

CS559 Deep Learning Project Proposal

Hamdi Alperen Cetin, Emre Dogan

Project Description

In a software project, it is an important task to identify an appropriate developer who could potentially fix the bug for a given bug report. This process is the primary task of a *bug triaging*, and it is simply a classification problem which bug title and bug description are the input and one of the available developers is the output.

There are already some studies employing different machine learning algorithms to solve this problem in the literature. The major difficulty faced in these studies is that input data -bug title, description etc.- is in text format and hard to represent. Mani et al.[1] brought a new approach to represent bug reports by extracting features with an attention based deep bidirectional recurrent neural network (DBRNN-A) model that learns syntactic and semantic features from long word sequences in an unsupervised manner. In our term project, we will benefit this study and try to explore more.

Dataset

Our dataset consists of bug reports from three open source systems: Google Chromium, Mozilla Core and Mozilla Firefox. The details of data collecting process are mentioned in the paper[1]. The authors provide the dataset available online, and we plan to use it directly. In total, there are 383,104 bug reports from Google Chromium, 314,388 bug reports from Mozilla Core, and 162,307 bug reports from Mozilla Firefox. All datasets have attributes *id*, *issue_id*, *issue_title*, *reported_time*, *owner*, *description* and *status*. Chromium dataset has another attribute named *type* and the other Mozilla datasets have another attribute named *resolution*. In DBRNN-A model, only *owner*, *title*, *description* and *status* attributes are used. We shared dataset links and brief explanation about dataset contents in our GitHub Repository.¹

Plan

Our brief plan for this term project will be in the following order:

1. Firstly, we will make the system work as it is given in the paper. Authors stated that they implemented the algorithm by using Keras. We intend to implement it by using PyTorch.
2. In the paper, authors suggest some future works. One of them is to predict whether a bug will be fixed or not by using DBRNN-A approach and *status* attribute of bug reports. We intend to put it into practice.
3. Authors applied transfer learning between different datasets and found it applicable. We, in addition to this, plan to ensemble models trained by different datasets and check whether learning process can be transferred to a completely new dataset.

References

[1] Senthil Mani, Anush Sankaran, and Rahul Aralikkatte. Deeptriage: Exploring the effectiveness of deep learning for bug triaging. *In Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, pages 171179. ACM, 2019.

¹<https://github.com/bilsengroup/bug-triaging-rnn>