**Convolutional Neural Network and word2vec-based Model for gestational diabetes prediction**

**Data source:** MIMIC iv

MIMIC (Medical Information Mart for Intensive Care) database is a public, large, single-center multi-parameter intensive care database provided by MIT, comprising physiological data of patients collected from ICU. MIMIC iv is the new release of MIMIC database. After more than 10 years of construction in multiple disciplines, the database has been successfully used in many research fields of clinical data mining and biomedical data science. All data resources in MIMIC are freely open to researchers around the world after deidentify process for patient privacy protection. and can be accessed through <https://physionet.org/content/mimiciv/0.4/>.

However due to confidentiality agreements, we cannot publish raw data we used in our project and the processed data after data processing.

**Data used in code:**

admissions.csv

diagnoses\_icd.csv

procedures\_icd.csv

prescriptions.csv

patients.csv

d\_items.csv

procedureevents.csv

**Folder:**

code: include all the codes

cleaned\_data: all the data after processing

word2vec\_model: word2vec model

output: accuracy and graphs

**Code:** (in the code folder. Please run in order)

**Initial\_ preprocessing.ipynb** -> **data preprocessing.** Load the data from raw mimic files to get all events\_data.csv for each patient

**date\_token\_event.ipynb** -> **data preprocessing.** Padding time token features into patients' event data

**train\_word2vec.ipynb** -> train word2vec model(website used in this code: <https://www.tensorflow.org/tutorials/text/word2vec>

https://radimrehurek.com/gensim/models/word2vec.html)

**divide\_case\_control.ipynb** -> **data preprocessing.** Divide patients and filter out patients and obtain case and control data

**classicModels** \*.ipynb -> train baseline prediction models **(including accuracy)**

**cnn\_withoutTime.ipynb** -> CNN analysis **(including graphs, accuracy, and final result)**

**List of website used in coding**

**Word2vec model code learn form these blog**

<https://www.analyticsvidhya.com/blog/2021/07/word2vec-for-word-embeddings-a-beginners-guide/>

<https://www.tensorflow.org/tutorials/text/word2vec>

<https://radimrehurek.com/gensim/models/word2vec.html>

<https://github.com/harmanpreet93/train-word2vec/blob/master/train-word2vec.ipynb>

**Cross-validation**

<https://scikit-learn.org/stable/modules/cross_validation.html>

**Logistic Regression**

<https://www.datacamp.com/tutorial/understanding-logistic-regression-python>

**SVM support vector machine**

<https://scikit-learn.org/dev/versions.html>

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html>

**Decision Tree**

<https://scikit-learn.org/stable/modules/tree.html>

**Random Forest**

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

**Convolutional Neural Network learn form this**

<https://github.com/leriomaggio/deep-learning-keras-tensorflow/blob/master/4.%20Convolutional%20Neural%20Networks/4.1%20Convolutional%20Neural%20Networks.ipynb>

**T-distributed Stochastic Neighbor Embedding to visualize the patient representations by CNN pooling layer**

<https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html>

**K-Means clustering to show clustered the representation from case patients**

<https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>