Premise selection with machine learning

Piotr Piękos

26 maja 2019

1 Introduction

2 General Description

3 Dataset creation

4 Models

5 Evaluation

6 Next Steps

- **Training LGBM on the whole dataset** - that simple thing could be made by using CLI (lesser memory requirements) or by training machine with more RAM, probably that single step would give a great improvement.
- **Iterative training** - theorems that are proved can later be used as samples for new training sets, which gets bigger with every iteration.
- **Hyperparameters** - There are a lot of parameters in the training. From lgbm parameters to 'meta-parameters':
  - Ratio of False to True samples in training dataset - setting it higher would probably improve results, as it would better resolve real scenario.
  - number of trees
  - trees depth
  - regularization
- **More stubborn prediction with dynamic number of premises** - For now the program doesn't even check whether solver gave up (timeout) or didn't manage to solve (not enough premises). It could check it

and increase number (start with small numbers) of premises until it finds proof or times out. That's important because for some theorems it may be sufficient to choose 2 premises, but for some many premises are needed. It's impossible to model with current architecture.

- **Better Features** - Using semantic dependencies between features or at least extracting more semantic features (as with autoencoder attempt).