Second Assignment of Studies on International Integrated Sciences

Yumeki Goto
student ID : 33F23012

July 11, 2023

Abstract

The correspondence between assignment requirements and sections in this report is the following. The applying different kernel functions of Support Vector Machine(SVM) is written in section 3, 4, and what kinds of parameter values I used are written in Section 1. Then, the analysis of my selection of features and visualization of the data by dimensional reduction are written in section 2.

1 Data: My selected features

property	features	Explanation
(given)	id	The identification number corresponding to the word
	word	The word(not used)
label	difficulty	The difficulty of the English word (label)
length	len	The length of the word
frequency	frequency	The frequency of the word in Wikipedia corpus
vowels&consonant	vowels	The number of vowels of the word
	consonant	The number of consonants of the word
Word2vec information	word2vec_0sim	The similarity of the first closest word
	word2vec_25sim	The similarity of the 26th closest word
	word2vec_50sim	The similarity of the 51th closest word
	word2vec_75sim	The similarity of the 76th closest word
	word2vec_100sim	The similarity of the 101th closest word

Table 1: CSV properties: The properties and their explanation in the used CSV file.

Table 1 shows an overview of the properties of the CSV file, which I used for the prediction. I used four kinds of properties to predict the difficulty of the English word based on the following four assumptions.

First, I assume that the difficult English word is likely to have long characters. For instance, the word 'juxtaposition', whose difficulty is 12, has 13 characters, on the other hand, the word 'men', whose difficulty is 1, has three characters. I obtained the length of the word as the property 'len'. The second assumption is that the word unlikely appears in the sentence as the word is more

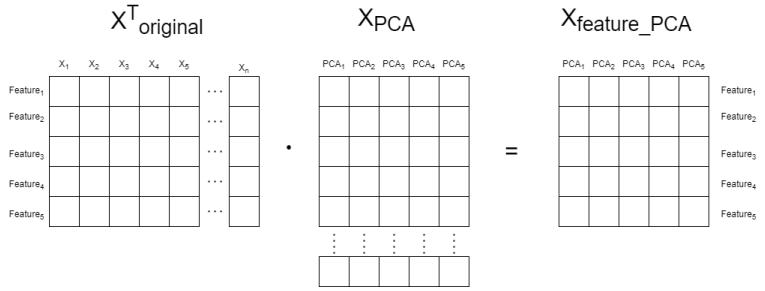

Figure 1: Calculation of the matrix on the importance of the features.

difficult. For example, the word 'resplendent' appears unlikely in the sentence, on the other hand, the word 'beautiful' is often used in the sentence. The frequency of the word was obtained by using the Wikipedia corpus. Third, I used the number of vowels and consonants as the properties 'vowels' and 'consonants' from the assumption that the difficult word has many consonants. Finally, as the fourth assumption, I assumed that the easy word is used in many situations, so it is substituted for few words because it is generally used. For example, the word 'have' has many meanings and is used in a lot of situations, so can be substituted for few words. To use this assumption, I used the similarity of the word by using Word2vec. Word2vec outputs similar words depending on the similarity when the word is fed into Word2vec. Then, to use the information on the similarity, I used the similarities of the first, 26th, 51st, 76th, and 101st words. Then, the cosine similarity between two words is used as the similarity.

2 Data: The analysis of my selected features by PCA

2.1 The importance of the features

In this section, I show two findings on my selected features in used data. First, I found at least three features are necessary to predict the difficulty of the word. This is found by analyzing the importance of the features by the Principal Component Analysis(PCA). Second, I tried to visualize the dataset on the 2-dimensional or 3-dimensional figures by using PCA.

First, I mention that at least three features(*frequency, len, similarity by Word2vec*) are necessary to predict the difficulty of the word. This result is led by the method analyzing how each feature and each PCA's value are dependent on each other [1].

I explain this method. First, this method calculates the same number of principal components X_{pca} as the original dataset $X_{original}$ by PCA. Then, the matrix, which shows the importance of the features, is calculated based on the calculation in Figure 1. Here, each element of the matrix is the inner product between a feature and a principal component. Thus, each element indicates how the feature depends on the principal component. If the element is close to 0, the feature is independent of the principal component. Then, in the case of the first principal component, it indicated the feature is less important. In the case

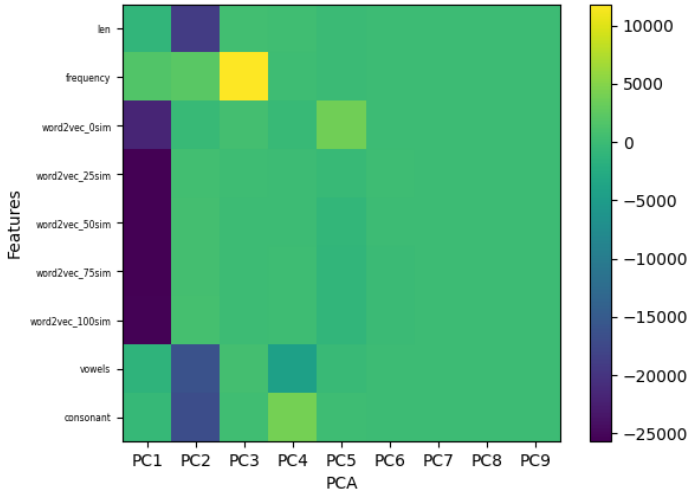

Figure 2: The matrix: The importance of the feature

the absolute of the element has a large value, the feature is important to predict the difficulty of the word because the feature strongly relates to the principal components. This is how I can identify which feature is important to predict the difficulty of the word by seeing the element of $X_{feature\_PCA}$.

Figure 2 shows the calculated matrix. As can be seen from the figure, the similarity by Word2vec is the most important by the inner product with the first principal component(PC1). Then, the length of the word and the frequency has a large value corresponding to PC2 and PC3. This is why the three features are necessary to predict, the similarity by Word2vec, the length of the word, and the frequency.

2.2 The dimensionality reduction to the data

I show the result of the dimensionality reduction of the data in Figure 3 and 4. As can be seen from Figure 3, it is difficult to predict the difficulty of the word by only two features. This is because the figure shows samples are not separated. On the flip side, as shown in Figure 4, the samples can be identified the difficulty by three features. This result is the same as the result in Section 2.1.

3 Model: The result of SVM prediction

I tied the five kernel functions to predict the difficulty of the word. Table 2 shows the result depending on the following five scenarios and five SVM settings. The first scenario uses all properties of data. Then, the scenarios from the second scenario to the fifth scenario use only one property, such as only Word2vec information, only length information, and so on. This property is corresponding to the Table 1. Here, the accuracy is computed by the test dataset, which is not overlapped with the training dataset.

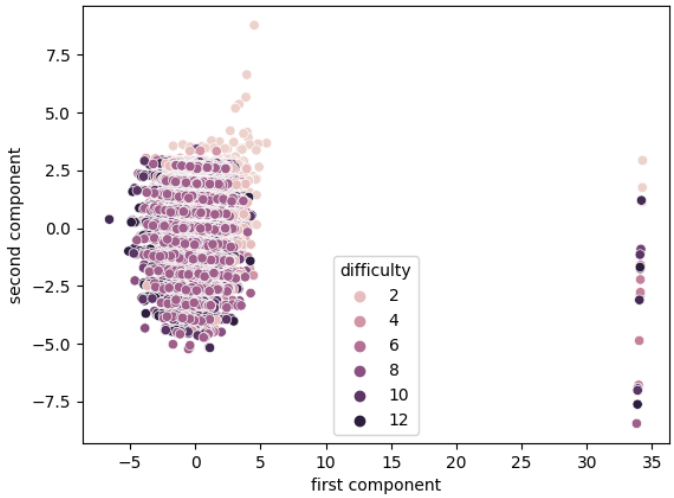

Figure 3: The result of dimensionality reduction: two dimensions

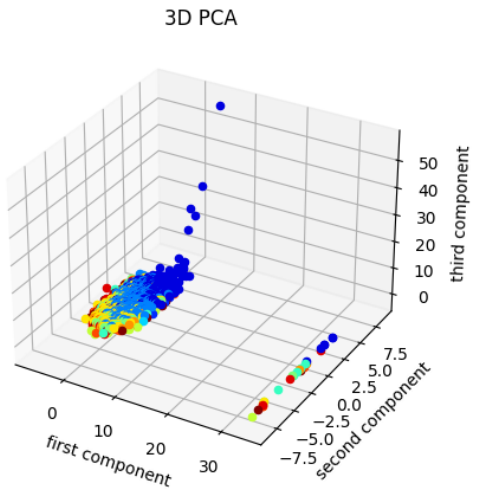

Figure 4: The result of dimensionality reduction: three dimensions

used properties	SVM1	SVM2	SVM3	SVM4	SVM5
all	0.179	0.118	0.15	0.157	0.159
only Word2vec information	**0.126**	0.086	0.126	0.127	0.127
only length information	0.121	0.100	0.127	0.128	0.124
only vowels consonant information	**0.130**	0.096	0.131	0.126	0.135
only frequency information	0.110	0.090	0.086	0.098	0.11

Table 2: Test accuracy depending on the scenarios and SVM settings

The SVM setting	Explanation
SVM1	liner kernel function
SVM2	polynomial kernel function ($degree = 2$)
SVM3	RBF kernel function ($gamma = 0.01$)
SVM4	RBF kernel function ($gamma = 0.05$)
SVM5	RBF kernel function ($gamma = 0.5$)

Table 3: The correspondence between the setting and kernel function

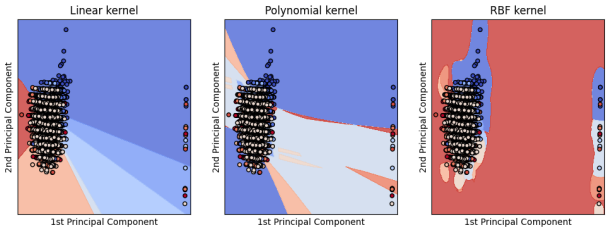

Figure 5: The analysis of SVM model depending on various kernel functions

Then, on the five SVM settings, I used different kernel functions in each setting. Table 3 shows the correspondence between the SVM settings and kernel functions.

Finally, I explain the result of the prediction. As shown in Table 2, the scenario, which used all properties of data, achieved the best test accuracy. Then, surprisingly, the scenario, which used only Word2vec information and only vowels&consonant information, achieves good accuracy.

Then, the kernel function for the low-dimensional dataset achieved better accuracy than the kernel functions. For example, the linear kernel function and RBF kernel function, whose gamma is large, achieved better accuracy in Table 2. Then, they are often used to predict with low-dimensional dataset.

As a conclusion, SVM by kernel function for low-dimensional data achieved the best accuracy when training with all properties. This indicates that there is no bad property, which deteriorates the prediction.

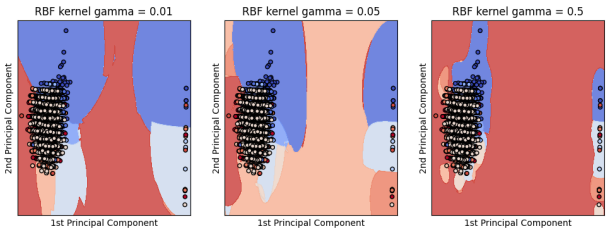

Figure 6: The analysis of SVM model depending on RBF kernel functions

4 Model: Discuss SVM prediction

I show the analysis of SVM models in Figure 5 and 6. As shown in Figure6, the RBF kernel function, whose gamma is large, works well to predict the difficulty of the word. This is because the SVM is generalized well. On the other hand, I am not sure why the liner kernel function also worked well. Because I can not find the crucial difference on Figure 5 between the liner kernel function and the polynomial kernel function, whose accuracy is bad.

A Code correspondence

I include the codes I used in the directory 'codes'. Actually, I used many codes and the flow is complicated. Thus, I show how I used codes in Figure 7.

First, I extracted 10,000 sentences(**wiki_en_out.txt**) from large wikipedia corpus by **make_wiki_corpus.py**. Then, by using these large sentences, I calculated the frequency of words and made a Word2vec model. Second, by **handle_data.py**, I made a CSV file, which is used for the prediction. Third, I execute the prediction by using three codes(**svm_analyze.py**, **svm_analyze_2.py**, and **svm_analyze_3.py**).

Finally, I analyze the data by **feature_importance_by_pca.py** and **dimensionality_reduction_2d_3d.py**.

References

[1] Kaushik, G.: Visualization tools for feature importance and principal component analysis (2018), https://medium.com/@gaurav_bio/creating-visualizations-to-better-understand-your-data-and-models-part-1-a51e7e5af9c0

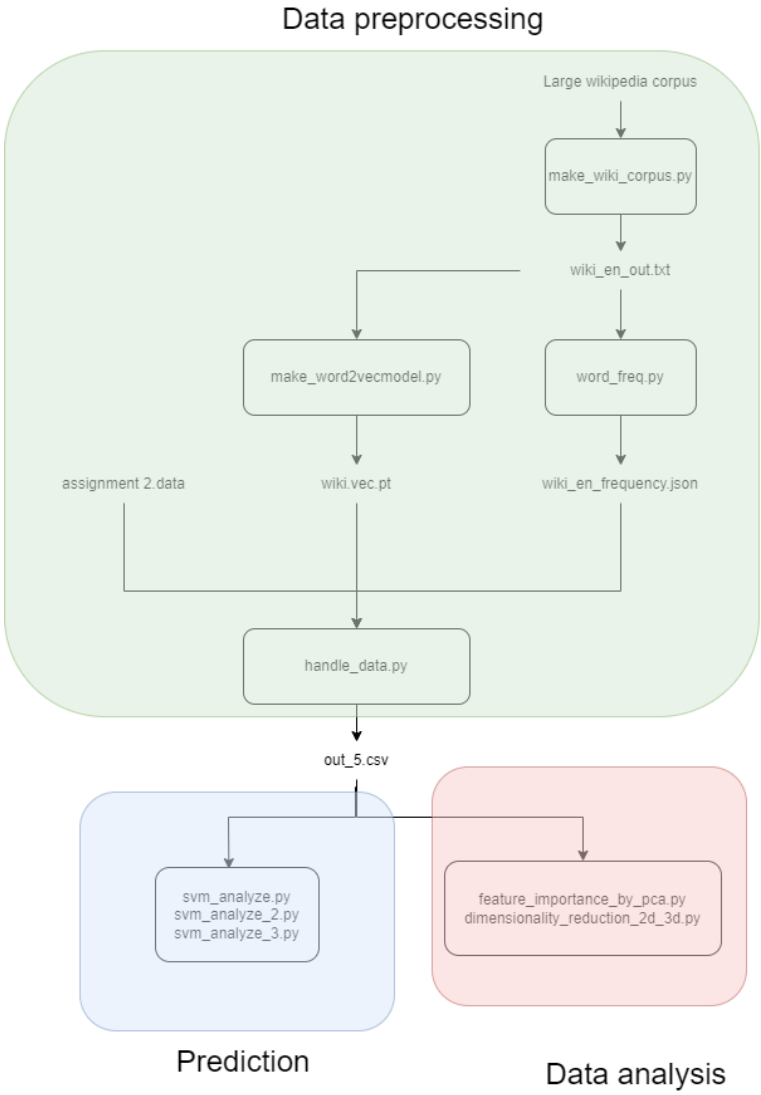

Figure 7: The flow of codes