**Program integrity tests**

This file consists of sample console output of the modules when the programs are run for testing purposes. It includes test prints, which are preceded by the words ‘TEST PRINT’, which do not appear in the code when it is normally run.

For ease of navigation, code which has been copied and pasted is in bold.

The machine learning tests are run on a dummy data set of 20 sentences (DummyDataAfter.csv, which is available in the same folder as this document), as that makes it easier to check if the results are in line with expectations.

The functionality which allows the user to specify the name of the dataset they wish to use is only tested once (in add-new-fields.py). This is because the functions used to achieve this are identical in all the modules to which this functionality has been added.

The testing includes entering invalid answers to multiple choice questions, to check that the programs works as they should.

Contents

[add-new-fields.py 1](#_Toc51675466)

[mcNemarTest.py 4](#_Toc51675467)

[ngram-only-tests.py 7](#_Toc51675468)

[non-ngram-only-tests.py 12](#_Toc51675469)

[ngram-and-non-ngram-tests-combined.py 23](#_Toc51675470)

[checkForDuplicateSentences.py 32](#_Toc51675471)

# add-new-fields.py

This module writes new data fields back to the original data file.

When a data file name other than the default data file name is selected, and the data file was present in the same folder as the program file, the file was written to and the expected console output produced:

**The default file name is original\_formality\_dataset.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name (including the file extension), then press enter Dummy\_Data\_Before.csv**

**Thank you. The file name has been changed to Dummy\_Data\_Before.csv**

**No of records uploaded: 20**

**Processing data. This can take a while. Please be patient.**

**[nltk\_data] Downloading package cmudict to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package cmudict is already up-to-date!**

**[nltk\_data] Downloading package punkt to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package punkt is already up-to-date!**

**[nltk\_data] Downloading package averaged\_perceptron\_tagger to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package averaged\_perceptron\_tagger is already up-to-**

**[nltk\_data] date!**

**[nltk\_data] Downloading package stopwords to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package stopwords is already up-to-date!**

**Number of records for new file 20**

**New data successfully written to file. Program complete.**

**Process finished with exit code 0**

This shows that, when a non-existent file name is entered instead, the user is given a message about it and the program ends:

**The default file name is original\_formality\_dataset.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name (including the file extension), then press enter thisFileDoesNotExist.csv**

**Thank you. The file name has been changed to thisFileDoesNotExist.csv**

**File not found: thisFileDoesNotExist.csv**

**Please ensure that the data file is in the same folder as the program file and try again.**

**Exiting program.**

**Process finished with exit code 0**

This shows that pressing return to indicate that the default file name is correct when the default data file is absent from the relevant folder causes the program to terminate cleanly and produces the correct output:

**The default file name is original\_formality\_dataset.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name (including the file extension), then press enter**

**Thank you. You have confirmed that the file name is correct: original\_formality\_dataset.csv**

**File not found: original\_formality\_dataset.csv**

**Please ensure that the data file is in the same folder as the program file and try again.**

**Exiting program.**

**Process finished with exit code 0**

The same process as above is now repeated, but with the default file name present in the folder, and the output was as expected:

**The default file name is original\_formality\_dataset.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name (including the file extension), then press enter**

**Thank you. You have confirmed that the file name is correct: original\_formality\_dataset.csv**

**No of records uploaded: 20**

**Processing data. This can take a while. Please be patient.**

**[nltk\_data] Downloading package cmudict to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package cmudict is already up-to-date!**

**[nltk\_data] Downloading package punkt to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package punkt is already up-to-date!**

**[nltk\_data] Downloading package averaged\_perceptron\_tagger to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package averaged\_perceptron\_tagger is already up-to-**

**[nltk\_data] date!**

**[nltk\_data] Downloading package stopwords to**

**[nltk\_data] C:\Users\water\AppData\Roaming\nltk\_data...**

**[nltk\_data] Package stopwords is already up-to-date!**

**Number of records for new file 20**

**New data successfully written to file. Program complete.**

**Process finished with exit code 0**

# mcNemarTest.py

This module is responsible for carrying out McNemar tests and printing the result to the console.

This is an example of a test where there is a statistically significant difference between the baseline values and the results obtained for a feature (the first set of data is random selections, and the second set is ‘informativeness’ as a feature, with Support Vector Machine as the classifier):

**For the first set of data, how many true positives are there? 322**

**For the first set of data, how many false positives are there? 346**

**For the first set of data, how many true negatives are there? 398**

**For the first set of data, how many false negatives are there? 340**

**For the second set of data, how many true positives are there? 486**

**For the second set of data, how many false positives are there? 176**

**For the second set of data, how many true negatives are there? 568**

**For the second set of data, how many false negatives are there? 176**

**SUMMARY OF DATA ENTERED**

**-------------------------**

**First data set - true positives:**

**First data set - false positives:**

**First data set - true negatives:**

**First data set - false negatives:**

**Second data set - true positives:**

**Second data set - false positives:**

**Second data set - true negatives:**

**Second data set - false negatives:**

**For the first set of data, there are 720 correct predictions and 686 incorrect predictions.**

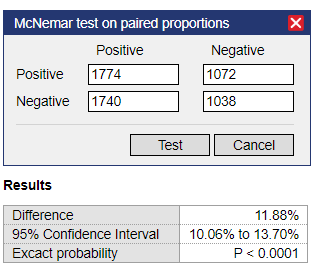
**For the second set of data, there are 1054 correct predictions and 352 incorrect predictions.**

**McNemar co-efficient value = 158.211, p-value = 0.000**

**The results ARE significantly different, based on a p value of 5 percent.**

**Process finished with exit code 0**

The online McNemar calculator at <https://www.scistat.com/statisticaltests/mcnemar.php> gives a p value for the test which corresponds with that given by the program:



Here is another example, which is for random selections as the first set of data and selections of ‘negative’ (i.e. informality) on all of the second set of data:

**For the first set of data, how many true positives are there? 322**

**For the first set of data, how many false positives are there? 346**

**For the first set of data, how many true negatives are there? 398**

**For the first set of data, how many false negatives are there? 340**

**For the second set of data, how many true positives are there? 0**

**For the second set of data, how many false positives are there? 0**

**For the second set of data, how many true negatives are there? 744**

**For the second set of data, how many false negatives are there? 662**

**SUMMARY OF DATA ENTERED**

**-------------------------**

**First data set - true positives:**

**First data set - false positives:**

**First data set - true negatives:**

**First data set - false negatives:**

**Second data set - true positives:**

**Second data set - false positives:**

**Second data set - true negatives:**

**Second data set - false negatives:**

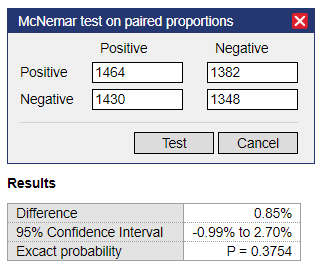
**For the first set of data, there are 720 correct predictions and 686 incorrect predictions.**

**For the second set of data, there are 744 correct predictions and 662 incorrect predictions.**

**McNemar co-efficient value = 0.786, p-value = 0.375**

**The results ARE NOT significantly different, based on a p value of 5 percent.**

The same McNemar calculator has used previously gives the same p value as was produced by the program:



# ngram-only-tests.py

This program runs machine learning tests purely involving n-grams, i.e. no other features included.

The output of the program running from start to finish will now be presented, which includes invalid choices for the multiple-choice questions (and does not include stop words):

**The default data file name is new\_formality\_data.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name, then press enter: Dummy\_Data\_After.csv**

**Thank you. The file name has been changed to Dummy\_Data\_After.csv**

**No of records uploaded: 20**

**The n-gram types are:**

**1 - Unigram**

**2 - Bigram**

**3 - Trigram**

**4 - Unigram and bigram combined**

**5 - Unigram, bigram and trigram combined**

**Choose an option by typing a number between 1 and 5 and then press 'enter': fgasfasdfasdf**

**Invalid selection. Please try again**

**The n-gram types are:**

**1 - Unigram**

**2 - Bigram**

**3 - Trigram**

**4 - Unigram and bigram combined**

**5 - Unigram, bigram and trigram combined**

**Choose an option by typing a number between 1 and 5 and then press 'enter': 1324123412342**

**Invalid selection. Please try again**

**The n-gram types are:**

**1 - Unigram**

**2 - Bigram**

**3 - Trigram**

**4 - Unigram and bigram combined**

**5 - Unigram, bigram and trigram combined**

**Choose an option by typing a number between 1 and 5 and then press 'enter': 1**

**The representation options are:**

**1 - Binary**

**2 - Non-Binary**

**3 - TF-IDF**

**Choose an option by typing a number between 1 and 3 and then press 'enter': fasdfasdfasdfasdf**

**Invalid selection. Please try again**

**The representation options are:**

**1 - Binary**

**2 - Non-Binary**

**3 - TF-IDF**

**Choose an option by typing a number between 1 and 3 and then press 'enter': 34123421323**

**Invalid selection. Please try again**

**The representation options are:**

**1 - Binary**

**2 - Non-Binary**

**3 - TF-IDF**

**Choose an option by typing a number between 1 and 3 and then press 'enter': 1**

**The stop word options are:**

**1 - Include stop words**

**2 - No, do not include stop words**

**Choose an option by typing either 1 or 2 and then press 'enter': 354123412341234**

**Invalid selection. Please try again**

**The stop word options are:**

**1 - Include stop words**

**2 - No, do not include stop words**

**Choose an option by typing either 1 or 2 and then press 'enter': fasdfasdfs**

**Invalid selection. Please try again**

**The stop word options are:**

**1 - Include stop words**

**2 - No, do not include stop words**

**Choose an option by typing either 1 or 2 and then press 'enter': 2**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': sdafasdfasd**

**That was not a valid selection. Please try again.**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': 3413242**

**That was not a valid selection. Please try again.**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': 4**

**You have selected Random Forest**

**TEST PRINT. Vocabulary list (NB: stop words are included if the user selected to include them in the test): ['aboard', 'abstain', 'annoyed', 'ask', 'away', 'book', 'class', 'clear', 'come', 'didn', 'die', 'disappeared', 'does', 'don', 'drinking', 'eat', 'faster', 'father', 'getting', 'guess', 'halt', 'helen', 'japan', 'let', 'll', 'maybe', 'minute', 'near', 'notebook', 'nut', 'pushes', 'quickly', 'religious', 'right', 'room', 'selfishness', 'shoot', 'sick', 'stay', 'step', 'stolen', 'study', 'time', 'tom', 'train', 'tries', 'try', 'view', 'want', 'watch', 'yesterday']**

**TEST PRINT. Vector shape (The first figure is number of sentences, the second figure is number of words in the entire corpus): (20, 51)**

**TEST PRINT. The internal representation of the n-gram specifications chosen. If 'stop\_words='english' is present, then stop words have been filtered out of the test and stop\_words=None means stop words are included:  
  
CountVectorizer(analyzer='word', binary=True, decode\_error='strict',**

**dtype=<class 'numpy.int64'>, encoding='utf-8', input='content',**

**lowercase=True, max\_df=1.0, max\_features=None, min\_df=1,**

**ngram\_range=(1, 1), preprocessor=None, stop\_words='english',**

**strip\_accents=None, token\_pattern='(?u)\\b\\w\\w+\\b',**

**tokenizer=None, vocabulary=None)**

**TEST SUMMARY**

**------------**

**unigram with binary representation and without stop words**

**You chose the following classifier: Random Forest**

**Please be patient - the program may take a while to run.**

**RESULTS SUMMARY**

**---------------**

**Feature tested: unigram with binary representation and without stop words**

**Classifier: Random Forest**

**Total predictions: 4**

**TRUE POSITIVES: 0**

**FALSE POSITIVES: 0**

**TRUE NEGATIVES: 2**

**FALSE NEGATIVES: 2**

**Accuracy: 0.50**

**Precision: N/A**

**Recall: N/A**

**False positive rate: N/A**

**AUC: 0.75**

**Balanced accuracy: 0.50**

**TEST PRINT. List of predictions in order: [False False False False]**

**TEST PRINT. List of actual whether sentence is formal in order: [False False True True]**

**Process finished with exit code 0**

# non-ngram-only-tests.py

This program purely runs machine learning programs not including n-grams. This is some sample output:

**The default file name is new\_formality\_data.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name, then press enter: Dummy\_Data\_After.csv**

**Thank you. The file name has been changed to Dummy\_Data\_After.csv**

**No of records uploaded: 20**

**You can add the following features:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average number of syllables per word**

**29 - Average word length**

**30 - Number of stop words**

**31 - Number of words with > seven characters**

**32 - Number of words with < 5 characters**

**33 - Average word frequency**

**34 - Number of words in 35 most common words in corpus**

**35 - VADER sentiment score**

**No features have been selected yet**

**Please choose the number of a feature you wish to add and then press 'enter': 35**

**You have just selected: VADER sentiment score**

**You can add the following features:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average number of syllables per word**

**29 - Average word length**

**30 - Number of stop words**

**31 - Number of words with > seven characters**

**32 - Number of words with < 5 characters**

**33 - Average word frequency**

**34 - Number of words in 35 most common words in corpus**

**You have previously added the following features:**

**VADER sentiment score**

**Please choose an additional feature and press 'enter'**

**or press C then 'enter' to select your classifier: 28**

**You have just selected: Average number of syllables per word**

**You can add the following features:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average word length**

**29 - Number of stop words**

**30 - Number of words with > seven characters**

**31 - Number of words with < 5 characters**

**32 - Average word frequency**

**33 - Number of words in 35 most common words in corpus**

**You have previously added the following features:**

**VADER sentiment score**

**Average number of syllables per word**

**Please choose an additional feature and press 'enter'**

**or press C then 'enter' to select your classifier: 32**

**You have just selected: Average word frequency**

**You can add the following features:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average word length**

**29 - Number of stop words**

**30 - Number of words with > seven characters**

**31 - Number of words with < 5 characters**

**32 - Number of words in 35 most common words in corpus**

**You have previously added the following features:**

**VADER sentiment score**

**Average number of syllables per word**

**Average word frequency**

**Please choose an additional feature and press 'enter'**

**or press C then 'enter' to select your classifier: asdfadsfsadfsa**

**You did not enter a number. Please try again.**

**You can add the following features:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average word length**

**29 - Number of stop words**

**30 - Number of words with > seven characters**

**31 - Number of words with < 5 characters**

**32 - Number of words in 35 most common words in corpus**

**You have previously added the following features:**

**VADER sentiment score**

**Average number of syllables per word**

**Average word frequency**

**Please choose an additional feature and press 'enter'**

**or press C then 'enter' to select your classifier: 1234123412341234123**

**You did not enter a valid number. Please try again.**

**You can add the following features:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average word length**

**29 - Number of stop words**

**30 - Number of words with > seven characters**

**31 - Number of words with < 5 characters**

**32 - Number of words in 35 most common words in corpus**

**You have previously added the following features:**

**VADER sentiment score**

**Average number of syllables per word**

**Average word frequency**

**Please choose an additional feature and press 'enter'**

**or press C then 'enter' to select your classifier: C**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': 213234123412341234**

**That was not a valid selection. Please try again.**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': aqsdfasdfadsfasdf**

**That was not a valid selection. Please try again.**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': 4**

**You have selected Random Forest**

**TEST PRINT. List of all formality classifications in dataset, in sentence order: [False, True, False, False, True, False, False, True, False, True, True, True, False, False, False, True, True, True, False, False]**

**TEST PRINT. Feature data, in sentence order: [[0.0, 1.0, 1.0], [-0.4939, 1.428571429, 1.166666667], [-0.5994, 1.454545455, 1.1], [-0.5457, 1.142857143, 1.0], [-0.296, 1.2, 1.111111111], [0.0, 1.363636364, 1.0], [0.0258, 1.5, 1.0], [0.0, 1.0, 1.142857143], [0.0, 1.3, 1.0], [0.0, 1.5, 1.0], [-0.2263, 1.6, 1.0], [0.0, 1.6, 1.0], [0.3818, 1.272727273, 1.1], [-0.296, 1.142857143, 1.0], [0.0, 1.75, 1.0], [0.0, 1.428571429, 1.0], [-0.4559, 1.333333333, 1.0], [0.0, 1.285714286, 1.0], [-0.6486, 2.0, 1.0], [0.0, 1.5, 1.0]]**

**Feature(s) tested: 'VADER sentiment score', 'Average number of syllables per word' and 'Average word frequency'**

**Classifier: Random Forest**

**Total predictions: 4**

**TRUE POSITIVES: 2**

**FALSE POSITIVES: 2**

**TRUE NEGATIVES: 0**

**FALSE NEGATIVES: 0**

**Accuracy: 0.50**

**Precision: 0.50**

**Recall: 1.00**

**False positive rate: 1.00**

**AUC: 0.00**

**Balanced accuracy: 0.50**

**TEST PRINT. List of predictions in order: [ True True True True]**

**TEST PRINT. List of actual whether sentence is formal in order: [False False True True]**

**Process finished with exit code 0**

# ngram-and-non-ngram-tests-combined.py

This program is for machine learning tests that combine n-grams and other features. Here is some output ( with stop words included – there will be a separate test for stop words excluded, to prove that the stop word functionality works):

**The default data file name is new\_formality\_data.csv**

**If this is the name of the data file you are using, simply press enter**

**Otherwise, please provide the correct name, and then press enter: Dummy\_Data\_After.csv**

**Thank you. The file name has been changed to: Dummy\_Data\_After.csv**

**No of records uploaded: 20**

**The n-gram types are:**

**1 - Unigram**

**2 - Bigram**

**3 - Trigram**

**4 - Unigram and bigram combined**

**5 - Unigram, bigram and trigram combined**

**Choose an option by typing a number between 1 and 5 and then pressing 'enter': sdafasdf**

**Invalid selection. Please try again**

**The n-gram types are:**

**1 - Unigram**

**2 - Bigram**

**3 - Trigram**

**4 - Unigram and bigram combined**

**5 - Unigram, bigram and trigram combined**

**Choose an option by typing a number between 1 and 5 and then pressing 'enter': 1**

**The representation options are:**

**1 - Binary**

**2 - Non-Binary**

**3 - TF-IDF**

**Choose an option by typing a number between 1 and 3 and then pressing 'enter': sdafasdf**

**Invalid selection. Please try again**

**The representation options are:**

**1 - Binary**

**2 - Non-Binary**

**3 - TF-IDF**

**Choose an option by typing a number between 1 and 3 and then pressing 'enter': 1**

**The stop word options are:**

**1 - Include stop words**

**2 - No, do not include stop words**

**Choose an option by typing 1 or 2 and then pressing 'enter': sdafasdfsa**

**Invalid selection. Please try again**

**The stop word options are:**

**1 - Include stop words**

**2 - No, do not include stop words**

**Choose an option by typing 1 or 2 and then pressing 'enter': 1**

**You can add the following features to the test:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average number of syllables per word**

**29 - Average word length**

**30 - Number of stop words**

**31 - Number of words with > seven characters**

**32 - Number of words with < 5 characters**

**33 - Average word frequency**

**34 - Number of words in 35 most common words in corpus**

**35 - VADER sentiment score**

**No features have been selected yet**

**Please choose the number of a feature to add: dsasdfsdf**

**You did not enter a number. Please try again.**

**You can add the following features to the test:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average number of syllables per word**

**29 - Average word length**

**30 - Number of stop words**

**31 - Number of words with > seven characters**

**32 - Number of words with < 5 characters**

**33 - Average word frequency**

**34 - Number of words in 35 most common words in corpus**

**35 - VADER sentiment score**

**No features have been selected yet**

**Please choose the number of a feature to add: 33**

**You have just selected: Average word frequency**

**You can add the following features to the test:**

**1 - ﻿Sentence ID**

**2 - HIT ID**

**3 - Formality**

**4 - Informativeness**

**5 - Implicature**

**6 - Length in Words**

**7 - Length in Characters**

**8 - F-score**

**9 - I-score**

**10 - Lexical Density**

**11 - Number of adjectives**

**12 - Number of verbs**

**13 - Number of adverbs**

**14 - Number of conjuctions**

**15 - Number of nouns**

**16 - Number of pronouns**

**17 - Number of modal verbs**

**18 - Number of prepositions**

**19 - Number of determiners**

**20 - Number of commas**

**21 - Number of exclamation marks**

**22 - Number of full stops**

**23 - Number of question marks**

**24 - Number of existential theres**

**25 - Number of proper nouns**

**26 - Number of capitalised words**

**27 - Number of interjections**

**28 - Average number of syllables per word**

**29 - Average word length**

**30 - Number of stop words**

**31 - Number of words with > seven characters**

**32 - Number of words with < 5 characters**

**33 - Number of words in 35 most common words in corpus**

**34 - VADER sentiment score**

**You have previously selected the following features:**

**Average word frequency**

**Please choose an additional feature and press 'enter'**

**or press C then 'enter' to select your classifier: C**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': sadfasdf**

**That was not a valid selection. Please try again.**

**The classifiers are:**

**1 - Support Vector Machine**

**2 - Logistic Regression**

**3 - Multinomial Bayes**

**4 - Random Forest**

**Please choose a classifier by typing a number between 1 and 4 and then press 'enter': 4**

**You have selected Random Forest**

**TEST SUMMARY**

**------------**

**unigram with binary representation and with stop words with the following non n-gram features:**

**'Average word frequency'**

**Your classifier is: Random Forest**

**The test may take a while. Please be patient.**

**TEST PRINT. Vocabulary list (whether stop words are included depends on what the user chose): ['aboard', 'abstain', 'all', 'and', 'annoyed', 'are', 'as', 'ask', 'at', 'away', 'book', 'class', 'clear', 'come', 'didn', 'die', 'disappeared', 'do', 'does', 'don', 'drinking', 'eat', 'enough', 'faster', 'father', 'first', 'from', 'get', 'getting', 'guess', 'halt', 'has', 'he', 'helen', 'her', 'here', 'his', 'if', 'in', 'is', 'it', 'japan', 'let', 'll', 'made', 'may', 'maybe', 'me', 'minute', 'my', 'near', 'no', 'not', 'notebook', 'nut', 'of', 'one', 'only', 'or', 'others', 'over', 'pushes', 'quickly', 'religious', 'right', 'room', 'selfishness', 'she', 'shoot', 'sick', 'some', 'stay', 'step', 'stolen', 'study', 'take', 'than', 'that', 'the', 'their', 'thin', 'this', 'time', 'to', 'tom', 'train', 'tries', 'try', 'us', 'view', 'want', 'was', 'watch', 'we', 'were', 'whenever', 'where', 'why', 'will', 'yesterday', 'you', 'your']**

**TEST PRINT. Vector shape (first figure is number of sentences, second is number of n-grams in corpus): (20, 102)**

**TEST PRINT - The internal representation of the n-gram specifications chosen. In the output below, stop\_words = 'stop\_words='english' means stop words are excluded and stop\_words=None means they are included:**

**CountVectorizer(analyzer='word', binary=True, decode\_error='strict',**

**dtype=<class 'numpy.int64'>, encoding='utf-8', input='content',**

**lowercase=True, max\_df=1.0, max\_features=None, min\_df=1,**

**ngram\_range=(1, 1), preprocessor=None, stop\_words=None,**

**strip\_accents=None, token\_pattern='(?u)\\b\\w\\w+\\b',**

**tokenizer=None, vocabulary=None)**

**TEST PRINT. List of all formality classifications in dataset, in order: [False, True, False, False, True, False, False, True, False, True, True, True, False, False, False, True, True, True, False, False]**

**TEST PRINT. List of feature data for the first sentence. Last three entries relate to the non n-gram features (as three of them were selected for the test):**

**[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.**

**0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0.**

**0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.**

**1. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0.**

**0. 0. 0. 0. 0. 0. 1.]**

**Feature tested:**

**unigram with binary representation and with stop words with the following non n-gram features:**

**'Average word frequency'**

**Classifier: Random Forest**

**Total predictions: 4**

**TRUE POSITIVES: 0**

**FALSE POSITIVES: 0**

**TRUE NEGATIVES: 2**

**FALSE NEGATIVES: 2**

**Accuracy: 0.50**

**Precision: N/A**

**Recall: N/A**

**False positive rate: N/A**

**AUC: 0.62**

**Balanced accuracy: 0.50**

**TEST PRINT. List of predictions in order: [False False False False]**

**TEST PRINT. List of actual whether sentence is formal in order: [False False True True]**

**Process finished with exit code 0**

# checkForDuplicateSentences.py

This program is used to check for duplicate sentences. The output was as follows:

**The default file name is new\_formality\_data.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name, then press enter: Dummy\_Data\_After.csv**

**Thank you. The file name has been changed to Dummy\_Data\_After.csv**

**No of records uploaded: 20**

**Good news! No duplicates were found.**

**Process finished with exit code 0**

Two duplicate entries were added to the data file. The output was as expected:

**The default file name is new\_formality\_data.csv**

**If this is the name of the data file, press enter**

**Otherwise, please provide the correct name, then press enter: Dummy\_Data\_After.csv**

**Thank you. The file name has been changed to Dummy\_Data\_After.csv**

**No of records uploaded: 20**

**Duplicate sentence: Let us do the first step.**

**Duplicate sentence: That book's over here.**

**Process finished with exit code 0**