Challenges in Building an Automatic ICD-10 Codes Recognition System

Yuting Hu
MSc Bioinformatics
Leiden University
Project Summary

- Company: Performation
- Data source: Diakonessenhuis 2018

Patients health records
- Text data
  - discharge letters
  - care activities
  ...
- Structured data
  - age
  - time of stay
  ...

ICD-10 codes
- Medical classification list by the WHO
- Unique code per disease
- Hierarchical structure
Project Summary

Previous Case Study
- Data of year 2017
- Only text data (discharge letters)
- Text cNN – 0.87 f1 scores
- Main diagnosis
- Only one specialty group (12k)

This Case Study
- Data of year 2018 (Latest at that time)
- Combine two types of data
- Mainly focus on non-deep classifiers
- Main diagnosis
- Only one specialty group (22k)

Group of diseases/ doctors
Different codes per hospital
Noted in the health records

Main diagnosis – 1 disease/ ICD-10
Secondary diagnosis – 0 or multiple
Data Collection & Analysis

• **Strict data access & Messy data**
  - Can only be used under the internal server
  - Managed and collected with SQL commands

• **Basic Analysis**

  **General info**
  - Largest specialty group 20% of whole dataset (22,255 out of 107,462 samples)
  - 204 classes (ICD-10)
  - 75% samples gathered at the top 40 classes (really long-tail)

  **Class label transformation**
Data Pre-processing

• Clean text data
• Encoding non-numeric structured data – one-hot encoding first
• Analyze the data again

Text Data
- Min, Max, Avg text length:
  7, 1881 and 152 words

Structured Data
- Super sparse (after one-hot encoding)
- Too many DBC codes types

• Apply Hash encoding on DBC codes
  - FeatureHasher() in scikit-learn
  - hash to 5 features

Medical care product code
One code per 120 days (1 or multiple)
Important for diagnosis
Data Pre-processing

- Split train-test-valid set (0.8/0.1/0.1)

- Resampling
  - random-oversampling
  - only on train set

Text Data

Vectorization
- tf-idf for shallow models
- word2vec for deep models

Structured Data

Max-min scaler
- important for some classifiers
- meaningless to numeric but nominal data
- fit on train, then transform on test/valid set

- Analyze the data after all pre-processing
  - samples in each class: same, 6102
  - samples in training set: 17,804 → 250,182
How to Combine Two Types of Data?

• Directly connect data together
  - tried in previous case study
decrease signal to noise ratio (text classifiers)

• Classifier Combiner
  - text classifiers + structure classifiers
  - two combining rules

  Combiner classifier: Logistic Regression classifier

Apply trained base-classifiers on both train and test sets
Apply trained base-classifiers only on test sets
Classifier Models

• Shallow models
  - Naïve Bayes Classifier: 2-gram, alpha
  - SVM Classifier: 2-gram, C, linear/SGD

• Deep models
  - Text cNN
    Convolutional neural networks for sentence classification.
    Yoon Kim, 2014.

• Boost models
  - Fast Text
  - Xgboost: only on structured data
Results

Optimal Model

• Previous model – 0.73 f1
• SVM(Linear) – 0.98 f1 score 😊
• Over-sampling improved a lot (0.74 f1 before)

Classifier Combiner

• Did not work well 😞 - less than 0.4 f1
• Possible reasons
  - ‘Prediction’ more than ‘Recognition’
  - Need more data types & more samples
Possible Future Work

• Extension of task
  - whole dataset/new dataset
  - secondary diagnosis

• The special DBC code format
  
  CHI-11-318-020107056

• Transfer learning?
Main Challenges

• Strict data access
• Low computing power
• Mixed & messy structured data
• High dimensionality & sparse
• Long-tail
• Combine two types of data