Overview of NTCIR-9 RITE

(Recognizing Inference in TExt)

RITE
Recognizing
Inference in
TExt@NTCIR9

Hideki Shima[*] Hiroshi Kanayama[**]
Cheng-Wei Lee[#] Chuan-Jie Lin[†]
Teruko Mitamura[*] Yusuke Miyao[‡]
Shuming Shi[+] Koichi Takeda[**]

[*]Carnegie Mellon University, USA [**]IBM Research – Tokyo, Japan
[#]Academia Sinica, Taiwan [†]National Taiwan Ocean University, Taiwan
[‡]National Institute of Informatics, Japan [+]Microsoft Research Asia, P.R. China

NTCIR-9 Workshop. Dec 8th, 2011

Outline

- Task Definition
- Task Organization Efforts
- Formal Run Results
- Review of Participant Works
- Conclusion

TASK DEFINITION

Overview of RITE

RITE is a generic benchmark task that addresses common semantic inference needs in various NLP/Information Access research areas.

- t_1: Taro was born in Tokyo.
- t_2: Taro was born in Japan.

- t_1: Yasunari Kawabata won the Nobel Prize
  in Literature for his novel “Snow Country”
- t_2: Yasunari Kawabata is the writer of “Snow Country”

Given t_1, can a computer infer that t_2 is most likely true?

Target languages: Japanese, Simplified Chinese, Traditional Chinese

Motivation

Information Access applications

- Question Answering; Information Retrieval; Information Extraction; Text Summarization; Automatic evaluation for Machine Translation, Text Summarization, Complex Question Answering

Success in previous shared tasks

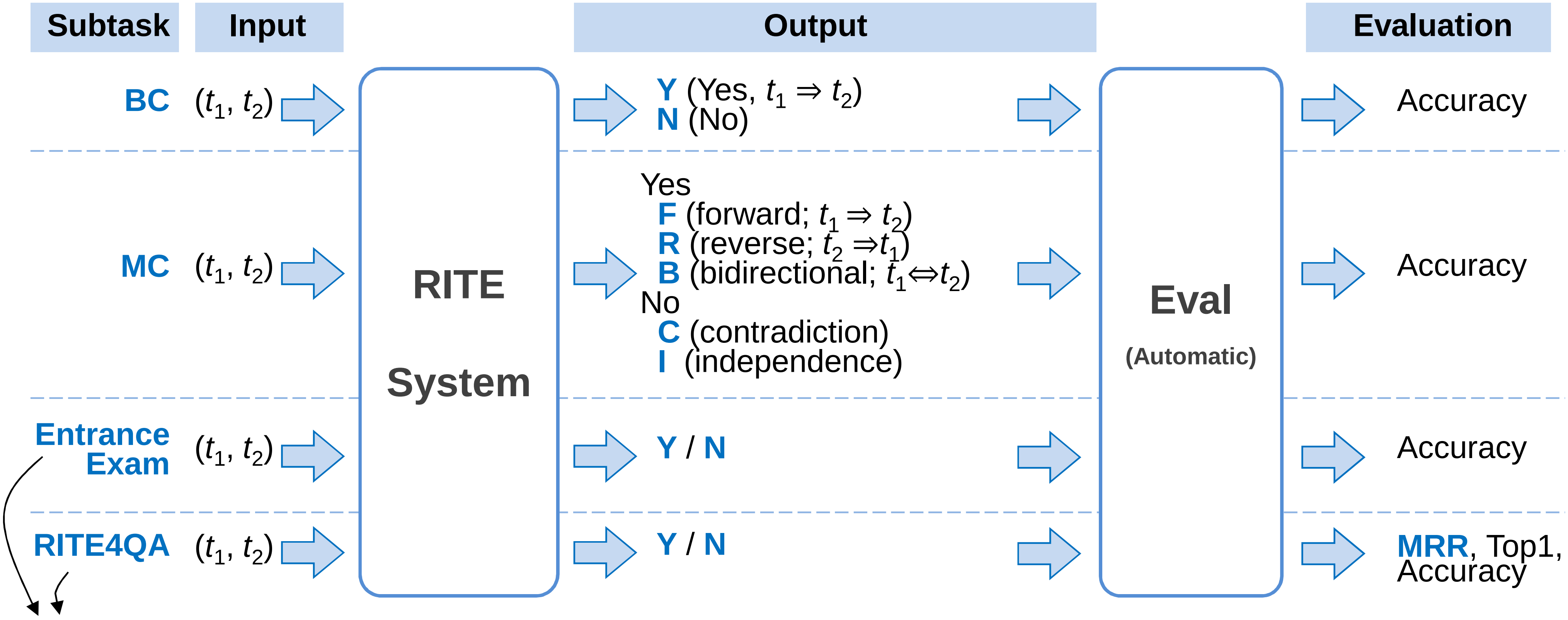
- TREC, CLEF and NTCIR are modern examples of the “Cranfield evaluation paradigm” (Voorhees, 2002)
  - Abstraction of a real Information Access (IA) task is done in a system-centric lab evaluation approach to avoid affects from uncontrollable variables.
  - We’d like to abstract away complexities further and focus on a key semantic processing need that commonly exists in various IA problems

RITE (Recognizing Inference in TExt)

t_1: Yasunari Kawabata won the Nobel Prize in Literature for his novel “Snow Country”.

t_2: Yasunari Kawabata is the writer of “Snow Country”.

Does t_1 entail (infer) t_2?



RITE System

Eval (Automatic)

Subtask	Input	Output	Evaluation
BC	(t_1, t_2)	**Y** (Yes, $t_1 \Rightarrow t_2$) **N** (No)	Accuracy
MC	(t_1, t_2)	Yes **F** (forward; $t_1 \Rightarrow t_2$) **R** (reverse; $t_2 \Rightarrow t_1$) **B** (bidirectional; $t_1 \Leftrightarrow t_2$) No **C** (contradiction) **I** (independence)	Accuracy
Entrance Exam	(t_1, t_2)	**Y** / **N**	Accuracy
RITE4QA	(t_1, t_2)	**Y** / **N**	**MRR**, Top1, Accuracy

application-oriented

Definition of Textual Entailment

- The premise t_1 entails the hypothesis t_2 if a human (with a common knowledge) reading t_1 would infer that t_2 is *most likely* true.

- Note that *logical entailment* and *textual entailment* are different.
  - – t_1: The temperature is only 5 degrees outside.
  - – t_2: It's cold outside.

Binary-class (BC) Subtask

Development process (JA)

1) RITE organizers proposed a small set of sample dataset on an online collaborative spreadsheet to participants.
2) Participants gave feedbacks to the samples and proposed additional samples.
3) Ten college students studied general trends from the sample, and then built training/test data. Sentences were collected from Mainichi newspaper corpus in (somewhat random) various topics. Minimum post-edits are allowed. Controlled to be difficult to solve.
4) Four students independently annotated labels on the collected pairs.
5) Pairs with agreement < 3 are discarded. Inter-annotator agreement: 0.829 (Fleiss' Kappa).
6) Organizers randomly split the dataset into dev and test, with label distribution balanced.

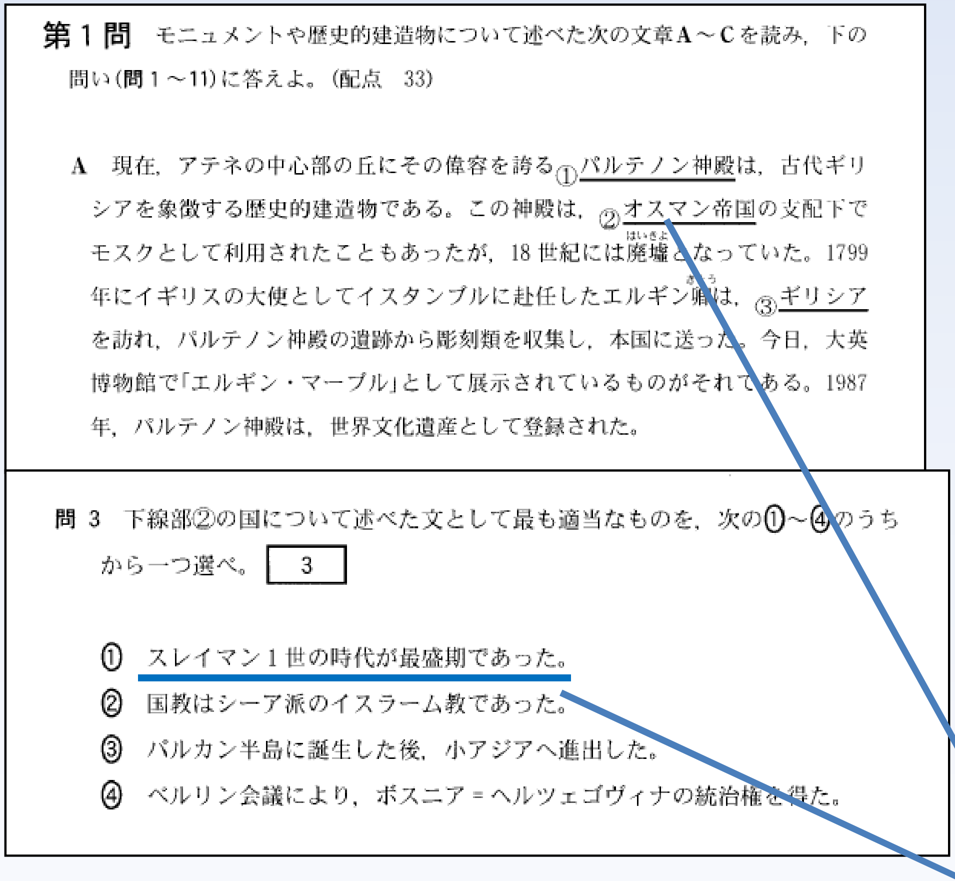
Multi-class (MC) Subtask

- A system needs to classify a pair into one of five categories considering entailment direction, paraphrase and contradiction.
- Output labels
  - **F**: forward entailment (t_1 entails t_2 AND t_2 does not entail t_1).
  - **R**: reverse entailment (t_2 entails t_1 AND t_1 does not entail t_2).
  - **B**: bidirectional entailment (t_1 entails t2 AND t_2 entails t_1).
  - **C**: contradiction (t_1 and t_2 contradict, or cannot be true at the same time).
  - **I**: independence (otherwise)
- Motivation: in Text Summarization, knowing textual entailment direction helps to choose one from multiple summary candidate sentences. Contradiction detection is also useful for finding contradicting opinions.
- Sentence length are controlled.

Entrance Exam Subtask

Entrance exam problem

National Center Test for University Admission (*Daigaku Nyushi Center Shiken*)



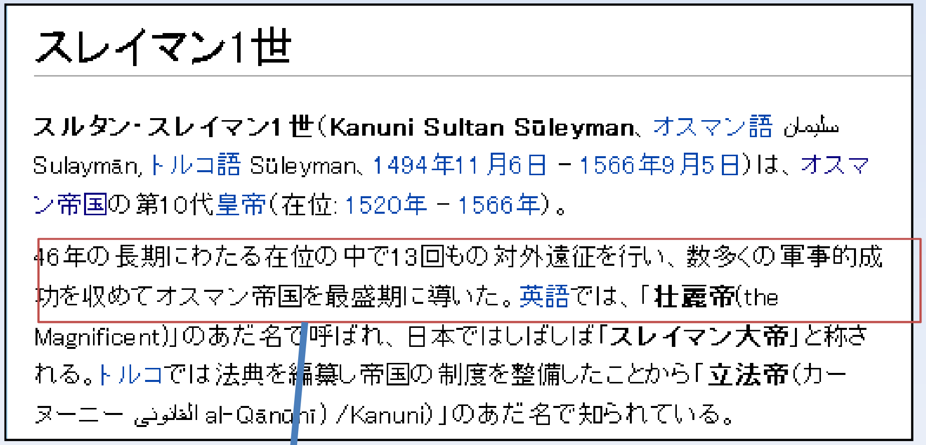
第１問　モニュメントや歴史的建造物について述べた次の文章Ａ～Ｃを読み，下の問い（問１～11）に答えよ。（配点　33）

Ａ　現在，アテネの中心部の丘にその偉容を誇る①パルテノン神殿は，古代ギリシアを象徴する歴史的建造物である。この神殿は，②オスマン帝国の支配下でモスクとして利用されたこともあったが，18 世紀には廃墟となっていた。1799 年にイギリスの大使としてイスタンブルに赴任したエルギン卿は，③ギリシアを訪れ，パルテノン神殿の遺跡から彫刻類を収集し，本国に送った。今日，大英博物館で「エルギン・マーブル」として展示されているものがそれである。1987 年，パルテノン神殿は，世界文化遺産として登録された。

問　３　下線部②の国について述べた文として最も適当なものを，次の①～④のうちから一つ選べ。 3

① スレイマン１世の時代が最盛期であった。
② 国教はシーア派のイスラーム教であった。
③ バルカン半島に誕生した後，小アジアへ進出した。
④ ベルリン会議により，ボスニア＝ヘルツェゴヴィナの統治権を得た。

Wikipedia



スレイマン1世

スルタン・スレイマン1世（**Kanuni Sultan Süleyman**、オスマン語 سليمان Sulaymān, トルコ語 Süleyman、1494年11月6日 － 1566年9月5日）は、オスマン帝国の第10代皇帝（在位: 1520年 － 1566年）。

46年の長期にわたる在位の中で13回もの対外遠征を行い、数多くの軍事的成功を収めてオスマン帝国を最盛期に導いた。英語では、「**壮麗帝**（the Magnificent）」のあだ名で呼ばれ、日本ではしばしば「**スレイマン大帝**」と称される。トルコでは法典を編纂し帝国の制度を整備したことから「**立法帝**（カーヌーニー القانوني al-Qānūnī）/Kanuni）」のあだ名で知られている。

t_1: スレイマン1世は数多くの軍事的成功を収めてオスマン帝国を最盛期に導いた. (Suleiman I contributed in a lot of military successes and led the Ottoman Empire to its peak.

t_2: オスマン帝国ではスレイマン1世の時代が最盛期であった.　(The Ottoman Empire’s peak was during the reign of Suleiman I).

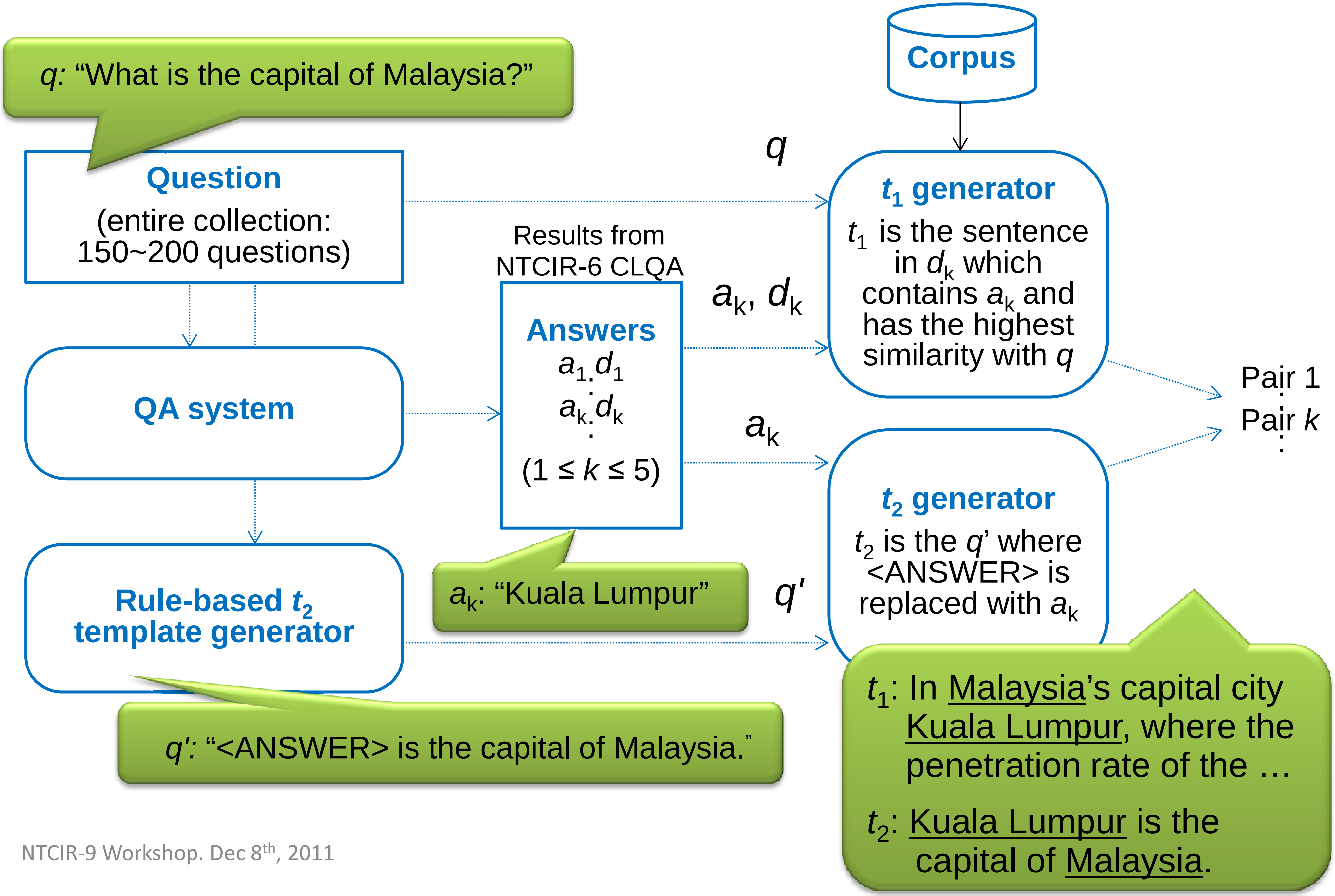
Entrance Exam Subtask

- Covers wide range subjects where different problem arises
  - Domestic and World History, Politics, Economy, and Modern Society
  - In History, geo-temporal reasoning may be required.
- Source difference in t_1-t_2 causes vocabulary mismatch (e.g. “bin Laden” and “bin Ladin”)
- Has a natural distribution of linguistic phenomena as seen in an exam-solver application
- Social impact - can wow the public

RITE4QA Subtask

- Can a RITE system rank a set of unordered answer candidates in QA?
- The dataset is created fully automatically from Japanese monolingual data at NTCIR-6 CLQA (Factoid Question Answering)
  - t1: answer-candidate-bearing sentence
  - t2: a question in an affirmative form
- A system is required to generate an additional confidence score used for the ranking process
- Also has a natural distribution of linguistic phenomena
- Uses QA evaluation metrics for result comparability

RITE4QA Subtask



q: “What is the capital of Malaysia?”

Question
(entire collection: 150~200 questions)

QA system

Rule-based t_2 template generator

q': “<ANSWER> is the capital of Malaysia.”

Results from NTCIR-6 CLQA

Answers
a_1, d_1
⋮
a_k, d_k
⋮
($1 \le k \le 5$)

a_k: “Kuala Lumpur”

Corpus

q

a_k, d_k

a_k

q'

t_1 generator
t_1 is the sentence in d_k which contains a_k and has the highest similarity with q

t_2 generator
t_2 is the q' where <ANSWER> is replaced with a_k

Pair 1
⋮
Pair k
⋮

t_1: In Malaysia’s capital city Kuala Lumpur, where the penetration rate of the …

t_2: Kuala Lumpur is the capital of Malaysia.

Dataset size

BC

	Total
JA (dev)	500
JA (test)	500
CS (dev)	407
CS (test)	407
CT (dev)	421
CT (test)	900

MC

	Total
JA (dev)	440
JA (test)	440
CS (dev)	407
CS (test)	407
CT (dev)	421
CT (test)	900

EXAM

	Total
JA (dev)	499
JA (test)	442

RITE4QA

	Total
JA (test)	964
CS (test)	682
CT (test)	682

Evaluation Metrics

- BC, MC and Entrance Exam

$$\text{Accuracy} = \frac{1}{\#\,\text{pairs}} \sum \left[\text{output label is correct}\right]$$

- RITE4QA

$$\text{Top1} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} [\text{top answer is correct}]$$

$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}$$

Comparison with Related Works

	Lang	Entailment	Entailment Direction	Paraphrase	Contradiction	Answer validation for QA
TAC RTE (2-way)	EN	X				
TAC RTE (3-way)	EN	X			X	
MSR Paraphrase Corpus	EN			X		
CLEF AVE	EN					X
Kurohashi Lab's	JA	X		(X)		
NTCIR-9 RITE	**JA** **CS** **CT**	**X**	**X**	**X**	**X**	**X**
SemEval-2012 CLTE	Cross-lingual	X	X	X		

Uniqueness of RITE

- BC, MC: Designed to be difficult so that character/word-overlap approach doesn't work.
- MC: Finer-grained classification categories
- Entrance Exam: Has potential to benchmark against a human in a problem familiar to everyone
- RITE4QA: Used QA evaluation metrics; dataset built automatically

TASK ORGANIZATION EFFORTS

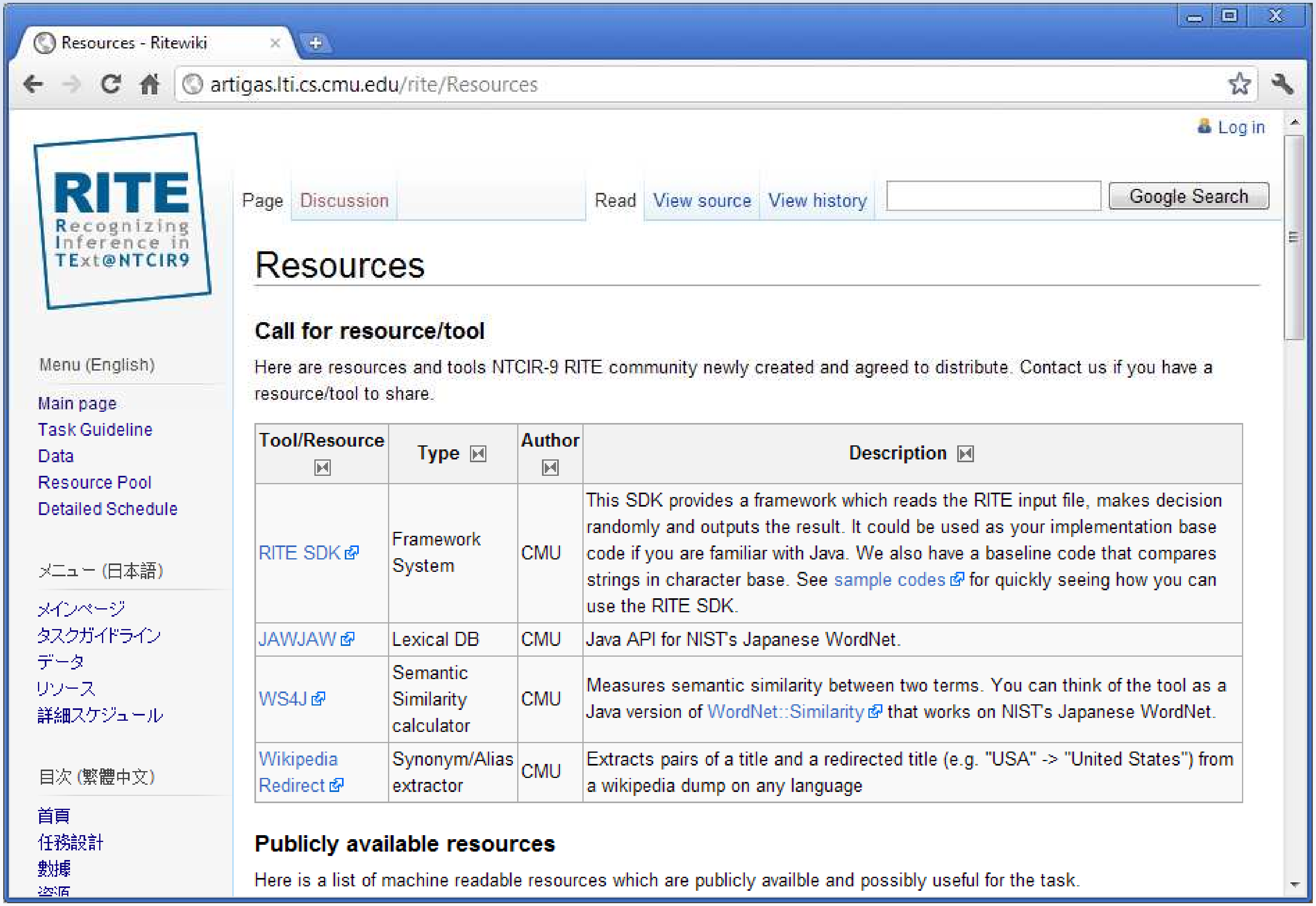
Task Organization Efforts

- Lowering Barrier to Entry
  - Provided **RITE-SDK** and **the resource pool** to help participants to quickly build a system. They can be used to improve reproducibility of a work.
- Ablation study
  - Removing one resource, tool, or algorithm at a time, and see its impact to the overall system
  - Taking advantage of automatic evaluation
  - Toward *building blocks* rather than a *black box*

Experiment result: all-but-one feature ablation [LTI]

Feature	BC		EXAM	
	Acc	Diff	Acc	Diff
All features	62.6%	N/A	68.9%	N/A
- Morpheme Overlap	61.0%	-1.6%	59.1%	-9.8%
- BE Overlap	54.2%	-8.4%	68.9%	0.0%
- Quote	61.4%	-1.2%	68.7%	-0.2%
- Polarity	59.8%	-2.8%	68.7%	-0.2%
- Quantification	62.2%	-0.4%	68.9%	0.0%
- Morpheme Diff	57.2%	-5.4%	68.7%	-0.2%

Resource Pool



RITE
Recognizing Inference in TExt@NTCIR9

Menu (English)

- Main page
- Task Guideline
- Data
- Resource Pool
- Detailed Schedule

メニュー (日本語)

- メインページ
- タスクガイドライン
- データ
- リソース
- 詳細スケジュール

目次 (繁體中文)

- 首頁
- 任務設計
- 數據
- 資源

Resources

Call for resource/tool

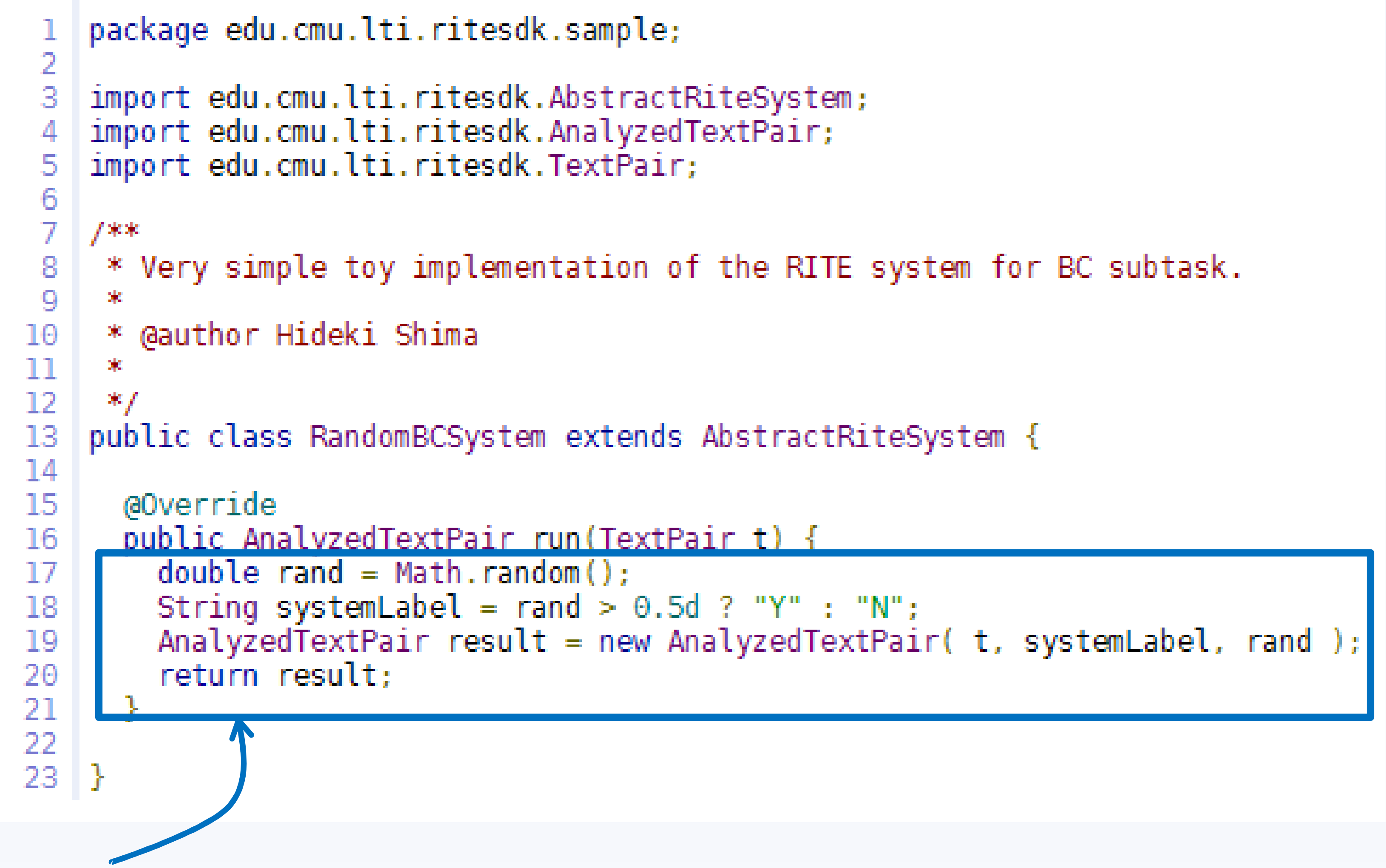
Here are resources and tools NTCIR-9 RITE community newly created and agreed to distribute. Contact us if you have a resource/tool to share.

Tool/Resource	Type	Author	Description
RITE SDK	Framework System	CMU	This SDK provides a framework which reads the RITE input file, makes decision randomly and outputs the result. It could be used as your implementation base code if you are familiar with Java. We also have a baseline code that compares strings in character base. See sample codes for quickly seeing how you can use the RITE SDK.
JAWJAW	Lexical DB	CMU	Java API for NIST's Japanese WordNet.
WS4J	Semantic Similarity calculator	CMU	Measures semantic similarity between two terms. You can think of the tool as a Java version of WordNet::Similarity that works on NIST's Japanese WordNet.
Wikipedia Redirect	Synonym/Alias extractor	CMU	Extracts pairs of a title and a redirected title (e.g. "USA" -> "United States") from a wikipedia dump on any language

Publicly available resources

Here is a list of machine readable resources which are publicly availble and possibly useful for the task.

RITE SDK



```
package edu.cmu.lti.ritesdk.sample;

import edu.cmu.lti.ritesdk.AbstractRiteSystem;
import edu.cmu.lti.ritesdk.AnalyzedTextPair;
import edu.cmu.lti.ritesdk.TextPair;

/**
 * Very simple toy implementation of the RITE system for BC subtask.
 *
 * @author Hideki Shima
 *
 */
public class RandomBCSystem extends AbstractRiteSystem {

  @Override
  public AnalyzedTextPair run(TextPair t) {
    double rand = Math.random();
    String systemLabel = rand > 0.5d ? "Y" : "N";
    AnalyzedTextPair result = new AnalyzedTextPair( t, systemLabel, rand );
    return result;
  }

}
```

Participants can simply focus on core part; the SDK takes care of the rest (e.g. data IO, evaluation).

Task Organization Efforts (Cont’d)

- Participant involvement
  - Sample data collaboratively built by participants and organizers
  - Mailing list provided for discussion

- Controlling the difficulty level by running a baseline
  - Assumption: Character-level overlap baseline performance correlates with difficulty of task
  - The entrance exam subtask dataset has been built given baseline feedbacks

FORMAL RUN RESULTS

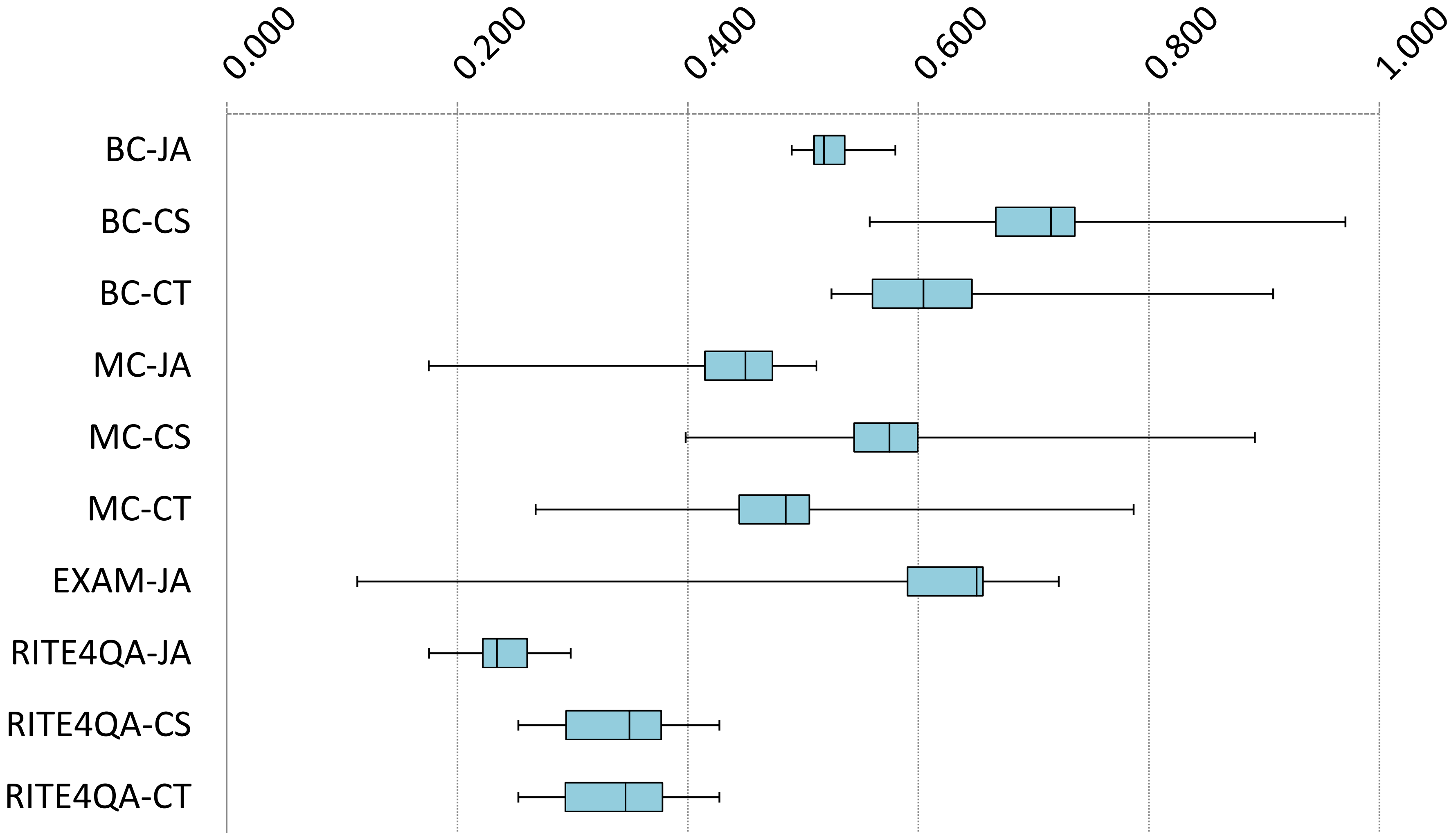
Formal Run Participation

Number of submitted runs

Subtask	Language			Total
	JA	CS	CT	
BC	24	33	32	**89**
MC	10	27	22	**59**
Entrance Exam	18	-	-	**18**
RITE4QA	13	17	16	**46**
Total	**65**	**77**	**70**	**212**

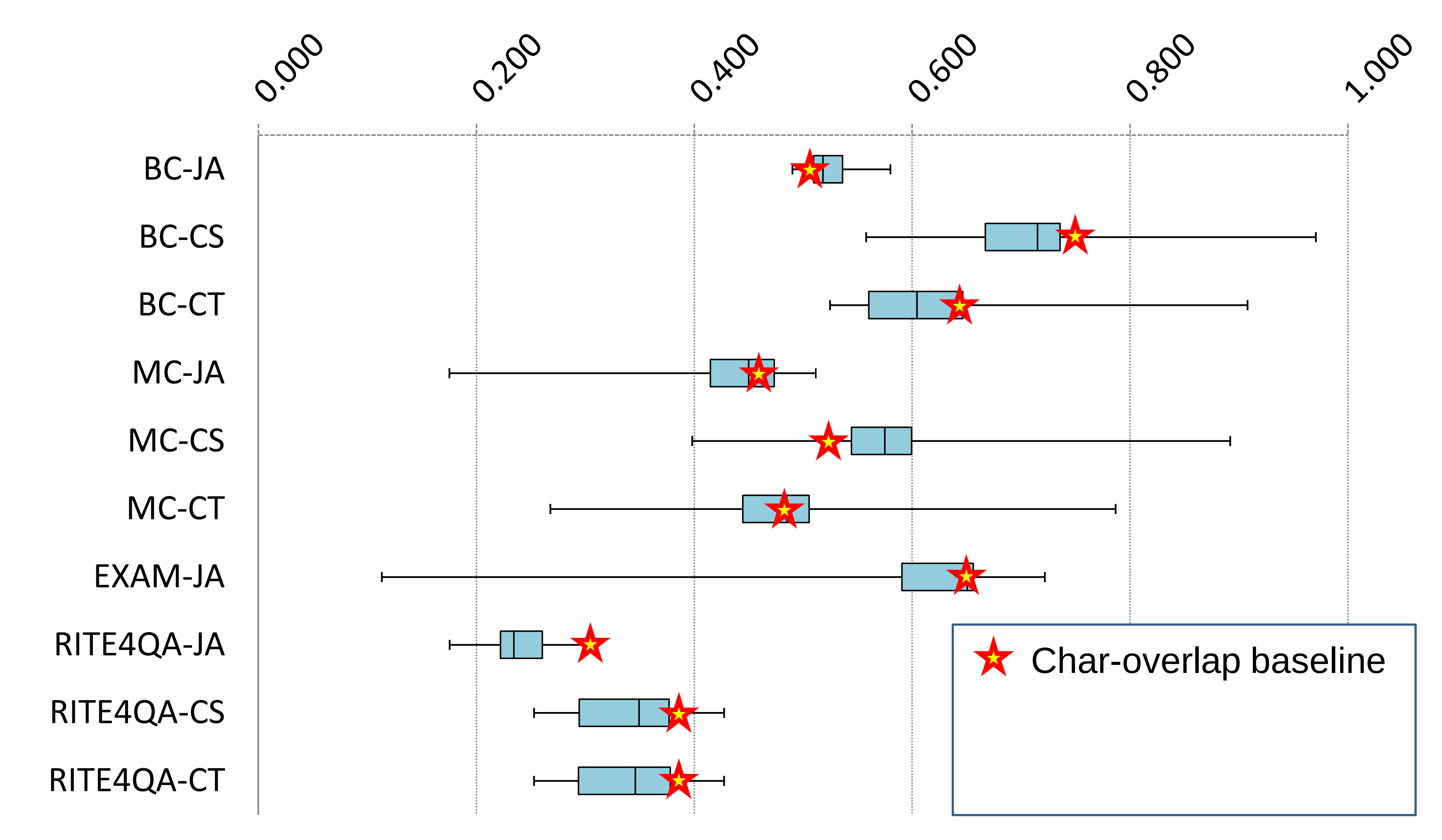
- 24 active participants from 5 countries

Score Distribution



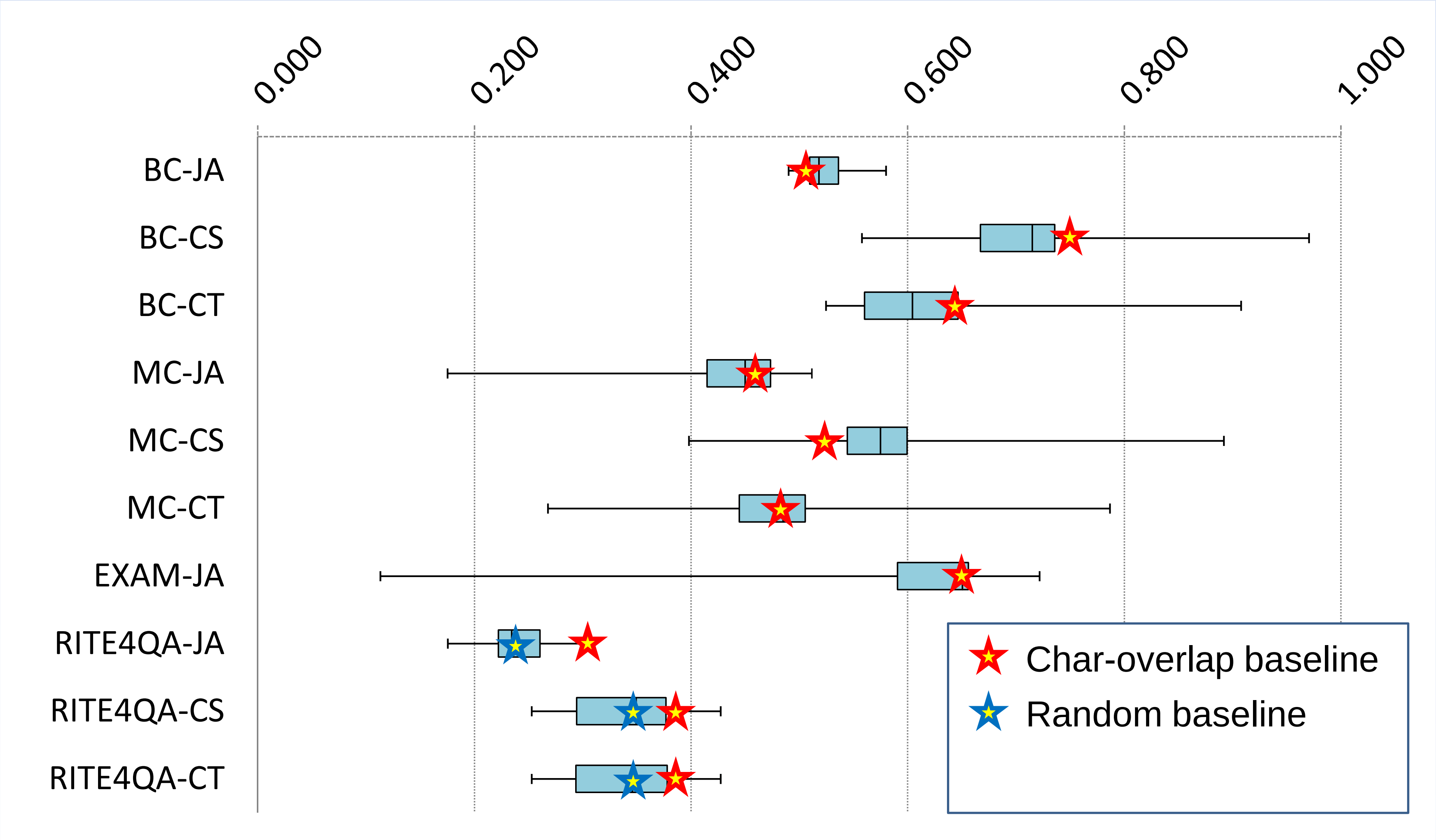
Percentiles: 0, 25, 50, 75, 100%

Score Distribution



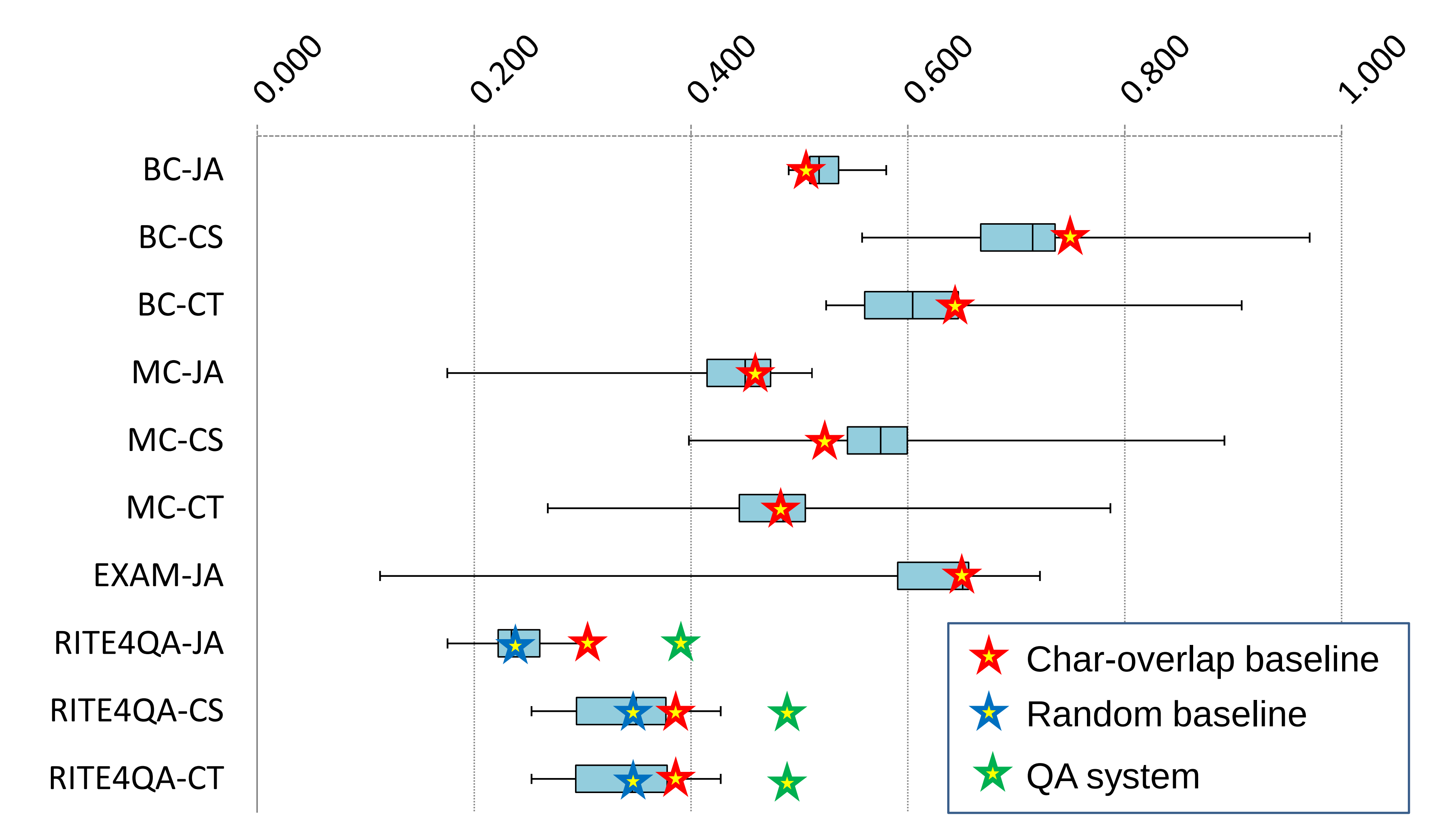
Percentiles: 0, 25, 50, 75, 100%

Score Distribution



Percentiles: 0, 25, 50, 75, 100%

Score Distribution



Percentiles: 0, 25, 50, 75, 100%

Result Highlights (BC)

JA

Run	Accuracy
JAIST-01	0.5800
JAIST-02	0.5660
JAIST-03	0.5520
NTTCS-03	0.5480
LTI-03	0.5460
LTI-02	0.5420
LTI-01	0.5340
NTTCS-01	0.5320
IBM-02	0.5260
FX-02	0.5240
Average	*0.5233*
Baseline (char overlap)	*0.5160*

CS

Run	Accuracy
UIOWA-01	0.9705
UIOWA-03	0.9631
UIOWA-02	0.9361
ICRC_HITSZ-03	0.7764
FudanNLP-02	0.7617
ICRC_HITSZ-02	0.7568
FudanNLP-01	0.7469
WHUTE-03	0.7371
NTU-01	0.7346
WHUTE-02	0.7322
WUST-01	0.7248
NTU-02	0.7224
NTU-03	0.7199
ZSWSL-01	0.7199
IASLD-01	0.7150
ICL-01	0.7150
Average	*0.7135*
Baseline (char overlap)	*0.7617*

CT

Run	Accuracy
UIOWA-01	0.9078
UIOWA-02	0.8844
IASLD-03	0.6611
IASLD-02	0.6533
III_CYUT_NTHU-02	0.6500
IASLD-01	0.6478
NTOUA-02	0.6422
Average	*0.6212*
Baseline (char overlap)	*0.6667*

Showing runs above the average.

Result Highlights (MC)

JA

Run	Accuracy
IBM-02	0.5114
KYOTO-03	0.4841
KYOTO-02	0.4795
IBM-01	0.4545
NTTCS-03	0.4523
NTTCS-01	0.4477
IBM-03	0.4455
Average	*0.4124*
Baseline *(char overlap)*	*0.4682*

CS

Run	Accuracy
UIOWA-01	0.8919
UIOWA-02	0.8919
UIOWA-03	0.8870
ICRC_HITSZ-03	0.6413
ICRC_HITSZ-02	0.6241
ZSWSL-02	0.6192
WHUTE-02	0.6093
Average	*0.5971*
Baseline *(char overlap)*	*0.5315*

CT

Run	Accuracy
UIOWA-01	0.7867
UIOWA-02	0.7744
UIOWA-03	0.7244
MCU-01	0.5356
IMTKU-01	0.5222
IMTKU-02	0.5067
Average	*0.5019*
Baseline *(char overlap)*	*0.4885*

Result Highlights (EXAM)

JA

Run	Accuracy
IBM-01	0.7217
TU-02	0.7183
TU-03	0.7042
IBM-02	0.6742
LTI-03	0.6674
KYOTO-02	0.6561
KYOTO-03	0.6561
LTI-02	0.6538
JAIST-02	0.6516
JAIST-03	0.6516
TU-01	0.6493
JAIST-01	0.6222
LTI-01	0.6018
KYOTO-01	0.5928
Average	*0.5863*
Baseline (char overlap)	*0.6516*

Result Highlights (RITE4QA)

JA

Run	Acc	MRR
LTI-03	0.6753	0.2982
JAIST-01	0.5602	0.2765
JAIST-03	0.6940	0.2731
JAIST-02	0.6763	0.2604
LTI-02	0.6411	0.2563
JUCS-01	0.5954	0.2490
Average	*0.6148*	*0.2424*
Baseline1 (char overlap)	*0.4180*	*0.3192*
Baseline2 (all yes)	*0.1100*	*0.1657*
Baseline3 (random)	*0.5000*	*0.2320*
Baseline4 (QA system)	*0.1100*	*0.3917*
Oracle	*1.0000*	*0.5326*

CS

Run	Acc	MRR
UIOWA-01	0.9010	0.4272
IMTKU-02	0.4090	0.3998
WHUTE-02	0.4876	0.3979
WHUTE-01	0.3886	0.3773
IMTKU-03	0.4716	0.3768
IMTKU-01	0.3319	0.3744
ICL-01	0.3231	0.3545
ICRC_HITSZ-01	0.6390	0.3520
WHUTE-03	0.3275	0.3494
ICRC_HITSZ-03	0.7293	0.3398
Average	*0.5192*	*0.3367*

CT

Run	Acc	MRR
UIOWA-01	0.9010	0.4272
IMTKU-03	0.4003	0.3992
NTOUA-03	0.6346	0.3824
NTOUA-01	0.5459	0.3803
IMTKU-01	0.3246	0.3772
IMTKU-02	0.3392	0.3736
NTOUA-02	0.5124	0.3572
ICRC_HITSZ-01	0.6390	0.3520
ICRC_HITSZ-03	0.7293	0.3398
Average	*0.5514*	*0.3352*
Baseline1 (char overlap)	*0.2317*	*0.3844*
Baseline2 (all yes)	*0.1906*	*0.2378*
Baseline3 (random)	*0.5000*	*0.3454*
Baseline4 (QA system)	*0.1906*	*0.4852*
Oracle	*1.0000*	*0.5906*

REVIEW OF PARTICIPANT WORKS

Summary of Ideas Explored

- Machine learning [many teams]
- Predicate-argument matching [KYOTO, LTI, NTTCS, SITLP, WHUTE, ZSWSL]
- Bilingual enrichment [JAIST, JUCS]
- Crowdsource-driven rule-based approach [UIOWA]
- Inference on Lexical Functional Grammar [FX]
- Alignment [TU]

Ideas NOT Explored…

- Monolingual Machine Translation

Summary of Features Explored

- Overlap (character, word, bigram, trigram, head-word, POS, NE, numerical expression)
- String Similarity (Jaro distance, Jaro–Winkler distance, Jaccard Coefficient, Chebyshev Distance, Dice Coefficient, Manhattan Distance, Longest Common Subsequence, Cosine similarity, Levenshtein Edit Distance, BLEU score)
- Structural matching (predicate-argument matching, subtree matching, Tree Edit Distance)
- Verbs number mismatch
- Antonyms
- Negation / Polarity matching
- Temporal matching (5% improvement in EXAM [IBM])
- Quantification (*all, only, most , every…*)
- Quote (something just said might not be true…)

: : :

Summary of Resources Explored

- Alexandria Digital Library
- Baidupedia
- CC-CEDICT
- GoiTaikei
- HowNet
- Hudong Wiki
- NAIST Japanese Dictionary
- REIKAI-SHOGAKU
- Tongyici Cilin
- Wikipedia
- WordNet (Japanese, Chinese)

: : :

Oral Presentations

- A Machine Learning based Textual Entailment Recognition System of JAIST Team for NTCIR9 RITE
  - Quang Nhat Minh Pham, Le Minh Nguyen, and Akira Shimazu (Japan Advanced Institute of Science and Technology, Japan)
- Predicate-argument Structure based Textual Entailment Recognition System of KYOTO Team for NTCIR9 RITE
  - Tomohide Shibata and Sadao Kurohashi (Kyoto University, Japan)
- UIOWA at NTCIR-9 RITE: Using the Power of the Crowd to Establish Inference Rules
  - Christopher G. Harris (The University of Iowa, USA)
- ICRC_HITSZ at RITE: Leveraging Multiple Classifiers Voting for Textual Entailment Recognition
  - Yaoyun Zhang, Jun Xu, Chenlong Liu, Xiaolong Wang, Ruifeng Xu, Qingcai Chen, Xuan Wang, Yongshuai Hou and Buzhou Tang (Harbin Institute of Technology, P.R.China)

CONCLUSION

Conclusion

- Best runs were able to outperform the strong character-overlap baseline
- Diverse techniques were explored – e.g. supervised machine learning, crowdsource-driven rule-based approach, predicate-argument matching, bilingual enrichment, LFG-based inference etc.
- Simple core challenge allowed participants to focus on developing textual entailment components that are potentially applicable to various IA problems
- Fast automatic evaluation enabled participants to report additional experimental results (e.g. ablation study).
- Attracted many participants including new comers as a first NTCIR task – indicating there's a research need.

RITE was successful as a first attempt in NTCIR!

THANK YOU!