Domain independent automatic feature engineering through Graph neural network

Oct. 20, 2020

1 BACKGROUND

Feature engineering is the process of using domain knowledge to extract features from raw data via data mining techniques[3]. The performance of a machine learning model depends to a large extent on feature engineering. This process requires expert knowledge and is typically time consuming. Due to limited human resources but steadily growing computing capacities, automating feature generation process becomes increasingly attractive. However, most of the Existing approaches are data set specific. For example [2] focus on stock price movement prediction and [1] needs to train new model for each data set. The core of our research is to build a model that can handle different data sets without further fine tune (or only with little effort).

2 BASIC IDEA

The basic idea is inspired by recommendation systems that call back based on content similarity calculations. In our research we represent the process of feature engineering with a graph, where each edge in the graph represents a transformation and the nodes connected to the edge indicate the accuracy before and after applying the transformation. We improve the generality of the model by removing specific information of the data set, such as number of columns, name of columns, etc., and instead use the improvement that each transformation brings to describe the data. Here, the graphic is the content and based on the content, we recommend the next transformation as in the recommendation system. However, there are still some challenges that need to be overcome:

- The definition of the available transformation set. Since we use transformations to describe data, this requires that our transformation theorem covers techniques in different domains, e.g. spatial, acoustic, etc.
- The design of the graph and graph neural network. There are many different types of graphs, such as directional graphs, direct acyclic graphs, hierarchical graphs, etc. Therefore, a key question is how to find the right graph type and graph representation.
- Evaluation. There is no benchmark for the feature engineering task. To evaluate the merits of our models, we have to select suitable data sets, replicate other useful models and compare them with our model on the selected data sets. In other words, we need to develop a benchmark for the feature engineering task.

3 METHODOLOGY

- Summarize different feature engineering transformations.
- Summarize and reproduce state of the art and baseline feature engineering methods.
- Summarize and reproduce state of art graph neural network methods.

- Go through open source data sets, define appropriate selection criteria and select multiple representatives in each domain.
- Model design and programming.
- Experiment and model finetune.

4 CONTRIBUTION

Three main contributions are realized in this work:

- Create a Feature engineering benchmark.
- Extend(or fix) the available feature transformation set.
- Propose a domain independent feature engineering algorithm with graph neural network.

References

[1] Udayan Khurana et al. "Cognito: Automated feature engineering for supervised learning". In: *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*. IEEE. 2016, pp. 1304–1307.

[2] Wen Long, Zhichen Lu, and Lingxiao Cui. "Deep learning-based feature engineering for stock price movement prediction". In: *Knowledge-Based Systems* 164 (2019), pp. 163–173.

[3] Andrew Ng. "Machine Learning and AI via Brain simulations". In: *Accessed: May* 3 (2013), p. 2018.