

# Using RNN to fix syntax errors in JavaScript files

Jordan Siaw, Justin Lew & Luiz Peres

SFU School of Computing Science

`['jsiaw', 'justin_lew', 'lperesde'].map(f => f + '@sfu.ca')`



## The Problem

Parsers often fail miserably in finding syntax errors

```
1 if (process.argv.length > 3) //{
2   console.error('Not enough args!');
3   process.exit(1);
4 }
```

Mozilla SpiderMonkey (2016)

```
wrong.js:4: SyntaxError: syntax error:
wrong.js:4: }
wrong.js:4: ^
```

Node with Google’s V8 JavaScript (2016)

```
/path/to/wrong.js:6 // line 6 does not exist!
});
^
SyntaxError: Unexpected token }
```

## GrammarGuru

GrammarGuru is a tool created by that seeks to find and fix single token syntax errors, using LSTM language models on 10,000 GitHub repos (Santos EA et al., 2017).

## Improving GrammarGuru suggestions

GrammarGuru works greatly for insertions and deletions, however it rarely work when a token is changed, e.g when a keyword is mistyped, e.g **functions** rather than **function**. We improve GrammarGuru by training a NN that reevaluates and reranks GrammarGuru scores by using a modification of the features of the ranking system ROSE [2]:

ID	Description
1-4	n-gram precision, n=1...4
5-8	n-gram recall, n=1...4
9-12	n-gram f-measure, n=1...4
13	average n-gram precision per code line
14	n-gram score at the document level
15-18	n-gram precision excluding common tokens, n=1...4
19-22	n-gram recall excluding common tokens, n=1...4
23-26	n-gram f-measure excluding stopwords, n=1...4
27	average n-gram precision excluding stopwords, n=1...4
28	pos distance of $hyp_1$ , $hyp_2$ against parser error $\{-1, 0, 1\}$
29	which of $hyp_1$ , $hyp_2$ equals parser error token $\{-1, 0, 1\}$
30	$hyp_1$ and/or $hyp_2$ fix the syntax error $\{-1, 0, 1\}$
31	which of $hyp_1$ , $hyp_2$ has best score $\{-1, 0, 1\}$

## Evaluation and Reranking

15 of the most unnatural tokens in the erroneous JS file are taken, evaluated and reranked by using the ROSE approach. We perform  $\binom{15}{2}$  comparisons per file....

WHAT ELSE :(

## References

[1] Santos EA, Campbell JC, Hindle A, Amaral JN. 2017. Finding and correcting syntax errors using recurrent neural networks. PeerJ Preprints 5:e3123v1 <https://doi.org/10.7287/peerj.preprints.3123v1>

[2] Xingyi Song and Trevor Cohn. 2011. Regression and ranking based optimisation for sentence level machine translation evaluation. In Proceedings of the Sixth Workshop on Statistical Machine Translation, pages 123–129. Association for Computational Linguistics

