EL331 Authorship Analysis
Team E
s1290042 Yahagi Takuya
s1290062 Miyazaki Souta
s1290067 Urade Rikuto
s1290073 Kondo Tatsuya

Activity 11

~perfect aspect/tense vs. and non-perfect aspect/tense~

<Description about assigned authorship marker>

The marker that is assigned to us is "perfect tense vs. non-perfect tense". We thought if a writer is good at English grammar, he/she can correctly use perfect tense. But if a writer is not good at English grammar, for example a beginner, he/she may not make a sentence of perfect tense. So, we calculate the frequency of perfect tense and non-perfect tense, and using this calculated value, we classify the authors.

<Comparison of another related marker>

The big difference between our marker and another related marker is focus point. We focus the most on the word after "have/has/had". However, in another marker, maybe the verb itself, auxiliary verb, or something are focused on. We think this is a difference.

<Pseudocode>

```
Step1. Preparering data.
        ・Get the set of text files from "/Data/Textdata/".
        ・Tokenize each text file.
        ・Each token is stored in a word list, also each pos_tag of token is stored in a Pos list.
        (Achieving feature of adverbs)
        ・In advance, make lists of words about place and time, and store to place list and time list, respectively.
        ・By using place and time lists, in each sentence, classify the word into two types, adverbial of place and time.
        ・After that, store the number of adverbs of place and time in P, T list, respectively.
        (Achieving feature of verbs)
        ・Look whether it is "have", "has", or "had". If it is true, then look whether the next word or the next next word is past participle. If it is also true, store the word in the v_p list. If the last condition is false, then the first word ("have". "has", or "had") is stored in the v_np list.
        ・If the word is not "have", "has", and "had", then look whether it is verbs that do not include -ing format or to verb format. If it is a verb, then it is stored in the v_np list.
        ・Then, store the number of v_p, v_np list in PerfectV, NonPerfectV list, respectively.
```

(Making dataframe and csv)

・Read csv from “/Data/Authordata/author.csv”.

・Initialize dataframe by the number of place and time adverbs(P and T).

・Then, add the number of adverbs(Num_Place + Num_Time) to the dataframe.

・Then, add the number of perfect tense and non perfect tense(PerfectV and NonPerfect V) to the dataframe.

・Also, add the number of verbs(PerfectV + NonPerfectV) to the dataframe.

・Then, calculate two feature values and add.

1. featureV1 = (Num_Place - Num_Time) / (Num_ad+1) “+1” is for escape dividing by 0.
2. featureV2 = (PerfectV - NonPerfectV) / (Num_Verbs+1)

・Finally, add the author to the dataframe.

・Transform the data frame to a csv file.

Step2.Training and Verification.

・Read a csv file that is made in Step1.

・The data that is used for predicating is featureV1 and feature V2 (X).

・The data that is used for the target is the Author (Y).

・Split randomly data into two parts, train and test, by a ratio of 9:1.

・Use the Random Forest Classifier and fit X_train, and Y_train.

・By using the data from the test, predict Y.

・print the verification of the classifier.

Step3. Trying to predict by new data.

・New data is a sentence that is string type.

・Like Step1, tokenize, get features, and make a data frame.

・Predict and print the result.