$\mathrm{nDCG@}K = \frac{\mathrm{DCG@}K}{\mathrm{iDCG@}K}, \tag{3}$$

where iDCG is DCG of the ideal ranked list of iUnits, which can be constructed by sorting all the iUnits for a query by their global importance.

Another evaluation metric is Q-measure proposed by Sakai [18]:

$$Q = \frac{1}{R} \sum_{r=1}^{M} \mathrm{IsRel}(u_r) \frac{\sum_{r'=1}^{r} (\beta G(u_{r'}) + \mathrm{IsRel}(u_{r'}))}{\beta \sum_{r'=1}^{r} G(u^*_{r'}) + r}, \tag{4}$$

where $\mathrm{IsRel}(u)$ is an indicator function that returns 1 if $G(u) > 0$; otherwise 0, R is the number of iUnits with non-zero global importance (*i.e.* $\sum_u \mathrm{IsRel}(u)$), M is the length of a ranked list, u^*_r is the r-th iUnit in the ideal ranked list of iUnits, and β is a patience parameter which we set to 1 following established standards [17]. Q-measure is used for ranking submitted runs since it can take into account the quality of the whole ranking.

Q-measure is a recall-based graded-relevance metric, while nDCG is a rank-based graded-relevance metric. Thus, we expect that using both metrics will enable us to measure the performance from different perspectives. Moreover, both of them were shown to be reliable [18].

4.2 iUnit Summarization Subtask

Runs submitted to the iUnit summarization subtask consists of the first layer $\mathbf{f}$ and second layers $S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$. The first layer $\mathbf{f}$ consists of iUnits and links (*e.g.* $\mathbf{f} = (u_1, u_2, l_1, u_3)$ where u_j is an iUnit and l_j is a link). Each link l_j links to a second layer $\mathbf{s}_j$. A second layer $\mathbf{s}_j$ is composed of iUnits (*e.g.* $\mathbf{s}_1 = (u_{1,1}, u_{1,2}, u_{1,3})$).

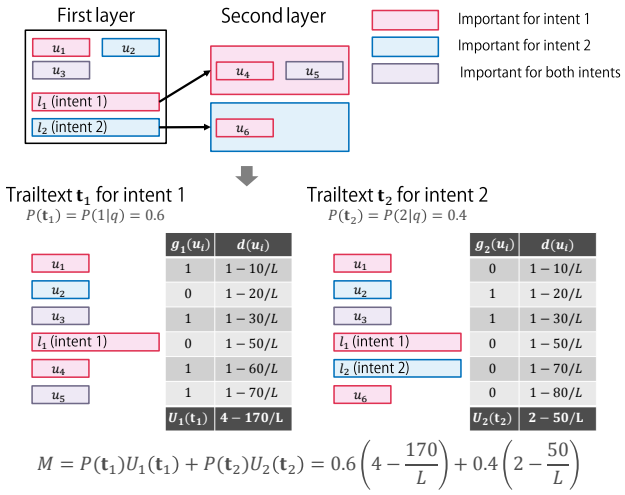


Figure 4: Illustration of M-measure computation. Two trailtexts are generated from a two-layered summary. U-measure of each trailtext is computed and summed with the probability of trailtexts (= intent probability). The length of iUnits is 10, while that of links is 20 in this example.

The principles of the iUnit summarization evaluation metric are summarized as follows:

(1) The evaluation metric is the expected utility of users who probabilistically read a summary.

(2) Users are interested in one of the intents by following the intent probability $P(i|q)$.

(3) Users read a summary following the rules below:

  (a) They read the summary from the beginning of the first layer in order and stop after reading L characters except symbols and white spaces.

  (b) When they reach the end of a link l_i, they click on the link and start to read its second layer if they are interested in the intent of l_i.

  (c) When they reach the end of a second layer s_j, they continue to read the first layer from the end of the link l_j.

(4) The utility is measured by U-measure proposed by Sakai and Dou [19], which consists of a position-based gain and a position-based decay function.

We then generate the user tails (or *trailtext*) according to the user model explained above, compute a U-measure score for each trailtext, and finally estimate the expected U-measure by combining all the U-measure scores of different trailtexts. *M-measure*, an iUnit summarization evaluation metric, is defined as follows:

$$M = \sum_{\mathbf{t} \in T} P(\mathbf{t}) U(\mathbf{t}), \tag{5}$$

where T is a set of all possible trailtexts, $P(\mathbf{t})$ is a probability of going through a trail $\mathbf{t}$, and $U(\mathbf{t})$ is the U-measure score of the trail. The computation of M-measure is illustrated in Figure 4.

A trailtext is a concatenation of all the texts read by a user, and can be defined as a list of iUnits and links in our case. According to our user model, a trailtext of a user who are interested in intent i can be obtained by inserting after the link of i a list of iUnits in its second layer. More specifically, trailtext $\mathbf{t}$ of intent i is obtained as follows:

(1) Let $\mathbf{f} = (\ldots, u_{j-1}, l_k, u_j, \ldots)$ where l_k is a link of intent i.

(2) Generate $\mathbf{t} = (\ldots, u_{j-1}, l_k, u_{k,1}, \ldots, u_{k,|\mathbf{s}_k|}, u_j, \ldots)$ for second layer $\mathbf{s}_k = (u_{k,1}, \ldots, u_{k,|\mathbf{s}_k|})$.

Note that a link in the trailtext is regarded as a non-relevant iUnit for the sake of convenience. Also note that only the first appearance of the same iUnit is relevant, while the other appearances are regarded as non-relevant.

As mentioned above, we can generate a trailtext for each intent, and do not need consider the other trailtexts as the way to read a summary only depends on the intent of users. In addition, the probability of a trailtext is equivalent to that of an intent for which the trailtext is generated. Thus, M-measure can be simply re-defined as follows:

$$M = \sum_{i \in I_q} P(i|q) U_i(\mathbf{t}_i). \tag{6}$$

The U is now measured in terms of intent i in the equation above, since we assume that users going through $\mathbf{t}_i$ are interested in i.

The utility is measured by U-measure proposed by Sakai and Dou [19], and is computed by the importance and offset of iUnits in a trailtext. The offset of iUnit u in a trailtext is defined as the number of characters between the beginning of the trailtext and the end of u. More precisely, the offset of the j-th iUnit in trailtext $\mathbf{t}$ is defined as $\mathrm{pos}_{\mathbf{t}}(u) = \sum_{j'=1}^{j} \mathrm{chars}(u_{j'})$ where $\mathrm{chars}(u)$ is the number of characters of iUnit u except symbols and white spaces. Recall that a link in the trailtext contributes to the offset as a non-relevant iUnit. According to Sakai and Dou's work [19], U-measure is defined as follows:

$$U_i(\mathbf{t}) = \frac{1}{\mathcal{N}} \sum_{j=1}^{|\mathbf{t}|} g_i(u_j) d(u_j), \tag{7}$$

where d is a position-based decay function, and $\mathcal{N}$ is a normalization factor (which we simply set to 1). The position-based decay function is defined as follows:

$$d(u) = \max\left(0, 1 - \frac{\mathrm{pos}_{\mathbf{t}}(u)}{L}\right), \tag{8}$$

where L is a patience parameter of users. Note that no gain can be obtained after L characters read, *i.e.* $d(u) = 0$. This is consistent with our user model in which users stop after reading L characters. In MobileClick-2, L is set to twice as many as X: 840 for English and 560 for Japanese, since $L = 500$ (or 250) for Japanese was recommended by a study on S-measure [21].

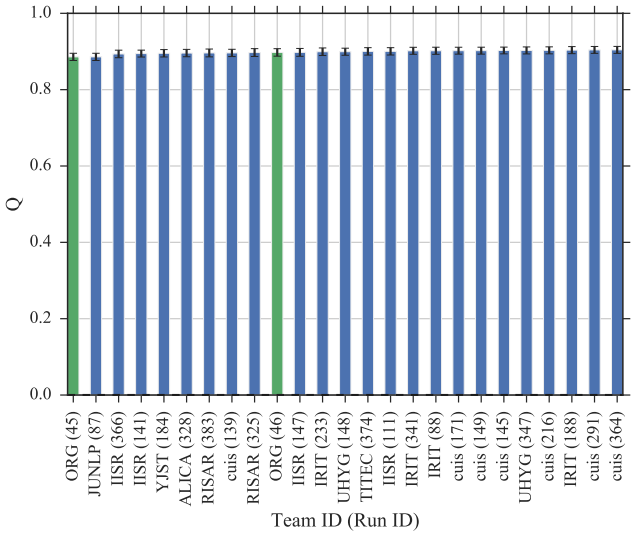
5. RESULTS

We report evaluation results for the iUnit ranking and summarization subtasks in this section.

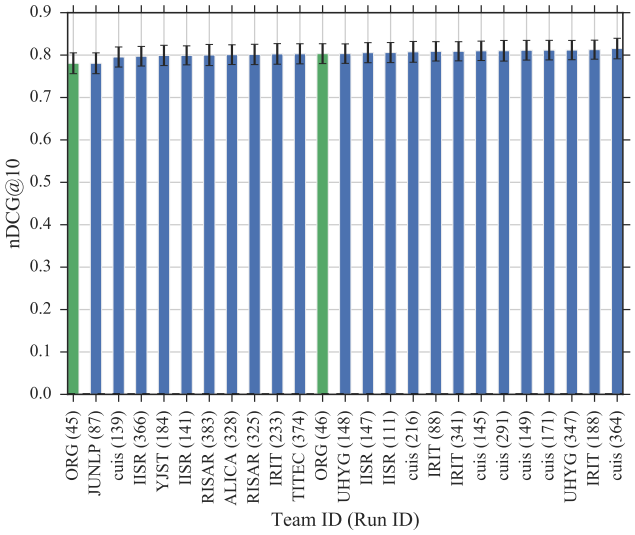
5.1 Results for iUnit Ranking Subtask

Figures 5 and 6 show evaluation results for English and Japanese iUnit ranking in terms of Q-measure and nDCG@10. The error bars indicate confidence intervals at $\alpha = 0.05$. The green bars represent results of baseline methods provided by the organizers. Tables 3 and 4 show iUnit ranking run ID pairs for which significant differences were found by randomized two-sided Tukey's HSD test at $\alpha = 0.05$.

In the English iUnit ranking subtask (Figure 5), **cuis**, **IRIT**, and **UHYG** performed well, though differences between the baseline



(a) Q-measure



(b) nDCG@10

Figure 5: Evaluation results for English iUnit ranking.

method (**ORG** (46)) and any runs were *not* statistically significant in terms of either Q-measure or nDCG@10. Overall, submitted runs demonstrated similar performance in the English iUnit ranking subtask.

In the Japanese iUnit ranking subtask (Figure 6), **UHYG**, **YJST**, and **rsrch** significantly outperformed the baseline method (**ORG** (48)). Among the top performers, there is a statistically significant difference between the best runs of **UHYG** and **rsrch**, while no statistically significant difference was found between the best runs of **UHYG** and **YJST**.

Figure 7 shows per-query evaluation results for English and Japanese iUnit ranking. Each line represents the maximum, mean, and minimum of Q-measure or nDCG@10 for a particular query. In both of the English and Japanese iUnit ranking subtasks, MC2-*-0001–0020 are CELEBRITY, MC2-*-0021–0040 are LOCAL, MC2-*-0041–0080 are DEFINITION, and MC2-*-0081–0100 are QA queries. It can be observed that (1) the difference between the maximum and minimum is large for the DEFINITION type of queries in the English iUnit ranking subtask, (2) the difference between the maximum and minimum is large for the LOCAL and QA types of queries in the Japanese iUnit ranking subtask, and (3) the difference between the maximum and minimum is relatively small for the CELEBRITY type of queries in both of the languages. Note that the results for MC2-J-0083 were anomalistically zero since no iUnits were provided due to an error in the document crawling.

Below, we briefly introduce the baseline methods and methods proposed by the participants.

Baselines (ORG). There are two types of baseline methods provided by the organizers: **ORG** (45) and **ORG** (48) are methods of outputting iUnits in random order, while **ORG** (46) and **ORG**

Table 3: Run ID pairs for which significant differences were found by randomized two-sided Tukey's HSD test at $\alpha = 0.05$ (English iUnit ranking).

(a) Q-measure

Run ID	Run IDs
45	88, 145, 149, 171, 188, 216, 291, 341, 347, 364
87	88, 145, 149, 171, 188, 216, 291, 341, 347, 364

(b) nDCG@10

Run ID	Run IDs
45	188, 364
87	188, 364

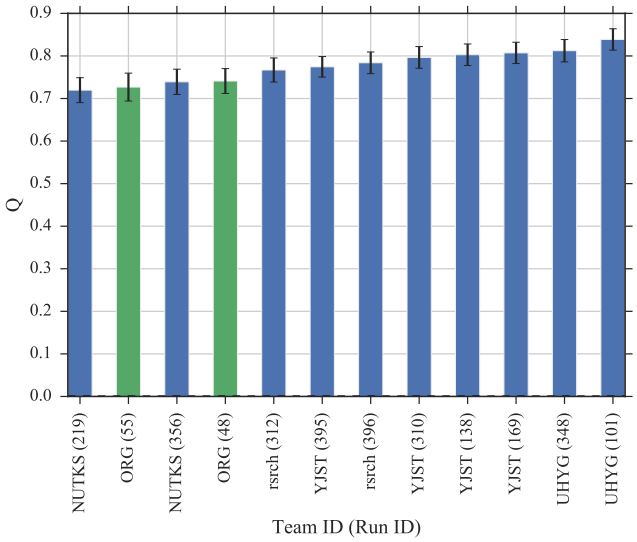
Table 4: Run ID pairs for which significant differences were found by randomized two-sided Tukey's HSD test at $\alpha = 0.05$ (Japanese iUnit ranking).

(a) Q-measure

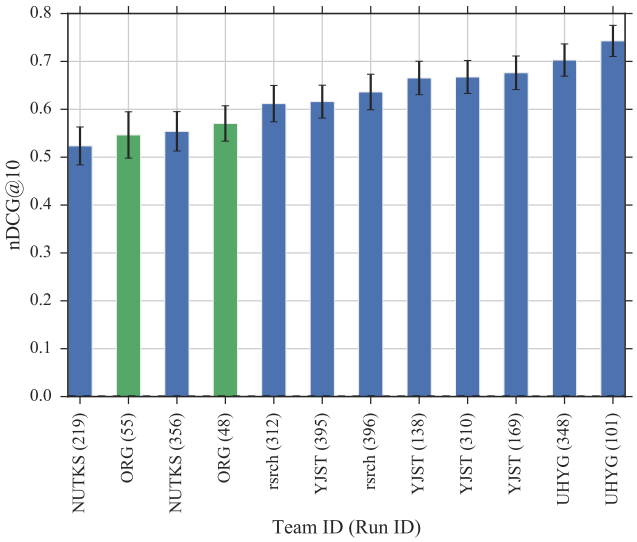
Run ID	Run IDs
48	101, 138, 169, 310, 348, 396
55	101, 138, 169, 310, 312, 348, 395, 396
101	219, 310, 312, 356, 395, 396
138	219, 356
169	219, 312, 356
219	310, 312, 348, 395, 396
310	356
312	348
348	356, 395
356	396

(b) nDCG@10

Run ID	Run IDs
48	101, 138, 169, 310, 348
55	101, 138, 169, 310, 348, 395, 396
101	138, 219, 310, 312, 356, 395, 396
138	219, 356
169	219, 356
219	310, 312, 348, 395, 396
310	356
312	348
348	356, 395
356	396

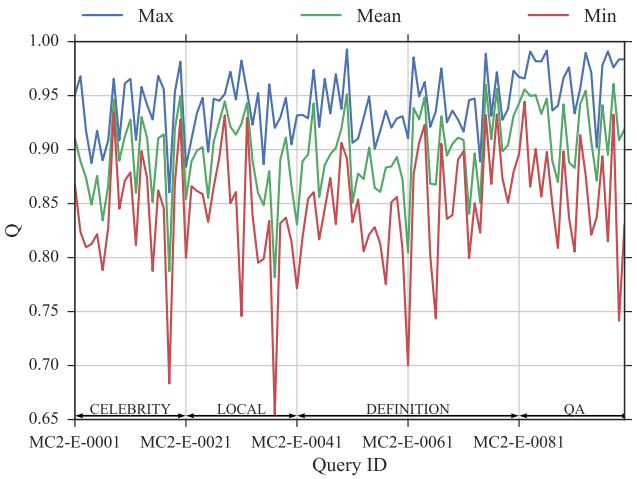


(a) Q-measure

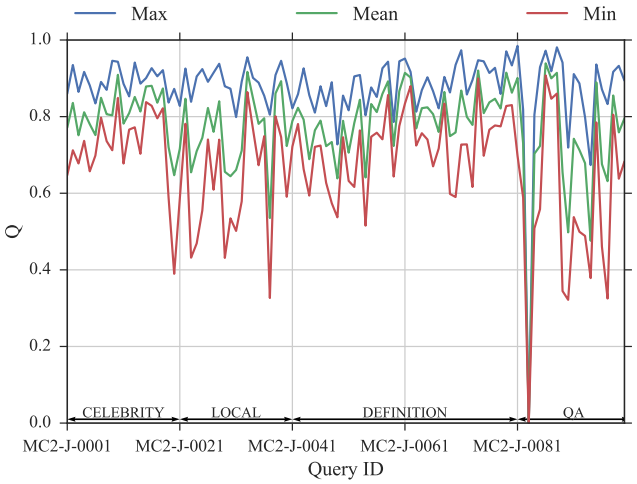


(b) nDCG@10

Figure 6: Evaluation results for Japanese iUnit ranking.



(a) English



(b) Japanese

Figure 7: Per-query evaluation results for iUnit ranking.

(55) are language-model-based methods that rank iUnits based on an odds ratio defined as follows:

$$\mathrm{OR}(u) = \sum_{w \in u} \frac{P_q(w)}{P_o(w)}, \tag{9}$$

where u is an iUnit for a query q, $P_q(w)$ is the probability of a word w in a document set retrieved by q, and $P_o(w)$ is the probability of a word w in document sets retrieved by queries other than q. $P_q(w)$ is maximum likelihood estimation of the word probability, *i.e.* $P_q(w) = n_{D_q,w}/n_{D_q,\cdot}$ where $n_{D,w}$ is the frequency of a word w in a document set D, and $n_{D,\cdot}$ is the number of words in D. $P_o(w)$ is defined as follows: $P_o(w) = n_{D_o,w}/n_{D_o,\cdot}$ where D_o is a set of documents retrieved by queries other than q.

IISR [4]. **IISR** improved the baseline methods by filtering out infrequent words, using the rank of webpages from which iUnits were extracted, etc. In addition, **IISR** utilized methods based on machine learning in which word vectors obtained by word2vec [14] and odd ratio in the baseline method were used as features.

UHYG [5]. **UHYG** constructed a iUnit-webpage bipartite graph, and applied link analysis algorithms including HITS and PageRank for estimating the iUnit importance.

TITEC [8]. **TITEC** utilized element-based retrieval for finding relevant elements in response to a query, and gave higher scores to iUnits more similar to the relevant elements. **TITEC** tackled the 1CLICK-2 task by a similar approach [9].

NUTKS [28]. **NUTKS** modeled search intents behind queries by Latent Dirichlet Allocation (LDA) [1], and ranked iUnits based on topics modeled by LDA.

cuis [10]. **cuis** also utilized LDA for modeling search intents, re-ranked webpages by topic distribution inferred from a query, and estimate the importance of iUnits based on re-ranked webpages

from which the iUnits were extracted.

JUNLP [3]. **JUNLP** employed SentiWordNet and SenticNet to obtain a concept-based positive, negative and neutral sense with their corresponding polarity score, and ranked iUnits based on the estimated sense.

IRIT [2]. **IRIT** ranked iUnits by the amount of information measured by Shannon's entropy, and word2vec-based similarity between a query and an iUnit [14].

RISAR [15]. **RISAR** took a learning-to-rank approach for iUnit ranking, and included various kinds of features including ones for query-biased summarization, and ones for non-factoid question answering.

YJST [16]. **YJST** extended the language-model-based baseline method by using several smoothing methods such as Dirichlet prior smoothing.

rsrch [27]. **rsrch** estimated the importance of iUnits by their neighbor iUnits in a word2vec space [14]. In addition, **rsrch** applied learning-to-rank to iUnit ranking.

ALICA [13]. **ALICA** used two tools developed in their research group: IR-n (a passage retrieval system) and COMPENDIUM (a summarization generator). Moreover, **ALICA** proposed a summarization method based on principal component analysis and applied it to the iUnit ranking problem.

5.2 Results for iUnit Summarization Subtask

Figure 8 shows evaluation results for English and Japanese iUnit summarization. The error bars indicate confidence intervals at $\alpha = 0.05$. The green bars represent results of baseline methods provided by the organizers. Table 5 shows iUnit summarization run ID pairs for which significant differences were found by randomized two-sided Tukey's HSD test at $\alpha = 0.05$.

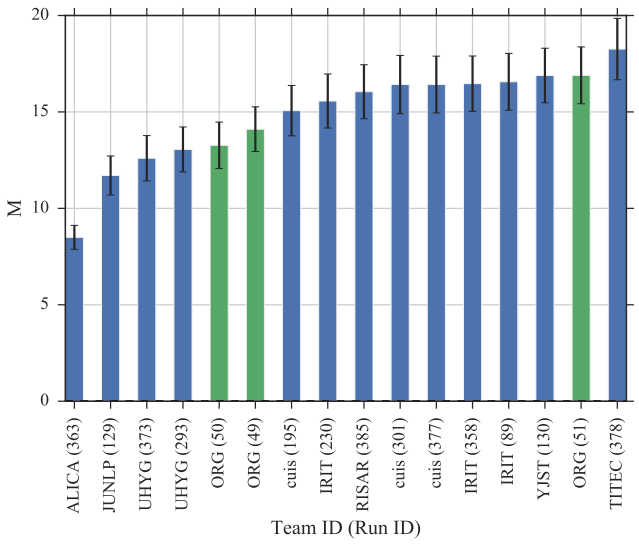
In the English iUnit summarization subtask (Figure 8(a)), **TITEC** and **YJST** are the top performers and not statistically distinguishable. **TITEC** (378) significantly outperformed **IRIT** (89) but is *not* significantly better than **ORG** (51). **TITEC**, **YJST**, **IRIT**, **cuis**, and **RISAR** were significantly better than a weak baseline method (**ORG** (49)).

In the Japanese iUnit summarization subtask (Figure 8(b)), **YJST** and **UHYG** are the top performers and not statistically distinguishable. **YJST** achieved significantly higher performance than the others but **UHYG** (232), and **UHYG** significantly outperformed the others but **YJST**. Only these two teams showed significantly better results than the baseline methods.

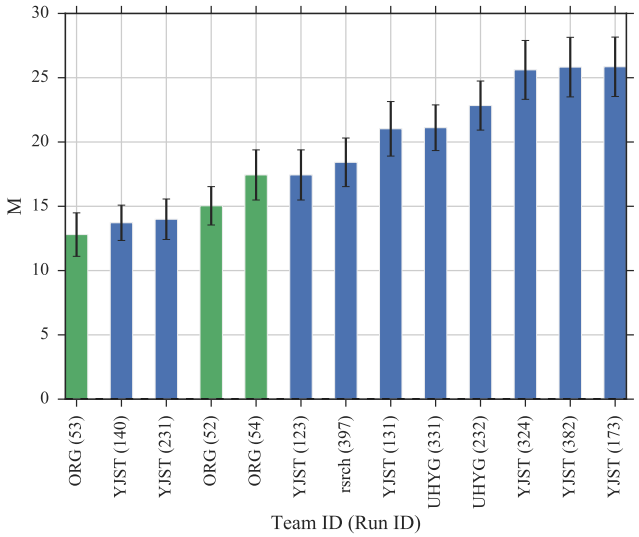
Figure 9 shows per-query evaluation results for English and Japanese iUnit summarization. Each line represents the maximum, mean, and minimum of M-measure for a particular query. In both of the subtasks, the difference between the maximum and minimum is relatively large for the CELEBRITY type of queries, while that is small for the LOCAL type of queries.

Below, we briefly introduce the baseline methods and methods proposed by the participants.

Baselines (ORG). There are three types of baseline methods provided by the organizers: (1) **ORG** (49) and **ORG** (52) are methods of outputting iUnits in random order, (2) **ORG** (50) and **ORG** (53) are language-model-based methods based on the odds ratio,



(a) English



(b) Japanese

Figure 8: Evaluation results for iUnit summarization.

and (3) **ORG** (51) and **ORG** (54) are also based the odds ratio. The difference between the baseline methods (2) and (3) is that the baseline method (2) outputs iUnits only in the first layer, while the baseline method (3) put iUnits into the second layer. All the baseline methods rank iUnits either randomly or in descending order of the odds ratio, and fill the first layer with the ranked iUnits. To rank iUnits for the second layer linked by intent i, the baseline method (3) utilizes a score function defined as follows:

$$\mathrm{Score}(u, i) = \mathrm{OR}(u)\mathrm{Sim}(u, i), \tag{10}$$

where $\mathrm{Sim}(u, i)$ is asymmetric similarity between iUnit u and intent i, *i.e.* $Sim(u, i) = |W_u \cap W_i|/|W_i|$ (W_u is a set of words in u and W_i is a set of words in i). The score becomes higher if the odds ratio is higher and similarity between an iUnit and an intent is higher. We designed this function so that important iUnits relevant to the anchor text of the second layer are included in the layer.

Table 5: Run ID pairs for which significant differences were found by randomized two-sided Tukey's HSD test at $\alpha = 0.05$ (iUnit summarization).

(a) English

Run ID	Run IDs
49	51, 89, 129, 130, 301, 358, 363, 377, 378, 385
50	51, 89, 130, 195, 230, 301, 358, 363, 377, 378, 385
51	129, 195, 293, 363, 373
89	129, 293, 363, 373, 378
129	130, 195, 230, 301, 358, 363, 377, 378, 385
130	195, 293, 363, 373
195	293, 363, 373, 378
230	293, 363, 373, 378
293	301, 358, 363, 377, 378, 385
301	363, 373, 378
358	363, 373, 378
363	373, 377, 378, 385
373	377, 378, 385
377	378
378	385

(b) Japanese

Run ID	Run IDs
52	131, 173, 232, 324, 331, 382, 397
53	54, 123, 131, 173, 232, 324, 331, 382, 397
54	131, 140, 173, 231, 232, 324, 331, 382
123	131, 140, 173, 231, 232, 324, 331, 382
131	140, 173, 231, 324, 382
140	173, 232, 324, 331, 382, 397
173	231, 331, 397
231	232, 324, 331, 382, 397
232	397
324	331, 397
331	382
382	397

UHYG [5]. **UHYG** proposed intent-sensitive PageRank on an iUnit-webpage bipartite graph, an extension of topic-sensitive PageRank, for the iUnit summarization subtask.

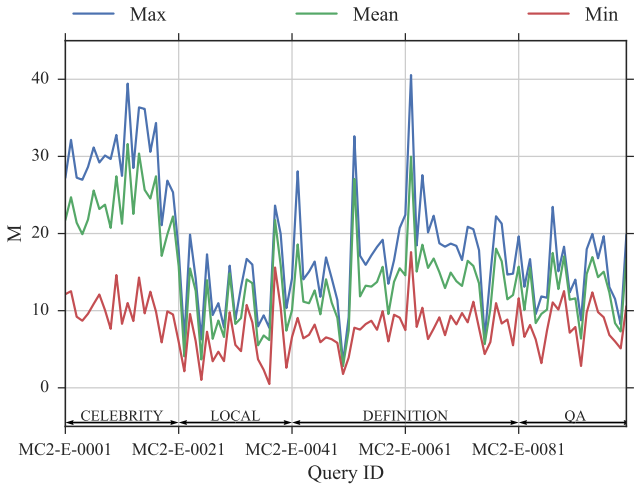
TITEC [8]. **TITEC** used a baseline method employed in MobileClick-1 [7] for iUnit summarization.

cuis [10]. **cuis** first estimated a relevant intent for each document, and then decided in which layers an iUnit should be presented by relevant intents of documents containing the iUnit.

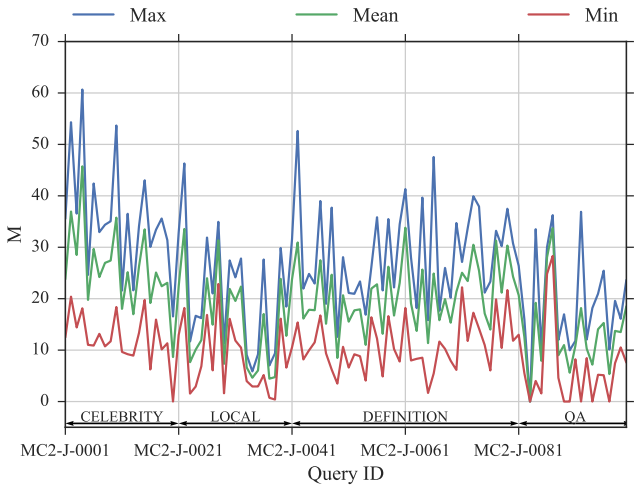
JUNLP [3]. **JUNLP** took graph-based approaches for iUnit summarization, which includes TextRank and a method based on WUP similarity [26].

IRIT [2]. **IRIT** examined two strategies to build a summary: a top-down approach where the first layer is filled first, and a bottom-up approach in which the second layer is filled first.

RISAR [15]. **RISAR** used the baseline methods with iUnits ranked by their own method.



(a) English



(b) Japanese

Figure 9: Per-query evaluation results for iUnit summarization.

YJST [16]. **YJST** utilized word2vec for measuring the similarity between iUnits and intents, and improved the similarity function in the baseline method.

rsrch [27]. **rsrch** took into account the length of iUnits in measuring scores, and used the standard deviation of similarity scores between an iUnit and intents as a criterion for deciding an appropriate layer for the iUnit.

ALICA [13]. **ALICA** used IR-n (a passage retrieval system) for estimating the relevance of iUnits and intents for a query.

6. CONCLUSIONS

This paper presents the overview of the MobileClick task at NTCIR-12. This task aims to develop a system that returns a concise summary of information relevant to a given query, and brings a structure into the summarization so that users can easily locate their desired information. In this paper, we mainly explained the task design, and evaluation methodology, and evaluation results.

Findings from the evaluation results are summarized below:

- In the English iUnit ranking subtask, most of the submitted runs demonstrated similar performance and are not statistically distinguishable.
- In the Japanese iUnit ranking subtask, **UHYG** and **YJST** are the top performers that significantly outperformed baseline methods.
- In the English iUnit summarization subtask, **TITEC** and **YJST** are the top performers but could not achieved significantly better results than a strong baseline method.
- In the Japanese iUnit summarization subtask, **YJST** and **UHYG** showed the best performance and significantly outperformed baseline methods.

7. ACKNOWLEDGMENTS

We thank the NTCIR-12 MobileClick participants for their effort in submitting runs. We appreciate significant efforts made by Yahoo Japan Corporation for providing quite valuable search query data. We also thank Dr. Young-In Song from Wider Planet for providing useful data.

8. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] A. Chellal and M. Boughanem. IRIT at the NTCIR-12 MobileClick-2 Task. In *NTCIR-12 Conference*, 2016.

[3] M. Dey, A. Mondal, and D. Das. NTCIR-12 MOBILECLICK: Sense-based Ranking and Summarization of English Queries. In *NTCIR-12 Conference*, 2016.

[4] W.-B. Han, H.-H. Wang, and T.-H. Tsai. NCU IISR System for NTCIR-12 MobileClick2. In *NTCIR-12 Conference*, 2016.

[5] S. Iizuka, T. Yumoto, M. Nii, and N. Kamiura. UHYG at the NTCIR-12 MobileClick Task: Link-based Ranking on iUnit-Page Bipartite Graph. In *NTCIR-12 Conference*, 2016.

[6] M. P. Kato, M. Ekstrand-Abueg, V. Pavlu, T. Sakai, T. Yamamoto, and M. Iwata. Overview of the NTCIR-10 1CLICK-2 Task. In *NTCIR-10 Conference*, pages 243–249, 2013.

[7] M. P. Kato, M. Ekstrand-Abueg, V. Pavlu, T. Sakai, T. Yamamoto, and M. Iwata. Overview of the NTCIR-11 MobileClick Task. In *NTCIR-11 Conference*, 2014.

[8] A. Keyaki, J. Miyazaki, and K. Hatano. XML Element Retrieval@MobileClick-2. In *NTCIR-12 Conference*, 2016.

[9] A. Keyaki, J. Miyazaki, K. Hatano, G. Yamamoto, T. Taketomi, and H. Kato. XML element retrieval @ 1CLICK-2. In *NTCIR-11 Conference*, 2013.

[10] K. P. Lai, W. Lam, and L. Bing. CUIS at the NTCIR-12 MobileClick2 Task. In *NTCIR-12 Conference*, 2016.

[11] J. Li, S. Huffman, and A. Tokuda. Good abandonment in mobile and pc internet search. In *SIGIR 2009*, pages 43–50, 2009.

[12] Y. Liu, R. Song, M. Zhang, Z. Dou, T. Yamamoto, M. Kato, H. Ohshima, and K. Zhou. Overview of the NTCIR-11 IMine task. In *NTCIR-11 Conference*, pages 8–23, 2014.

[13] F. Llopis Pascual, E. Lloret, and J. M. GÃşmez. University of Alicante at the NTCIR-12: Mobile Click. In *NTCIR-12 Conference*, 2016.

[14] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS 2013*, pages 3111–3119, 2013.

[15] K. Ong, R.-C. Chen, and F. Scholer. RMIT at the NTCIR-12 MobileClick-2: iUnit Ranking and Summarization Subtasks. In *NTCIR-12 Conference*, 2016.

[16] Y. Ozawa, T. Yatsuka, and S. Fujita. YJST at the NTCIR-12 MobileClick-2 Task. In *NTCIR-12 Conference*, 2016.

[17] T. Sakai. On penalising late arrival of relevant documents in information retrieval evaluation with graded relevance. In *Workshop on Evaluating Information Access (EVIA 2007)*, pages 32–43, 2007.

[18] T. Sakai. On the reliability of information retrieval metrics based on graded relevance. *Information processing & management*, 43(2):531–548, 2007.

[19] T. Sakai and Z. Dou. Summaries, ranked retrieval and sessions: a unified framework for information access evaluation. In *SIGIR 2013*, pages 473–482, 2013.

[20] T. Sakai, Z. Dou, T. Yamamoto, Y. Liu, M. Zhang, R. Song, M. Kato, and M. Iwata. Overview of the NTCIR-10 INTENT-2 task. In *NTCIR-10 Conference*, pages 94–123, 2013.

[21] T. Sakai and M. P. Kato. One click one revisited: Enhancing evaluation based on information units. In *AIRS 2012*, pages 39–51, 2012.

[22] T. Sakai, M. P. Kato, and Y.-I. Song. Click the search button and be happy: Evaluating direct and immediate information access. In *CIKM 2011*, pages 621–630, 2011.

[23] T. Sakai, M. P. Kato, and Y.-I. Song. Overview of NTCIR-9 1CLICK. In *NTCIR-9*, pages 180–201, 2011.

[24] J. Sim and C. C. Wright. The kappa statistic in reliability studies: use, interpretation, and sample size requirements. *Physical therapy*, 85(3):257–268, 2005.

[25] R. Song, M. Zhang, T. Sakai, M. P. Kato, Y. Liu, M. Sugimoto, Q. Wang, and N. Orii. Overview of the NTCIR-9 INTENT task. *NTCIR-9*, pages 82–105, 2011.

[26] Z. Wu and M. Palmer. Verbs semantics and lexical selection. In *ACL 1994*, pages 133–138, 1994.

[27] S. Yokoyama, S. Nakamura, R. Kitajima, and Y. Hirate. Ranking and Summarization using word-embedding at NTCIR-12 MobileClick Task. In *NTCIR-12 Conference*, 2016.

[28] T. Yoshioka and T. Yukawa. NUTKS at NTCIR-12 MobileClick2: iUnit Ranking Subtask Using Topic Model. In *NTCIR-12 Conference*, 2016.