Improved Grammatical Error Correction through Neural Networks and Genetic algorithms

Li Jin Tao, Joey
Niharika Shrivastava
Aditya Ganesh Kumar
Lee Rong Jieh
Adi Kamaraj

Our contributions

- Modified ESC architecture that uses 1D Convolution layers to decide on a proposed edit[1].
- Implement an ensemble of our modified ESC architecture with the highest baseline BART.
- Using genetic algorithm as the optimization method on Neural Networks
    - Works Performance is more robust, less likely to be stuck in local optima[2]

[1]https://github.com/nusnlp/esc
[2]https://www.kaggle.com/code/zzettrkalpakbal/genetic-algorithm-tutorial-of-pygad?scriptVersionId=101358507

Related work

Grammatical Error Correction

- Frustratingly Easy System Combination for Grammatical Error Correction[1] provides the code to combine Grammatical Error Correction (GEC) models to produce better predictions with just the models' outputs.
- Intelligent Error Correction of College English Spoken Grammar Based on the GA-MLP-NN Algorithm[2]
    - The article stated that genetic algorithm on multilayer perceptron neural network for the intelligent correction of spoken grammar is very fast and accurate.

[1]https://github.com/nusnlp/esc
[2]Yining Du. (August 2021). *Intelligent Error Correction of College English Spoken Grammar Based on the GA-MLP-NN Algorithm.* Computational Intelligence and Neuroscience Volume 2021, Article ID 7371416, 9 pages. https://doi.org/10.1155/2021/7371416

Baseline

Established using neural network configuration proposed in the paper.

```
class Model(nn.Module):
    def init(self, feature_length):
        self.linear =
        nn.Linear(feature_length,1)
    def forward(self, x):
        x = torch.sigmoid(self.linear(x))
        return x
```

Addition of 1D Convolutions

Convolution layers provide additional context to decide if an edit must be performed or not.

```
Conv1d[(3x3) kernel, 'same' padding]
ReLU
MaxPool1d[(5x5) kernel, stride 2]
Conv1d[(3x3) kernel, 'same' padding]
ReLU
MaxPool1d[(5x5) kernel, stride 2]
Linear[ceil(feature_length/4) - 3 -> 1]
```

Adam optimizer[learning rate 1e-04]
200 epochs

Ensemble of Modified ESC with BART

- Highest baseline for GEC.
- Uses a series of Transformers to implement:
  - masked-token generation
  - next-sentence prediction
  - along with fine-tuning on a text-summarization task.
- BART adds extra determinants in the final output.
- Lower overall performance on CoNLL Shared Task.

Results

Results from various approaches on CoNLL-2014 Restricted

Metric	Baseline	CNN	Ensemble
Precision	81.98	82.43	69.63
Recall	43.36	43.45	52.01
F0.5	69.51	69.89	65.21

Results from various approaches on CoNLL-2014 Alternate Annotations

Metric	Baseline	CNN	Ensemble
Precision	84.68	85.09	70.01
Recall	45.73	45.81	51.35
F0.5	72.28	72.63	66.25

Results

Results from various approaches on BEA 2019

Metric	Baseline	CNN
Precision	86.6	61.16
Recall	60.9	72.58
F0.5	79.9	63.15

Differences between ground truth and Predictions

- In retrospect , its is also one 's duty to ensure that he or she undergo _______ periodic healthchecks on their own .
- In retrospect , _ it is also one 's duty to ensure that he or she _____ undergoes periodic healthchecks on their own .

- Therefore , the social media is truly good for people .
- Therefore , ___ social media is truly good for people .

- Secondly , the social media sites can not only connect people with their friends closely .
- Secondly , ___ social media sites can not only connect people with their friends closely .

- When we are diagnosed with _ certain genetic diseases ___ , are we suppose ____ to disclose this result to our relatives ?
- When we are diagnosed with a certain genetic ____ disease , are we ___ supposed to disclose this result to our relatives ?

Character-Level Error Corrections

- Generate synthetic training data using a combination of addition, deletion, swap, substitution of noise characters (called "edits").

- 2 major challenges :
  - Deciding the underlying distribution of characters to choose for the edits

  - The frequency at which to inject the edits

- Approach discarded

Genetic Algorithm

To optimize the logistic regression models

- At each step, the algorithm selects models to be the parents and produce new offspring for the next generation by making slight modifications to the parents. Eventually, we would achieve better fitted models after generations of 'evolution'.

- Recent research article[1] pointed out that GA provides excellent accuracy when it is used to optimization a multilayer perceptron neural network model for spoken grammar error correction.

- Utilise the PyGAD package[2].

[1]Yining Du. (August 2021). *Intelligent Error Correction of College English Spoken Grammar Based on the GA-MLP-NN Algorithm.*
[2]https://pygad.readthedocs.io/en/latest/

Genetic Algorithm

Configurations of our GA

- Parent selection
  - steady-state selection
  - random selection
  - tournament selection
  - roulette wheel selection
  - stochastic universal selection
  - rank selection
- Crossover
  - single point crossover
  - two point crossover
  - uniform crossover
  - scattered crossover
- Number of generations:
  - 1000
- Number of parents mating:
  - 5
- Percentage Mutation:
  - 10%
- Fitness value:
  - 1 / BCE loss
- Neural network model:
  - single layer logistic regression model

Results

Genetic Algorithm

Selection	Crossover	F0.5
rws	single_point	0.6910
rws	two_points	0.6913
rws	uniform	0.6914
rws	scattered	0.6907
sss	single_point	0.6913
sss	two_points	0.6919
sss	uniform	0.6919
sss	scattered	0.6921
sus	single_point	0.6905
sus	two_points	0.6910
sus	uniform	0.6913
sus	scattered	0.6907
rank	single_point	0.6913
rank	two_points	0.6913
rank	uniform	0.6921
rank	scattered	0.6916
random	single_point	0.6907
random	two_points	0.6922
random	uniform	0.6921
random	scattered	0.6910
tournament	single_point	0.6919
tournament	two_points	0.6921
tournament	uniform	0.6921
tournament	scattered	0.6921

- The highest F0.5 score achieved by genetic algorithm is 0.6922, which is slightly lower than the F0.5 score of 0.6926 achieved by the stochastic gradient descent optimizer on the same neural network model.
- GA's performance on ESC CNN model is significantly worse, with a F0.5 score of between 0.6800 to 0.6810.

Findings and Discussion

Genetic Algorithm

Selection	Crossover	F0.5
rws	single_point	0.6910
rws	two_points	0.6913
rws	uniform	0.6914
rws	scattered	0.6907
sss	single_point	0.6913
sss	two_points	0.6919
sss	uniform	0.6919
sss	scattered	0.6921
sus	single_point	0.6905
sus	two_points	0.6910
sus	uniform	0.6913
sus	scattered	0.6907
rank	single_point	0.6913
rank	two_points	0.6913
rank	uniform	0.6921
rank	scattered	0.6916
random	single_point	0.6907
random	two_points	0.6922
random	uniform	0.6921
random	scattered	0.6910
tournament	single_point	0.6919
tournament	two_points	0.6921
tournament	uniform	0.6921
tournament	scattered	0.6921

- In general, tournament selection and uniform crossover has the best F0.5 score, when tested against CoNll-14 dataset.
- However, we also notice that the highest F0.5 score is achieved by random selection, this might suggest that we could achieve better results using different sets of configurations or optimizer, such as stochastic gradient descent.

Discussion

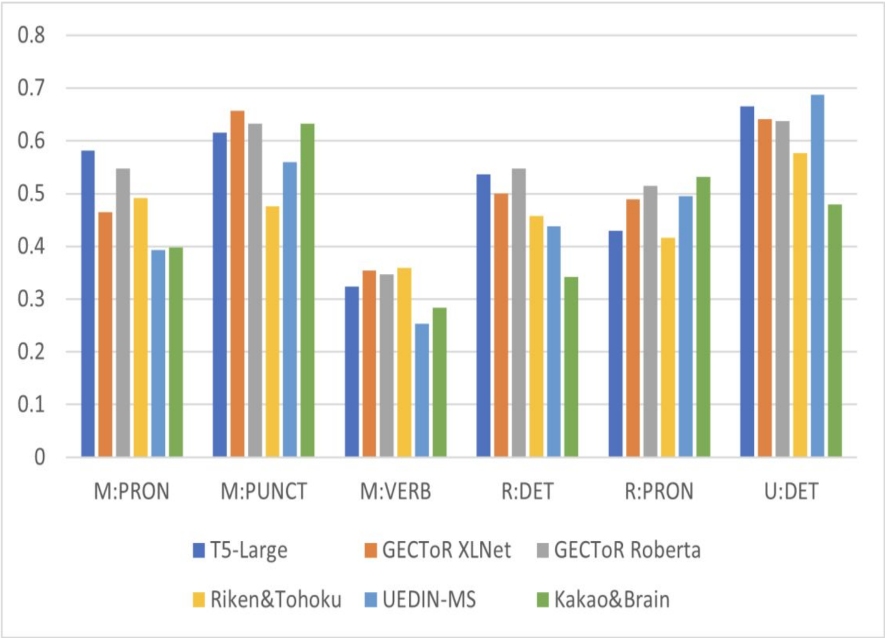

Figure 1: The $F_{0.5}$ scores of the base GEC systems that we use in our experiments on selected error types in the BEA-2019 development set.

- 0.3 points gain on CoNLL-14 using ESC-CNN.
- ESC extended with BART: model performance deteriorates due to the addition of unnecessary determinants. Perform an ensemble with a model such as UEDIN-MS that gives less weightage to extra determinants in the final output.
- The CNN architecture doesn't work well on the BEA-Shared Task. We believe it's because our model is overfitting the CoNLL-2014 Shared task.
- Due to the large amount of training data vocabulary and limited memory, using Language Models wasn't a scalable approach.

Image taken from: Muhammad Qorib, Seung-Hoon Na, and Hwee Tou Ng. 2022., Frustratingly easy system combination for grammatical error correction. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies

THE END

Thank you for your attention!