DIGITAL MEDICINE

CASE1:

OBESITY DETECTION

GROUP3

葉詠富 310554031

游智鈞 310551059

高承翰 310551106

CONTENTS

01 Introduction

02 Text Preprocessing

03 Method

04 Result

01 INTRODCUTION

TARGET AND DATASET

Target

Use a doctor's diagnosis certificate to determine whether the patient is obese.

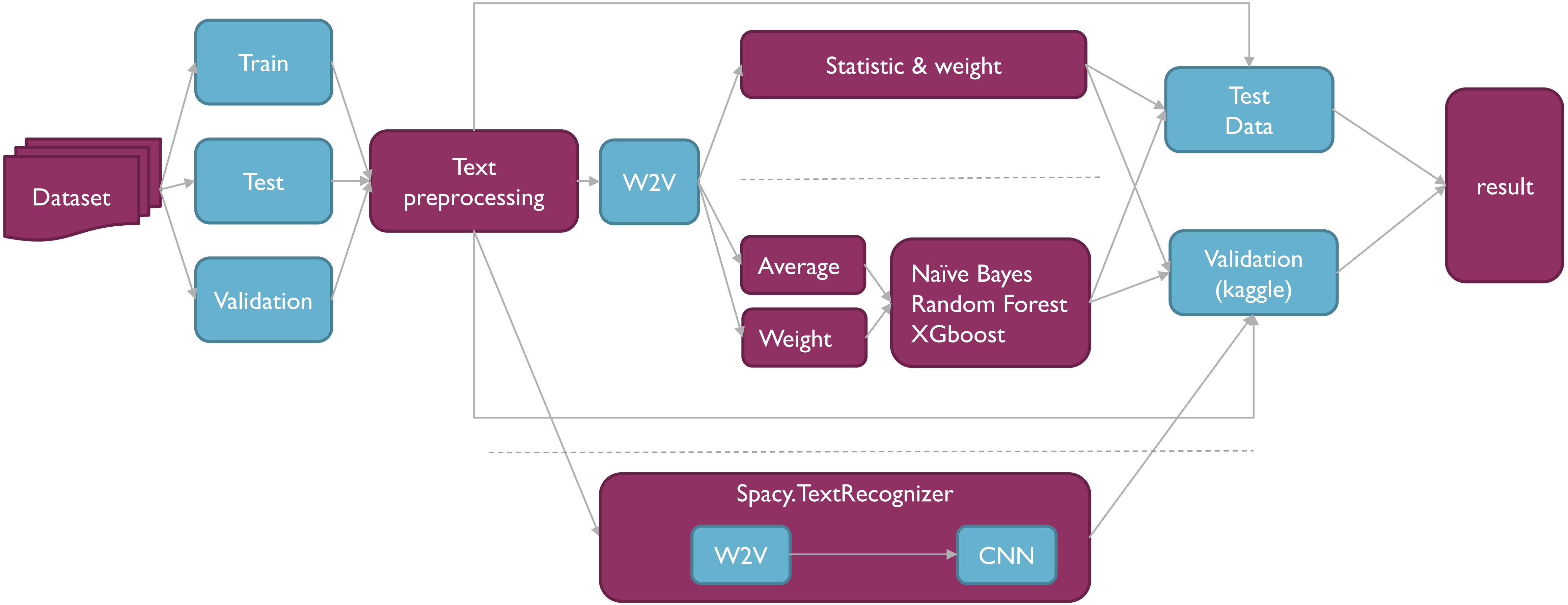
Train/Test Dataset

1. Training data based on textual judgement
   - Textual judgement: 200 cases obesity vs. 200 cases unmentioned.
2. Testing data based on intuitive judgement
   - Intuitive judgement: 200 cases obesity vs. 200 cases absence

Validation Dataset

Validation data (50 cases) based on textual judgement

DATA PIPELINE



02 TEXT PREPRECESSING

TEXT CLEAR

a. Remove punctuation

Remove punctuation and numbers to make word split more precise.

b. Word tokenize

The process of splitting a large sample of text into words.

c. Remove stopword

Used to improve the quality of text features or reduce the dimensionality of text features.

d. Lemmatize

Lemmatization is to remove the affixes of the word and extract the main part of the word.
For example, the word "cars" after lemmatize is "car", and the word "ate" after lemmatize is "eat".

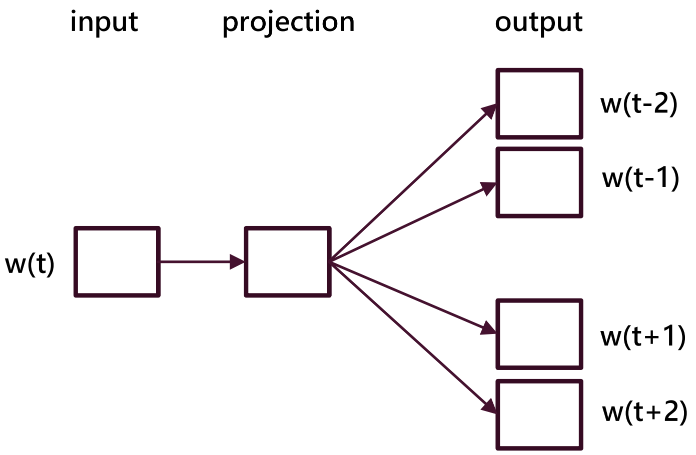
W2V

Advantage

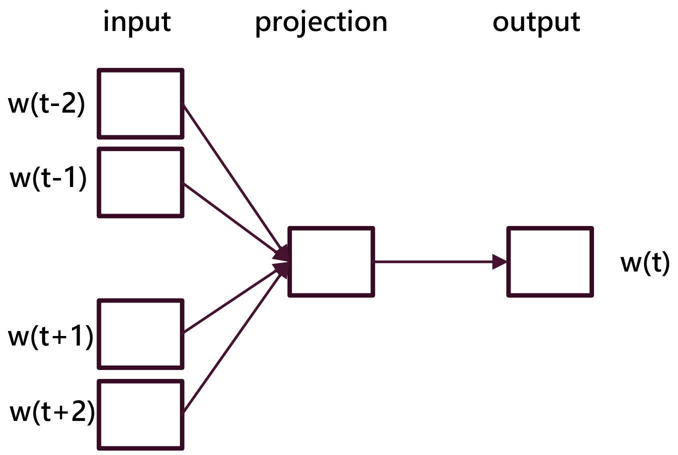
Compared with one hot encoding, Word to vector can consider a word in the context of the article.

Algorithm

word to vector contains two algorithms, Skip-gram and CBOW.
Skip-gram uses the central word to predict the context, and CBOW uses the context to predict the central word.



Skip-gram



CBOW

03 METHOD

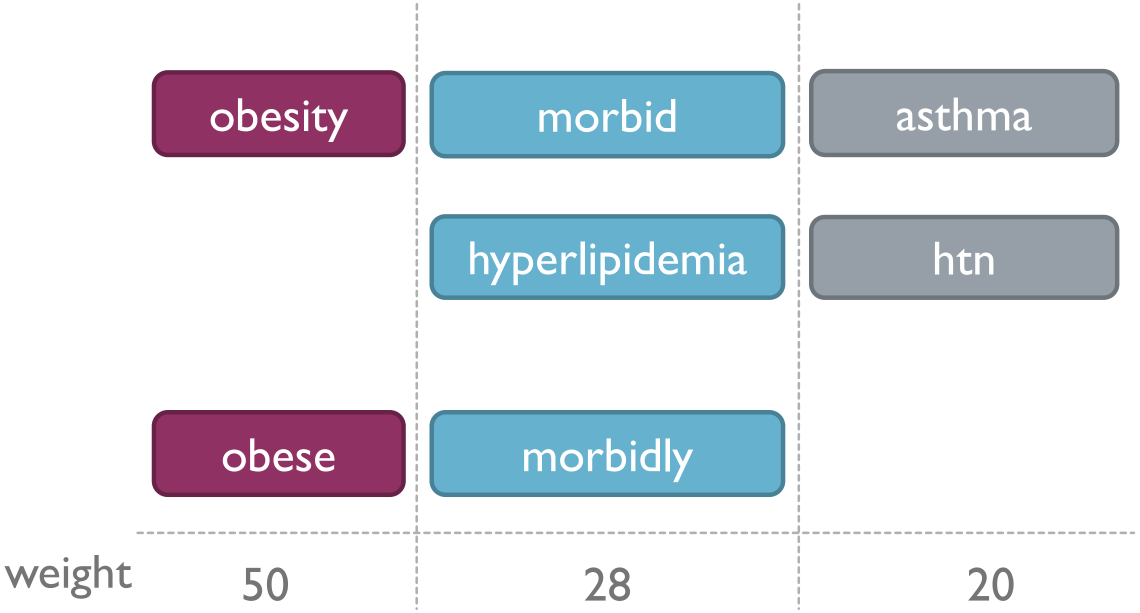
STATISTIC METHOD

a. Most similar

Use cosine to calculate the angle and find the most similar word of “obesity” and “obese”.

b. Weight

1. obesity and obese are key words for obesity, so 50 points are given for evaluation.
2. Morbidly, morbid, hyperlipidemia and obesity-related words are the most similar.
3. Asthma and htn are not so close, so give a weight of 20 points.



	obesity	morbid	asthma
		hyperlipidemia	htn
	obese	morbidly	
weight	50	28	20

c. Criterion

Count the weight of an article, weight greater than 50 points is obesity.

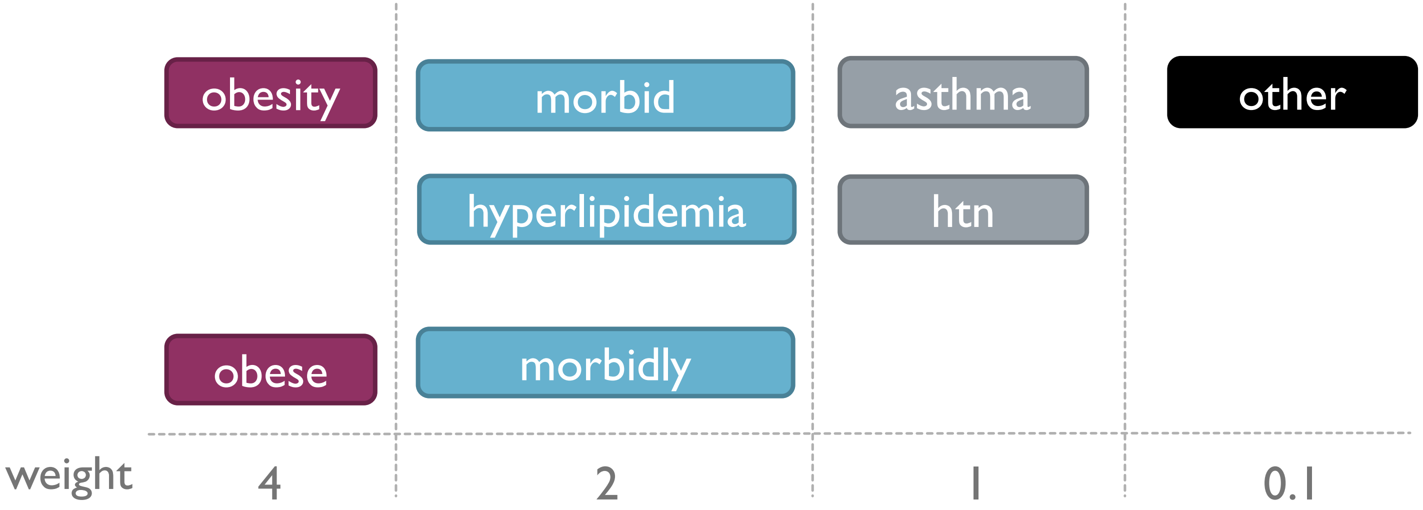
MACHINE LEARNING ARTICLE VECTOR

A. Average

Calculate the average vector of the article and use it as the article vector.

B. Weight

Calculate the weight vector of the article, give the weight to the key words, and use it as the article vector.



weight	4	2	1	0.1
	obesity	morbid	asthma	other
		hyperlipidemia	htn	
	obese	morbidly		

MACHINE LEARNING CLASSIFICATION ALGORITHM

Naïve Bayes

Naive Bayes is a classification model based on calculating the probability of conditions. By assuming that each event is independent, the probability under each condition can be calculated to obtain the probability of the event (category) occurring

Random Forest

The (random forest) algorithm establishes the outcome based on the predictions of the decision trees. It predicts by taking the average or mean of the output from various trees. Increasing the number of trees increases the precision of the outcome.

XGboost

XGboost (Extreme Gradient Boosting) is a Gradient Boosted Tree (GBDT) that keeps the original model unchanged every time, and adds a new function to the model to correct the error of the previous tree to improve the overall model. Mainly used to solve the problem of supervision is learning, can be used for classification can also be used for regression problems.

DEEP LEARNING SPACY.TEXTRECOGNIZER

W2V

Same as above

CNN

Convolutional neural network (CNN, or ConvNet) is a class of artificial neural network, most commonly applied to analyze visual imagery. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on the shared-weight architecture of the convolution kernels or filters that slide along input features and provide translation equivariant responses known as feature maps.

Spacy.TextRecognizer

The model supports classification with multiple, non-mutually exclusive labels. By default, the TextCategorizer class uses a convolutional neural network to assign position-sensitive vectors to each word in the document. The TextCategorizer uses its own CNN model, to avoid sharing weights with the other pipeline components. The document tensor is then summarized by concatenating max and mean pooling, and a multilayer perceptron is used to predict an output vector of length nr_class. The value of each output neuron is the probability that some class is present.

04 RESULT

STATISTIC RESULT

a. Test dataset

Precision	0.985
Recall	0.96
Accuracy	0.975
f1	0.975

b. Validation

f1	0.543

MACHINE LEARNING RESULT (A)

Naïve Bayes

Random Forest

XGboost

a. Test dataset

Naïve Bayes

Precision	0.70
Recall	0.63
Accuracy	0.70
f1	0.66

Random Forest

Precision	0.72
Recall	0.68
Accuracy	0.73
f1	0.70

XGboost

Precision	0.73
Recall	0.73
Accuracy	0.75
f1	0.73

b. Validation

Naïve Bayes

f1	0.543

Random Forest

f1	0.485

XGboost

f1	0.514

MACHINE LEARNING RESULT (B)

Naïve Bayes

Random Forest

XGboost

a. Test dataset

Precision	1.0
Recall	0.9
Accuracy	0.95
f1	0.95

Precision	1.0
Recall	0.95
Accuracy	0.975
f1	0.974

Precision	1.0
Recall	0.95
Accuracy	0.975
f1	0.974

b. Validation

f1	0.543

f1	0.57

f1	0.48

DEEP LEARNING RESULT

a. Test dataset

Precision	0.72
Recall	0.72
Accuracy	0.72
f1	0.72

b. Validation

f1	0.514

CONCLUSION

a. Overfitting

no	Problem	Improve
1	Train dataset is too small.	More train dataset.
2	Bad weight design.	More dataset to reference.
3	Test dataset vs validation dataset too different.	Pick data sets more evenly.

b. More try

1. Redesign and reduce word vector.
2. Word Clustering by K-Means、DBCAN.

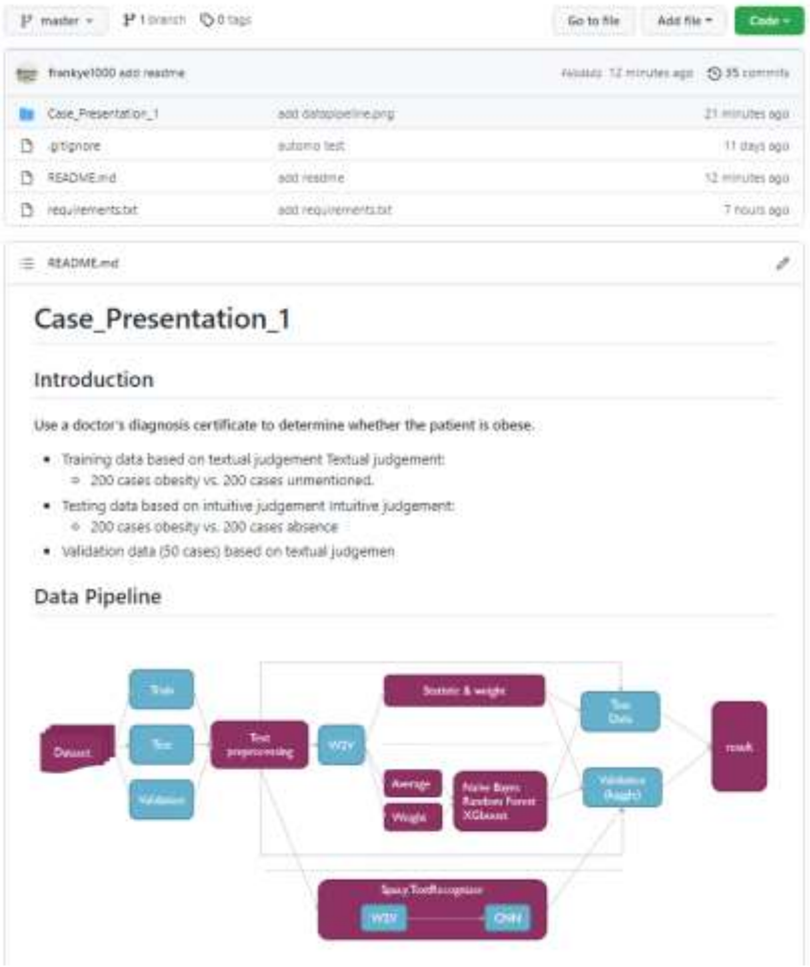
THANK YOU

GITHUB

GITHUB

Case presentation 1

https://github.com/frankye1000/NYCU-DigitalMedicine



frankye1000 add readme [illegible] 12 minutes ago 35 commits

Case_Presentation_1	add datapipeline.png	21 minutes ago
.gitignore	automo test	11 days ago
README.md	add readme	12 minutes ago
requirements.txt	add requirements.txt	7 hours ago

README.md

Case_Presentation_1

Introduction

Use a doctor's diagnosis certificate to determine whether the patient is obese.

- Training data based on textual judgement Textual judgement:
  - 200 cases obesity vs. 200 cases unmentioned.
- Testing data based on intuitive judgement Intuitive judgement:
  - 200 cases obesity vs. 200 cases absence
- Validation data (50 cases) based on textual judgemen

Data Pipeline

REFERENCE

REFERENCE

[1]python 去除所有的中文 英文标点符号(https://blog.csdn.net/weixin_38819889/article/details/105389248)

[2]Python處理中文標點符號大集合(https://codertw.com/%E7%A8%8B%E5%BC%8F%E8%AA%9E%E8%A8%80/356827/)

[3]英文自然語言處理的經典工具 NLTK(https://clay-atlas.com/blog/2019/07/30/nlp-python-cn-nltk-kit/)

[4] Word2Vec的参数解释(https://blog.csdn.net/laobai1015/article/details/86540813)

[5] NLP入门（三）词形还原（Lemmatization）(https://www.cnblogs.com/jclian91/p/9898511.html)

[6] Word2Vec的簡易教學與參數調整指南(https://www.kaggle.com/jerrykuo7727/word2vec)

[7]新手村逃脫！初心者的 Python 機器學習攻略 1.0.0 documentation(https://yaojenkuo.io/ml-newbies/07-performance.html)

[8] Word2vec from scratch (Skip-gram & CBOW)(https://medium.com/@pocheng0118/word2vec-from-scratch-skip-gram-cbow-98fd17385945)

REFERENCE

[9] ML入門（十七）隨機森林(Random Forest) (https://medium.com/chung-yi/ml%E5%85%A5%E9%96%80-%E5%8D%81%E4%B8%83-%E9%9A%A8%E6%A9%9F%E6%A3%AE%E6%9E%97-random-forest-6afc24871857)

[10] scikit-learn Machine Learning in Python (https://scikit-learn.org/stable/index.html)

[11] Introduction to Random Forest in Machine Learning(https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/)

[12]機器學習之XGBoost分類器XGBClassifier-- xgb使用sklearn介面(https://www.itread01.com/content/1545533828.html)

[13] Python xgboost 模块，XGBClassifier() 实例源码(https://codingdict.com/sources/py/xgboost/12189.html)

[14] Introduction to Naive Bayes: A Probability-Based Classification Algorithm(https://blog.paperspace.com/introduction-to-naive-bayes/)

CONTRIBUTION OF GROUP MEMBERS

CONTRIBUTE

Name	Responsible	contact
葉詠富	1. Data clear 2. Word to vector 3. Try Statistic method 4. Try Machine learning method 5. PPT & github design	frankye100.c@nycu.edu.tw
游智鈞	1. Data clear 2. Word to vector 3. Try Statistic method 4. Try Deep learning method 5. PPT design	thomas91714@gmail.com
高承翰	1. Data clear 2. Word to vector 3. Try Statistic method 4. PPT design	climnehcc234@gmail.com