Enhance An Evaluation of Parser Robustness for Ungrammatical Sentences

Jianqiang Li

*** Computer Science, Queens College, CUNY**

Abstract

For NLP robust related topic there are a lot of research and the parser is one of the most important topics. In the paper we already know the parser algorithms can overlook ungrammatical sentence to produce a parser that closely analysis for the correct sentence. The paper introduce a robust score concept by using the F1 micro score. In order to compare different parser algorithms robust over ungrammatical sentences, the robust score evaluation is the key to compare them. In our paper, I do some slightly modify the robust evaluation according to the F1 score calculation method. The score is an evaluation metric of accuracy of the parser from the TurboParser on two ungrammatical domains: learner English (ESL) and machine translation outputs (MT).

1. Introduction

Since parsing is an essential component of NLP applications, there are many ungrammatical sentences input for these applications. We need to develop a method to know which parser algorithms are more robust than others against sentences not well-formed.

A "gold-standard free" alternative is to compare the parser for the incorrect sentence with the parser of the corresponding correct sentence. We simulate the golden standard method through comparing the parser output for an ungrammatical sentence with the automatically generated parser of the corresponding correct sentence.

In this paper, we develop the F1 score by the formulas as below and Figure 1 show the F1 conception.

$$\text{Precision} = \frac{true\ positives}{false\ positive + true\ positives}$$

$$= \frac{\text{\# of shared dependencies}}{\text{\# of dependencies of the ungrammatical sentence}}$$

$$\text{Recall} = \frac{true\ positives}{false\ negatives + true\ positives}$$

$$= \frac{\text{\# of shared dependencies}}{\text{\# of dependencies of the grammatical sentence}}$$

$$\text{Robustness F1 score} = \frac{2\ Precision * Recall}{Precision + Recall},$$

which is the harmonic mean of precision and recall.

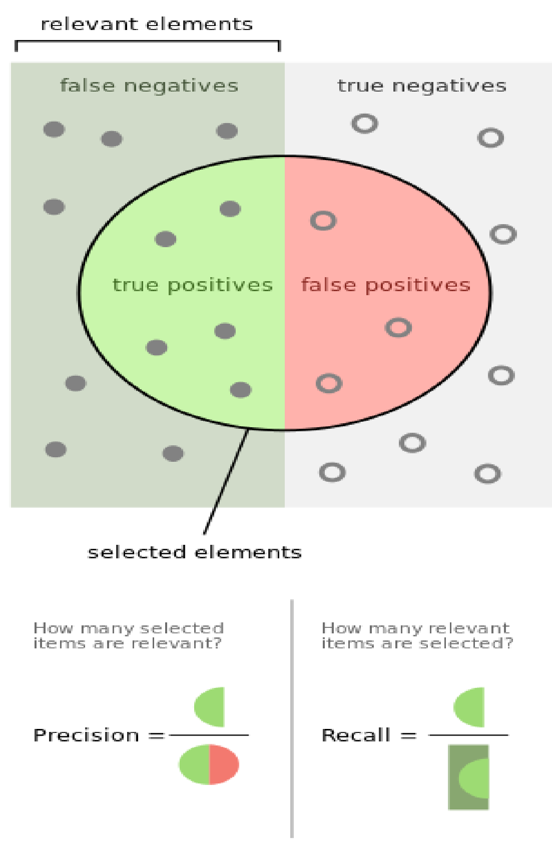


Figure 1 F1 score formula combine

The F1 score of each state-of-art parsers is the mean of all the sentences F1 score, so this is a F1 micro score.
If we give parameters to the F-measure that sets the tradeoff between precision and recall, we have a Weighted F-measure:

F_β score = $\frac{(1+\beta^2)* \ Precision*Recall}{\beta^2*Precision+Recall}$,

0<β<1 gives more weight to the precision while β>1 gives more weight to the recall.

We need to have trained a lot of corpus sentences and ESL/MT sentences to decide the parameter beta. In this experiment, we are simple to set β=0.9 to give more weight to precision, so the weighted F-measure becomes:

$$F_{\beta=0.9} \text{ score} = \frac{(1+0.9^2)* \ Precision*Recall}{0.9^2*Precision+Recall}$$

$$= \frac{1.81 * \ Precision*Recall}{0.81* Precision+Recall},$$

which means the precision is as important as the recall.
The overall performances of all parsers are measured by it. We choose TurboParser to calculate the $F_{\beta=0.9}$ score with an enhanced precision and recall values.

2. Related (previous work)

In the previous paper, a set of empirical analyses of eight leading dependency parsers on two domains of ungrammatical text: English as a Second Language (ESL) learner text and machine translation (MT) outputs. The parsers are trained with the PennTreebank and Tweebank (a Treebank on tweets).
The robustness F1 score is made up of two different parts: Precision and Recall. In the previous method, the authors use the formulas as below:

$$\text{Precision} = \frac{true\ positives}{false\ positive+true\ positives}$$

$$= \frac{\text{\# of shared dependencies}}{(\text{\# of dependencies of the ungrammatical sentence} - \text{\#of error related dependencies of the ungrammatical sentence})}$$

$$\text{Recall} = \frac{true\ positives}{false\ negatives+true\ positives}$$

$$= \frac{\text{\# of shared dependencies}}{(\text{\# of dependencies of the grammatical sentence} - \text{\#of error related dependencies of the grammatical sentence})}$$

Which denominators are FP + TP - TF and FN + TP – TF respectively.
Then they are not the F1 micro score method since the denominators do not include the TF part. Additionally, the precision of the dependencies measures how many dependencies of ungrammatical sentences are shared out of how many were found. Recall dependencies measures the coverage (from the dependencies of grammatical sentences that should have been retrieved, how many were found) Hence we use the normal F1 formula instead of previous method.

$$\text{Precision} = \frac{true\ positives}{false\ positive+true\ positives}$$

$$= \frac{\text{\# of shared dependencies}}{(\text{\# of dependencies of the ungrammatical sentence})}$$

$$\text{Recall} = \frac{true\ positives}{false\ negatives + true\ positives}$$

$$= \frac{\#\text{ of shared dependencies}}{(\#\text{ of dependencies of the grammatical sentence})}$$

3. Experiments

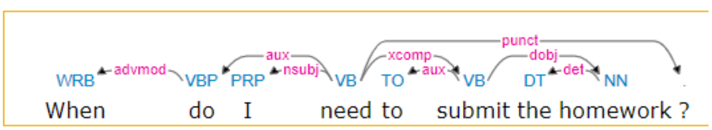
1) Parser algorithm

We choose Turbo parser algorithm to do the experiments because this algorithm works well on both ESL (tweets) and MT input data.

For example:

- When do I need to submit the homework? (grammatical sentence)

Tree View (explanation)

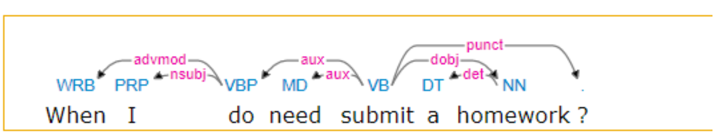


CoNLL Format

```
1 When     _ WR    WRB   _   2   advmod
2 do       _ VB    VBP   _   4   aux
3 I        _ PRP   PRP   _   4   nsubj
4 need     _ VB    VB    _   0   null
5 to       _ TO    TO    _   6   aux
6 submit   _ VB    VB    _   4   xcomp
7 the      _ DT    DT    _   8   det
8 homework _ NN    NN    _   6   dobj
9 ?        _ .     .     _   4   punct
```

- When I do need submit a homework? (ungrammatical sentence)

Tree View (explanation)



CoNLL Format

```
1 When     _WR    WRB   _   3   advmod
2 I        _PRP   PRP   _   3   nsubj
3 do       _VB    VBP   _   5   aux
4 need     _MD    MD    _   5   aux
5 submit   _VB    VB    _   0   null
6 a        _DT    DT    _   7   det
7 homework _NN    NN    _   5   dobj
8 ?        _ .    .     _   5   punct
```

2) Data

This folder named data includes sample data for two domains of ESL and MT.

ESL data contains the ungrammatical sentences written by English as second language learners and their corresponding error corrections.

ESL includes:

- 10 sentences POS Good and 10 sentences POS Bad
- 10 sentences Turbo Good and 10 sentences Turbo Bad

MT data contains French-to-English machine translation outputs and their human post-editions.

Machine Translation (MT) includes:

- 10000 sentences POS Good and 10000 sentences POS Bad
- 10000 sentences Turbo Good and 10000 sentences Turbo Bad

4. Results

Here are some ESL and MT data results from the console output, the whole output please reference ESL_output.txt and MT_output.txt:

1) ESL output:

```
[S, 1, 2, conclusion, UD, 3, 4, , S, 9, 10,
disadvantages, MD, 15, 15, the, AGV, 16, 17, are,
MD, 19, 19, the]
{but=CC, TV=NN, In=IN, whole=NN, ,=,, .=., both=DT,
conclusion=NN, the=DT, advantages=NNS,
disadvantages=NNS, are=VBP, and=CC, than=IN,
has=VBZ, greater=JJR, on=IN}
parser: data/ESL/FCE_10_Turbo.Good.txt
# of sentences: 10
matched: 160
un-matched: 161
count_all_goodsent_dependencies: 167
count_all_badsent_dependencies: 166
Precision/Recall based on each error type:
 # of Sents    Precision      Recall  F-score

All errors     10      96.3855421686747
       95.80838323353294     96.0960960960961
       96.98795180722891     96.40718562874252
       96.6966966966967
# of error 1   7       99.01960784313727
       99.01960784313727     99.01960784313727
       100.0  100.0  100.0
# of error 2   1       100.0  100.0  100.0  100.0
       100.0  100.0
# of error 3   0       NaN    NaN    NaN    NaN
       NaN    NaN
# of error 4   0       NaN    NaN    NaN    NaN
       NaN    NaN
# of error 5   1       90.0   85.71428571428571
       87.80487804878048     90.0
       85.71428571428571     87.80487804878048
# of error 6   1       85.0   85.0   85.0   85.0
       85.0   85.0
# of error 7   0       NaN    NaN    NaN    NaN
       NaN    NaN
```

2) MT output:

```
[D, 6, 6, that, S, 6, 7, lasts, I, 11, 12, , I, 13,
14, , I, 14, 15, , S, 15, 16, be, S, 16, 17,
released, S, 19, 20, next, S, 21, 22, and, S, 22,
23, will, S, 23, 24, then]
{``=``, next=IN, lasts=VBZ, TV=NN, be=VB, for=IN,
two-and-a-half=CD, The=DT, that=WDT, cinemas=NNS,
Friday=NNP, and=CC, four=CD, than=IN, of=IN,
released=VBN, ''='', a=DT, hours=NNS, will=MD,
in=IN, more=JJR, film=NN, then=RB, Mayo=NNP, .=.,
Telemadrid=NNP, Sangre=NNP, series=NN, become=VB}
parser: data/MT/HTER_10000_Turbo.Good.txt
# of sentences: 10000
matched: 175805
un-matched: 184467
count_all_goodsent_dependencies: 235049
count_all_badsent_dependencies: 239201
*************** Using Edit Distance to find errors
Precision/Recall based on each error type:
 # of Sents    Precision      Recall  F-score
All errors     10000   73.49676631786657
       74.79504273577 74.14022140221402
       77.11798863717125     78.48023178145833
       77.79314707432789
# of error 1   684     94.73180909328002
       95.43702925205815     95.08311155708739
       95.4533599930453      96.16395165528114
       95.80733824876751
# of error 2   974     89.84854456393221
       91.14169855502884     90.49050184292601
       91.3462079837847      92.66091724256097
       91.99886589169265
# of error 3   1024    84.3294143261485
       86.21111791219185     85.25988498472884
       86.34334857391683     88.26999054031495
       87.29604050298545
# of error 4   952     81.03410700664705
       82.92723724560915     81.9697428972967
       83.79644764084124     85.75411207136882
       84.76397806497837
# of error 5   889     78.31431079894644
       80.21131905805655     79.25146458616764
       81.33779631255487     83.30804248861912
       82.31113085487407
# of error 6   812     76.11802140584322
       77.97664908433592     77.03612623689912
       79.28261498409024     81.21851478693772
       80.23888986474617
# of error 7   740     74.11855639455378
       75.55151515151515     74.82817611572976
       77.91188536773886     79.41818181818182
       78.65782286383146
# of error 8   643     72.41688552943437
       74.1700564230416      73.28298706770816
       76.20908333867145     78.05406114683112
       77.12053933166952
# of error 9   548     70.50225693200545
       71.49084568439406     70.99310991666967
       74.65787776742853     75.70473699505958
       75.17766314346524
# of error 10  2729    63.022860588783516
       63.913536593501895    63.465073783116836
       67.97872923633572     68.93944448150866
       68.45571632834452
```

5. Discussion

We analysis the robust of different state-of-the-art dependency parsers by comparing the ungrammatical sentences parser and the corresponding grammatical parser . Even if the “gold-standard” is not absolutely correct to represent the robust of the parser algorithm. Since the sentences may have two or more meanings for the same structure and words, the parser can only express the sentence meaning without context.

For the purpose to evaluate parser algorithm robustness to ungrammatical sentences, we already have a modified metric for the dependencies connecting to unmatched (extra or missing) error words are ignored. The formal definition is :

- Shared dependency is a mutual dependency between two trees;
- Error-related dependency is a dependency connected to an extra word1 in the sentence;

We use these two definitions to evaluate the Precision, Recall, and Robustness F1 value respectively.

About the F1-score, we know it is used to measure a test's accuracy. In statistical analysis of binary classification, it considers both the precision p and the recall r of the test to compute the score. Precision p is the number of correct positive results divided by the number of all ungrammatical positive results, which are mutual dependencies between two dependency parsers divided by the total number of dependencies in ungrammatical sentence. Recall r is the number of correct positive results divided by the number of positive results, which are mutual dependencies between two parsers divided by the total number of dependencies in grammatical sentence.

And the F1 score can be interpreted as a weighted average of the precision and recall values, where an F1 score reaches its best between [0,1] range. The best value is at 1 while the worst is at 0.

F_β score = $\frac{(1+\beta^2) * \ Precision*Recall}{\beta^2 * Precision + Recall}$,

0<β<1 gives more weight to the precision while β>1 gives more weight to the recall.

Since the grammatical and ungrammatical sentences have many alignments and confusions, we cannot just use one F1 score to evaluate the robustness of the parsers. We may need to develop a more meaningful merit method to do the evaluation in the future. One idea is to give different weight beta to precision and recall variables on the dependency, so we can consider the sentences' deeply meaning in the context.

References

Homa B. Hashemi and Rebecca Hwa. An Evaluation of Parser Robustness for Ungrammatical Sentences. In proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP), 2016.