

Machine Learning Engineer Nanodegree

Capstone Proposal

Linggih Saputro

October 1st, 2017

Proposal

(approx. 2-3 pages)

Domain Background

The project is about classifying clinically actionable genetic mutations from Kaggle. This competition is actually in precision medicine, which about the genetic testing of cancer. Most of the data is based on the text-based clinical literature, so this will be a combination of text analysis or a natural language processing problem with a classification problem of semisupervised learning. SSL is half supervised and half unsupervised learning and it divides the data into labeled and unlabeled data sets (Chapelle et al., 2009)

Natural Language Processing (NLP) is the study of mathematical and computational modeling of various aspects of language and the development of a wide range of systems. (Joshi AJ., 1991)

This problem motivates me the most because I study in bioinformatics and I have a particular interest in precision medicine. I hope by taking this project I will get more insights about the domain and more opportunities to explore.

Joshi, A. J. (1991). Natural language processing. *Science*, 1242-1249.

Chapelle, O., Scholkopf, B., & Zien, A. (2009). Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3), 542-542.

Problem Statement

The main goal in this project is to eliminate the manual work of annotating the classes of the cancer based on evidence of the genetic mutations from the text-based clinical literature. Because of this problem is related to natural language processing as they have a lot of text-based literature, it is a quantifiable by tokenizing and weighing the words.

Datasets and Inputs

(approx. 2-3 paragraphs)

In this competition you will develop algorithms to classify genetic mutations based on clinical evidence (text).

There are nine different classes a genetic mutation can be classified on.

This is not a trivial task since interpreting clinical evidence is very challenging even for human specialists. Therefore, modeling the clinical evidence (text) will be critical for the success of your approach.

Both, training and test, data sets are provided via two different files. One (training/test_variants) provides the information about the genetic mutations, whereas the other (training/test_text) provides the clinical evidence (text) that our human experts used to classify the genetic mutations. Both are linked via the ID field.

Therefore the genetic mutation (row) with ID=15 in the file training_variants, was classified using the clinical evidence (text) from the row with ID=15 in the file training_text

- training_variants - a comma separated file containing the description of the genetic mutations used for training. Fields are ID (the id of the row used to link the mutation to the clinical evidence), Gene (the gene where this genetic mutation is located), Variation (the aminoacid change for this mutations), Class (1-9 the class this genetic mutation has been classified on)
- training_text - a double pipe (||) delimited file that contains the clinical evidence (text) used to classify genetic mutations. Fields are ID (the id of the row used to link the clinical evidence to the genetic mutation), Text (the clinical evidence used to classify the genetic mutation)

- test_variants - a comma separated file containing the description of the genetic mutations used for training. Fields are ID (the id of the row used to link the mutation to the clinical evidence), Gene (the gene where this genetic mutation is located), Variation (the aminoacid change for this mutations)
- test_text - a double pipe (||) delimited file that contains the clinical evidence (text) used to classify genetic mutations. Fields are ID (the id of the row used to link the clinical evidence to the genetic mutation), Text (the clinical evidence used to classify the genetic mutation)
- submissionSample - a sample submission file in the correct format

Solution Statement

The solution I propose is to use the scikit-learn module in NLP to tackle this. Then after all the data is transformed from text to array, the data will be splitted to training and testing data. Support vector machine will be used to classifying the class.

Benchmark Model

For the benchmark model, naive bayes will be used to compare it with the solution.

Evaluation Metrics

(approx. 1-2 paragraphs)

I will implement the cross validation method using GridSearchCV, to prevent overfitting, a situation that create a perfect score but tends to fail with the new data.

Project Design

The theoretical workflow will start with the input data.

1. Preprocessing Input data
2. Transform the data into the vectors
3. Perform the cross-validation
4. Classify using the support vector machine.