Evaluation of really good Grammatical Error Correction

Robert Östling
Katarina Gillholm
Murathan Kurfalı*
Marie Mattson
Mats Wirén

Department of Linguistics
*Department of Psychology/RISE
Stockholm University

Executive Summary

The issue

As grammatical error correction (GEC) systems become more capable, how do we evaluate their performance?

Executive Summary

The issue

As grammatical error correction (GEC) systems become more capable, how do we evaluate their performance?

The setting

Improvement of Swedish second language learner texts, using a palette of different GEC methods.

Executive Summary

The issue

As grammatical error correction (GEC) systems become more capable, how do we evaluate their performance?

The setting

Improvement of Swedish second language learner texts, using a palette of different GEC methods.

Results

Automatic metrics have different biases, we suggest evaluating by measuring changes during post-editing.

Approaches

Discrete errors

- Goal: identify, categorize and correct discrete errors.
- GEC tools: fits well with traditional NLP pipeline of POS tagging, morphological analysis, parsing.
- Characteristics: conservative, any change that is not an official 'error' counts as a mistake.
- Evaluation: comparison to error annotations.

Approaches

Discrete errors

- Goal: identify, categorize and correct discrete errors.
- GEC tools: fits well with traditional NLP pipeline of POS tagging, morphological analysis, parsing.
- Characteristics: conservative, any change that is not an official ‘error’ counts as a mistake.
- Evaluation: comparison to error annotations.

Holistic improvement

- Goal: make the text more native-like.
- GEC tools: more suited for generative neural models.
- Characteristics: may edit the text considerable, multiple very different outputs may all be considered appropriate.
- Evaluation: that’s the question...

Evaluations

- Reference-based
  - Requires one or more target versions of each text
  - Sometimes includes additional error annotations
  - GLEU: text similarity to reference(s), similar to MT
  - ERRANT: similarity of the *types* of edits made, compared to the reference(s)

Evaluations

- Reference-based
  - Requires one or more target versions of each text
  - Sometimes includes additional error annotations
  - GLEU: text similarity to reference(s), similar to MT
  - ERRANT: similarity of the *types* of edits made, compared to the reference(s)
- Reference-free
  - Focus on the end result: is the final text acceptable?
  - Typically judged by a language model
  - Separate module needed to guard against changing semantics (meaning preservation)
  - SOME: weighs together fluency, grammaticality and meaning preservation scores from different models
  - Scribendi: LM score + surface level check for similarity in content

Evolution of evaluations

- Consider the failure modes of traditional GEC systems:
  - failure to detect an error, which is left unchanged (no effect on precision)
  - false positive, making a **small** unnecessary change
- Favored by most reference-based metrics (conservative bias)

Evolution of evaluations

- Consider the failure modes of traditional GEC systems:
  - failure to detect an error, which is left unchanged (no effect on precision)
  - false positive, making a **small** unnecessary change
- Favored by most reference-based metrics (conservative bias)
- Compare to failure modes of generative neural models:
  - hallucination or omission of content
  - subtle changes of meaning
- Favored by most reference-free metrics (anti-conservative bias)

Evolution of evaluations

- Consider the failure modes of traditional GEC systems:
  - failure to detect an error, which is left unchanged (no effect on precision)
  - false positive, making a **small** unnecessary change
- Favored by most reference-based metrics (conservative bias)
- Compare to failure modes of generative neural models:
  - hallucination or omission of content
  - subtle changes of meaning
- Favored by most reference-free metrics (anti-conservative bias)
- How do these biases affect actual GEC systems? What can we do about them?

The playing field

- Three categories of GEC systems:
  - Rule-based (Granska)
  - Translation-based (from artificially corrupted text)
  - LLM-based (GPT-3.5)
- Human 'systems':
  - Minimal: minimal edits to achieve grammaticality
  - Fluent: minimal edits to achieve native-like fluency
  - Free: any edits to achieve native-like fluency and idiomaticity

Human evaluation

- Anchor point: human minimal (only one with access to context beyond sentence)
- Humans annotate the following for each system output:
  - Grammaticality score (1–4)
  - Fluency score (1–4)
  - Meaning preservation score (1–4)
  - Post-edited output, which must achieve the maximal score for all three scores above

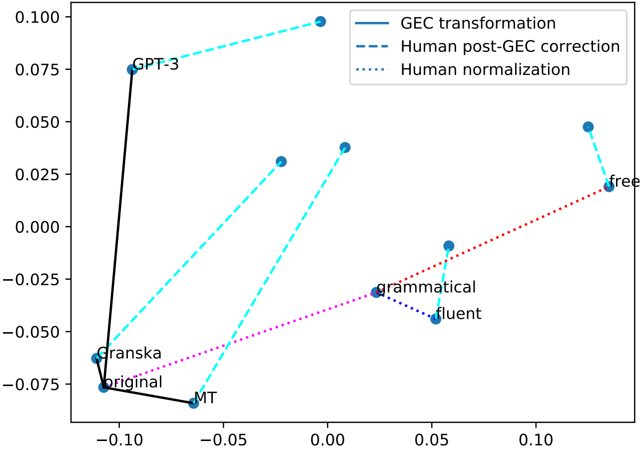
What metric to use?

- Our suggestion: distance to post-edited version
- One 'reference' per system output
- Including human corrections (human free/fluent) provides a baseline for differences of opinion
- We use normalized Levenshtein distance for simplicity
- Others could be used, including ERRANT-like for more detailed semi-automatic analysis
- Main disadvantage: expensive (but LLMs could be useful in some settings)

Normalized Levenshtein distance

	CEFR level			
System	All	A	B	C
Granska	0.126	0.119	0.180	0.079
MT	0.113	0.095	0.158	0.087
GPT-3.5	0.076	0.068	0.112	0.050
Human fluent	0.034	0.034	0.045	0.022
Human free	0.029	0.030	0.034	0.025

Map of edits



Pairwise normalized Levenshtein distance matrix reduced to 2-D by MDS

Reference-based (GLEU)

	CEFR level			
System	All	A	B	C
Uncorrected	0.44	0.29	0.17	0.53
Granska	0.47	0.35	0.24	0.55
MT	0.57	0.48	0.38	0.63
GPT-3.5	0.63	0.60	0.52	0.65
Human minimal	1.0	1.0	1.0	1.0

Reference-free (Scribendi)

System	CEFR level All	A	B	C
Uncorrected	0	0	0	0
Granska	0.03	0.08	0.11	-0.01
MT	0.51	0.57	0.68	0.43
GPT-3.5	0.69	0.70	0.83	0.65
Human minimal	0.68	0.67	0.77	0.65

Conclusions

- Different types of automated metrics have their problems:
  - Reference-based: penalize highly capable systems
  - Reference-free: too much emphasis on fluency

Conclusions

- Different types of automated metrics have their problems:
  - Reference-based: penalize highly capable systems
  - Reference-free: too much emphasis on fluency
- Manual post-editing is an appealing alternative:
  - does not suffer from biases of automated metrics
  - possibility of fine-grained analysis
  - labor intesive, but could be LLM-assisted in some cases

Conclusions

- Different types of automated metrics have their problems:
  - Reference-based: penalize highly capable systems
  - Reference-free: too much emphasis on fluency
- Manual post-editing is an appealing alternative:
  - does not suffer from biases of automated metrics
  - possibility of fine-grained analysis
  - labor intesive, but could be LLM-assisted in some cases
- LLM-based GEC sets a new state of the art for Swedish

Resources

- Annotation tool and guidelines
- Annotated and post-edited Swedish GEC system outputs
- https://github.com/robertostling/gec-evaluation