# Course Code: AIML427

# Course Title: Big Data

# Assignment 3 – Individual Part 2

George Wiles

Student ID: 300586465

Due Date: 8 June 2021, 17:00 pm

Date of Submission: 8 June 2021

# Individual Machine Learning Task – AG News Classification Dataset

1. Task Description and Overview  
   According to Anand [1], AG is a collection of more than 1 million new articles gathered by the academic news search engine ComeToMyHead. The dataset contains 3 columns; the class index or response variable, and 2 text columns representing the article title and the article description.

This is an example of a multiclass classification problem, where the response variable (‘Class Index) is a value that represents 4 new classifiers (1-World, 2-Sports, 3-Business, 4-Science and Technology), that represents the title and sentence description of the each news article instance.

Original Dataset Size  
The AG News training set contains 120,00 training dataset instances (29 MB), and the test set contains 7600 testing samples (1.8 MB). It is assumed a typical text article can contain 500-800 words, which when applied to a TF-IDF (term frequency-inverse document frequency) numerical statistic a vectorized feature model containing both title and description fields could contain up to 1500 unique words or features. The Kaggle dataset [1] retrieved at <https://www.kaggle.com/amananandrai/ag-news-classification-dataset> has been pre-processed to ensure that there are no missing columns, the text and description columns contain quotes, commas and new line characters that have been escaped.

Expected Output  
This assignment uses two algorithms, logical regression and decision tree, executed on a spark Hadoop cluster to predict the multiclass response variable of the test set against the trained model using the training data set.

1. Pre-Processing Step

As noted in section (a), the AG News dataset selected [1] has been pre-processed in terms of ensuring there are no missing data instances in the column data and the textual data for Title and Description has been largely escaped so it can be successfully read as a CSV file without any pre-processing of the initial dataset. In terms of testing it locally a small random split of the data was made to test the program before running on the cluster. These files were added the Hadoop input dfs drive under wilesgeor/resources.

While the original dataset contains 2 features being the Title and Description text fields, the decision tree and linear regression programs take a feature count as a parameter, this was used to limit the word vectorization of Title and Description respectively represent these text fields as 500-800 features each respectively. The respective word features are cleaned using regex, stop words transformers to reduce the non-word/irrelevant features dimensionality and combining the word features using a Spark vector assembler.

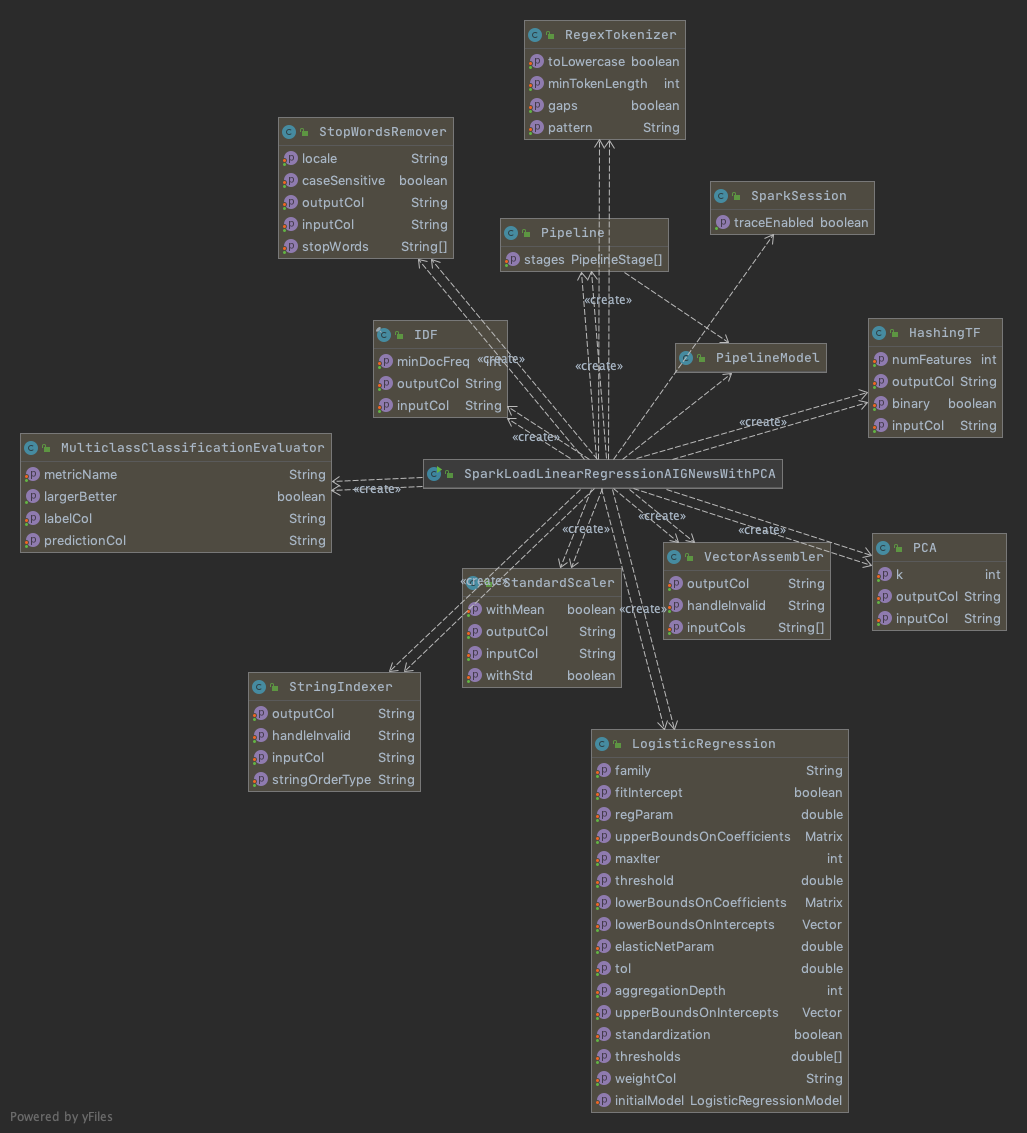
* 1. 1. Regex Tokenizer to
     1. Remove white space characters
     2. Set all word/features as lower case to remove duplication
     3. Set a minimum length token of 1 character to reduce punctuation.  
        *|Reuters - Stocks ...| - to - |[reuters, stocks,...|*
  2. Stop words remover, by default the Spark ML StopWordsRemover contains a list of common stop words (and, of, to et al).

*|stocks, assets, of| - to -.|[stocks, assets, nati...|*

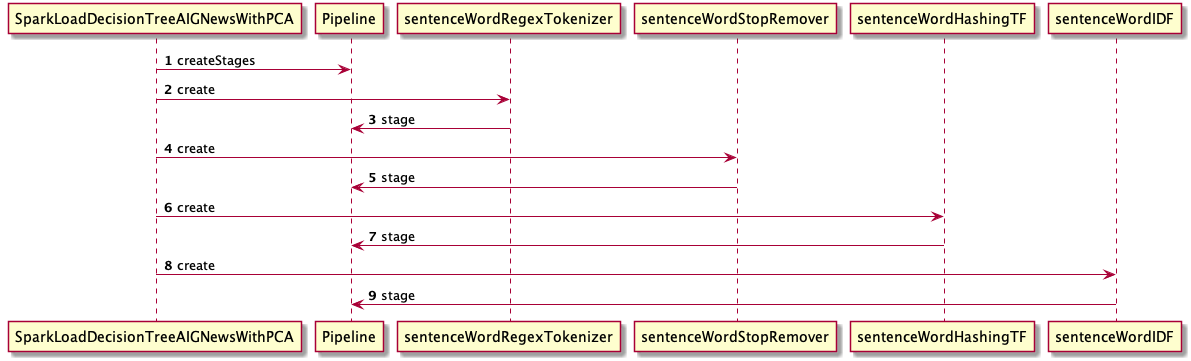
* 1. Hashing term frequency (HashingTF vs CountVectorizer, both were investigated with as a pre-fitting processing data transform step.

1. Program Description for both Linear Regression and Decision Tree Programs

The Linear Regression classification program for the AIG dataset can be seen in *Diagram 1, Linear Regression AIG classification program.* The main program executed on the spark clusters is called *SparkLoadDecisionTreeAIGNewsWithPCA,* This class contains a number of pre-fitting data processing functions that are executed by the pipeline in a deterministic order and accepts a number of configuration parameters used to tune the model.

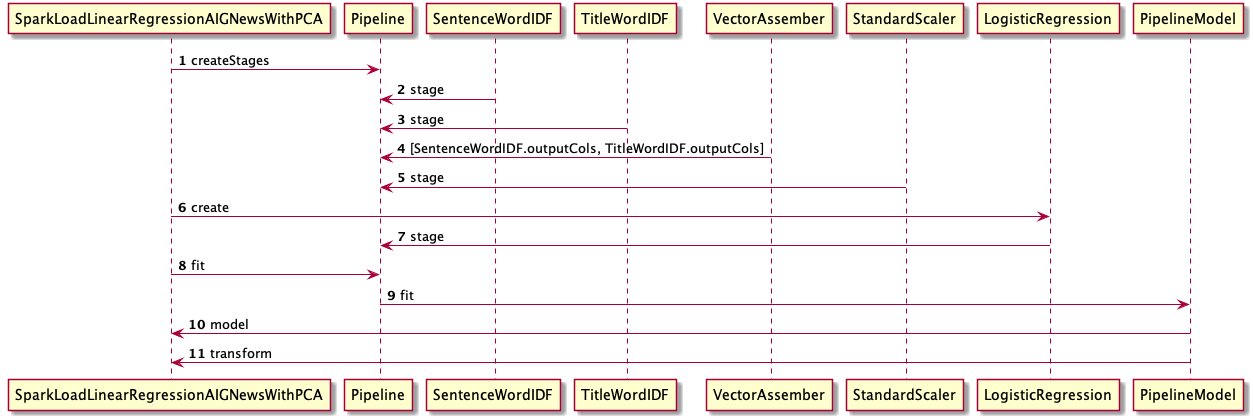


*Diagram 1 Linear Regression AIG classification program.*

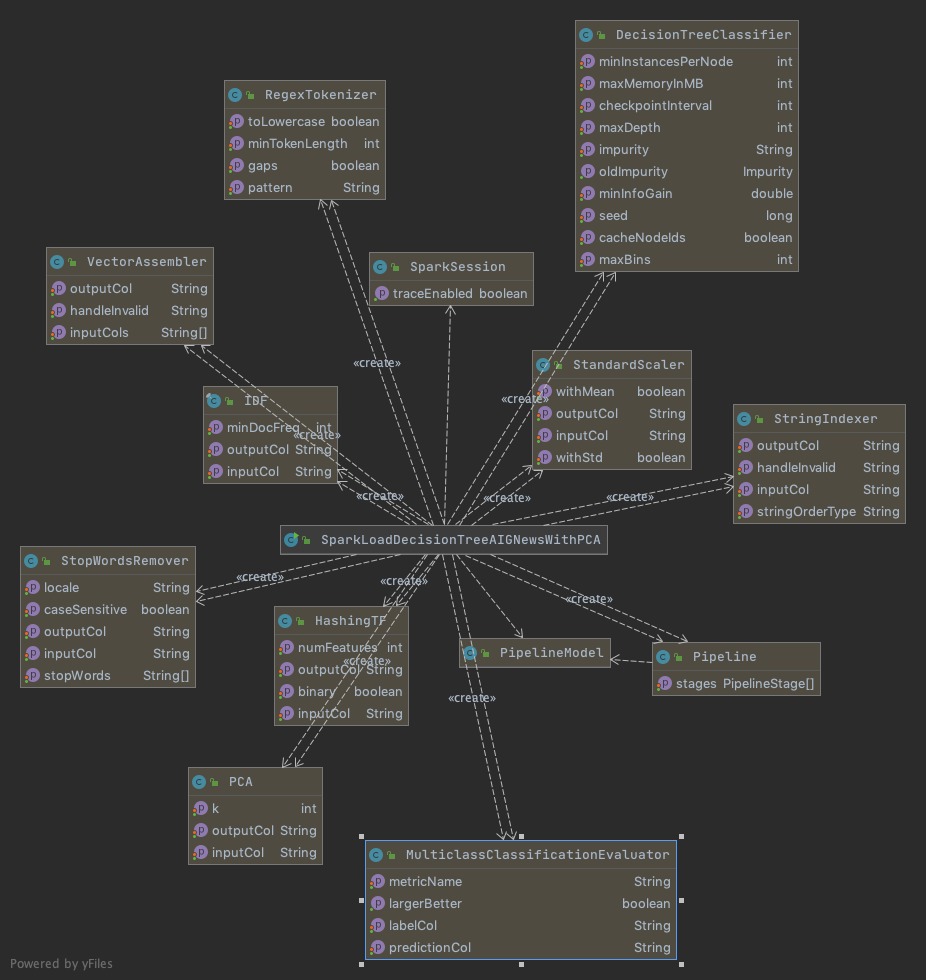
Both Full Text input features Title and Sentence follow the same Pipeline pattern, in that the following Spark ML transformers are used in the word pre-fitting pipeline data feature processing pipeline steps (RegexTokenizer, WordStopRemover, HashingTF and IDF). After which both Titles and Sentences/Descriptions are combined using a VectorAssembler Spark ML pipeline Step.  
  


Once the Title and Description/Sentence word features are staged, the following sequence diagram illustrates the pipeline stages that orchestrate the creation of the LogisticRegression models which are then able to fit the training set creating a trained model from which training and test set accuracies can be evaluated against the generated model. Both the DecisionTree program and LogisticRegression programs follow this pattern.

PCA is used in the pipeline as a command line argument that will apply dimensionality reduction on the these large dimensioned word vector feature sets.



The following diagram illustrates the class relationships for the DecisionTree program. The Decision Tree program follows the same pattern of pre-processing the features utilizing the Spark ML pipeline discussed for the LogisticRegression program above.



1. Installing and Running the Programs
2. Compare and Discuss Results
3. Compare and Discuss Results with PCA

# Appendix

[1] Aman Anand: AG’s News Topic Classification Set. Retrieved 7 June 2021: https://www.kaggle.com/amananandrai/ag-news-classification-dataset