NLP ASSIGNMENT-2 REPORT

Student id: 210151488

1. Observed 12 null values in all_train_data. Dropped the null values and proceed with splitting

Training data has been split into 90% train and 10% validation data using first 360 lines from the training split and first 40 lines from validation split.

Pre-processing used:

a. Lower casing all the words.
b. Using different pattern to split properly with the help of regular expression:
   - Didn’t use nltk default tokenization because it separate I’ll as I and ‘ll as two different words.
   - Splitting using patter helps tokenizing properly words like I’ll and “….you”. are handled properly.
c. Using stop words and punctuation to remove very common words and extra punctuations.
d. Converting numeric to word eg: 25 as twenty five.
e. Converting a word to its base form using lemmatization

Result: Achieved mean rank of 2.375 after using above pre-processing

2. Following table is the summary of different feature used:

Features Used	Feature format	Mean Rank
Count + ONE previous word+ POS tag	'Count_3', 'PRE_ashley', 'POS_NN',	2.0
Count + ONE previous word+ ONE next word+ POS tag	'Count_2', 'PRE_ ', 'POST_ask', 'POS_JJ',’PRE_hey’,’POST_you’, ’POS_JJ’	1.875
Count +ONE previous word + ONE next word	'Count_2', 'PRE_surprised', 'POST_	2.25
Count +Two previous word(Bi-grams)+ POS tag	'Count_14', "PRE1_how's", 'PRE2_ ', 'POS_VBG'	2.187
Count + ONE previous word+ ONE next word+ POS tag(only one)	'Count_2', 'PRE_ ', 'POST_ask', 'POS_JJ',’PRE_hey’, ’POST_you’	1.812

Observation: It is observed for a word almost have same POS tag each time it is repeated in a sentence. Therefore I tried using just one POS tag instead of multiple POS in “Count + ONE previous word+ ONE next word+ POS tag” feature mean rank reduced little from 1.875 to 1.812

Result: Achieved mean rank of 1.812

3. As per question I have made a column “pre_current_post” in data frame which includes scene info, previous line and post line for a current character line provided they are in same scene and are different character.

Some pre- processing such as stop word and punctuations removal have been done in “Line” column as we are focusing on the number of counts as feature in this question.

Reason to take number of counts:

a. Since this is drama one character calling other character name in the line is common. It becomes easier to know at least the current character won’t be repeating his/her name again and again in whole drama.
b. Easy to count scene_info and find a pattern which character has more chance of playing the scene

Format example:

a. DESERTED_CAR_PARK_EXT_NIGHT, NONE, Look, ya, mark, ya, And, think, 're, unlucky, man, POST_Shirl, POST_..., _EOL_ → When current line is first line in scene
b. R&R_INT_NIGHT,PRE_Okay,Are,alright,You,'ve,bit,since,got,POST_Are,POST_alright,_EOL_ → General format when it satisfies our condition

Result: Achieved mean rank of 1.5

4. TF-IDF were used in the following features:

Features Used	Feature format	Mean Rank (TF-IDF)	Mean Rank
Scene_info + previous line + current line + post line	R&R_INT_NIGHT,PRE_Okay,Are,alright, You,'ve,bit,since,got,POST_Are, POST_alright,_EOL_	1.062	--
Count + ONE previous word+ POS tag	'Count_3', 'PRE_ashley', 'POS_NN',	4.4375	2.0
Count + ONE previous word+ ONE next word+ POS tag	'Count_2', 'PRE_ ', 'POST_ask', 'POS_JJ',’PRE_hey’,’POST_you’, ’POS_JJ’	4.0	1.875
Count +ONE previous word + ONE next word	'Count_2', 'PRE_surprised', 'POST_	2.125	2.25
Count +Two previous word(Bi-grams)+POS tag	'Count_14', "PRE1_how's", 'PRE2_ ', 'POS_VBG'	4.625	2.187
Count + ONE previous word+ ONE next word+ POS tag(only one)	'Count_2', 'PRE_ ', 'POST_ask', 'POS_JJ',’PRE_hey’, ’POST_you’	1.625	1.812

Observation: Tried TF-IDF in all the combination of q2 and the mean rank increased for almost all feature except the one where we had received the best mean rank i.e “Count + ONE previous word+ ONE next word+ POS tag(only one)”

Result: Achieved mean rank of 1.062

5. As seen from above questions we got the best performance on “Scene_info + previous line + current line + post line (WITH TF-IDF)” as 1.062 therefore applying similar approach to our test data.

After using training_data (first 400 lines per character maximum) and final testing on the test file (using the first 40 lines per character maximum) following result was observed.

Feature Used	Mean Rank
Scene_info + previous line + current line + post line (WITH TF-IDF)	1.0
Scene_info + previous line + current line + post line (WITHOUT TF-IDF)	1.1875

Result: Achieved mean rank of 1 on test data.