Detecting Contradiction in Text by Using Lexical Mismatch
and Structural Similarity

Daniel Andrade, Masaaki Tsuchida, Takashi Onishi, Kai Ishikawa

Knowledge Discovery Research Laboratories, NEC Corporation, Japan

Overview of the features used by our system

- Clipped precision of single characters.

- Match of temporal expressions. Temporal expressions are normalized using normalizeNumexp.

- Tree edit distance considering free word-order of siblings in dependency trees.
  （リンゴを八百屋で買った。＝　八百屋でリンゴを買った。）
  (buy apple at shop = buy at shop apple)

- Degree of contradiction.　　Main focus of this talk

Empowered by Innovation　**NEC**

Detecting contradiction – Main Idea

Observation:
Contradiction on **lexical level** does not necessarily imply contradiction on sentence level.

Examples:　(Taro gives flowers to Hanko)
T1：「太郎さんは花子さんに花を**あげた。**」
T2：「花子さんは太郎さんに花を**もらった。**」　　　　　no contradiction
(Hanko receives flowers from Taro)

(In year 2008, Taro went to Tokyo)
T1：「**2008年**太郎さんは東京に行った。」
T2：「**2009年**太郎さんは花子さんに花をあげた。」　　　no relation
(In year 2009, Taro gave flowers to Hanako)

(In year 2008, Taro went to Tokyo.)
T1：「2008年太郎さんは**東京**に行った。」　　　(possible) contradiction
T2：「2008年太郎さんは**京都**に行った。」
(In year 2008, Taro went to Kyoto.)

Idea:
The more similar the tree structure of T1 and T2 is, the more likely that lexical contradiction propagates to sentence contradiction as a whole.

Detecting contradiction – Processing Steps

1. **Split complex sentences into simple sentences**
    that contain only one predicate each.

2. **Calculate the minimum cost alignment for each pair of simple sentences from T1 and T2.**
    => Alignment-Cost Matrix

3. **Find minimum cost alignment**
    using Alignment-Cost Matrix between each simple sentence in T1 and T2.

4. **Calculate degree of lexical contradiction**
    between each minimum aligned simple sentence.

5. **Calculate the degree of contradiction on sentence level between T1 and T2**
    using the alignment costs and degree of lexical contradiction from previous steps.

1. Split complex sentence into simple sentences

1.1. Use SynCha to detect predicates and their arguments.

1.2 Split into simple sentence such that each simple sentence contains exactly one predicate and its arguments.

Example:

(Taro went to the supermarket and bought Apples）
"太郎はスーパーに行って、リンゴを買った。"

(Taro goes to supermarket）　　　　　　　　　　　　(Taro buys apple.）
＝＞　"太郎はスーパーに行く。"　、　"太郎はリンゴを買う。"

2. Calculate alignment costs between simple sentences.

▊ Calculate alignment cost for each pair of simple sentence from T1 and T2 using the following definition:

Words in T1 and T2	Alignment costs
same	0
Synonym, Antonym	low (e.g. 0.2)
Related (e.g. same parent in WordNet)	middle (e.g. 0.8)
otherwise	high (e.g. 1.0)

▊ Example:
Simple Sentence from T1: "太郎はリンゴを買う。" (Taro buys apple)
Simple Sentence from T2: "太郎はリンゴを売る。" (Taro sells apple)

「太郎は」 (Taro) ⇔ 「太郎は」 (Taro) alignment costs = 0.0 (same base form)

「リンゴを」(apple) ⇔ 「リンゴを」(apple) alignment costs = 0.0 (same base form)

「買う」(buy) ⇔ 「売る」(sell) alignment costs = 0.2 (can be antonyms)

=> Alignment cost between the two simples sentences = 0.2

Empowered by Innovation **NEC**

3. Find minimum cost-alignment of simple sentences

Using the alignment-cost matrix calculated from the previous step we find the best (minimum-cost) alignment between the simple sentences using the Hungarian Algorithm.

Example simple sentence from T1 and T2:

(Taro goes to supermarket)   (Taro buys apple)

T1: **太郎はスーパーに行く。" 、 "太郎はリンゴを買った。"**

Alignment cost: 0.2   Alignment cost: 1.0

T2: **"太郎はリンゴを売る。" 、 "太郎はレストランに走る。"**

(Taro sells apple)   (Taro runs to restaurant)

Empowered by Innovation NEC

4. Calculate Lexical Contradiction

Calculate lexical contradiction between aligned simple sentence pair. Lexical contradiction definition:

Words in T1 and T2	Lexical Contradiction
Antonyms, different place names	high (e.g. 1.0)
Same head but different suffixes	middle (e.g. 0.5)
otherwise	0

Example simple sentence from T1 and T2:

(Taro goes to supermarket)　　　　　　　　(Taro buys apple)

T1: 太郎はスーパーに行く。”　、　“太郎はリンゴを買う。”

Lexical contradiction: 1.0　　　　　　　　　Lexical contradiction: 0.0

T2: "太郎はリンゴを売る。"、　"太郎はレストランに走る。"

(Taro sells apple)　　　　　　　　(Taro runs to restaurant)

5. Calculate Degree of contradiction between T1 and T2

Degree of contradiction =

Total lexical contradiction

$$\frac{\sum_{all\ aligned\ simple\ sentences\ (s1,s2)} lexical\ contradiction\ in\ (s1,s2)}{\sum_{all\ aligned\ simple\ sentences\ (s1,s2)} alignment\ costs\ of\ (s1,s2) + d}$$

Total alignment costs

some constant
(here set to 10)

Example from previous two slides:

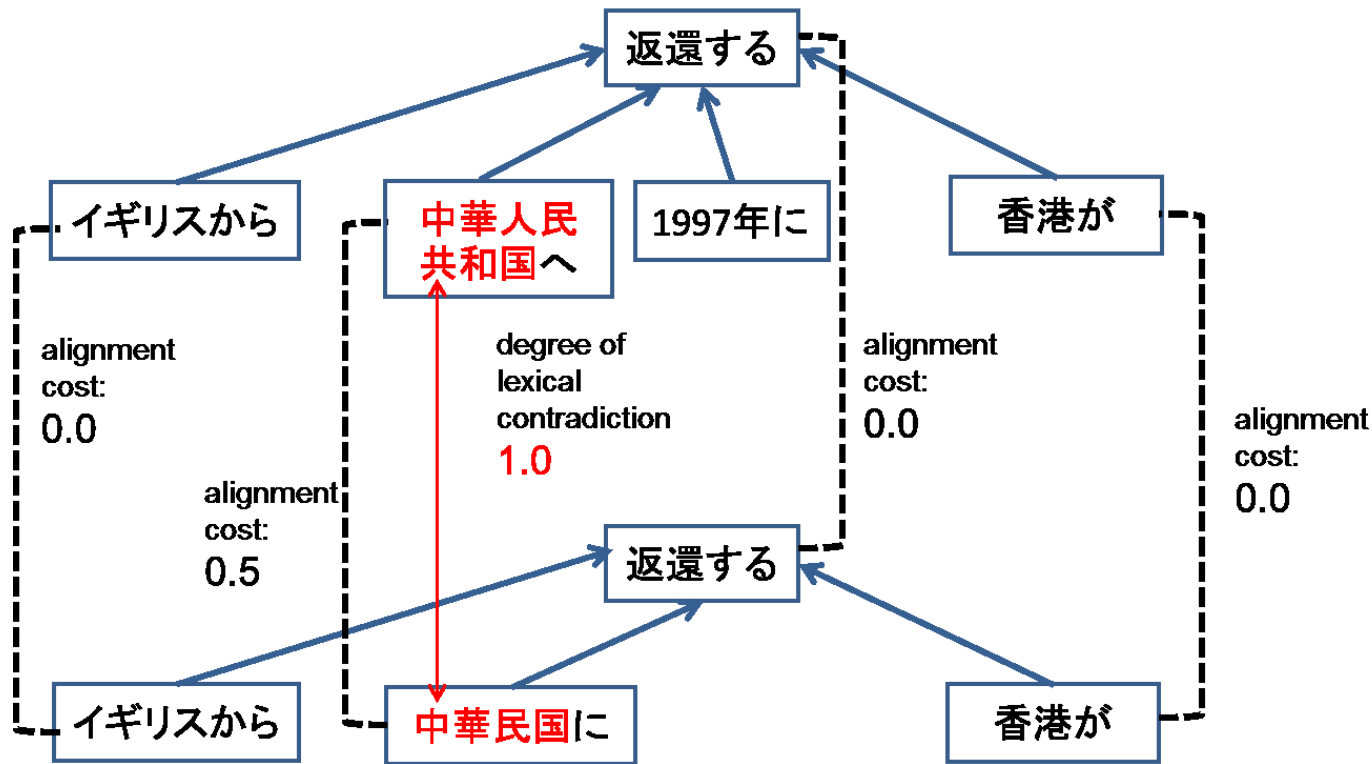$$degree\ of\ contradiction = \frac{1.0 + 0.0}{0.8 + 1.0 + 10} = 0.085$$

Empowered by Innovation   NEC

Example from the training data – **Low Alignment Cost and High Lexical Contradiction**

<pair id="351" label="N">
<t1>1997 **年に香港がイギリスから中華人民共和国へ返還された。**</t1>
<t2>**イギリスから中華民国に香港が返還された。**</t2>

T11: 1997年に香港がイギリスから中華人民共和国へ返還された。

返還する

イギリスから   中華人民共和国へ   1997年に   香港が

alignment cost: 0.0

degree of lexical contradiction 1.0

alignment cost: 0.0

alignment cost: 0.0

alignment cost: 0.5

返還する

イギリスから   中華民国に   香港が

T21: イギリスから中華民国に香港が返還された。

Alignment Costs = 0.5 (bunsetsu alignment)
+ 0.0 (node deletion T21) = 0.5

Lexical Contradiction = 1.0 (different place names)

=> **Contradiction T1 and T2 = 0.10**

Empowered by Innovation   NEC
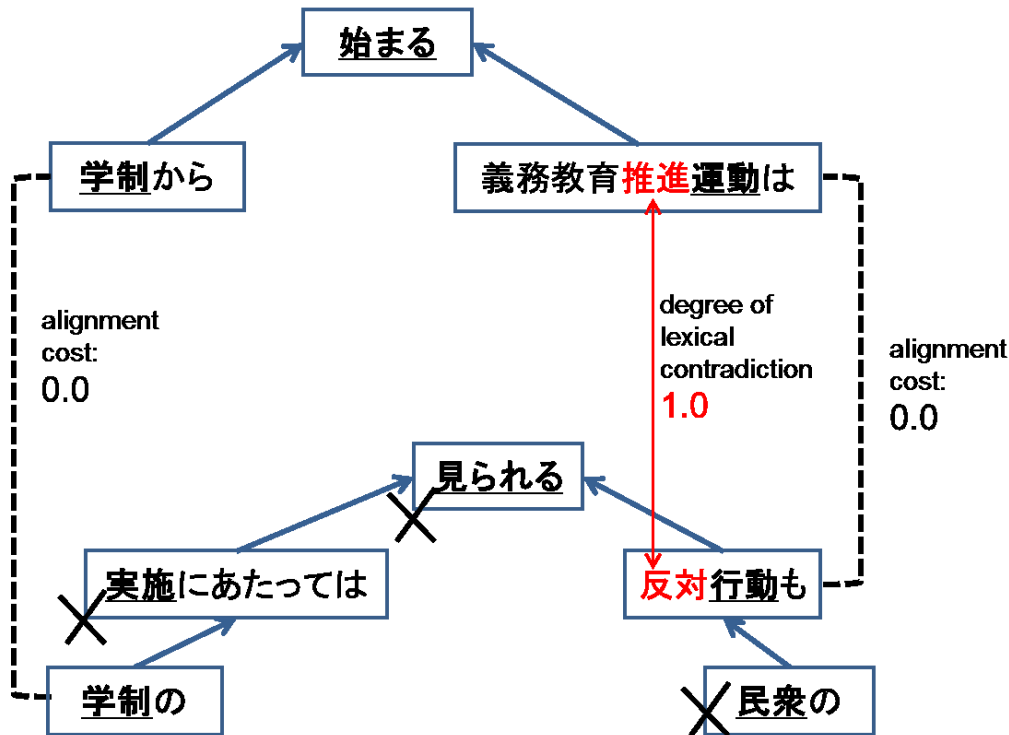
Example from the training data – **High Alignment Cost and High Lexical Contradiction**

<pair id="9" label="Y">
<t1>**一揆をおこした農民は徴兵以外にも、新政のいろいろに不満をもっていたが、**1872**年に施行された学制に対するそれも大きく、学制から始まった義務教育推進運動は、当初は授業料徴収があったためになかなか効果を上げなかった。**</t1>
<t2>**学校の建設費や授業料が民衆の負担とされたため，学制の実施にあたっては民衆の反対運動もみられた。**</t2>

T15: 学制から始まった義務教育推進運動は、

```
                        始まる

        学制から              義務教育推進運動は

  alignment                          degree of          alignment
  cost:                              lexical            cost:
  0.0                                contradiction      0.0
                                     1.0
                    見られる

        実施にあたっては          反対行動も

        学制の                    民衆の
```

T22: 学制の実施にあたっては民衆の反対運動もみられた。

Alignment Costs =
0.0 (bunsetsu alignment)
+ 3.0 (node deletion) = 3.0

Lexical Contradiction =
1.0 (antonyms)

=> Contradiction T1 and T2 = 0.08

Contradiction on sentence level is lower than before due to high alignment costs

Experiments – Exam BC (formal run, official result)

Team	MacroF1	Acc.	Correct Answer Ratio	Y-F1	Y-Prec.	Y-Rec.	N-F1	N-Prec.	N-Rec.
BnO-JA-ExamBC-02	67.15	70.31	55.56	56.96	64.71	50.87	77.34	72.76	82.55
BnO-JA-ExamBC-03	66.97	68.75	57.41	59.30	59.65	58.96	74.64	74.37	74.91
KDR-JA-ExamBC-02	66.90	68.75	51.85	59.06	59.76	58.38	74.73	74.19	75.27
BnO-JA-ExamBC-01	66.86	69.87	57.41	56.87	63.57	51.45	76.84	72.73	81.45
KDR-JA-ExamBC-03	66.64	68.30	47.22	59.20	58.86	59.54	74.09	74.36	73.82
WSD-JA-ExamBC-01	64.90	67.86	52.78	54.72	60.00	50.29	75.09	71.62	78.91
WSD-JA-ExamBC-03	64.71	67.63	52.78	54.55	59.59	50.29	74.87	71.52	78.55
SKL-JA-ExamBC-02	64.04	65.63	49.07	56.50	55.25	57.80	71.59	72.66	70.55
WSD-JA-ExamBC-02	63.96	67.63	51.85	52.46	60.61	46.24	75.47	70.57	81.09
KDR-JA-ExamBC-01	63.31	64.51	49.07	56.68	53.61	60.12	69.94	72.83	67.27
SKL-JA-ExamBC-01	61.65	67.63	29.63	46.49	64.29	36.42	76.80	68.57	87.27
SKL-JA-ExamBC-03	60.47	63.17	42.59	50.15	52.53	47.98	70.80	68.97	72.73
KitAi-JA-ExamBC-01	59.84	63.17	36.11	48.28	52.74	44.51	71.40	68.21	74.91
KitAi-JA-ExamBC-03	59.05	61.38	45.37	49.27	50.00	48.55	68.83	68.21	69.45
JAIST-JA-ExamBC-02	59.04	63.39	41.67	45.70	53.49	39.88	72.39	67.40	78.18
IBM-JA-ExamBC-03	58.76	60.94	38.89	49.28	49.42	49.13	68.24	68.12	68.36
JAIST-JA-ExamBC-03	58.65	64.96	42.59	42.49	58.00	33.53	74.80	66.95	84.73
JAIST-JA-ExamBC-01	57.55	63.17	40.74	42.11	53.57	34.68	73.00	66.37	81.09
KitAi-JA-ExamBC-02	57.16	58.71	39.81	49.04	46.84	51.45	65.29	67.44	63.27
KYOTO-JA-ExamBC-02	56.82	62.05	43.52	41.78	51.26	35.26	71.85	65.96	78.91
NTTD-JA-ExamBC-02	55.57	55.58	34.26	54.88	45.15	69.94	56.26	71.11	46.55
IBM-JA-ExamBC-01	55.17	57.59	38.89	44.77	45.03	44.51	65.58	65.34	65.82
IBM-JA-ExamBC-02	55.17	57.59	38.89	44.77	45.03	44.51	65.58	65.34	65.82
NTTD-JA-ExamBC-03	53.12	54.02	34.26	46.63	42.25	52.02	59.61	64.68	55.27
NTTD-JA-ExamBC-01	52.02	58.93	31.48	33.81	44.76	27.17	70.23	63.27	78.91
JUNLP-JA-ExamBC-01	50.46	50.89	30.56	45.81	39.91	53.76	55.10	62.79	49.09
*TKDDI-JA-ExamBC-03	49.08	62.50	28.70	22.94	55.56	14.45	75.22	63.28	92.73
TKDDI-JA-ExamBC-01	48.62	62.28	26.85	22.12	54.55	13.87	75.11	63.12	92.73
TKDDI-JA-ExamBC-02	48.62	62.28	26.85	22.12	54.55	13.87	75.11	63.12	92.73
THK-JA-ExamBC-01	43.77	62.28	26.85	11.52	61.11	6.36	76.03	62.33	97.45
KYOTO-JA-ExamBC-03	38.57	61.38	21.30	1.14	50.00	0.58	76.01	61.43	99.64
KYOTO-JA-ExamBC-01	37.86	60.94	20.37	0.00	0.00	0.00	75.73	61.21	99.27
Baseline-JA-ExamBC-01	54.77	56.47	32.41	45.98	44.15	47.98	63.55	65.38	61.82

Table 15: Results on Exam BC subtask (JA).

Our system is 2nd best system of all participants, with only small margin (0.25 percent points) difference to best system, but large marge (2.0 percent points) to third best system.

Empowered by Innovation **NEC**

Experiments – Exam BC   Analysis

Features	CV Training Data		Test Data	
	Accuracy	Macro-F1	Accuracy	Macro-F1
BC (KDR-JA-ExamBC-02)	73.69	72.59	68.75	66.90
BC - Contradiction	70.38	69.27	68.08	66.77
BC - Tree-edit distance	72.93	71.32	68.75	66.50
BC - Character Overlap	59.73	50.09	61.16	49.68
BC - Temporal Expressions	71.94	70.81	67.63	65.83

Contradiction feature is helpful for training data (+ 3.32 percent points)
but only little contribution to test data performance (+ 0.13 percent points)

Experiments – Search BC (formal run, unofficial result)

Team	MacroF1	Acc.	Correct Answer Ratio	Y-F1	Y-Prec.	Y-Rec.	N-F1	N-Prec.	N-Rec.
*KDR-JA-ExamSearch-02	58.12	64.51	32.41	41.76	57.00	32.95	74.48	66.67	84.36
*KDR-JA-ExamSearch-01	57.59	63.84	33.33	41.30	55.34	32.95	73.87	66.38	83.27
*KDR-JA-ExamSearch-03	57.39	63.17	34.26	41.70	53.64	34.10	73.08	66.27	81.45
NTTD-JA-ExamSearch-01	55.02	58.04	25.93	43.37	45.28	41.62	66.67	65.05	68.36
*BnO-JA-ExamSearch-02	54.77	56.47	31.48	45.98	44.15	47.98	63.55	65.38	61.82
*BnO-JA-ExamSearch-01	52.45	54.91	26.85	41.62	41.62	41.62	63.27	63.27	63.27
*BnO-JA-ExamSearch-03	51.78	51.79	31.48	51.57	42.12	66.47	52.00	66.86	42.55
NTTD-JA-ExamSearch-02	49.15	49.33	25.93	52.21	41.06	71.68	46.08	66.44	35.27
KYOTO-JA-ExamSearch-01	46.57	62.95	28.70	17.00	62.96	9.83	76.15	62.95	96.36
KYOTO-JA-ExamSearch-02	45.41	62.28	26.85	15.08	57.69	8.67	75.75	62.56	96.00

Table 16: Results on Exam Search subtask (JA).

Our system is best system of all participants in unofficial run, with large margin (3.01 percent points) to second best system.

System for Search BC uses same features as system for Exam BC plus additionally:
- Ratio of Named Entities between T2 and T1 (recognized with Cabocha).
- Tsubaki Search Engine score of T1 candidate.
- Word Overlap (clipped precision of morphemes)

Features are extracted from Top-1 and Top-2 search results (T1 candidates) from Text book and Wikipedia.

Experiments – Search BC　Analysis

Using Top-1 and Top-2 Search Result from Text Book and Wikipedia				
	CV Training Data		Test Data	
Features	Accuracy	Macro-F1	Accuracy	Macro-F1
BC	59.84	55.22	62.50	57.91
BC - Contradiction	58.89	54.00	60.27	55.55
BC + Tsubaki Score	61.03	56.97	61.61	56.46
BC + Tsubaki Score + Word Overlap	60.60	55.12	60.94	55.31
BC + Tsubaki Score + Word Overlap + Named Entity (KDR-JA-ExamSearch-02)	64.13	59.18	64.51	58.12

Ablation test indicates that contradiction feature can greatly improve performance also on test data (+ 2.36 percent points for system that was submitted)

Empowered by Innovation　NEC

Summary and Conclusions

Main Idea:

The more similar the tree structure of T1 and T2 is, the more likely that lexical contradiction propagates to sentence contradiction as a whole.

Contradiction = Lexical contradiction / Alignment Costs
High contradiction if lexical contraction is high and alignment costs are low.

Experiments:
Indicate that contradiction feature can be helpful (+ 2.36 percent points for Search Exam)

Empowered by Innovation   **NEC**