A method for GeoTime information retrieval based on question decomposition and question answering

--- Yokohama National University at NTCIR-8 GeoTime ---

Tatsunori Mori

Yokohama National University, Japan

mori@forest.eis.ynu.ac.jp

Introduction

- We participated in the Japanese mono-lingual (JA-JA) task.
- Our proposed method for GeoTime information retrieval is based on
  - Question decomposition and
  - Question answering.
    - GeoTime information retrieval can be regarded as one special case of IR4QA, because a query submitted to a system is a natural language question in typical situations.
    - We may straightforwardly consider documents that have good answer candidates as documents relevant to the query.
- We developed a system that utilize a question-answering system.

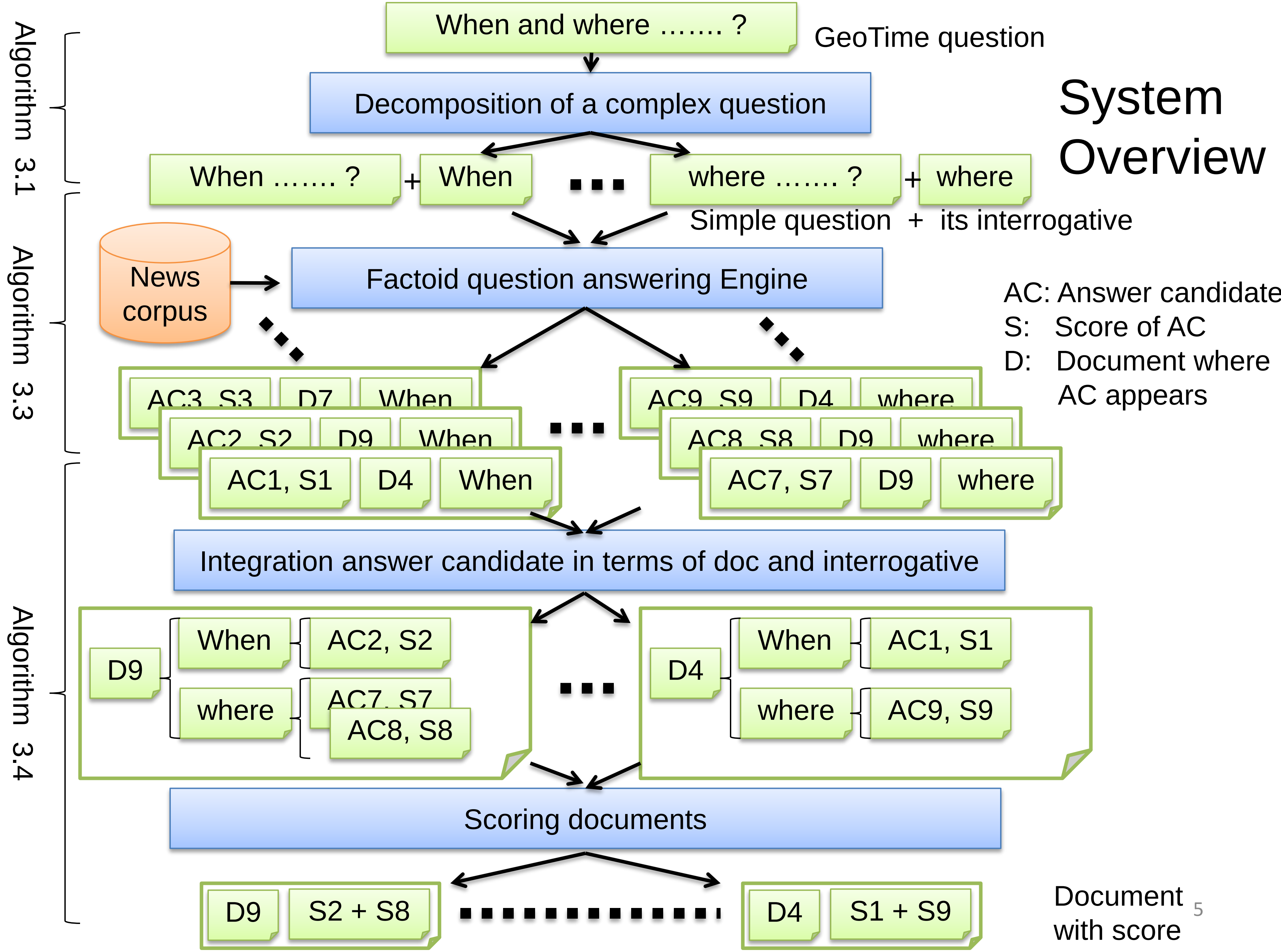
Related work

- GeoTime information retrieval may be regarded as a special case of IR4QA.
  - Many approaches to IR4QA introduce some extensions to treat natural sentence questions or question types.
  - Their foundation are information retrieval systems[Sakai et al. 2008].
- There are some text processing method based on the result of question answering system.
  - [Mori et al. 05] proposed a method for multi-answer-focused summarization using a question-answering engine.
    - Importance of each sentence is calculated based on the scores of answer candidates appeared in the sentence.
- Our approach to GeoTime information retrieval takes the same kind of approach as the latter researches.
  - In these researches, the scores of answer candidates are used to weight sentences.
  - In our GeoTime information retrieval, documents are weighted according to the score.

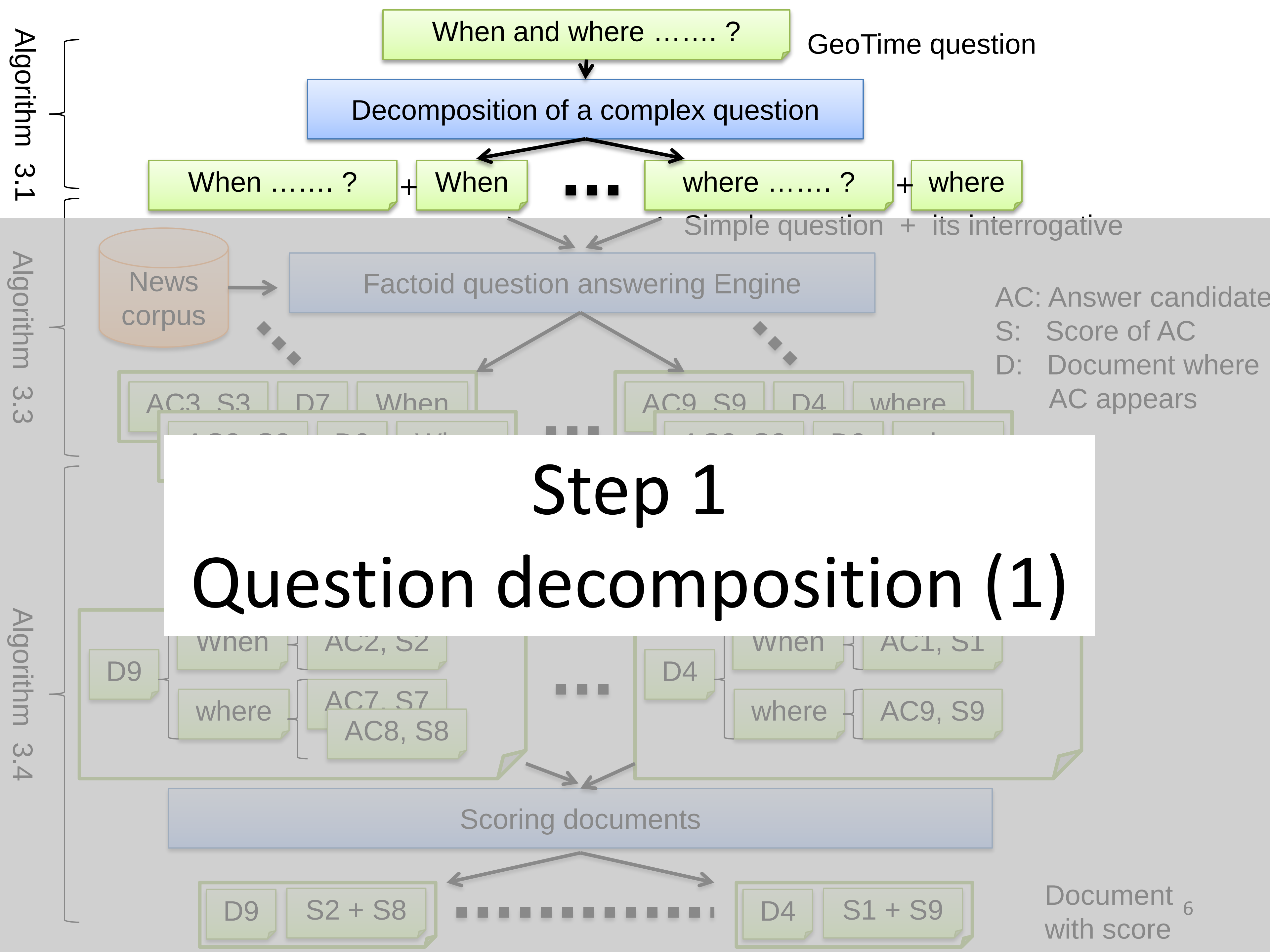
Proposed method

- The proposed method consists of the following three procedures.
  1. Decomposing a complex GeoTime question into a set of simple factoid questions,
  2. Factoid question-answering for the simple questions, and
  3. Scoring documents according to the scores of answer candidates in each document.

System Overview



Step 1
Question decomposition (1)



Question decomposition (2)

- GeoTime questions are usually complex questions, which have multiple interrogatives, like when, where, etc.
- We suppose that each GeoTime question is able to be decomposed into a set of simple factoid questions.
  - <*Q*, *interrog*>
    - *Q* is a simple question with one interrogative *interrog*.
- These simple factoid questions may be handled a factoid question-answering system.

Algorithm 3.1: DECOMPOSEQUESTION(Qc)

---

comment: returns a set of tuples of $\langle Q, interrog \rangle$, where Q is a simple question with one interrogative $interrog$, which is obtained by the decomposition of an inputted complex GeoTime question Qc.

global $InterrogPats$
comment: $InterrogPats$ is a set of patterns that match with interrogatives in question sentences.

procedure PATTERNMATCH($Str, Pats$)
  comment: returns a set of tuples of position $\langle PosS, PosE \rangle$, where $PosS$ and $PosE$ are the start and end positions of a substring of Str matched with one of patterns $Pats$.
  return ($\{\langle PosS, PosE \rangle\}$)

procedure SUBSTR($Str, \langle PosS, PosE \rangle$)
  comment: returns a sub-string $SubStr$ of Str that starts from position $PosS$ and ends at position $PosE$.
  return (SubStr)

procedure DELSUBSTRS($Str, Matches$)
  comment: returns a string $Str1$ that is obtained by deleting all substring expressed by $Matches$ from a string Str.
  return ($Str1$)

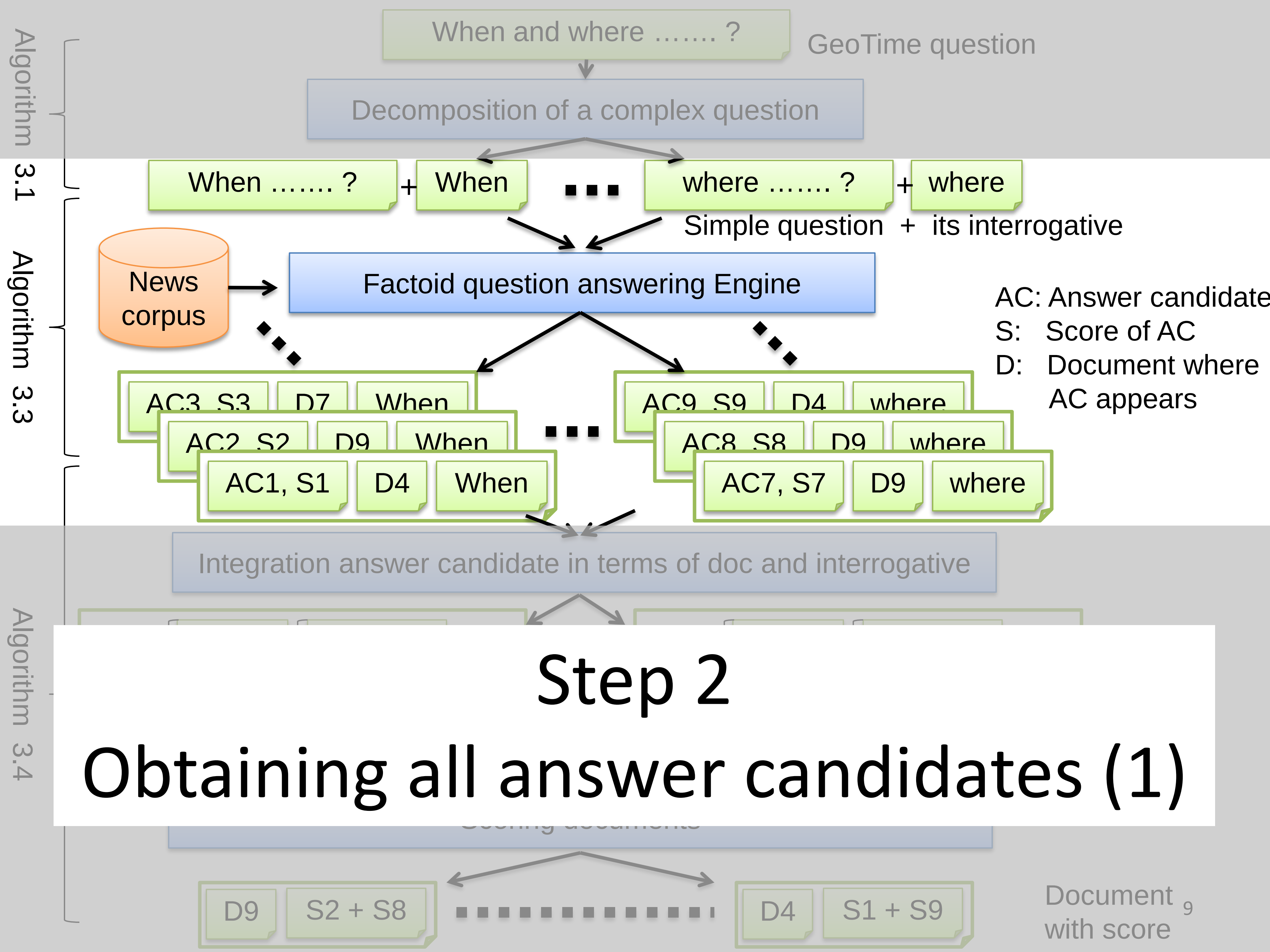
main
  $Ms \leftarrow$ PATTERNMATCH($Qc, InterrogPats$)
  $Qs \leftarrow \bigcup_{M \in Ms} \{\langle \text{DELSUBSTRS}(Qc, Ms \backslash \{M\}), \text{SUBSTR}(Qc, M) \rangle\}$
  return (Qs)

Question decomposition (3)

- Example GeoTime qestion (GeoTime-0003)
  - いつ、どこでポール・ニッツは、亡くなりましたか？
    - (When and where did Paul Nitze die?)
- Decomposed questions
  1. いつポール・ニッツは、亡くなりましたか？
     - (When did Paul Nitze die?)
  2. どこでポール・ニッツは、亡くなりましたか？
     - (Where did Paul Nitze die?)
- Our current implementation of question decomposition is based on a simple pattern-match.

Step 2

Obtaining all answer candidates (1)



Obtaining all answer candidates (2)

- The algorithm calls a factoid question-answering system to obtain answer candidates and their scores for all of the simple questions.
- <*D*,*interrog*,*AC*,*Sr*,*Sw*>
  - *D*: document
  - *Interrog*: interrogative
  - *AC*: answer candidate
  - *Sr*: raw score of *AC*
  - *Sw*: weighted score of *AC*

Algorithm 3.3: GetAllAnsCands(Qs)

---

comment: returns a set of tuples of $\langle D, interrog, AC, Sr, Sw\rangle$, where AC and D are an answer candidate and a document in which the answer candidate appears. $interrog$ is the interrogative asked in a decomposed question. Sr and Sw are the raw and weighted score of the answer candidate. The inputs Qs is a set of decomposed questions.

procedure QA(Q)

  comment: returns a set of tuples of $\langle AC, D, Sr, Sw\rangle$ for the question Q by using a factoid question-answering system.

  return $(\{\langle AC, D, Sr, Sw\rangle\})$

main

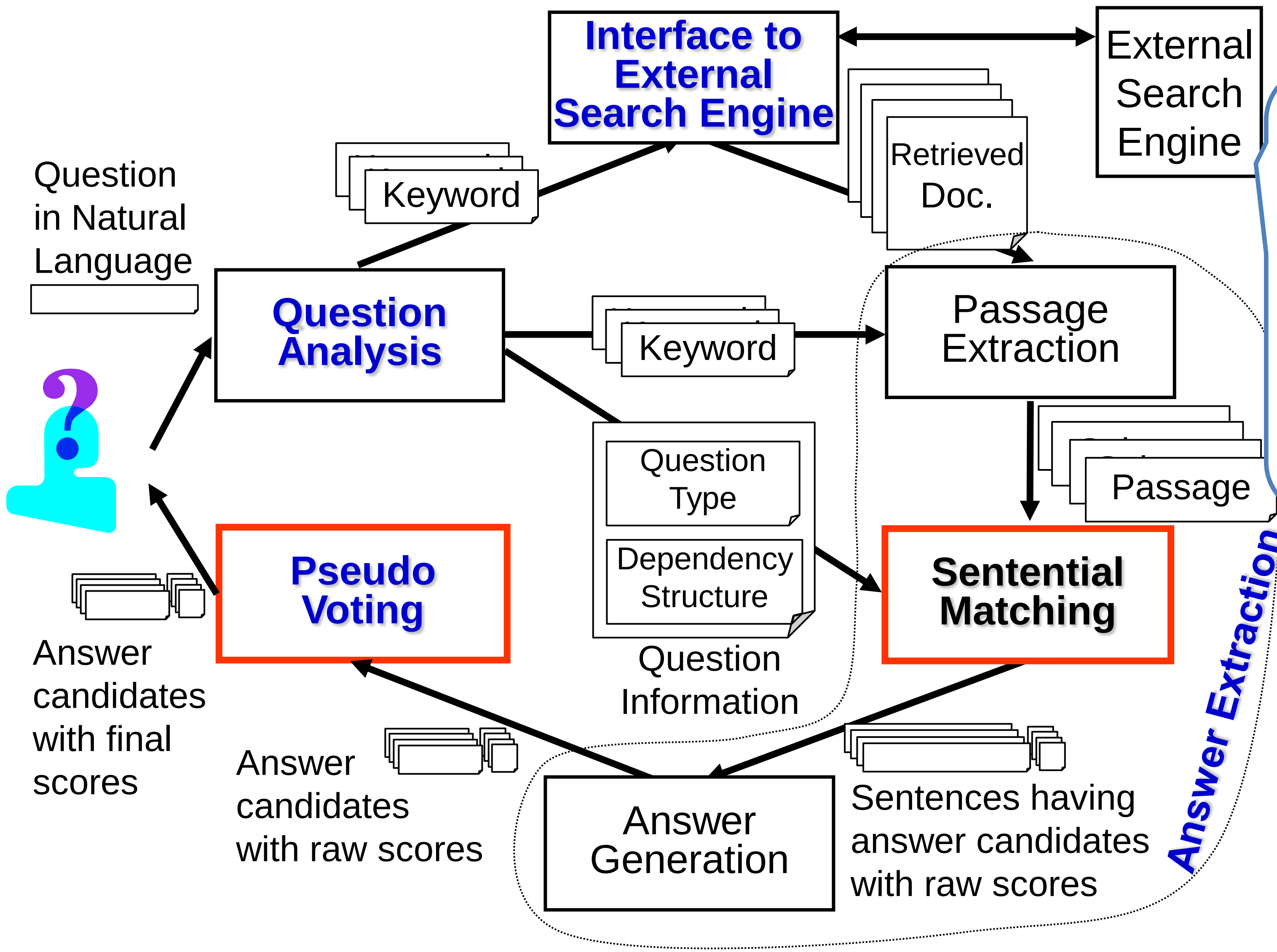
  $ACs \leftarrow \{\}$

  for each $\langle Q, interrog\rangle \in Qs$

    do $\begin{cases} As \leftarrow \text{QA}(Q) \\ \textbf{for each } \langle AC, D, Sr, Sw\rangle \in As \\ \quad \textbf{do } ACs \leftarrow ACs \bigcup \{\langle D, interrog, AC, Sr, Sw\rangle\} \end{cases}$

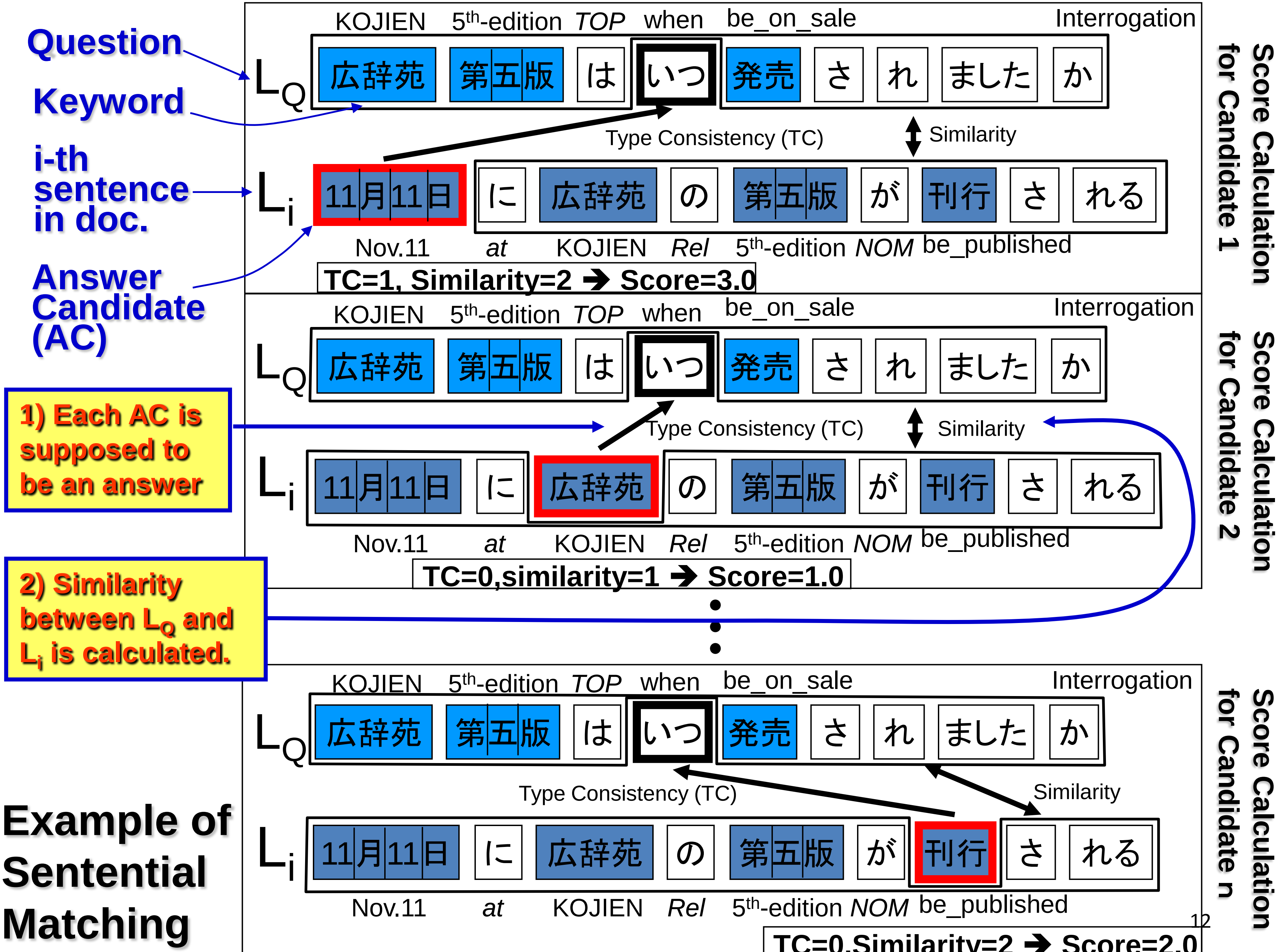
  return (ACs)

Basic factoid-type QA system (1)



Ordinary search engine based on tf*idf and the vector space model. We do not introduce no feedbacks.

Example of Sentential Matching



Basic factoid-type QA system (3)

Raw score for answer candidates

- **It is difficult to make QA systems high precision with one monolithic method.**
  - There is a trade-off relation between informativeness and robustness of analysis in each processing technique.
    - More informative ←→ Less informative
    - Less robust ←→ More robust
- We employ **multiple complementary methods** in order for our QA system to have a variety of informativeness and robustness.
- Implementation: **Raw score** for an answer candidate $\boldsymbol{AC}$ in the $\boldsymbol{i}$-th retrieved sentence $\boldsymbol{L_i}$ with respect to a question sentence $\boldsymbol{L_q}$.

$$S(AC, L_i, L_q) = \underbrace{Sb(AC, L_i, L_q)}_{\text{Bi-gram}} + \underbrace{Sk(AC, L_i, L_q)}_{\text{keyword}} + \underbrace{Sd(AC, L_i, L_q)}_{\text{Dependency between an answer candidate and a keyword}} + \underbrace{St(AC, L_i, L_q)}_{\text{Question type consistency}}$$

Basic factoid-type QA system (4)

Pseudo voting method in search scheme

- Many existing QA systems exploit global information about answer candidate.
  - **Voting method --- boosting the score for answers that occur multiple times** [Clarke 01, Xu 03].
- Pseudo voting [Mori 05]
  - Since our method continues searching for answers until scores of ***n different*** answers are fixed in n-best search, the system may find ***other answer candidates that have same surface expression***.
  - We can **use the partial frequency information** with regard to found answer candidates.
  - **Weighted score** $S^v(AC,L_q)$ for an answer candidate AC is:

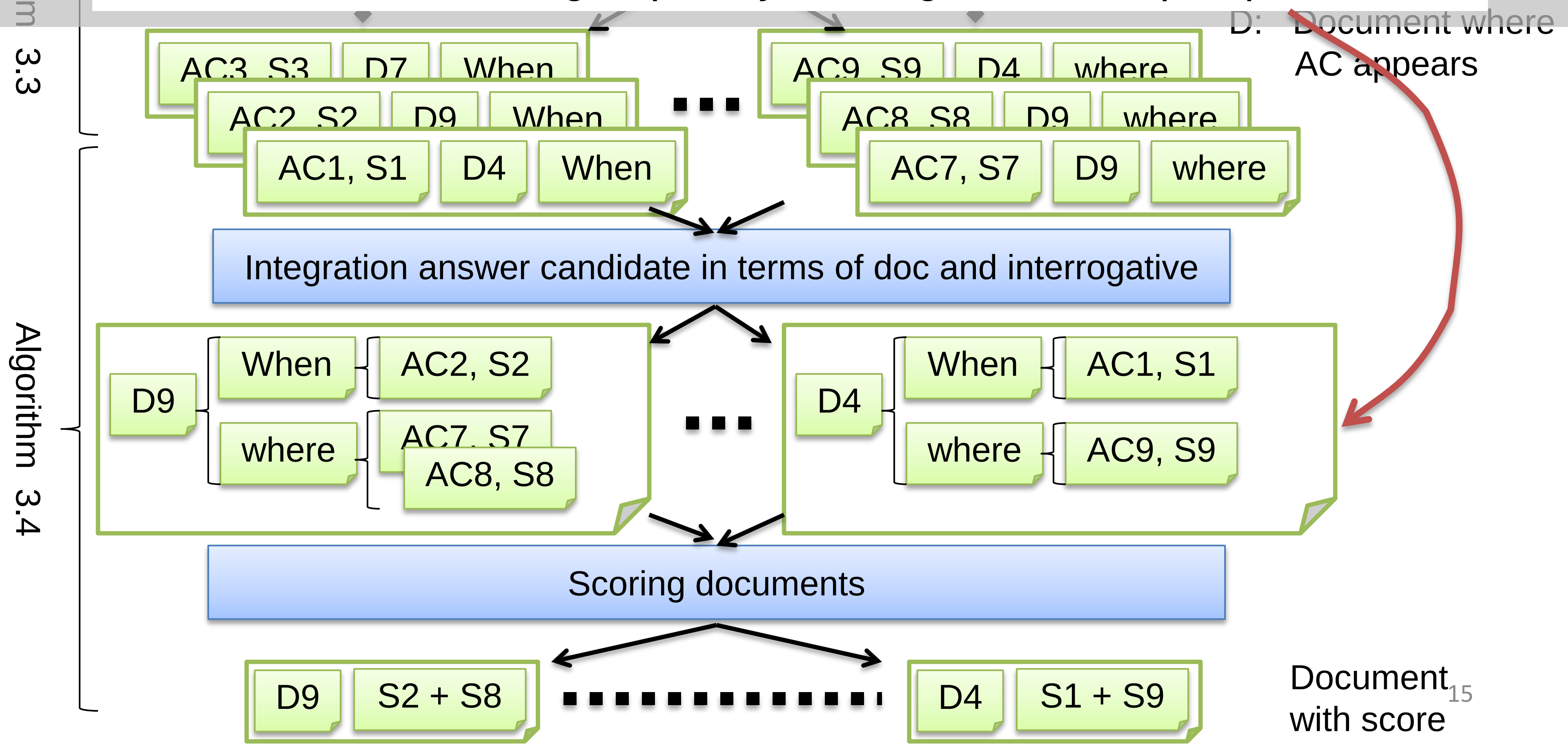
$$S^v(AC,L_q) = (\log_{10}(\boxed{freq(AC, AnsList)}) + 1) \cdot \max_{L_i} \boxed{S(AC, L_i, L_q)}$$

Frequency: $freq(AC, AnsList)$; Raw score: $S(AC, L_i, L_q)$

where $AnsList$ is the list of answer candidates whose scores are fixed.

Step 3
Scoring documents (1)

First, all answer candidates (ACs) are grouped by document, and then ACs in a document are grouped by interrogative of simple question.



Scoring documents (2)

- All answer candidates (ACs) are grouped by document, and then ACs in a document are grouped by interrogative of simple question.
- We define the **sub-score of document** in terms of an interrogative as the **maximum score of answer candidates that associated with the interrogative**,
- and finally define the **score of document** as the **summation of the sub-scores** over all interrogatives.
- Since we have two types of scores of ACs, namely weighted scores and raw scores, two scoring strategies, **Strategy 1** (**weighted score**) and **Strategy 2** (**raw score**), are prepared, respectively.

Algorithm 3.4: ScoreDocs($ACs, Strategy$)

comment: returns a set of tuples of $\langle D, S\rangle$, where S is the score of document D.

procedure Docs(ACs)
  comment: returns a set of all documents appeared in ACs.
  return $(\{D\})$

procedure Interrogs(ACs)
  comment: returns a set of all interrogatives appeared in ACs.
  return $(\{Interrogative\})$

procedure ScoreDoc1(D, ACs)
  return $(\sum_{i \in \text{INTERROGS}(ACs)} \max_{\langle D,i,AC,Sr,Sw\rangle \in ACs} Sw)$

procedure ScoreDoc2(D, ACs)
  return $(\sum_{i \in \text{INTERROGS}(ACs)} \max_{\langle D,i,AC,Sr,Sw\rangle \in ACs} Sr)$

main
  $DSs \leftarrow \{\}$
  for each $D \in$ Docs(ACs)
  do {
    if $Strategy == 1$
      then $DSs \leftarrow DSs \bigcup \{\langle D, \text{SCOREDOC1}(D, ACs)\rangle\}$
    else if $Strategy == 2$
      then $DSs \leftarrow DSs \bigcup \{\langle D, \text{SCOREDOC2}(D, ACs)\rangle\}$

Experimental Results (1)

Settings

- We conducted four runs shown in Table 3.
- The difference among the runs is due to:
  - Scoring strategy and
  - Parameter settings of the question-answering system.

Table 1: Description of system parameters

a:	Number of answers to be searched.
d:	Number of documents to be retrieved.
ppd:	Maximum number of passages retrieved from one document.
p:	Number of passages to be considered in the retrieved documents.
pwin:	Number of sentences in one passage.

Table 2: Common parameter settings of the question-answering system

d	pwin	ppdoc
250	3	3

Table 3: Submitted runs

Run ID	Strategy	a	p
FORST-JA-JA-01-D	1 (weighted score)	10	30
FORST-JA-JA-02-D	2 (raw score)	10	30
FORST-JA-JA-03-D	1 (weighted score)	20	60
FORST-JA-JA-04-D	2 (raw score)	20	60

Experimental Results (2)

Overall evaluation

- Strategy 2 (raw score) is superior to Strategy 1 (weighted score).
  - In GeoTime retrieval, documents with answer candidats for both ‘when’ and ‘where’ are imortant.
  - The weighted score scheme may give wrongly high value to documents that have only one kind of answer candidates.
  - We need more detailed analysis.
- The parameter settings of question answering do not seriously affect to the effectiveness in GeoTime retrieval.
- There are no statistically significant difference among runs according to the Wilcoxon matched pairs signed rank sum test.

Table 4: Mean of each evaluation metric

Run ID	mean AP	mean Q	mean nDCG
FORST-JA-JA-01-D	0.233	0.259	0.332
FORST-JA-JA-02-D	0.286	0.284	0.372
FORST-JA-JA-03-D	0.206	0.238	0.324
FORST-JA-JA-04-D	0.276	0.287	0.377

Table 3: Submitted runs

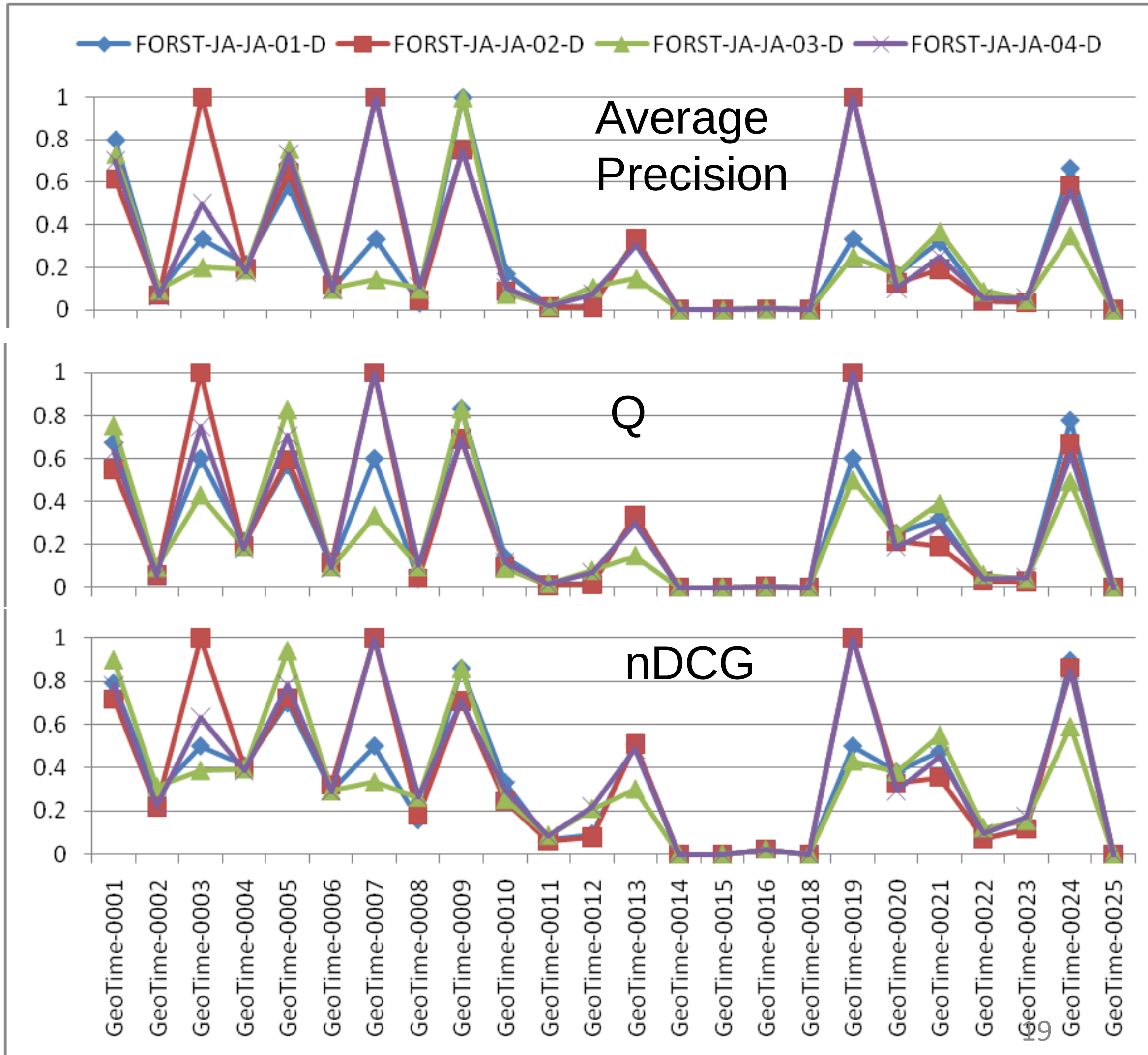
Run ID	Strategy	a	p
FORST-JA-JA-01-D	1 (weighted score)	10	30
FORST-JA-JA-02-D	2 (raw score)	10	30
FORST-JA-JA-03-D	1 (weighted score)	20	60
FORST-JA-JA-04-D	2 (raw score)	20	60

a: Number of answers to be searched.
p: Number of passages

Experimental Results(3)

Per-topic evaluation

- There are some topics that cannot be appropriately handled by our method.
  - The method lacks in robustness in terms of variety of queries.
- Especially, the question decomposition module failed to decompose GeoTime questions in some cases as follows.



Failure analysis (1)

in question decomposition

- Failures because of lack of patterns.
  - GeoTime-0010:いつITERの設置とその建設予定地が決定しましたか？
    - (When was the decision made on siting the ITER and where is it to be built?)
  - GeoTime-0018: 2002年に合衆国がある国に侵攻したのは何月何日でしたか？
    - (What date was a country was invaded by the United States in 2002?)

Failure analysis (2)
in question decomposition

- Failures because the given questions consist of two separate questions.
  - They cannot be handled by our question-answering systems.
  - We need a system for information access dialogue (IAD) task like NTCIR-5 QAC.
  - GeoTime-0015:どのアメリカンフットボールチームが、2002年のスーパーボウルで優勝しましたか、また、試合はどこで開催されましたか？
    - (What American football team won the Superbowl in 2002, and where was the game played?)
  - GeoTime-0020:もっとも最近に国連に加盟したのはどの国ですか、また、加盟したのはいつですか？
    - (What country is the most recent to join the UN and when did it join?)
  - GeoTime-0023:欧州連合の最大の規模拡大が生じたのはいつですか、また、どの国がメンバーになりましたか？
    - (When did the largest expansion of the European Union take place, and which countries became members?)

Conclusion

- We proposed a method of GeoTime information retrieval based on question decomposition and question answering.
- We demonstrated that the proposed method is able to accept GeoTime questions and retrieve relevant documents to some extent.
- However, there is still room to improve the effectiveness of retrieval.
  - Question decomposition, etc.

Thank you very much!!