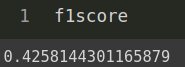
**Artificial Intelligence In Medical Applications and Services**

**Homework3 Report**

**1. Score Result**

****

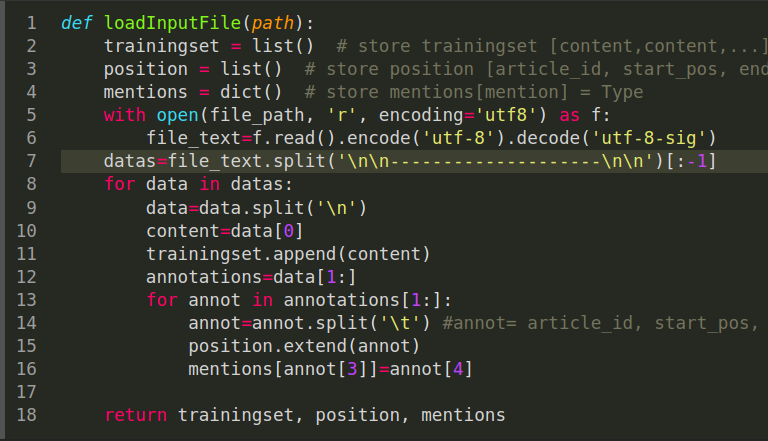
**2. Process**

As the TA mentioned, we can seperate the whole project into 2 parts which are

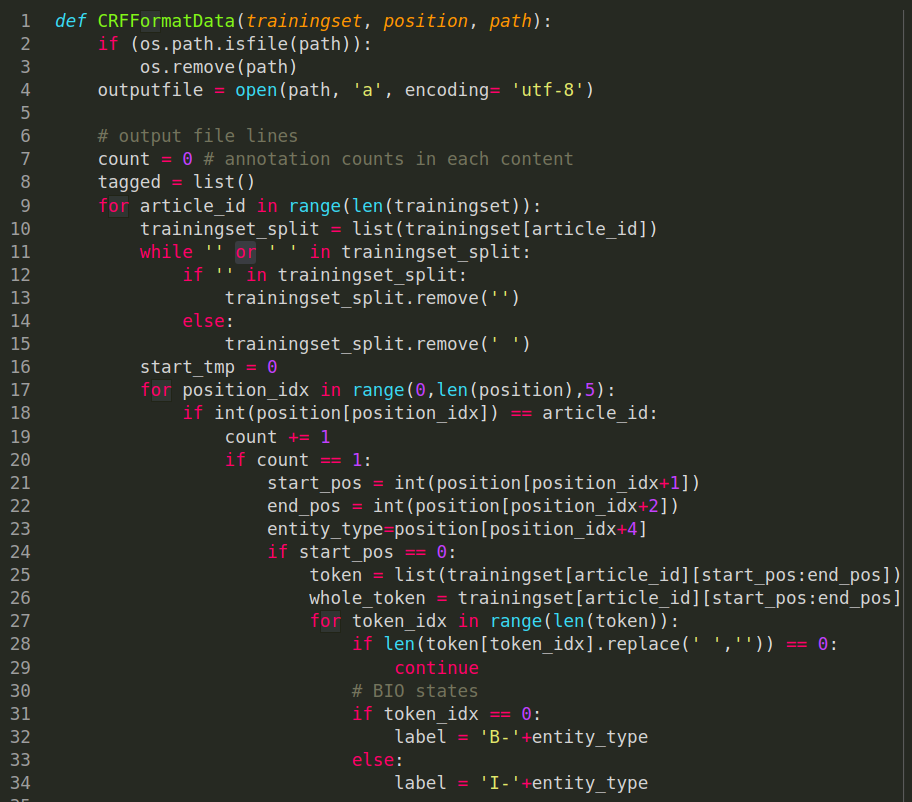
**Preprocessing** and **Model\_Training.**

1. **Preprocessing**

First we use **loadInputFile** function toload the data given by TA (SampleData\_deid.txt).

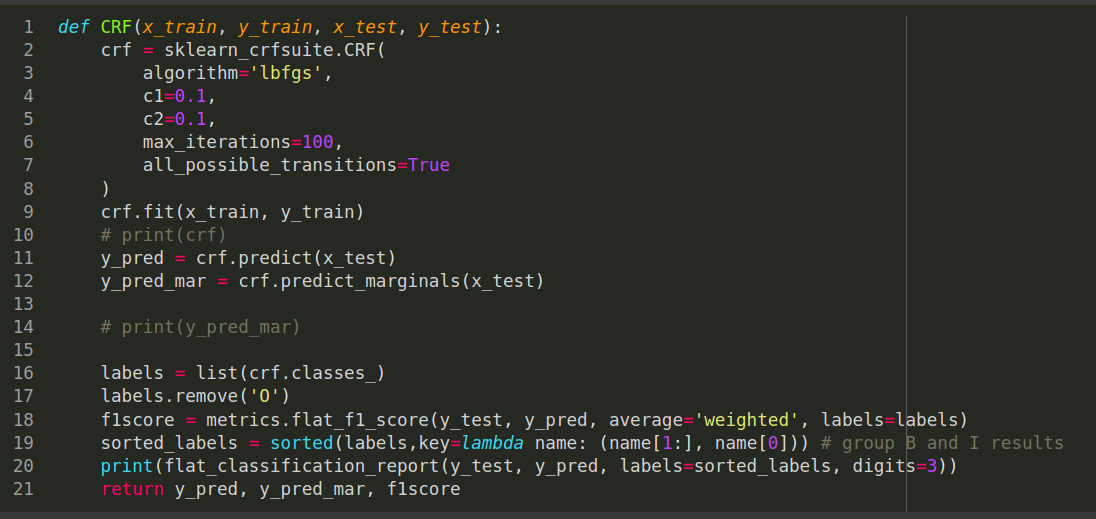


Then use CRFFormatData to change the input data into a corresponding format (clear the space, adding 'B' 'O' 'I' label)

****

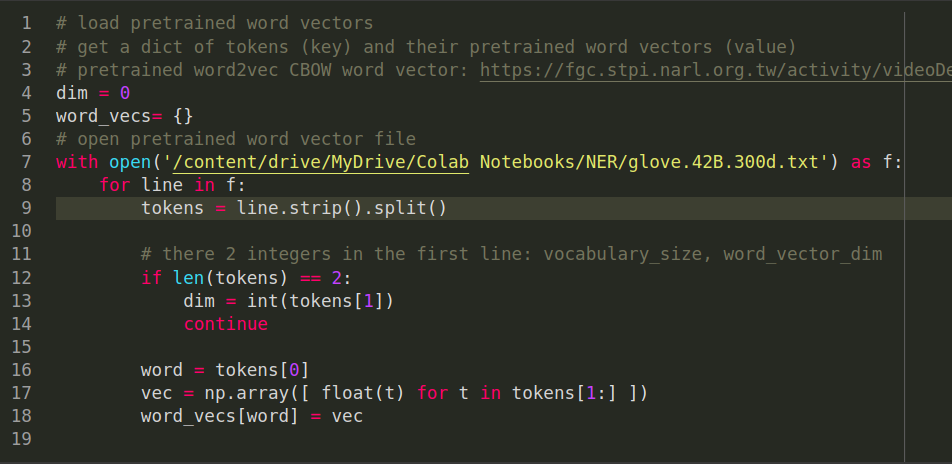
1. **Model\_Training**

Create a model to fit and predict, and don’t forget to flatten to get final f1score and print the prediction result

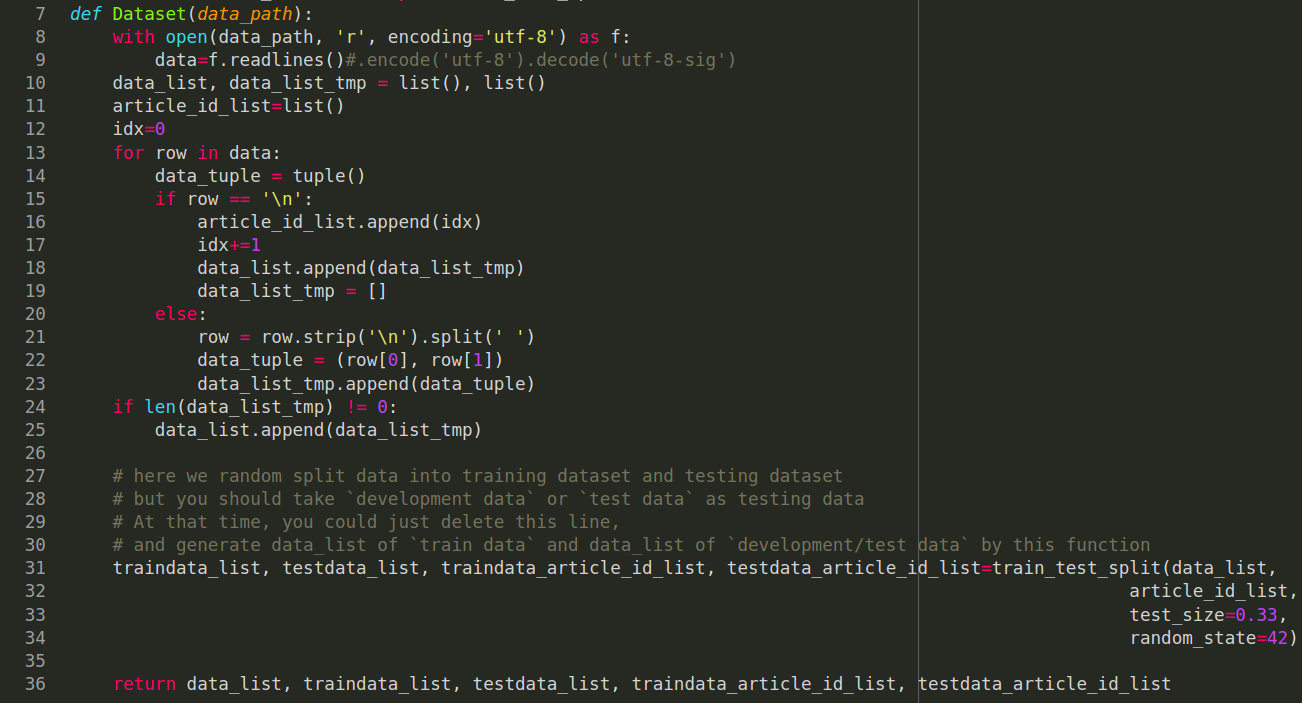


This part is very important. We need to prepare a pretrained word vector file to let it find corresponding key value of the token that get from input data.

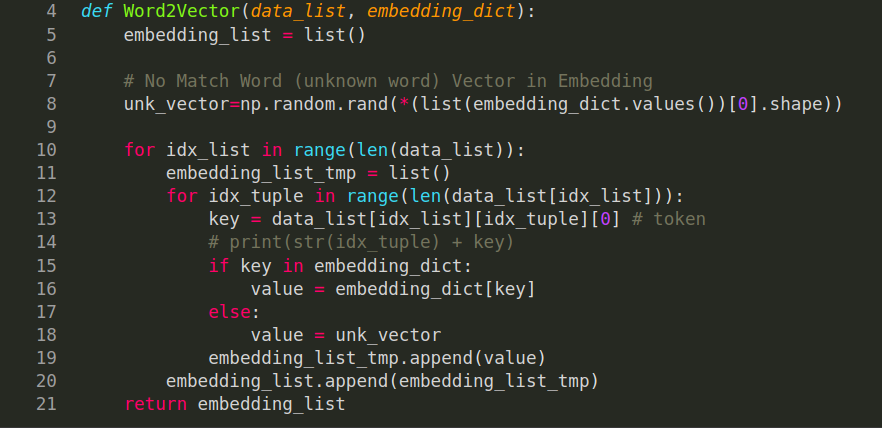
The first two tokens are the parameters which are vocabulary\_size and it’s dimension. And keep the rest tokens into **vec** array in float type, and return word\_vecs dictionary.



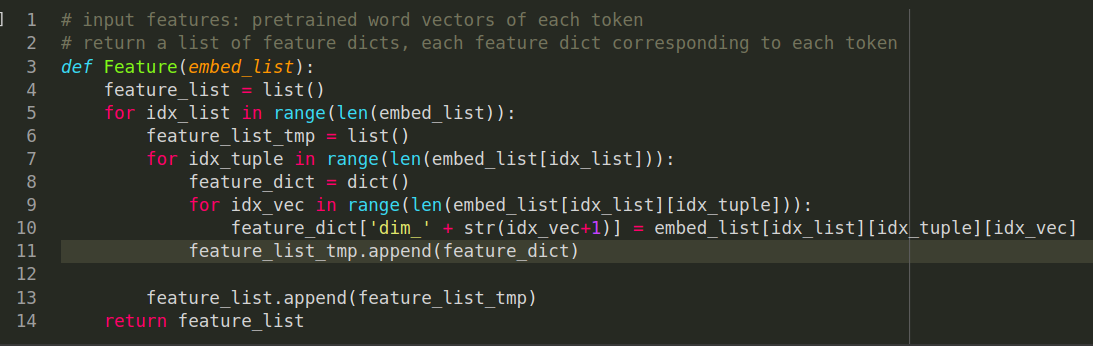
Prepare dataset and split them into training\_set and testing\_set with test\_size = 0.33



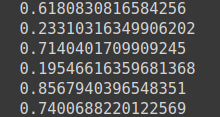
Use the pretrained word vector that we just imported as embedding\_dict to find the input data\_list’s corresponding value with three for loops and keep them as a list.



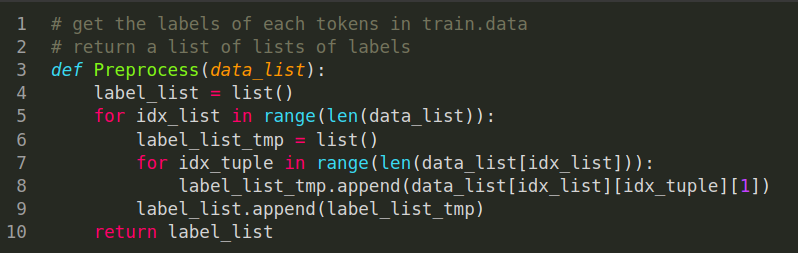
Extract all the features of embedding\_list and keep also(three loops go through 3 dimension of embed\_list)



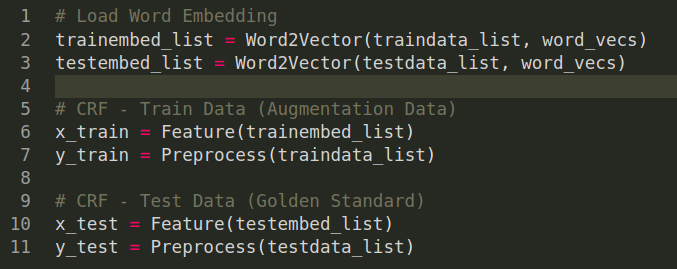
This is how embed\_list[idx\_list][idx\_tuple][idx\_vec] looks like:



Prepare a Label list by set the third dimension of data\_list to **1**



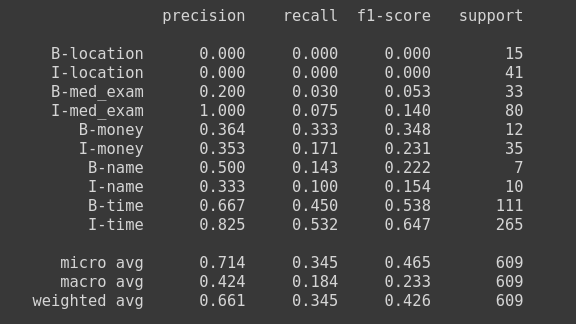
Call the functions



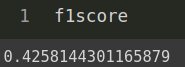
Fit and predict



Result

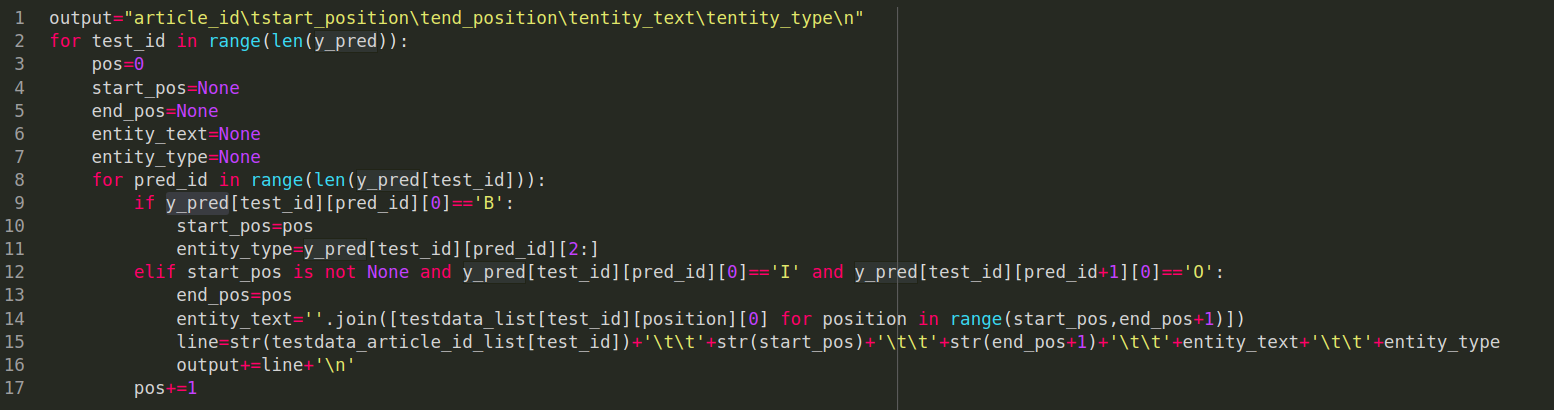


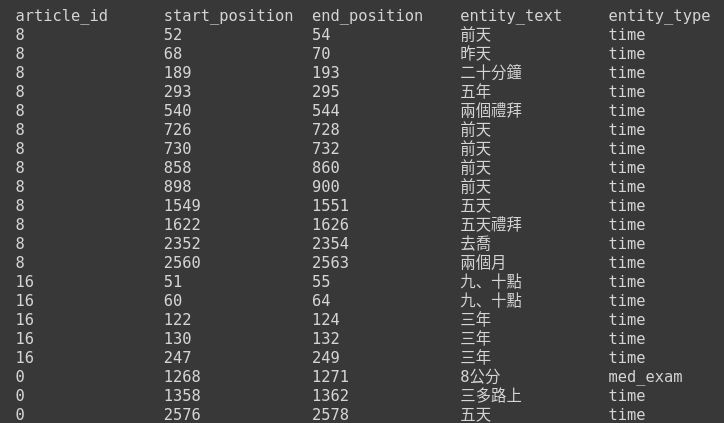
F1Score



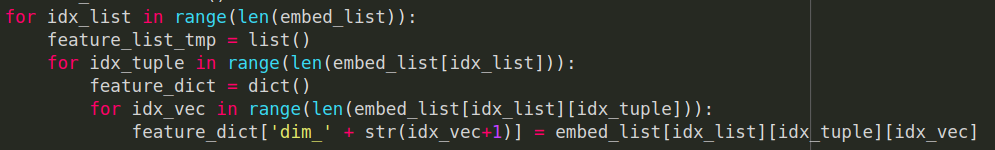
Change the output format and save into output.tsv file

Here we show the entity text’s id, start\_position, ending\_position, content and type.

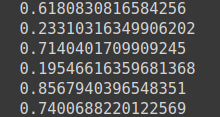




**3. Features**

****

This is how embed\_list[idx\_list][idx\_tuple][idx\_vec] looks like:



**4. Experience**

* From this homework, I had learned what are **NER** and **CRF model**,and how computer learn a high-level human language.
* As we know, **f1score** show our model performance, but the point is how to get higher score! This is definitely different to the typical ML model, because it is not only the input\_data but also how we processing the input\_data into tokens and label is decisively important !
* There is no so much parameters to let you adjust to get better model performance, so a great processing\_data and completely pretrained word vector file will help your model to become better!
* Interesting topic for us to explore how great ML is in this generation.
* Thanks for Prof and TA! :)