Improved Correction Detection in Revised ESL Writings

Huichao Xue, Rebecca Hwa
Department of Computer Science, University of Pittsburgh

Correction Detection

Compare a sentence with its revision, to detect corrections of individual mistakes

As a result, when the earthquake that happened in Lushan, Jiangxi province this year, some measures, such as temporal traffic management and emergency stocks mobilization, became more efficient, and more effective.

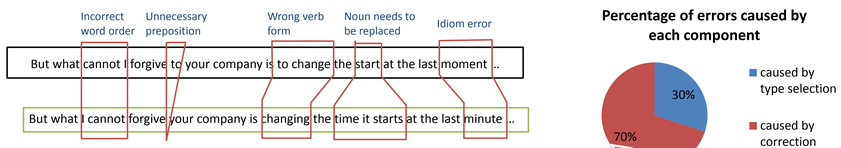
As a result, when the earthquake happened in Lushan this year, some measures, such as temporal traffic management and emergency stocks mobilization became more efficient, and more effective.

| Quote |

Correction detection

As a result, when the earthquake ~~that~~ happened in Lushan, ~~Jiangxi province~~ this year, some measures, such as temporal traffic management and emergency stocks mobilization, ~~,~~ became more efficient, and more effective.

Swanson and Yamagil (2012): comparing two versions automatically



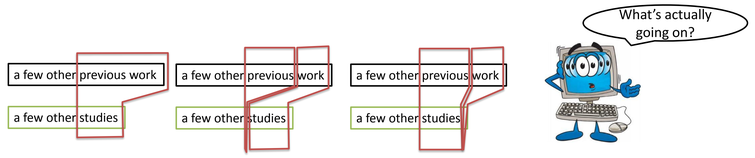
S&Y's system involves two components:

- Correction detection
- Error type selection: determining the error type that was addressed.

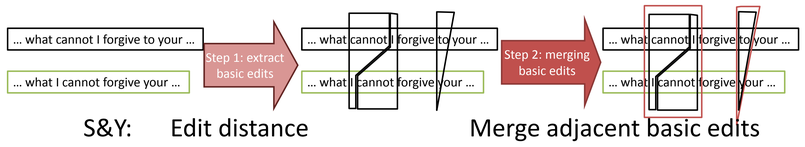
Our focus

The challenge of correction detection – Ambiguity

There are often multiple ways to interpret one revision



S&Y's uses a deterministic algorithm for correction detection



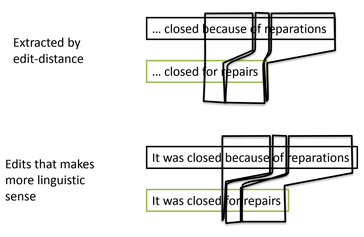
The heuristics were developed from one single data collection

- Lacks the flexibility to adapt to a specific context
- May not work as well for other guidelines

S&Y's algorithm makes mistakes when there are ambiguities

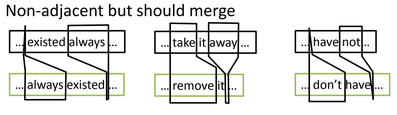
First step's mistakes

The extracted basic edits might not match our linguistic intuition

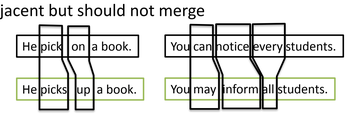


Second step's mistakes

Non-adjacent but should merge



Adjacent but should not merge

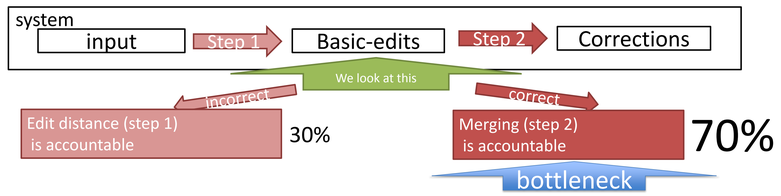


Our contributions

- We show empirically that a major challenge in correction detection is to determine if two edits address the same error – do we need to merge them
- We have developed a merging model that reduces mis-detections by 1/3
- We show the merging model is generalizable – we have conducted experiments across multiple corpora

How to improve correction detection? – An error analysis of S&Y's system

Which step is accountable for more errors?



This highlights the difference between correction detection and previous work

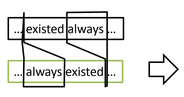
- Previous work focus on alignment -- step 1
  - Previous tasks also involve identify corresponding phrases between two sentences (Koehn et al., 2003, Cohn et al. 2008)
  - A bigger concern in previous work is to guarantee the extracted phrase pairs are indeed translations or paraphrases (Snover et al., 2009; Heilman and Smith, 2010)
- Correction detection concerns the granularity of the extracted phrase pairs – focuses on step 2

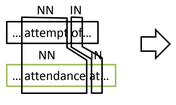
We propose: A classifier to determine if consecutive basic edits address the same mistake

Intuition: certain patterns indicate whether two edits address the same writing mistake

We encode these patterns into a classifier

- Input: features extracted from two consecutive basic edits
- Output: whether we should merge them



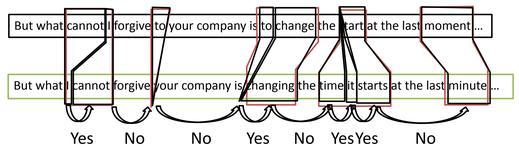


Classifier with Contextual Features	
Does merging the two basic edits match the pattern for a single correction?	gap-between-edits
	Tense-change
	Word-order-error
	Same-word-set
	Revised-to
Does one basic-edit address one single error?	Editdistance=1
	Orig-word-not-in-dict
	Same-POS
	Both-prepositions

Yes

no

Constructing training samples



Experiments

Task: detect corrections from revisions, on different corpora

Evaluation Metric: Precision/Recall/F_1-measure

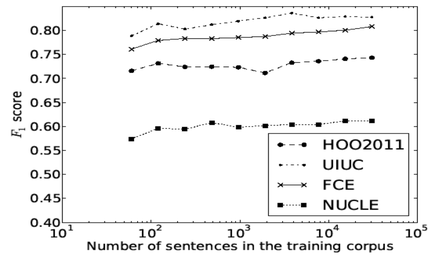
Did additional contextual information help?

Method	Corpus	Correction Detection F1	Overall F-score
S&Y	FCE	70.40%	57.10%
MaxEntMerger	FCE	**80.96%**	**66.36%**
S&Y	NUCLE	61.18%	39.32%
MaxEntMerger	NUCLE	**63.88%**	**41.00%**
S&Y	UIUC	76.57%	65.08%
MaxEntMerger	UIUC	**82.81%**	**70.55%**
S&Y	HOO2011	68.73%	50.95%
MaxEntMerger	HOO2011	**75.71%**	**56.14%**

How did our method generalize over revisions from different sources?

Training \ Testing	FCE	NUCLE	UIUC	HOO2011
S&Y	70.44%	61.18%	76.57%	68.73%
FCE	**80.96%**	61.26%	**83.07%**	75.43%
NUCLE	74.53%	**63.88%**	78.57%	74.73%
UIUC	77.25%	58.21%	82.81%	70.83%
HOO2011	71.94%	54.99%	71.19%	**75.71%**

- FCE corpus is a comparably good resource for training correction detection models.
- Big data size benefits model training



Conclusions

The merging step accounts for 70% errors in correction detection

We proposed a merging model:

- Reduces 1/3 errors in correction
- Leads to significant overall system performance
- Generalizes over corpora