Documents Classification Report

1. Introduction
   1. This task includes classifying documents for their content.
   2. The Classification problem was whether the documents discussing company purchasing or not.
   3. The input data includes CSV containing:
      1. URL for the document
      2. Description
      3. Title
      4. Content
      5. Classification
2. Approach
   1. This task is formally a binary classification problem.
   2. Several classification models were tested:
      1. Decision tree
      2. Random forest
      3. KNN
   3. Every one of these models was tested against each of the fields separately with small changes in model properties for optimization:
      1. Title
      2. Description
      3. Content
3. Data Insights
   1. Contains 3 languages:
      1. English
      2. Italian
      3. Spanish
   2. The data is imbalanced (~2500 T / ~17,500 F)
      1. 
4. Preprocessing
   1. Includes Data cleaning:
      1. Lower casing
      2. Stripping multiple white spacing
      3. Removing stop words:
         1. English
         2. Italian
         3. Spanish
      4. Stripping alphanumeric characters
      5. Stripping numeric characters
      6. Stripping Punctuation
   2. Creating indices dictionary using all of the documents.
   3. Preparing indices matrix from all of the documents
5. Model training
   1. There are several approaches for handling data imbalance, I will try several methods for solving this issue:
      1. Using different algorithms
      2. Resampling the data-set
         1. Removing negative classifications
         2. Duplicating Positive classifications
         3. Both of the above
   2. I have created the indices matrix using doc2idx what made it optional for me using the vector for model training.
   3. Simple decision tree classifier with original data set:
      1. Attached below the results for decision tree classifier for the fields => title, description, content:
      2. 
   4. Random forest classifier with original data set:
      1. Attached below the results for random forest classifier for the fields => title, description, content (300 sub trees):
      2. 
   5. KNN Classifier:
      1. First was finding best k for the model, attached k measurements for original data set for fields => title, description, content:
      2. 
      3. Attached below the results for KNN classifier using best k for the original dataset and fields => contents, titles, description
      4. 