KSU Team's Multiple Choice QA System at the NTCIR-12 QA Lab-2 Task

Tasuku Kimura
Kyoto Sangyo University, Japan
i1658047@cc.kyoto-su.ac.jp

Ryosuke Nakata
Kyoto Sangyo University, Japan
g1244820@cc.kyoto-su.ac.jp

Hisashi Miyamori
Kyoto Sangyo University, Japan
miya@cc.kyoto-su.ac.jp

ABSTRACT

This paper describes the systems and results of the team KSU for QA Lab-2 task in NTCIR-12. In each phase of the task, we developed three automatic answering systems for world history questions in the National Center Test for University Admissions. In order for QA systems using document retrieval to answer questions correctly, it is important to estimate exact question types, and to utilize knowledge sources and query generation methods in accordance with these types. Therefore, we designed systems that focus on knowledge sources and query generations using the underlined texts in given exams. Scores of the formal runs were 20 correct answers(49%) and 68 points with KSU-JA-02@PH1 system in phase-1, 26 correct answers(41%) and 70 points with KSU-JA-01@PH2 system in phase-2 and 14 correct answers(39%) and 38 points with KSU-JA-01@PH3 system in phase-3.

Team Name

KSU

Subtasks

Phase-1 National Center Test (Japanese)
Phase-2 Mock Examination of National Center Test (Japanese)
Phase-3 National Center Test (Japanese)

Keywords

underlined text, adaptive query generation, knowledge source

1. INTRODUCTION

Conventional search systems return multiple documents for given queries, meaning that users have to look into them and select an appropriate answer meeting their information needs by themselves. In contrast, question-answering systems provide a single answer for given questions written in natural language, which enables users to access a correct piece of information efficiently. Thus, user can get a information efficiently and accurately.

In QA Lab task at NTCIR-11[2], the participants were required to develop automatic answering systems for world history questions in the National Center Test for University Admissions and in the second-stage exams for national and public universities in Japan. Questions were available both in Japanese and in their English translations. In the National Center Test, it is essential for the system to correctly choose answers from given multiple choices. In the second-stage exams, it is necessary for the system to summarize the given sentences and to generate sentences describing correct answers. In QA Lab-2 at NTCIR-12[1], the participants were given new opportunities to have their systems take part in some actual mock exams for university entrance examinations, in conjunction with Todai Robot Project. In QA Lab-2, the systems were evaluated in three different phases.

This paper describes the systems and results of the team KSU for QA Lab-2 task in NTCIR-12. In each phase, we developed three automatic answering systems for world history questions in the National Center Test for University Admissions. In order for QA systems using document retrieval to answer questions correctly, it is important to estimate exact question types, and to utilize knowledge sources and query generation methods in accordance with these types. Therefore, we designed systems that focus on knowledge sources and query generations using the underlined texts in given exams. Scores of the formal runs were 20 correct answers(49%) and 68 points with KSU-JA-02@PH1 system in phase-1, 26 correct answers(41%) and 70 points with KSU-JA-01@PH2 system in phase-2 and 14 correct answers(39%) and 38 points with KSU-JA-01@PH3 system in phase-3.

2. SYSTEM DESCRIPTION

This section explains the proposed systems based on query generation methods and different knowledge sources. When the system tries to answer the world history questions correctly in the National Center Test, two points are important; the generation of queries with less unnecessary terms, and the use of precise and comprehensive knowledge sources, because the answer candidates are often obtained by document retrieval. Therefore, we implemented several functions such as query generation corresponding to question types, query generation with particular kinds of words including named entities, adaptive query generation based on the underlined texts in given questions, and utilization of various knowledge sources like textbooks, Wikipedia, and ontologies describing only historical events. Also, we developed three systems with various combinations of these functions in each phase. The systems were implemented by modifying and improving the baseline system provided for QA Lab-2 by the organizers. Figure 1 ∼ Figure 3 show the configuration of the proposed systems in each phase.

2.1 Reading questions

2.1.1 Creating user dictionary specialized in world history

The text data of a given exam are morphologically analyzed and used in the successive modules. Thus, we aimed to improve the accuracy of morphological analysis by creating a user dictionary specialized in world history, which was also expected to enhance the performance of the whole system.

Total of 19,415 named entities were registered on the user dictionary; 14,622 of them were contained in the textbooks published by Tokyo Shoseki, Co., Ltd. with 32 kinds of annotated tags, and 4,793 of them were included in the event ontology EVT. Some surface strings of the newly registered words become substrings of words already registered in the original dictionary. In such cases, the cost of the word with longer character length was set higher priority. Also, some words were attached different kinds of named-entity tags; for example, the word "Ireland" has "government" and "location" tags. In this paper, such word was given the same single cost, regardless of their different tags.

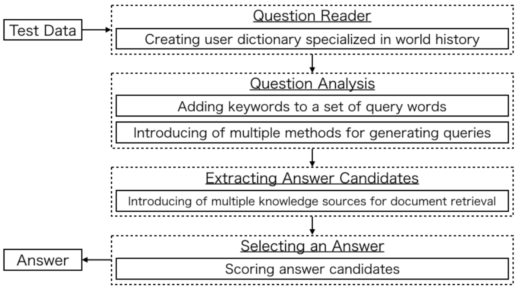

Test Data
Question Reader
Creating user dictionary specialized in world history
Question Analysis
Adding keywords to a set of query words
Introducing of multiple methods for generating queries
Extracting Answer Candidates
Introducing of multiple knowledge sources for document retrieval
Selecting an Answer
Answer
Scoring answer candidates

Figure 1: Basic system configuration in Phase-1

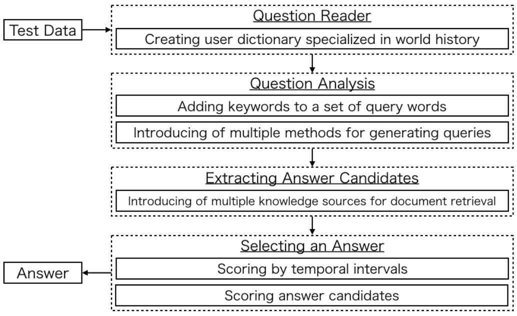

Test Data
Question Reader
Creating user dictionary specialized in world history
Question Analysis
Adding keywords to a set of query words
Introducing of multiple methods for generating queries
Extracting Answer Candidates
Introducing of multiple knowledge sources for document retrieval
Selecting an Answer
Answer
Scoring by temporal intervals
Scoring answer candidates

Figure 2: Basic system configuration in Phase-2

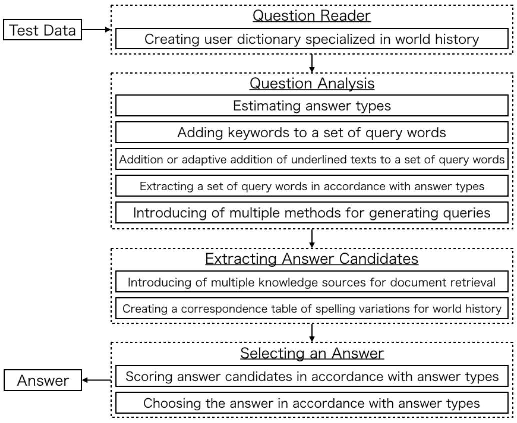

Test Data
Question Reader
Creating user dictionary specialized in world history
Question Analysis
Estimating answer types
Adding keywords to a set of query words
Addition or adaptive addition of underlined texts to a set of query words
Extracting a set of query words in accordance with answer types
Introducing of multiple methods for generating queries
Extracting Answer Candidates
Introducing of multiple knowledge sources for document retrieval
Creating a correspondence table of spelling variations for world history
Selecting an Answer
Answer
Scoring answer candidates in accordance with answer types
Choosing the answer in accordance with answer types

Figure 3: Basic system configuration in Phase-3

2.2 Estimating answer types

2.2.1 *Estimating answer types*

Answer types were estimated with particular string patterns appearing in each sub-question.

The following world history questions were used as the training data; National Center Tests from 1997 to 2009 biennially, i.e. for seven times as a whole.

Each sub-question in the training data was first manually examined to obtain particular string patterns. Then, the string patterns were summarized in the form of regular expressions for each answer type. Table 2 shows an example of frequent string patterns found in each answer type.

Table 2: An example of frequent string patterns in each answer type

Answer type	String pattern
TERM	として正しい
SENTENCE	について述べた文として正しい
TF	正誤の組み合わせとして正しい
COMBO	の組み合わせとして正しい
UNIQUE	年代の古いものから順に正しく配列されている
MAP	次の地図中の (1)~(4) のうちから一つ

2.3 Creating queries

2.3.1 *Adding keywords to a set of query words*

Keywords were identified by extracting words (e.g., "キリスト教" or "人物") from phrases (e.g., "キリスト教に関連して" or "次の人物について述べた文") in sub-questions with the regular expressions used in section 2.2.1, and were added to a set of query words.

2.3.2 *Addition or adaptive addition of underlined texts to a set of query words*

The underlined texts in sub-questions were morphologically analyzed to extract only nouns, verbs, and adjectives, which were added to the set of query words.

Some underlined texts were unnecessary to answer sub-questions. Therefore, we determined whether the underlined texts were unnecessary or not by the classifier built from the training data, and implemented another function to adaptively adding only the necessary underlined texts to the query words. For more details, refer to section 3.

2.3.3 *Extracting a set of query words in accordance with answer types*

Figure 4 shows some examples of extracted query words for each answer type.

Only nouns, verbs and adjectives were extracted and added to the query words for each answer type. When the answer type was TERM or SENTENCE, queries were generated by each answer choice. When the answer type was TF, queries were generated by each of the corresponding texts required for true-or-false judgement. When the answer type was COMBO, queries were generated by each choice from the corresponding texts including blanks in question and from the words contained in each choice. When the answer type was UNIQUE, queries were created by each of the corresponding texts required to sort correctly in chronological order. When the answer type was MAP, no queries were generated and the sub-question was ignored by the system.

Table 1: Answer types of subquestions

Answer type	Description
TERM	A choice is in the form of a name of people, event, etc.
SENTENCE	A choice is in the form of a sentence.
TF	A choice is composed of a combination of "true" or "false".
COMBO	A choice is composed of a combination of symbols used in slot-filling problems.
UNIQUE	A choice is in the form of a chronological sequence of events.
MAP	Processing of images such as maps or photos are required to choose the right answer.

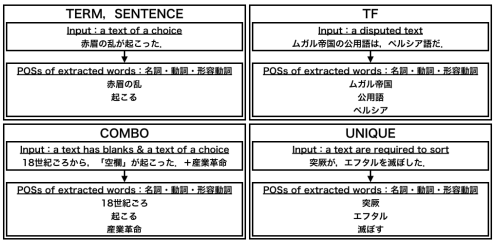

TERM, SENTENCE
Input : a text of a choice
赤眉の乱が起こった.
POSs of extracted words : 名詞・動詞・形容動詞
赤眉の乱
起こる
TF
Input : a disputed text
ムガル帝国の公用語は，ペルシア語だ.
POSs of extracted words : 名詞・動詞・形容動詞
ムガル帝国
公用語
ペルシア
COMBO
Input : a text has blanks & a text of a choice
18世紀ごろから，「空欄」が起こった．＋産業革命
POSs of extracted words : 名詞・動詞・形容動詞
18世紀ごろ
起こる
産業革命
UNIQUE
Input : a text are required to sort
突厥が，エフタルを滅ぼした.
POSs of extracted words : 名詞・動詞・形容動詞
突厥
エフタル
滅ぼす

Figure 4: Example of query words extracted for each answer type

2.3.4 Introducing multiple methods for generating queries

Figure 5 shows an example of queries generated by each method introduced in this section.

Four methods of generating queries were implemented: OR query method, which searches documents including any of the given query words, AND query method, which find documents containing at least particular named entities among the given query words, VS query method, which locates documents by both the OR and AND query method, and CLASS query method, which transforms the query words to their super-ordinate concept labels and searches only from documents having particular labels.

In CLASS query method, it is necessary to convert each query word to their super-ordinate concept label. We used the super-ordinate concept labels included in the event ontology EVT[1] and the named-entity tags attached in NE_Tokyoshoseki. The event ontology EVT was developed by Kawazoe et al., and is an ontology to support judging the authenticity of sentences written in natural languages. As of its version 1.3, the ontology has the total of 4,793 important events and people described in high school textbooks, which are classified in their super-ordinate concept categories such as "nation and dynasty", "social systems", and "technology and invention". Also, each of the classified named entities are annotated by various tags such as "evt:alias", which means the alias of another word, "wikipedia", which indicates it has the related header texts of the Wikipedia article, and "evt:location", suggesting the location where the event took place. NE_Tokyoshoseki is a set of named-entity tags used in the textbooks published by Tokyoshoseki with annotated named entities, provided by the organizers. It contains 14,622 named entities annotated with 32 kinds of tags. Each entity was labeled with at least one or more tag such as "person type, social role", "social system", and "historical event".

We manually created a correspondence table for the super-ordinate labels and the named-entity tags. If a target word was contained in the user dictionary and was attached one of the named-entity tags used in NE_Tokyoshoseki, the super-ordinate concept label was obtained from the correspondence table, and was added to the query words. Therefore, the CLASS query method only searches for documents generated from the event ontology EVT. The details of the knowledge sources used for document retrieval are described in section 2.4.1.

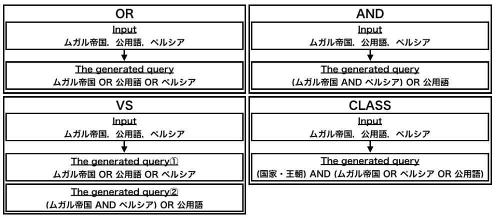

OR
Input
ムガル帝国，公用語，ペルシア
The generated query
ムガル帝国 OR 公用語 OR ペルシア
AND
Input
ムガル帝国，公用語，ペルシア
The generated query
(ムガル帝国 AND ペルシア) OR 公用語
VS
Input
ムガル帝国，公用語，ペルシア
The generated query①
ムガル帝国 OR 公用語 OR ペルシア
The generated query②
(ムガル帝国 AND ペルシア) OR 公用語
CLASS
Input
ムガル帝国，公用語，ペルシア
The generated query
(国家・王朝) AND (ムガル帝国 OR ペルシア OR 公用語)

Figure 5: Example of queries generated by each method

2.4 Document retrieval

2.4.1 Introducing multiple knowledge sources for document retrieval

This section describes multiple knowledge sources used in obtaining answer candidates through document retrieval.

First, the "textbook" set of documents, hereafter referred to as "T", was generated from several textbooks of world history for high school students published by Tokyo Shoseki Co., Ltd. and Yamakawa Shuppansha Ltd., and by registering each paragraph as a document.

Then, three sets of documents were created from Japanese Wikipedia articles with the title containing particular keywords; "Wikipedia per Doc" was built by registering one Wikipedia article as a document, "Wikipedia per Paragraph" by one paragraph of the article as a document, and "Wikipedia per Sentence" by one sentence as a document. Each of the three sets are hereafter referred to as "WD", "WP", and "WS", respectively. The keywords used to choose the articles were the 14,622 named entities included in the textbooks published by Tokyo Shoseki Co., Ltd. with annotated named-entity tags. Also, if the article was a redirect, the redirected page was obtained and added to the set of documents.

Also, the textbook set of documents "T" and each of the three sets of Wikipedia articles, "WD", "WP", and "WS", were merged as the mixed sets of documents, which were hereafter referred to as "TWD" for T+WD, "TWP" for T+WP, and "TWS" for T+WS, respectively.

Furthermore, another set of documents were made from the event ontology EVT, where is hereafter referred to as

[1] イベントオントロジー EVT - Researchmap : `http://researchmap.jp/zoeai/event-ontology-EVT/`

"Evt". The event ontology EVT contains a number of named entities used in world history, and sometimes include the textual descriptions about one of the named entities, which were extracted from Wikipedia. If such descriptions were found, the documents were added to "Evt", whose title was the corresponding named entity and whose body comprised of the descriptions.

Lastly, a chronology database was built as the knowledge source to be used only when the answer type was unique. The database was made by extracting the name of an event, its start year, and its end year from the chronology data available on the Web [2] [3] and the event ontology EVT.

2.4.2 *Creating a correspondence table of spelling variations for world history*

Synonyms were extracted by the following three methods from Wikipedia and the event ontology EVT.

The first method was based on the redirect of Wikipedia, where the title of the redirect and that of the redirected page were obtained as synonyms.

The second method was based on the characteristics of the first paragraph of Wikipedia articles. The first paragraph summarizes the whole article and often contains the descriptions about the aliases and abbreviations of the title. Thus, the certain phrases were identified by particular patterns, and the words were extracted which appeared in specific positions corresponding to each pattern. The title of the article and the extracted words were obtained as synonyms.

The third method utilized the tags attached in the event ontology EVT. The event ontology EVT include the tags indicating the aliases for some of the named entities. When a named entity has such corresponding aliases, they were extracted as synonyms.

2.5 Scoring answer candidates

2.5.1 *Scoring by temporal intervals*

The answer candidates were scored based on the number of the overlapped temporal intervals between the expressions in a sub-question and those in each answer choice. First, the temporal intervals were extracted for the words or phrases in a sub-question and those in each choice, which were registered in the chronology database in section 2.4.1 or were the general expressions such as "12th century". Then, the number of the overlapped temporal intervals and that of the non-overlapped intervals were calculated between the sub-question and the answer choice. The point of 1.0 for each overlapped intervals and that of -0.25 for each non-overlapped intervals were added to get the final score.

2.5.2 *Scoring answer candidates in accordance with answer types*

Solr[4] was used as a full-text document retrieval engine for every answer type. The scores and the ranking of the retrieval results were decided according to cosine similarity by TF-IDF.

If the answer type was TERM, SENTENCE, COMBO, or TF, the top ten documents were retrieved by the given query, and the highest score among the ten documents were set as the score for the answer candidate.

If the answer type was UNIQUE, the top ten documents were retrieved from the chronology database by the given query, and the start year of the top document was set as the start year for the query.

If the answer type was MAP, the system ignore the sub-question.

2.5.3 *Choosing the answer in accordance with answer types*

Figure 6 shows an example of choosing the answer in accordance with answer types.

If the answer type was TERM, SENTENCE, or COMBO, the choice with the highest score was selected as the answer.

If the answer type was TF, each sentence to be checked its authenticity was judged as true when its score exceeded a certain threshold and as false otherwise. The threshold was set to 10, because the preliminary experiment showed that the query generated from the sentence was highly related to the retrieved documents when the score of the query was more than 10. The answer choice having the same combination of the true-or-false results was selected as the answer.

If the answer type was UNIQUE, the events in question were chronologically sorted by the start year, and the choice matching the same chronological sequence with the sorted result was selected as the answer.

If the answer type was MAP, the system ignore the sub-question.

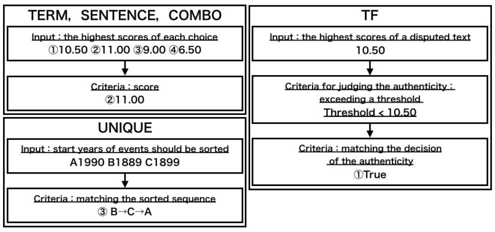

TERM, SENTENCE, COMBO
Input : the highest scores of each choice
①10.50 ②11.00 ③9.00 ④6.50
Criteria : score
②11.00
UNIQUE
Input : start years of events should be sorted
A1990 B1889 C1899
Criteria : matching the sorted sequence
③ B→C→A
TF
Input : the highest scores of a disputed text
10.50
Criteria for judging the authenticity :
exceeding a threshold
Threshold < 10.50
Criteria : matching the decision
of the authenticity
①True

Figure 6: Example of choosing the answer in accordance with answer types

2.6 Systems Configuration

This section describes the configuration of the systems in each phase. For convenience and clarity, each phase is represented as "PH1", "PH2", and "PH3", respectively, and included in the system's ID below.

2.6.1 *Phase-1*

All of the three systems developed for Phase-1 did not include the estimation of the answer types. They were based on the baseline system provided by the organizers, and modified and improved by adding the following functions:

(1) Creating user dictionary specialized in world history
(2) Adding keywords to a set of query words
(3) Introducing multiple methods for generating queries
(4) Introducing multiple knowledge sources for document retrieval

Table 3 shows the difference of the system's configurations for Phase-1 in (3) and (4).

2.6.2 *Phase-2*

All of the three systems developed for Phase-2 were based on the baseline system provided by the organizers, and modified and improved by adding the following functions:

[2] 世界史年表：`http://www.h3.dion.ne.jp/~urutora/sene.htm`

[3] 世界史年表 [1-5]（理解する世界史）：`http://www2s.biglobe.ne.jp/~t_tajima/nenpyo-[1-5]/nenpyo-[1-5].htm`

[4] Apache Solr：`http://lucene.apache.org/solr/`

Table 3: The comparison of the system configuration for Phase-1

System Id	(3)	(4)
KSU-JA-01@PH1	VS	WP
KSU-JA-02@PH1	VS	WS
KSU-JA-03@PH1	OR	WS

(1) Creating user dictionary specialized in world history
(2) Adding keywords to a set of query words
(3) Introducing multiple methods for generating queries
(4) Introducing multiple knowledge sources for document retrieval
(5) Scoring by temporal intervals

Table 4 shows the difference of the system's configurations for Phase-2 in (3) and (4).

Table 4: The comparison of the system configuration for Phase-2

System Id	(3)	(4)
KSU-JA-01@PH2	VS	WP
KSU-JA-02@PH2	VS	WS
KSU-JA-03@PH2	OR	WS

2.6.3 Phase-3

All of the three systems developed for Phase-3 were based on the baseline system provided by the organizers, and modified and improved by adding the following functions:

(1) Creating user dictionary specialized in world history
(2) Estimating answer types
(3) Adding keywords to a set of query words
(4) Addition or adaptive addition of underlined texts to a set of query words
(5) Extracting a set of query words in accordance with answer types
(6) Introducing multiple methods for generating queries
(7) Introducing multiple knowledge sources for document retrieval
(8) Creating a correspondence table of spelling variations for world history
(9) Scoring answer candidates in accordance with answer types
(10) Choosing the answer in accordance with answer types

Table 5 shows the difference of the system's configurations for Phase-3 in (4), (6), and (7).

3. ADDITION OR ADAPTIVE ADDITION OF UNDERLINED TEXTS TO A SET OF QUERY WORDS

Some underlined texts were unnecessary to answer sub-questions. Therefore, we developed the classifier which determined whether the underlined texts were unnecessary or not, and adaptively added only the necessary underlined texts to the query words.

Table 5: The comparison of the system configuration for Phase-3

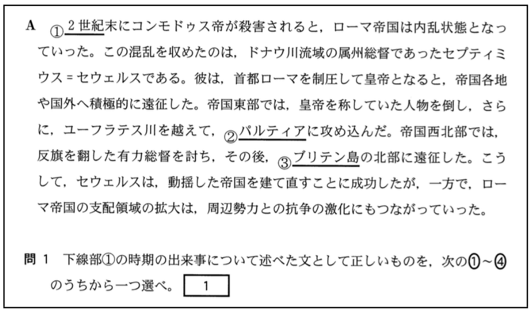
System Id	(4)	(6)	(7)
KSU-JA-01@PH3	No	VS	WP
KSU-JA-02@PH3	No	CLASS	Evt
KSU-JA-03@PH3	Yes	VS	WP

3.1 Target Data

Shown below are some examples of the underlined texts when they are necessary and unnecessary to answer the sub-question, respectively.

3.1.1 When the underlined texts are necessary to answer the question

Figure 7 shows the example of a sub-question where the underlined texts are necessary to answer it. The phrase "時期の出来事", which literally means "the event during the period", is ambiguous by itself and insufficient to answer the sub-question correctly. It is indispensable for the system to specify that "時期の出来事" represents "2 世紀の出来事", which literally means "the event during the 2nd century", by referring to the underlined text in the context sentences. Likewise, the underlined text is determined to be necessary to answer the question, when the descriptions in the sub-question or in the answer choice are short of specific words or phrases, and when only the underlined texts include such words or phrases.



A　①2世紀末にコンモドゥス帝が殺害されると，ローマ帝国は内乱状態となっていった。この混乱を収めたのは，ドナウ川流域の属州総督であったセプティミウス＝セウェルスである。彼は，首都ローマを制圧して皇帝となると，帝国各地や国外へ積極的に遠征した。帝国東部では，皇帝を称していた人物を倒し，さらに，ユーフラテス川を越えて，②パルティアに攻め込んだ。帝国西北部では，反旗を翻した有力総督を討ち，その後，③ブリテン島の北部に遠征した。こうして，セウェルスは，動揺した帝国を建て直すことに成功したが，一方で，ローマ帝国の支配領域の拡大は，周辺勢力との抗争の激化にもつながっていった。

問 1　下線部①の時期の出来事について述べた文として正しいものを，次の①～④のうちから一つ選べ。 1

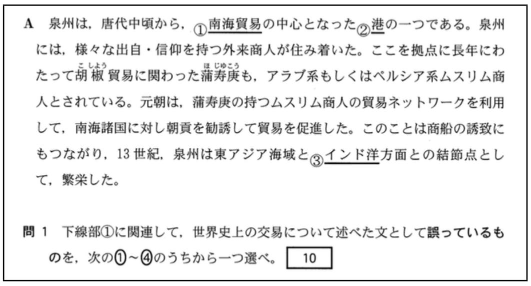
Figure 7: When the underlined texts are necessary to answer the question

3.1.2 When the underlined texts are unnecessary to answer the question

Figure 8 shows the example of a sub-question where the underlined texts are not necessary to answer it. The phrase "世界史上の交易", which literally means "international commerce in the world history", is sufficient to answer the sub-question correctly. The underlined text "南海貿易", which literally means "trade around South Sea", is not necessarily required and might be the cause to bring an incorrect answer. Likewise, the underlined text is determined to be unnecessary to answer the question, when the descriptions in the sub-question or in the answer choice contain sufficient words or phrases, and when the underlined texts are not necessarily required.

3.1.3 Creating the training data and test data

The following exams provided by the organizers were used as the training data for classifying the underlined texts.



A　泉州は，唐代中頃から，①南海貿易の中心となった②港の一つである。泉州には，様々な出自・信仰を持つ外来商人が住み着いた。ここを拠点に長年にわたって胡椒貿易に関わった蒲寿庚も，アラブ系もしくはペルシア系ムスリム商人とされている。元朝は，蒲寿庚の持つムスリム商人の貿易ネットワークを利用して，南海諸国に対し朝貢を勧誘して貿易を促進した。このことは商船の誘致にもつながり，13 世紀，泉州は東アジア海域と③インド洋方面との結節点として，繁栄した。

問 1　下線部①に関連して，世界史上の交易について述べた文として誤っているものを，次の①～④のうちから一つ選べ。 10

Figure 8: When the underlined texts are unnecessary to answer the question

- National Center Tests from 1997 to 2009 biennially, i.e. for seven times as a whole.
- Benesse Trial Exams in 2014 and 2015
- Yoyogi Seminar Trial Exams in 2013

The correct labels for classification were manually given by each answer column for all the above exams. The answer columns were determined as "Yes" when the underlined texts were necessary to answer the question, and as "No" when they were not, considering the content of the underlined text, the sub-question, and the answer choices, according to section 3.1.1 and 3.1.2. Also, some answer columns were labeled as "undefined" when there was no underlined text in the first place, when a certain image processing was necessary to answer the question or when the answer columns could be judged as either "Yes" or "No". In this manner, the training data were made comprising of 136 "Yes", 215 "No", and 55 "undefined". Incidentally, 55 data of "undefined" were composed of 50 cases when there was no underlined text in the first place, two cases when a certain image processing was necessary to answer the question, and three cases when the answer columns could be judged as either "Yes" or "No".

3.2 Features used for classification

In this section, features used for classification are explained. Table 6 shows the definition of symbols used in calculating the features.

Table 6: Definition of symbols for calculating features

Symbol	Definition
M_U	the set of morphemes in an underlined text
M_S	the set of morphemes in a sub-question
N_U	the set of nouns in an underlined text
N_S	the set of nouns in a sub-question
NG_U	the set of N-grams in an underlined text
NG_S	the set of N-grams in a sub-question
NOC_U	the number of characters in an underlined text
NOC_S	the number of characters in a sub-question
W_U	the set of words in an underlined text
W_S	the set of words in a sub-question

3.2.1 Overlap ratio of morphemes

The overlap ratio of morphemes is calculated, as follows:

$$MorphemeOverlap = \frac{|M_U \cap M_S|}{|M_U \cup M_S|}$$

3.2.2 Overlap ratio of nouns

The overlap ratio of nouns is calculated, as follows:

$$NounOverlap = \frac{|N_U \cap N_S|}{|M_U \cup M_S|}$$

3.2.3 Cosine Similarity

Two kinds of document vectors were generated for calculating cosine similarity, based on each of the the following term-weighting methods:

(1) term frequency (tf)
(2) tf-idf

Moreover, in (2), the idf was calculated in the following two ways.

- Regarding the union of the document sets made from the underlined text, the sub-question, and the answer choices as the whole document set
- Regarding each document set made from the underlined text, the sub-question, and the choices as the respective whole document set

In this paper, the former is called "$tfidf_1$", and the latter is called "$tfidf_2$".

In view of the above, the five kinds of cosine similarity were calculated, as follows:

- $Sim_1 = CosineSimirality(\overrightarrow{D_{tf,U}}, \overrightarrow{D_{tf,S}})$
- $Sim_2 = CosineSimirality(\overrightarrow{D_{tfidf_1,U}}, \overrightarrow{D_{tfidf_1,S}})$
- $Sim_3 = CosineSimirality(\overrightarrow{D_{tfidf_2,U}}, \overrightarrow{D_{tfidf_2,S}})$
- $Sim_4 = CosineSimirality(\overrightarrow{D_{tf,C}}, \overrightarrow{D_{tf,S}})$
- $Sim_5 = CosineSimirality(\overrightarrow{D_{tfidf_1,C}}, \overrightarrow{D_{tfidf_1,S}})$

where $CosineSimirality$ is defined, as follows:

$$CosineSimirality(\vec{X}, \vec{Y}) = \frac{\vec{X} \cdot \vec{Y}}{|\vec{X}||\vec{Y}|}$$

Also, $\overrightarrow{D_{tf,U}}$ denotes the document vector of the tfidf in the underlined text, the subscripts of U, S, and C represent the underlined text, the sub-question, and the choice, respectively.

3.2.4 Overlap ratio of N-gram(N = 1, 2, 3)

The overlap ratio of N-gram is calculated, as follows:

$$NgramOverlap = \frac{|NG_U \cap NG_S|}{|NG_S|}$$

3.2.5 Number-of-characters ratio

The number-of-characters ratio is calculated, as follows:

$$PropotionOfNumberOfCharacters = \frac{NOC_U}{NOC_U + NOC_S}$$

3.2.6 Jaccard coefficient

The jaccard coefficient is calculated as follows:

$$JaccardCoefficient = \frac{|W_U \cap W_S|}{|W_U \cup W_S|}$$

3.2.7 Simpson coefficient

The simpson coefficient is calculated as follows:

$$SimpsonCoefficient = \frac{|W_U \cap W_S|}{min(|W_U \cup W_S|)}$$

3.3 Constructing classifiers

The classifiers were constructed based on the features shown in section 3.2. The morphological analysis for the underlined texts, the sub-question, and the answer choices were done by MeCab, and the classification was carried out by Random Forest implemented on Weka.

4. RESULTS

This section gives the results of the systems in each phase.

4.1 Phase-1

Table 7 shows the results of our systems in Phase-1. The table only indicates the accuracy, because the systems neither estimated the answer types nor carried out the process dependent on each answer type.

Table 7: Results of our runs for Phase-1

System ID	Accuracy
KSU-JA-01@PH1	0.49(20/41)
KSU-JA-02@PH1	0.49(20/41)
KSU-JA-03@PH1	0.46(19/41)

4.2 Phase-2

Table 8 shows the results of our systems in Phase-2. The table only indicates the accuracy, because the systems did not carry out the process dependent on each answer type, whereas they estimated the answer types.

Table 8: Results of our runs for Phase-2

System ID	Accuracy
KSU-JA-01@PH2	0.41(26/63)
KSU-JA-02@PH2	0.37(23/63)
KSU-JA-03@PH2	0.33(21/63)

4.3 Phase-3

Table 9 shows the results of our systems in Phase-3.

5. DISCUSSION

In this section, we discuss the systems developed for Phase-3. The basic system KSU-JA-01@PH3 is compared with KSU-JA-02@PH3 and KSU-JA-03@PH3.

5.1 Comparison between KSU-JA-01@PH3 and KSU-JA-02@PH3

The differences between KSU-JA-01@PH3 and KSU-JA-02@PH3 are the query generation method and the knowledge sources used. The system KSU-JA-01@PH3 was based on the VS query method and used "Wikipedia per paragraph" as the knowledge source. Meanwhile, the system KSU-JA-02@PH3 was based on the CLASS query method and utilized "Evt" made from the event ontology EVT as the knowledge source which can be searched by the superordinate concept for each word.

Table 9 shows the accuracy of KSU-JA-01@PH3 was higher than that of KSU-JA-02@PH3. For KSU-JA-01@PH3, the different retrieval results were obtained for each of the answer choices, whereas for KSU-JA-02@PH3, the retrieval results often showed no difference for each of the choices. That is to say that the search function of KSU-JA-02@PH3 was hardly appropriate to answer correctly. One of the reasons for this is thought to be the smaller number of documents in Evt, which has 4,024 documents, whereas WP has 79,607 documents.

5.2 Comparison between KSU-JA-01@PH3 and KSU-JA-03@PH3

The difference between KSU-JA-01@PH3 and KSU-JA-03@PH3 is the way of handling the underlined texts for the query words. In KSU-JA-01@PH3, each underlined text was always added to the set of query words, while in KSU-JA-03@PH3, each underlined text was adaptively added to the set of query words.

Table 9 indicates that the accuracy of KSU-JA-01@PH3 and KSU-JA-03@PH3 showed no difference. There was a tendency for the score of the queries in KSU-JA-03@PH3 to become higher than that in KSU-JA-01@PH3. However, the query words in KSU-JA-03@PH3 were only one or two words less than those in KSU-JA-01@PH3, whereas the total number of the query words were six to ten words in both systems. Therefore, the adaptive addition of the underlined texts could not show sufficient effect on the answer selection, which turned out that they almost always selected the same choice as a result.

5.3 Discussion for systems with other configuration

This section discusses how the accuracy of the systems change depending on the different combinations of each function.

Several systems were developed with the different configurations of the following functions:

- Addition or adaptive addition of underlined texts to a set of query words introduced in section 2.3.2
- Four kinds of multiple methods for generating queries in section 2.3.4
- Eight kinds of multiple knowledge sources for document retrieval described in section 2.4.1

Table 10: Systems developed with different configurations. The system ID is composed of X-Y, where X indicates the query generation method and Y denotes the knowledge source.

	OR	AND	VS	CLASS
T	1-1	2-1	3-1	
WD	1-2	2-2	3-2	
WP	1-3	2-3	3-3, 4-3[1]	
WS	1-4	2-4	3-4	
TWD	1-5	2-5	3-5	
TWP	1-6	2-6	3-6	
TWS	1-7	2-7	3-7	
Evt	1-8	2-8	3-8	4-1

[1] Only this system adaptively adds the underlined texts to a set of query words.

Table 10 shows the 26 systems developed with different configurations. In this table, the system 3-3 is identical to KSU-JA-01@PH3. The systems 4-1 and 4-3 are identical to KSU-JA-02@PH3 and KSU-JA-03@PH3, respectively. For an experiment, each system answered the test data for Phase-3.

Figure 9 shows the result of the experiment. Also, the result of the baseline system, hereafter referred to as "Baseline", provided by the organizers was added to figure 9.

Table 9: Results of our runs for Phase-3

System ID	TERM	SENTENCE	TF	COMBO	UNIQUE	MAP	Accuracy
KSU-JA-01@PH3	2/2	5/20	3/6	2/4	2/3	0/1	0.39(14/36)
KSU-JA-02@PH3	2/2	2/20	3/6	2/4	2/3	0/1	0.31(11/36)
KSU-JA-03@PH3	2/2	5/20	3/6	2/4	2/3	0/1	0.39(14/36)

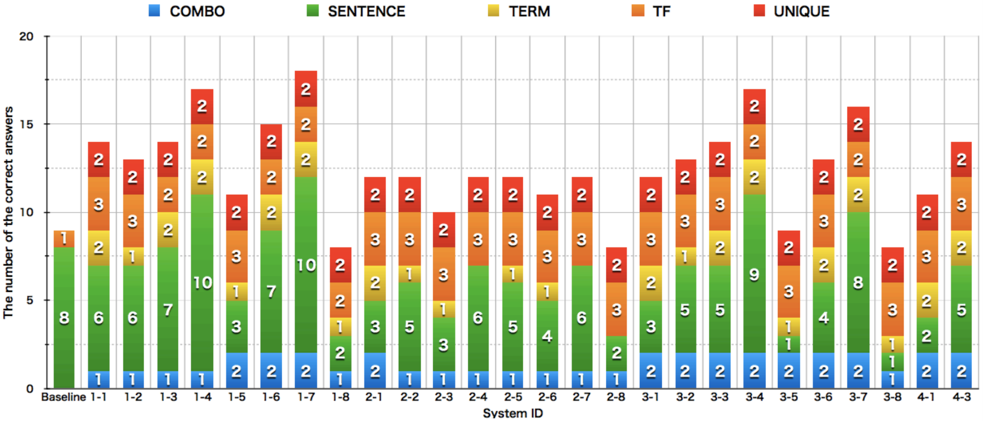


Figure 9: Comparison of the number of correct answers between the systems with different configurations

Figure 9 shows that the system 1-7 achieved the highest score of 18 correct answers, i.e. the accuracy of 50%, which used the OR query method and TWS. It also indicates that the system 1-4 and 3-4 accomplished the second highest scores of 17 correct answers, i.e. 47.2%, which used WS, and that the system 3-7 the third highest scores of 16 correct answers, i.e. 44.4%, which used TWS. Meanwhile, the score of Baseline was 9 correct answers, i.e. the accuracy of 25.0%, suggesting that the system 1-7 achieved improvements by 25 points. It was confirmed that the Baseline often gave wrong answers for the sub-questions with the answer type of TF and UNIQUE, whereas the system 1-7 chose correct answers for them. Thus, it seems reasonable to conclude that the query generation method in accordance to the answer types and other modifications successfully improved the accuracy.

For those systems using the OR query method and the VS query method, namely, the systems 1-1∼1-8 and 3-1∼3-8, respectively, the smaller the granularity of documents from Wikipedia became, the higher the accuracy of the systems got. This happened because 55.6% of all the sub-questions had the answer type of SENTENCE, and because it was easier to agree with the queries when searching for the sentence-based documents than for the paragraph- or page-based documents. Also, the sub-questions having the answer type of SENTENCE in the National Center Test almost always ask whether each answer choice is true or false, and these choices often have a brief description about historical events. Therefore, the sentence-based documents seem to have more documents which are more similar to those descriptions used in the answer choices.

For those systems using the AND query method, i.e. the systems 2-1∼2-8, the difference of the accuracy between the knowledge sources are smaller than those for systems using the OR or VS query. It is considered that the AND query method always returned the documents containing particular words, meaning that the scores of such documents became relatively high regardless of the granularity of documents. Thus, the systems using the AND query seem to have more often selected the same answer as a result. It was confirmed that the systems using the AND query actually selected the same answer more often than those using the OR or VS query.

6. CONCLUSION

This paper described the systems and results of the team KSU for QA Lab-2 task in NTCIR-12. In each phase of the task, we developed three automatic answering systems for world history questions in the National Center Test for University Admissions. In order for QA systems using document retrieval to answer questions correctly, it is important to estimate exact question types, and to utilize knowledge sources and query generation methods in accordance with these types. Therefore, we designed systems that focus on knowledge sources and query generations using the underlined texts in given exams. Scores of the formal runs were 20 correct answers(49%) and 68 points with KSU-JA-02@PH1 system in phase-1, 26 correct answers(41%) and 70 points with KSU-JA-01@PH2 system in phase-2 and 14 correct answers(39%) and 38 points with KSU-JA-01@PH3 system in phase-3.

7. ACKNOWLEDGEMENTS

A part of this work was supported by Kyoto Sangyo University Research Grants.

8. REFERENCES

[1] S. Hideyuki, S. Kotaro, I. Madoka, F. Akira, K. Yoshionobu, M. Teruko, M. Tatsunori, and K. Noriko. Task overview for ntcir-12 qa lab-2 (draft). In *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies*, 2016.

[2] H. Shibuki, K. Sakamoto, Y. Kano, T. Mitamura, M. Ishioroshi, K. Y. Itakura, D. Wang, T. Mori, and N. Kando. Overview of the ntcir-11 qa-lab task. In *NTCIR*, 2014.