Report

Question 1

Tokenization:
The input text is tokenized using the word_tokenize function of the Natural Language Toolkit (NLTK).

Normalization:
All tokens are converted to lowercase.

Stop-word removal:
Stop words, common words that often do not contribute much to the overall meaning, are removed from the list of tokens. The set of English stop words is sourced from NLTK.

Lemmatization:
The remaining tokens undergo lemmatization using the WordNetLemmatizer from NLTK.

Output:
The function returns a list of pre-processed and lemmatized tokens.

Question 2

The `extract_ngrams` is defined that converts unigram tokens to n-grams

In `to_feature_vector_dictionary`, bigrams are extracted from the 'character_doc' tokens passed to the function. A feature dictionary 'feature_dict' is initialized that updates the (key, value) pairs from bigram token frequency dictionary. The function returns this dictionary

TfidfTransformer object is initialized. In `create_document_matrix_from_corpus`, we fit the TfidfTransformer class to the train_feature_matrix that we initially created using DictVectorizer and use it to transform the train_feature_matrix and val_feature_matrix

Question 3

In `create_character_document_from_dataframe` we group the dataframe on the episode number, scene number and character name such that each row includes all the lines spoken by that character in that particular scene to incorporate the context of the line spoken by the character in terms of the lines spoken by other characters in the same scene

Question 4

A grid search is performed on all possible n-grams combinations i.e [(1,), (2,), (3,), (1, 2), (1, 3), (2, 3), (1, 2, 3)] where the number corresponds to the particular n-gram (unigram, bigram, trigram) and we create a feature dictionary for each combination. We then fit the training_feature_matrix for each combination and tranform the training and validation_feature_matrix and calculate the mean_rank for each combination and store it along with the combination. The combination with lowest mean_rank is considered the best i.e. bigrams only combination which gives a mean_rank of 1.1 and accuracy of 0.9

Question 5

From the similarity heat map, it is clear that 90% of the character vectors in the validation set i.e. 9/10 were matched accurately with character vectors of themselves in the training set. Character vectors of Chandler Bing are more similar to Joey Tribbiani and Ross Geller than the female characters Phoebe Buffay and Rachel Green. This is because the language use and speaking style of men are different from women resulting in more similar n-gram features between men characters. Phoebe Buffay has a much more unique style of speaking which is suggested by the huge difference in similarity scores of her testing vectors with other characters' training vectors. The lines spoken by all characters in union denoted by #ALL# has a similarity score very different from other character vectors as it confuses the algorithm of who actually spoke it.