Project Report
(Homework 10)

Python Programs Clustering

Yehor Atell Krasnopolskyi
Oliver Vainikko

Business understanding (Task 2)

Identifying our project goals

The data set for our project has been collected here, at the University of Tartu, by Reimo Palm, thus our project, besides being of scientific interest is also going to benefit the University of Tartu and its instructors by developing a tool to analyse similarities between the students' works that involve coding with Python. **The goal is therefore to develop a working python program classification tool.**

Assessing the situation

One of the main challenges we face here is that the data is absolutely not numerical. It is code, that is, it is basically text with a certain strict structure to it, which we need to analyse. Our approach is based on Python's abstract syntax trees (ASTs) that we first use to build sequences of tokens for each program in the data set, and then train a vectorizer on top of the latter. Overall, it is not a very computationally or monetary costly approach which can be run on a laptop.

The set of libraries we are going to use is relatively small: we need a certain submodule of `sklearn`, and a couple of tools for processing the ASTs (i.e. `visitors`). There's still a chance that this approach is not good enough. In this case, if the method will be proven to be insufficient for our purposes, we will need to modify it.

Our main assumption about the data should be clear at this point: we assume that analysing the structure of the ASTs with visitors is enough to quantify the difference between different pieces of code. In other words, we hope that there's enough information in this structure to make any further automated decisions.

Addressing the visitors' technique, these are the objects that produce sequences of tokens out of ASTs. In a nutshell, visitors are algorithms that walk through the tree in a certain way and collect node types into a single sequence, which we can then analyse.

Data-mining goals

There are numerous ways we can adjust the so-called "visitors". The main difference between different visitors is which types of nodes they save. One of the crucial goals for data-mining therefore is to determine which features (which types of nodes in an AST) contain the most relevant information for us.

A true success would be to develop an approach that:

1. Detects similarities between programs of the same structure but with different variable names
2. Detects similarities between programs of the same structure but with different constants or values
3. For any two programs that are clustered separately, they should have different structures.

Understanding the data (Task 3)

The data set is essentially a set of Python programs, grouped by purpose. It consists of Python programs, submitted by the students as the solutions to the homework tasks in the introductory Computer Programming course. The programs are grouped by homework number and task number; there are 13 homeworks and 33 tasks. The average number of submissions per task is around 300; the average program size is around 600 B.

Project planning (Task 4)

1. Make the vectorisation work (this includes building the abstract syntax trees, using a visitor on them, and training the vectorizer out of `sklearn`). This includes implementing the loading of the dataset.
2. Test different visitors and training approaches. Determine whether we should pre-train a part of our "model" or train it on each task separately, meaning that we can even have a different vocabulary on each set of submissions.
3. Develop the clustering part, choose an algorithm (or several) that would cluster programs given the similarity matrix.
4. Edit the code to make it reusable (maybe in the form of a Python module). Write the readme file and documentation.