QAque: Faceted Query Expansion Techniques for Exploratory Search using Community QA Resources

Atsushi Otsuka
Graduate School of Library, Information and Media Studies
University of Tsukuba
1-2 Kasuga, Tsukuba, Ibaraki, Japan
aotsuka@slis.tsukuba.ac.jp

Yohei Seki
Faculty of Library, Information and Media Science
University of Tsukuba
1-2 Kasuga, Tsukuba, Ibaraki, Japan
yohei@slis.tsukuba.ac.jp

Noriko Kando
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan
kando@nii.ac.jp

Tetsuji Satoh
Faculty of Library, Information and Media Science
University of Tsukuba
1-2 Kasuga, Tsukuba, Ibaraki, Japan
satoh@ce.slis.tsukuba.ac.jp

ABSTRACT

Recently, query suggestions have become quite useful in web searches. Most provide additional and correct terms based on the initial query entered by users. However, query suggestions often recommend queries that differ from the user's search intentions due to different contexts. In such cases, faceted query expansions and their usages are quite efficient. In this paper, we propose faceted query expansion methods using the resources of Community Question Answering (CQA), which is social network service (SNS) that shares user knowledge. In a CQA site, users can post questions in a suitable category. Others answer them based on the category framework. Thus, the CQA "category" makes a "facet" of the query expansion. In addition, the time of year when the question was posted plays an important role in understanding its context. Thus, such seasonality creates another "facet" of the query expansion. We implement two-dimensional faceted query expansion methods based on the results of the Latent Dirichlet Allocation (LDA) analysis of CQA resources. The question articles deriving query expansion are provided for choosing appropriate terms by users. Our sophisticated evaluations using actual and long-term CQA resources, such as "Yahoo! CHIEBUKURO" demonstrate that most parts of the CQA questions are posted in periodicity and in bursts.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Query formulation, Search process*

Keywords

query expansion, faceted search, Community QA
Latent Dirichlet Allocation, category, seasonality

WWW 2012 Companion, April 16-20, 2012, Lyon, France
ACM 978-1-4503-1230-1/12/04.

1. INTRODUCTION

The World-Wide-Web (Web) is one of the most valuable tools for collecting information, especially with search engines. Since web search engines require queries of keywords, users have to input keywords extracted from their information needs. Web search engines provide query suggestions, which are helpful tools for web searches. Query suggestions propose query candidates based on the initial queries. Users match their information needs and query candidates to obtain various web pages using query suggestions without a keyboard. However, query suggestions often recommend queries that differ from the user's search intentions. For example, the keyword "virue" is ambiguous. When initial query "virus" refers to "computer viruses", search engines suggest "antiviral software", "influenza virus", "RSvirus", and "MACvirus"[1]. "Antiviral software" and "MACvirus" denote computer viruses, but "influenza virus" and "RSvirus" refer to diseases. "RSvirus" and "macvirus" are especially rare words. Those queries are arranged without classification. Thus, users might select words about diseases even though they are interested in computers.

To solve this problem, user intent-based information retrieval has gained attention. A "faceted search" confirms the user intention by providing various aspects of search results that must be diverse. In this paper, we propose a diversified faceted navigation system using a query expansion approach. Our system provides diversified queries based on user intentions. A facet expresses intentions based on a topic and a time. Our system matches user intentions and queries by browsing topics and time. The most important point of our study is that our expanded queries contain information needs written in natural language, which is easy for users to understand. Our expanded queries provide the expansion reason by taking examples of information needs satisfied by expansion queries. Users select an expansion query

[1]Actually, those words were suggested by a Japanese web search engine.

that adequately matches their intentions from candidates by confirming the information needs behind the queries.

We used the Community Question Answering corpus to create queries that contain reasons. Community Question Answering (CQA) is a social website intended for knowledge sharing. CQA users post questions in natural language. We believe that question articles can replace the information needs of web search users, because, questions can also be answered by web searches [12] in many cases. Question articles verbalize latent information needs. We assume that question articles have information that can satisfy user needs. Web search users extract keywords based on their information needs. We simulate this process by extracting keywords from question articles. On the other hand, CQA is categorized by subject, and each article has a posted date-time. We employ this information for faceted searches. Switching the category or the date-time changes the queries to other candidates. We split the CQA dataset into category and season. Expanded queries are created from each bit of partitioned data using Latent Dirichlet Allocation (LDA).

This remainder of our paper is organized as follows. Section 2 reviews related works. Section 3 discusses our query suggestion system and queries with question articles. Section 4 proposes an implementation of our method.

In Section 5, we report our experimental results and provide discussions, and we conclude our paper in Section 6.

2. RELATED WORK

Query expansion has been widely used in information retrieval. Traditionally, queries are expanded from top-k ranked search results as typified by pseudo relevance feedback (PRF). But, due to the growth of the web, such external resources as query logs and snippets are often used for query expansion [15]. Xu et al. [14] used Wikipedia for PRF feedback data and demonstrated that this method was effective in ambiguous queries. Paul et al. [3] evaluated a query's diversity by click entropy and query reformulations. Lin et al. [8] proposed a query expansion method based on social annotation and used tag data in social bookmark sites for query expansion. Zha et al. [17] proposed visual query suggestion, that used social photo sharing data; it suggested queries with images to understand the queries.

In particular, context aware query suggestions have gained attention. Cao et al. [2] proposed a context aware query suggestion method by constructing suffix trees from clickthrough logs and session data. Semgstock et al. [11] developed a context aware query suggestion system from six million query logs. In their system, users can configure the influence of the domain and the hour using a slider interface. Queries change depending on the domain and the hour. Guo et al. [5] proposed query expansion with social annotation data. They created relational graphs of click logs and social tag data, clustered queries based on graphs, and provided an annotation-based query recommendation system. Annotation labels was created using a natural language processing approach by Reisinger et al. [10], who extracted label data using the IsA relation, clustered query log and label data, and used a Probabilistic Context-Free Gramma (PCFG) model to bridge the query and the label.

In a faceted search, the search results are altered by various topics. Yoon et al. [16] proposed search result categorization using CQA categories and assumed that user intentions could be replaced by CQA categories and the nouns

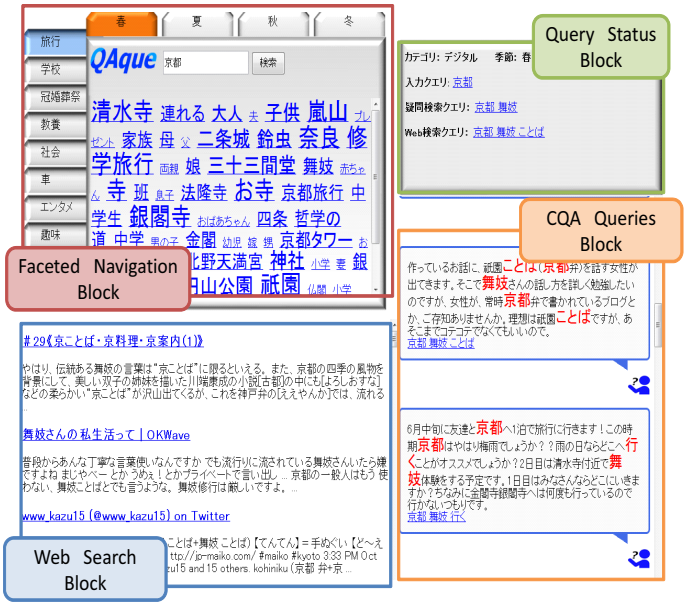

Figure 1: Our Faceted Navigation System by Query Suggestions using CQA

of question articles. Hearst [7] presented an interface design for a layered, faceted navigation system and provided breadcrumb navigation in faceted searches.

In this work, we develop a diversified faceted navigation system based on query suggestions, only using the CQA corpus. Our work is different from the above works because both query annotation and facets are created from CQA resources.

3. FACETED NAVIGATION SYSTEM USING CQA

A screenshot of our faceted navigation system is shown in Fig. 1, and Fig. 2 shows a block diagram. Our system consists of four blocks. The Faceted Navigation Block generates tag-clouds for expansion queries. The Query Status Block shows the queries being used. In our system, three types of queries are generated in navigation. The initial query is the first one entered by the user. The question retrieval query, which is generated when users select a related word in the Faceted Navigation Block, is used for question article retrieval. CQA queries are expansion queries with question articles generated in the CQA Queries Block, which generates expansion queries from question articles and shows them. The Web Search Block shows the web search results. Those queries with "initial query," , "question retrieval query," and "CQA query" contain a different number of keywords. Users have to select queries for web searches depending on the concreteness of web search.

In this section, we introduce our faceted navigation system using CQA. First, we describe the tab and the tag-cloud interface in the Faceted Navigation Block. Next, we explain CQA queries.

3.1 Interface for faceted navigation

Faceted search provides various standpoints to users. In this paper, we propose two facets: categories and seasons.

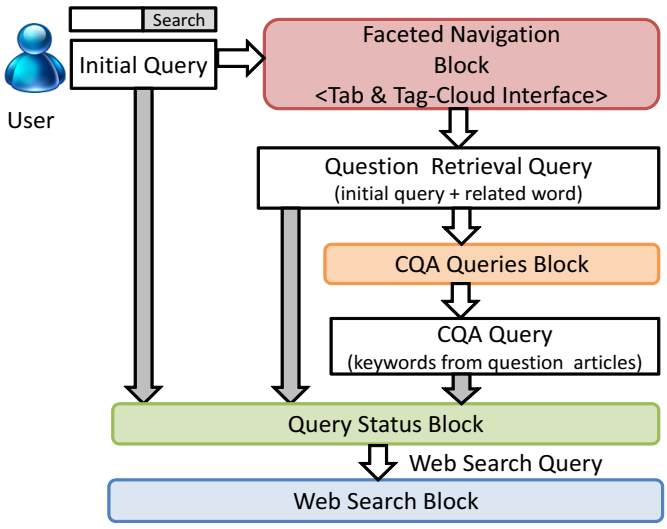

Figure 2: Interface Block Diagram of Faceted Navigation System

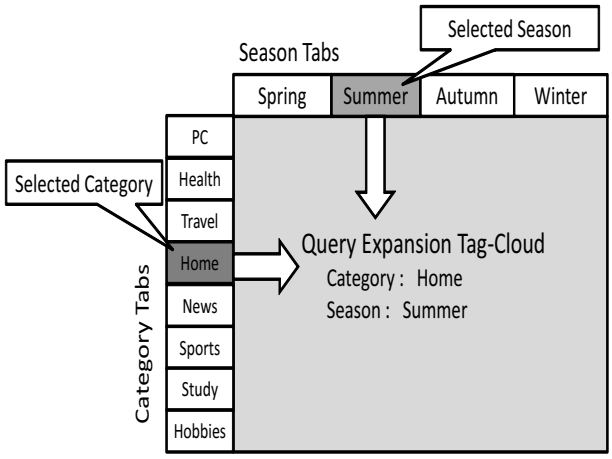

Figure 3: User Interface Block Diagram of Faceted Navigation System

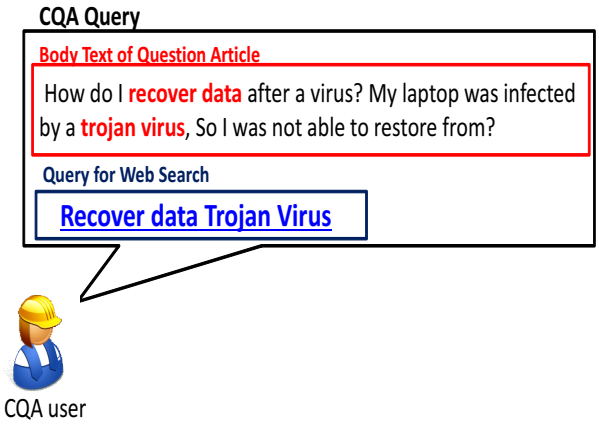

Figure 4: Ideal model of CQA query

Categories are classified by a set of keywords from a single topic. Even if the keywords are the same, their meanings might be different, depending on the topic. For example, the meaning of keyword "virus" changes depending on whether the topic is health or computers. Various information needs are hidden behind the query. Even if the query is "Kyoto" the expected queries will differ among users who seek visitor information or history of the city. Hence a faceted search has to differentiate among various topics relevant to the query. The second facet is the seasons of the year. They influence the information needs and the meaning of words. Suggesting queries about summer tourist spots in winter is absurd. Since viruses also differ by seasons, seasons are effective facets for diversity.

Categories and seasons have high affinity with CQA. In CQA, since users have to select a category when they post a question article, question articles are already categorized in CQA. We use these categories as facets and focus on CQA periodicity. In this paper, we assume that question articles consist of various information needs from many users. However, many questions have with similar meanings in CQA, explaining why CQA users emphasize by writing question articles than retrieving existing question articles using queries. Such questions have similar contents, but the posted dates differ by year or month. We presume CQA periodicity, and could make the same claim about web searches. We match web search and CQA periodicity by simultaneously, providing the questions and queries to match the user intentions. The most common cycle is year. Many are held on an annual basis, and the lifestyle is also based on a year. It is appropriate to suggest context changes by year. We employ common year periodicity and separate annual quarters into "seasons".

In this paper, we developed a faceted navigation interface, using categories and season as facets. Faceted Navigation Block interface is represented schematically in Fig. 3. Our system can change the categories and the seasons by switching tabs. The tag-cloud contents are changed by category and season. Tag-clouds, which show related words for initial queries, are an effective tool for showing many words. Generally, the sequence of words in tag-clouds is alphabetical [6]. But our tag-clouds arrange words in order of the degree of the association in the latent topics model. The font size of the word in the tag-cloud depends on the number of question articles contained in an initial query and a related word. Question retrieval queries are generated by appending selected related words to the initial query.

3.2 CQA Queries

CQA queries are generated from question articles, which are retrieved by question retrieval queries. Fig. 4 illustrates them. CQA queries consist of the body of question articles and the queries for web searches. The queries are extracted from the body of a question article. Note that CQA queries provide not only keywords but also the body of the question article. Since private questions from individual experiences are sometimes included in question articles, it is occasionally difficult to understand the meaning of queries that only consist of keywords. However, the body of question articles is written in natural language, which is easy to understand. Users can easily confirm their needs by reading the question articles. "A question retrieval query" only consists of

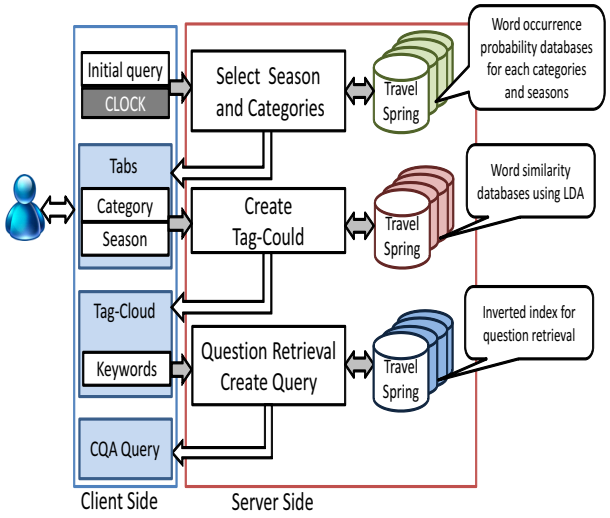

Figure 5: System Configuration Diagram

keywords. However, CQA queries retrieved by question retrieval queries show examples of needs. By confirming the examples, users can systematically learn the needs of the keyword even if its meaning is unknown.

4. METHODS

We show a diagram of our system configuration in Fig. 5. Our system consists of three processing methods and three databases. In this section, we discuss the implementation of our faceted navigation system. We first present the dataset using the CQA corpus and then describe all of the method ' s blocks in the subsections.

4.1 Data Set

Our faceted navigation system provides topical and seasonal facets using only CQA resources: a Yahoo!CHIEBUKURO (Japanese Yahoo! Answers) corpus, that contains three years (2006 to 2008) of data. First, we separated them into categories to create topical facets. Yahoo!CHIEBUKURO has more than 100 categories and three hierarchical structures. We consolidated the second-level category data and created "navigation categories" (topical facets) for our query expansion method by analyzing the corresponding contents. Although the navigation categories almost correspond to the top-level concept categories, they are more correctly divided by averaging the contents size when top-level concept categories contain multiple topics. For example, top-level category "*Hobby & Entertainment*" has many more contents than the other top-level categories. Therefore, we divide it into "*Hobbies*" and "*Entertainment*". In this way, we created twenty navigation categories compared with fourteen top-level categories.

In this paper, we also propose seasonality for query expansion techniques and assume that the question contents are seasonally shifted. We group the question data into the following three-month clusters that reflect the four seasons:

- **Spring**: question data from March to May.
- **Summer**: question data from June to August.
- **Autumn**: question data from September to November
- **Winter**: question data from December to February.

Finally, we created eighty units of data: twenty navigation categories and four seasons. As an example, Table. 1 shows the compartments of the navigation categories and the number of questions. The column label, "Second-Level Categories, " shows the category names contained in the navigation category. Column labels "Spring," "Summer," "Autumn," and "Winter" show the number of questions in each season.

4.2 Season and Category Tabs

Our system provides seasonal and topical facets on the tab interfaces. Seasonal facets are initially decided by the access time. Since the system automatically retrieves the dataset of the access time's season, users do not have to select it.

The topical facet in the Faceted Navigation block is decided depending on the initial query. Up to ten relevant categories are ordered and suggested to the user from 20 navigation categories. The relevance between the initial query and the navigation categories is expressed as a probability of document occurrence. We assume that the number of question documents $N_{C,w}$ includes word w in category C, and the total number of question documents in category C is N_C. This document occurrence is defined as:

$$P_{C,w} = \frac{N_{C,w}}{N_C} \tag{1}$$

4.3 Tag-Cloud and Question Retrieval

In our system, we employed a probabilistic language model approach proposed by Ponte and Croft [9]. Such approaches are recently receiving more attention to improve information retrieval. In particular, a query likelihood model is expected to replace the tf-idf method. Additionally, such probabilistic topic models as Probabilistic Latent Semantic Indexing (PLSI) and LDA are often used for dimension reduction. In the following subsections, we discuss methods for question retrieving and creating tag-clouds using a query likelihood model and LDA.

4.3.1 Query Likelihood Model for Question Retrieval

Our system retrieves question article data with the "Question Retrieval & Create Query" block in Fig. 5. When users have selected keyword from tag-cloud, the system recommends questions related with selected keywords and an initial query. Additionally, the system creates an expanded query by extracting keywords from the question content. Query Likelihood Model calculate the probability $P(Q|D)$ that query Q is occurred from document(question) D. Creating a maximum-likelihood query corresponds to finding an optimum combination of keywords that $P(Q|D)$ takes max probability. We retrieve questions by a query likelihood model, and create CQA queries using $P(Q|D)$.

The goal of a probabilistic model is to solve $P(D|Q)$, which is the probability that query Q expresses document D. The query likelihood model solves this problem using the following Bayes' theorem:

$$P(D|Q) = \frac{P(Q|D)P(D)}{P(Q)} \propto P(Q|D)P(D) \tag{2}$$

$P(Q)$ can be omitted because $P(Q)$ is independent of the document, and $P(D)$ entails previous knowledge. In the

Table 1: Dataset Example

Navigation Category	Second-Level Categories	Spring	Summer	Autumn	Winter
Hobby	Anime, Comics, Books, Toys, Divination, Lot, Fancywork	46,281	55,793	67,941	89,893
Entertainment	Movies, Musics, Artists, Musicals, TV/Radio, Traditional Culture	77,018	85,588	94,995	106,841
Digital Equipment	PCs, Digital Cameras, Internet, Software, Audio/Visual, Phone	92,981	91,920	96,057	110,821
Travel	Domestic Tours, Foreign Travel, Transportation Guides, Railroads, Amusement Parks	85,900	96,284	99,627	107,634
Human Relationship	Love, Life Counseling, Relationships	14,9830	16,9239	19,6009	22,0074

query likelihood model, each document is ranked in the order of the likelihood:

$$P(D|Q) \propto P(Q|D) = \prod_{w \in Q} P(w|\theta_D)^{c(w,Q)} \quad (3)$$

where w denotes a term in q. $c(w, Q)$ gives the frequency of w in query Q. θ_D denotes a unigram linguistic model, assuming that the words are independent in a document. $P(Q|D)$ can be computed with Dirichlet smoothing as follows:

$$P(w|\theta_D) = \frac{c(w,D)}{|D|+\mu} + \frac{\mu}{|C|(|D|+\mu)} \sum_{D \in C} \frac{c(w,D)}{|D|} \quad (4)$$

where μ denotes a hyperparameter with a positive value and $c(w, D)$ is the frequency of w in document D. $|D|$ indicates the number of words in document D. C is the document collection, and $|C|$ denotes the number of documents in the collection. Questions are ranked based on likelihood $P(D|Q)$.

4.3.2 Probabilistic Topic Models for Tag-Clouds

Tag-clouds show the related words for initial queries in Fig. 5. We retrieve related words without a particular dictionary like a thesaurus with a probabilistic topic model that creates latent topics by analyzing the document and word relationship. The combination of words that almost have the same probability in the same topics are related.

Blei et al. [1] proposed LDA, which is often used for dimension reduction. A document-term model consists of n documents, and m terms are expressed by a $n \times m$ matrix. LDA generates r latent topics, and document-term models are represented by $n \times r$ and $r \times m$ matrices. The general idea of LDA is that documents are expressed in mixture topic distribution, and topics are expressed by the probabilistic distribution of the terms. LDA introduces a Dirichlet prior on the multinomial distribution over the topics for the documents. The LDA generative process is as follows:

1. For all documents d sample $\theta_d \sim$ Dirichlet(α)
2. For all topics t sample $\phi \sim$ Dirichlet(β)
3. For each N_d word w_i in document d:
   (a) Sample topic $z_i \sim$ Multinomial (θ_d)
   (b) Sample topic $w_i \sim$ Multinomial (ϕ_z)

where α and β are the hyperparameters for the Dirichlet prior. To generate an LDA model, we must estimate topic collection Z. We used a collapsed Gibbs sampling method [4]. Probability $P(z_i = k|Z_{-i}, W)$ in which the nth term in document d belongs to topic $z_i = k$ can be computed as follows:

$$P(z_i = k|Z_{-i}, W) = \frac{N^d_{k-i} + \alpha}{N^d_{-i} + T\alpha} \frac{N^v_{k-i} + \beta}{N_{k-i} + W\beta} \quad (5)$$

where i means the nth term in document d. N^d_{k-i} denotes the number of assignments of topic k in document d without term i, N^d_{-i} counts the terms in document d without term i, N^v_{k-i} represents the frequency of term v in topic k without term i, and N_{k-i} denotes the number of terms in topic k without term i. T and W are the number of topics and the vocabulary.

Term-topic distribution ϕ and topic-document distribution θ are estimated from topic collection Z computed by collapsed Gibbs sampling. Probability $\hat{\phi^w_k}$, whose term t is generated from topic k, and $\hat{\theta^k_d}$, whose topic k is generated from document d, are estimated as follows:

$$\hat{\theta^k_d} = \frac{N^d_k + \alpha}{N^d + T\alpha} \quad (6)$$

$$\hat{\phi^w_k} = \frac{N^v_k + \beta}{N_k + W\beta} \quad (7)$$

Topic k is expressed as the occurrence probability of words. All terms have probability in all topics. Thus, terms have probability distribution of the topics. If the probability distribution of the topics is similar, these words are related because similar words are gathered to the same topics in LDA. We characterize the topics as a basis vector reduced dimension in the matrix and apply cosine similarity for relevance computation. The similarity of words w_1 and w_2 can be computed as follows:

$$sim(w_1, w_2) = cos(P_{w_1}, P_{w_2}) = \frac{P_{w_1} \cdot P_{w_2}}{|P_{w_1}||P_{w_2}|} \quad (8)$$

where P_{w_1} and P_{w_2} are the probability distribution of the topics. Related words are arranged in descending order of the cosine similarity.

5. EVALUATIONS

Our system provides seasonality facets to separate a dataset on a quarterly basis and assumes that the popular trends of questions shift from season to season. In this section, we evaluate seasonality in CQA by surveying different word occurrence probabilities in question contents. We first retrieve the periodicity word candidates from each category in CQA. Then, words with high occurrence probability in one season but low probability in other seasons are extracted as "seasonality words" from the candidates. We assume that CQA

has seasonality, if we can find seasonality words. In this section, we describe a method that extracts seasonality words and show our evaluation results.

5.1 Seasonality Words Extraction

We extract seasonality words that have high occurrence probability in a particular season but low probability in other seasons. In other words, the occurrence probabilities of seasonality words are highly variant. We calculate the variance, which is the distribution of the word occurrence probability in each month. We use Yahoo!CHIEBUKURO datasets for this evaluation and compute the monthly word occurrence probability in each navigation category between 2006 and 2008 and variance of the probability distribution. Formally, we compute the "Coefficient of Variation ($C.V$)" which normalizes the variance using the mean of the probability distribution:

$$C.V = \frac{\sqrt{\sigma^2}}{\overline{x}} \tag{9}$$

where $\overline{x}$ denotes the mean of word x occurrence probability, and $\sqrt{\sigma^2}$ is the standard deviation.

In our approach, we also exclude words that inconstantly burst from seasonality word candidates. Burst words shows high occurrence probability once or twice in a particular circumstances and events and have little relation with seasonality. To eliminate burst words, we employ *Vlachos*'s method [13] that, extracts burst queries from query logs using "Moving Average(MA)." Their extraction process is as follows:

1. Calculate Moving Average MA_w of length w for sequence $t = (t_1, ..., t_n)$
2. Set threshold : $cutoff = mean(MA_w) + x * std(MA_w)$
3. Burst = $\{\ t_i | MA_w(i) > cutoff\ \}$

where sequence t denotes a set of monthly word occurrence probabilities, and its length is three years($w = 36$). x is a weight parameter for standard deviation set $x = 2.5$ that was obtained by repeated experiments. w, which means the window size for MA, is set to $w = 3$ because our dataset is partitioned in three months. $mean()$ and $std()$ are the mean and standard deviation. We compare the intervals between bursts using the above method discussed. Periodicity words burst at a constant frequency. We define words that burst constantly as seasonality, and eliminate other words that burst inconstantly.

We apply this seasonality word extraction method to a dataset and evaluate three categories, (*Digital Equipment*, *Travel* and *Human Relationship*), that which contain many questions. We show the extraction results :

- Table. 2 shows the extracted words in the "Digital Equipment" category. "年賀状 (New Year's card)" means Japanese New Year's greeting cards that are usually written in November or December. "湿る (humidity)," "除 (dehumidify)," and "冷房 (air conditioning)" are relevant to the hot and humid summer days.
- *Travel* category in Fig. 3 contains such month words as "2 月 (February)" and "4 月 (April)." "雪 (snow)" is a weather word in winter, and "桜 (cherry blossoms)" bloom in spring.
- Table. 4 suggests words extracted in the *Human Relationship* category, and "チョコ (chocolate)," "お返し (return)," and "義理 (courtesy)" are relevant to *St. Valentine*'s Day, when women usually give chocolate to men, and "White Day," when men return candy to women.

Table 2: Keywords from Digital Equipment Category

Rank	Word	Month of burst
1	年賀状 (New Year's card)	November, December
2	湿る (humidity)	June, July
3	除 (dehumidify)	June, July, August
4	4 月 (April)	March, April
5	冷房 (air conditioning)	June, July, August

Table 3: Keywords from Travel Category

Rank	Word	Month of burst
1	2 月 (February)	January, February
2	GW(Golden Week holidays)	April
3	雪 (snow)	January, February
4	4 月 (April)	February, March
5	桜 (cherry blossoms)	March, April

Table 4: Keywords from Human Relationship Category

Rank	Word	Month of burst
1	チョコ (chocolate)	February
2	バレンタイン (Valentine's Day)	January, February
3	お返し (return)	February, March
4	義理 (courtesy)	February
5	正月 (New Year's Holidays)	January

We also show three chronological shifts of the occurrence probability of three words in Figs. 6. Blue dotted line is the occurrence probability of "年賀状 (New Year's card)" in the *Digital Equipment* category, red dashed line denotes "桜 (cherry blossoms)" in the *Travel* category and green solid line shows the occurrence probability of "チョコ (chocolate)" in the *Human Relationship* category. All graphs denote the same tendency; the occurrence probability in a given month is the highest, and the probabilities in other months are nearly zero. Additionally, the bursts occurred in the same month every year. This result means that the trend of the CQA questions moved periodically. In particular, words whose occurrence probability increased at the same time every year are seasonality words.

5.2 Evaluation of Seasonality

Our tag-cloud suggests the highest 100 keywords that are ordered by LDA and does not display keywords that cannot retrieve questions. Thus, tag-clouds tend to suggest more keywords for more questions. We evaluated the seasonality of our system by comparing the number of suggested

Table 5: Keywords Suggested from Tag-Clouds

Initial Query[Category]	Season	All Words	Duplication	Unique Rate	Example keywords
年賀状 (New Year's card) [Digital Equipment]	Spring	35	30	0.14	*Black, Send, Paper size*
	Summer	26	20	0.23	Laser, Paint
	Autumn	52	38	0.26	dpi, Scan, Image
	Winter	62	39	0.37	Ink Refill, *Hudegurume*, Preview
桜 (cherry blossoms) [Travel]	Spring	45	8	0.82	Blue sheets, gazing, karaoke
	Summer	13	0	1.00	*Miyazu*, *Hunajima*
	Autumn	21	3	0.86	Leaf peeping, Festa
	Winter	22	7	0.68	Plum grove, Garden
チョコ (chocolate) [Human Relationship]	Spring	57	36	0.37	Money, Pay, Louis Vuitton
	Summer	28	20	0.29	Letter, Perfume, Flower
	Autumn	27	23	0.14	Cooking, Wine, Amulet
	Winter	69	39	0.43	Main love, Zippo, Love letter

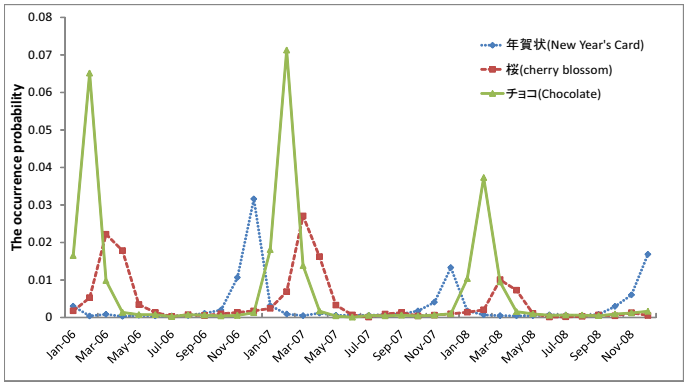

Figure 6: Shift of "年賀状 (New Year's card)", "桜 (cherry blossoms)" and "チョコ (chocolate)" Occurrence in each Categories

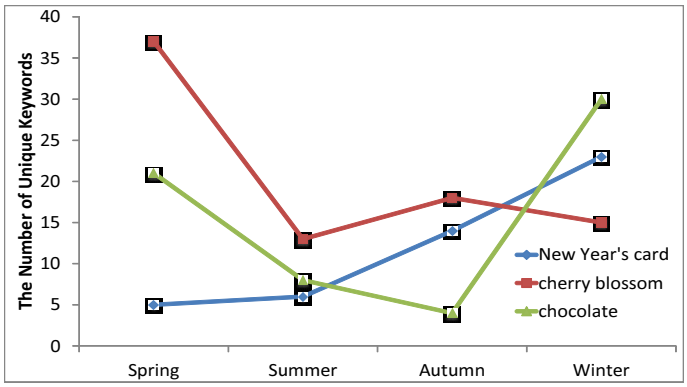

Figure 7: Unique Keywords in Each Season

keywords for each season. Table. 5 shows the number of keywords that the system suggested using three seasonality queries in Section 5.1. "All Words" shows the number of suggested keywords in the tag-clouds, and "Duplication" denotes the number of keywords duplicated with keywords suggested in other seasons. "Unique Rate" expresses the rate of the unique keywords of all the suggested keywords relative to other seasons. "Example keywords" shows keyword examples of tag-cloud. Table. 5 shows that the keywords are suggested more frequently in the season relevant to the initial queries than in other seasons. Additionally, Unique queries are also retrieved more frequently in the relevant season. The comparison results of the number of unique keywords in each season appear in Fig. 7, which reveals that the number of unique keywords disproportionately emphasize a specific season. The high number of unique keywords expresses diversity; variety of questions are posted in the season. The experimental results clearly show that our system can express seasonality by partitioning the question data into four seasons.

5.3 Outputs for CQA Queries

Finally, we show two examples of CQA queries in Table. 6. CQA queries consist of the body of the question articles and the queries for a web search. Table. 6 shows two CQA query examples as part of all CQA queries actually suggested by our system when a user input "桜 (cherry blossoms)" as a initial query and selected "花見 (gazing)" from the tag-clouds. The first example concerns a place to view cherry blossoms in the US, and the other example wants a restaurant appropriate for such cherry gazing. The contents of the two questions are different. Additionally, the question content of the first example includes such auxiliary words as "imported from Japan" and "Potomac River near Washington, DC " without keywords for the query. It is helpful for users to understand the query meanings, and they can actually use those words for subsequent searches. These examples confirmed that the question data were optimum for increasing the amount of query information accessorily.

6. CONCLUSION

In this paper, we proposed a faceted query expansion system using a CQA dataset. Our system suggests topical and seasonal keywords using CQA categories and posted times. The tab/tag-cloud interface allows diversity and context aware expansion of queries by tab switching. The popular trends of questions shift from season to season. We seasonally evaluated CQA by bursts of word occurrence. Additionally, we proposed a CQA query that contained the body of the question data with query keywords. Question data written in natural language help users understand the meaning of queries and directly express specific intent.

Table 6: Examples of CQA query

Tag-Cloud			CQA query	
Question query	Season	Category	Body of question	Keywords
桜 花見 (cherry, gazing)	Spring	Travel	アメリカ、ワシントン D.C. のポトマック川周辺に日本から輸入された桜があるそうですが、花見行ったことある人いますか? (In the US, there is cherry imported from Japan in the Potomac River near Washington, DC. Have you ever done bloom gazing?)	桜 花見 アメリカ (cherry gazing America)
			室内でお花見（桜）出来るレストラン等、ご存知の方教えてください。(Please tell me a restaurant place where we can do bloom gazing)	桜 花見 レストラン (cherry gazing restaurant)

7. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, pages 993–1022, 2003.

[2] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li. Context-aware Query Suggestion by Mining Click-through and Session Data. *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining(KDD'08)*, pages 875–883, 2010.

[3] P. Clough, M. Sanderson, M. Abouammoh, S. Navarro, and M. Paramita. Multiple Approaches to Analysing Query Diversity. *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval(SIGIR'09)*, pages 734–735, 2009.

[4] T. L. Griffiths and M. Steyvers. Finding Scientific Topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235, 2004.

[5] J. Guo, X. Cheng, G. Xu, and H. Shen. A Structured Approach to Query Recommendation with Social Annotation Data. *Proceedings of the 19th ACM international conference on Information and knowledge management(CIKM'10)*, pages 619–628, 2010.

[6] Y. Hassan-montero and V. Herrero-solana. Improving Tag-Clouds as Visual Information Retrieval Interfaces. *International Conference on Multidisciplinary Information Sciences and Technologies(InScit2006)*, pages 25–28, 2006.

[7] M. Hearst. Design Recommendations for Hierarchical Faceted Search Interfaces. *ACM SIGIR Workshop on Faceted Search*, 2006.

[8] Y. Lin, S. Jin, H. Lin, Y. Ma, and K. Xu. Social Annotation in Query Expansion a Machine Learning Approach. *Proceedings of the 34nd international ACM SIGIR conference on Research and development in information retrieval(SIGIR'11)*, pages 405–414, 2011.

[9] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval(SIGIR'98)*, pages 275–281, 1998.

[10] J. Reisinger and M. Pasca. Fine-Grained Class Label Markup of Search Queries. *Proceedings of The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011.

[11] C. Sengstock and M. Gertz. CONQUER: A System for Efficient Context-aware Query Suggestions. *Proceedings of the 20th International Conference on World Wide Web(WWW'11)*, pages 265–268, 2011.

[12] N. Takata, H. Ohshima, and K. Tanaka. Social Search Based on Mutual Complements of Web and QA Contents. *WebDB Forum 2010(Japanese)*, 2010.

[13] M. Vlachos, C. Meek, Z. Vagena, and D. Gunopulos. Identifying similarities, periodicities and bursts for online search queries. *Proceedings of the 2004 ACM SIGMOD international conference on Management of data(SIGMOD'04)*, pages 131–142, 2004.

[14] Y. Xu, G. J. Jones, and B. Wang. Query Dependent Pseudo-relevance Feedback Based on Wikipedia. *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval(SIGIR'09)*, pages 59–66, 2009.

[15] Z. Yin, M. Shokouhi, and N. Craswell. Query Expansion Using External Evidence. *Proceedings of the 31st European Conference on Information Retrieval*, pages 362–374, 2009.

[16] S. Yoon, A. Jatowt, and K. Tanaka. Intent-Based Categorization of Search Results Using Questions from Web Q&A Corpus. *Proceedings of the 10th International Conference on Web Information Systems Engineering(WISE'09)*, pages 145–158, 2009.

[17] Z.-J. Zha, L. Yang, T. Mei, M. Wang, Z. Wang, T.-S. Chua, and X.-S. Hua. Visual Query Suggestion: Towards Capturing User Intent in Internet Image Search. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, pages 13:1–13:19, 2010.