ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΤΕΧΝΙΚΕΣ ΕΞΟΡΥΞΗΣ ΔΕΔΟΜΕΝΩΝ
2016 - 2017

ΕΡΓΑΣΙΑ 1

Γιαννούδης Αστέριος - 1115201200025
Κολυβάς Κωνσταντίνος – 1115201200066

Table of Contents:

Introduction:

1. Γνωστά λάθη υλοποίησης:

   i. Δεν μπορέσαμε να βρούμε πώς θα υλοποιούταν το ROC curve του αλγορίθμου K-NN λόγω της μη ικανότητας εύρεσης της πιθανότητας κατηγορίας για το κάθε document.

   ii. Τα αποτελέσματα στο αρχείο output/EvaluationMetric_10fold.csv δεν είναι στην μορφή που ζητείται από την εκφώνηση. Δηλαδή, στην πρώτη στήλη βρίσκεται το όνομα του αλγορίθμου, και στις υπόλοιπες οι μέσοι όροι των μετρικών συναρτήσεων.
   Για τυχόν διευκόλυνση, παραδίδεται επίσης το αρχείο output/EvaluationMetric_10fold_proper.csv όπου έχουν γραφεί τα αποτελέσματα με τον τρόπο που ζητείται, **χειρόγραφα**, δηλαδή δεν παράγονται αυτόματα από κάποια εκτέλεση του προγράμματος.

1. Classification:

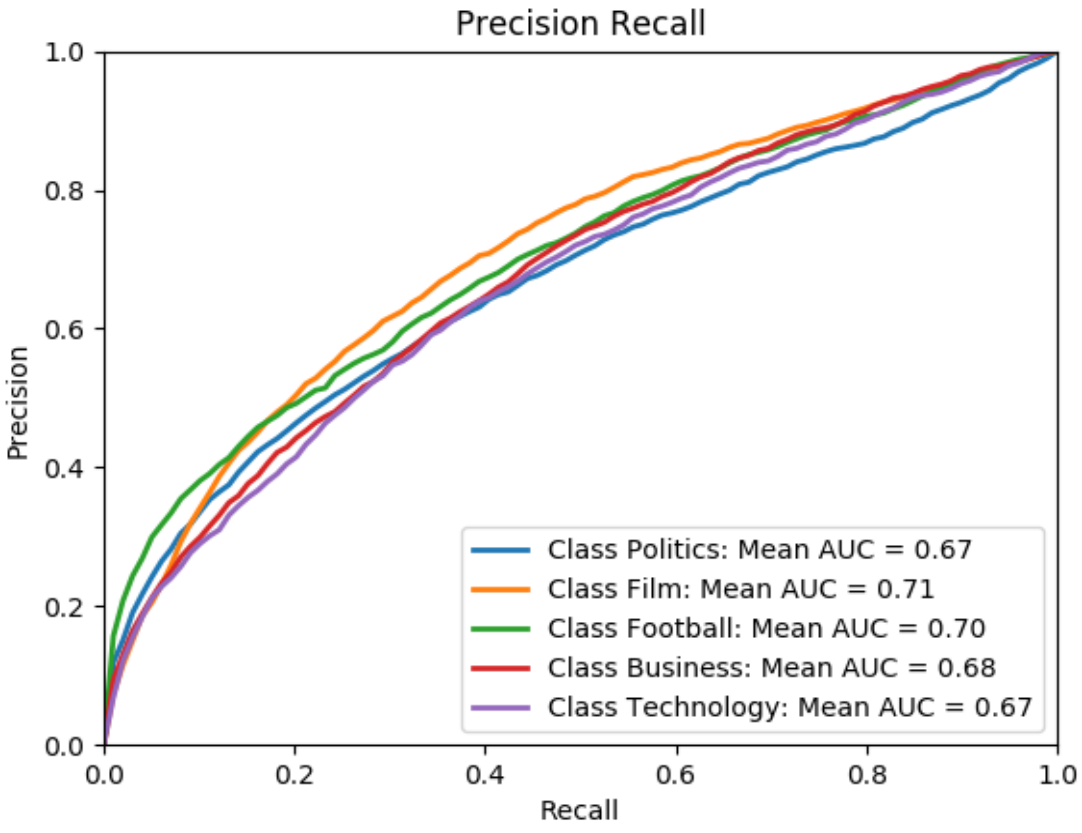
Γενικά:

Σε όλους τους αλγορίθμους, χρησιμοποιήθηκε ο TfidfVectorizer του sklearn για την μείωση των features σε 200, ο οποίος βρέθηκε να είναι ο βέλτιστος αριθμός features για ταχύτητα **σε σχέση** με την ακρίβεια αποτελεσμάτων. Επίσης χρησιμοποιήθηκε TruncatedSVD για την μείωση των διαστάσεων. Τέλος, αυτά εφαρμόστηκαν μέσω pipelining στα δεδομένα.

1. Naive Bayes:

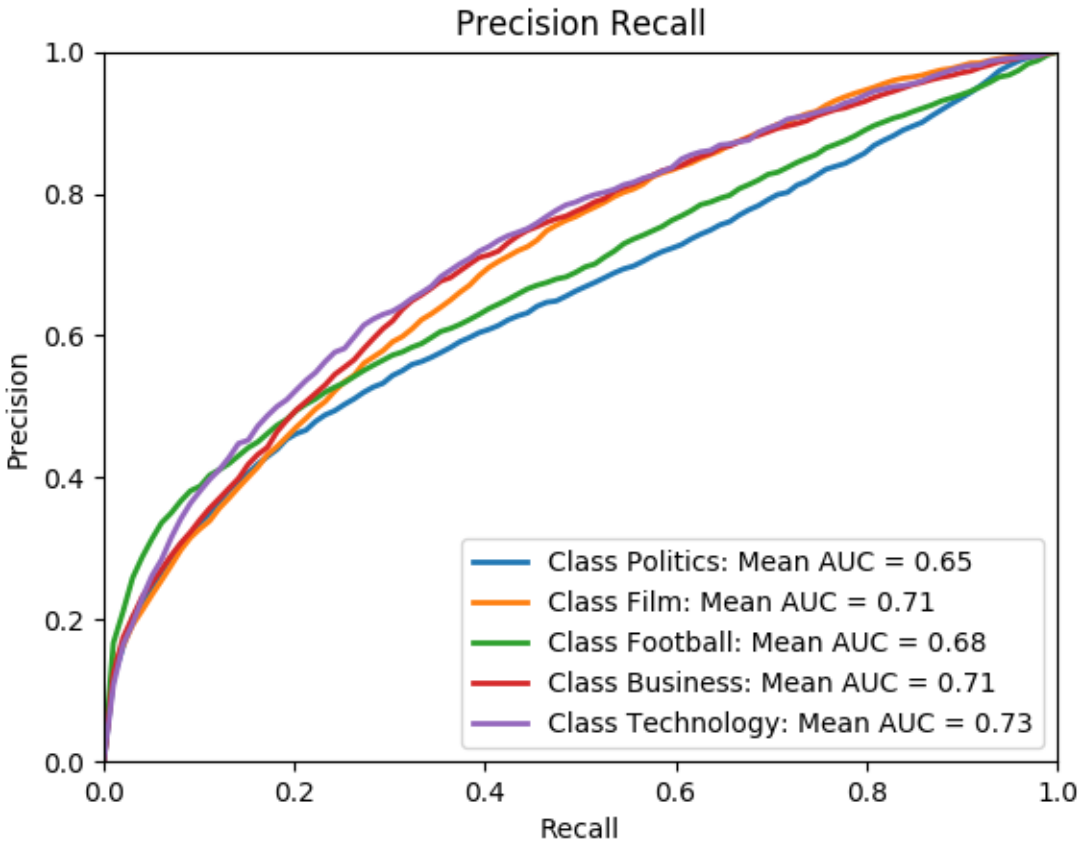
Ως υλοποίηση του αλγορίθμου Naive Bayes χρησιμοποιήθηκε ο Bernoulli Naive Bayes. Ο αλγόριθμος αυτός έδωσε χειρότερα αποτελέσματα συγκριτικά με τους υπόλοιπους.

Παρακάτω φαίνεται το ROC curve μετά από 10-fold cross validation:



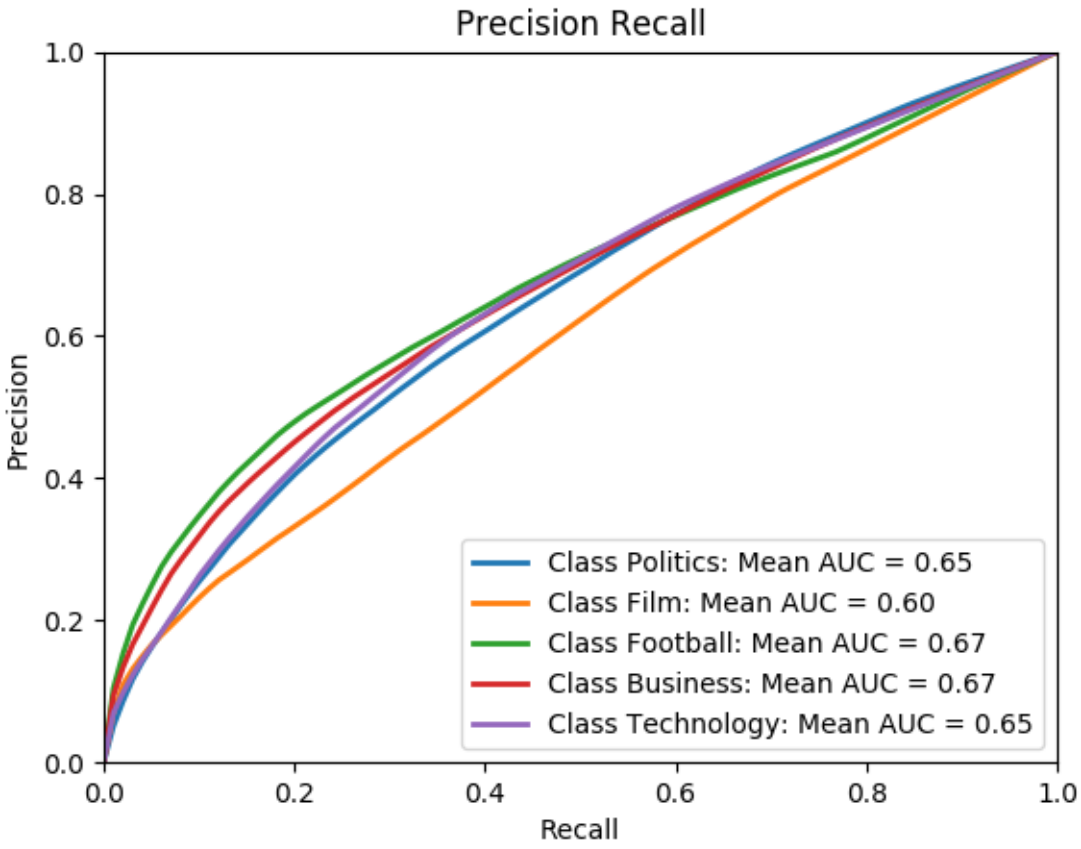
2. SVM (Support Vector Machines):

Για τον αλγόριθμο SVM, βρέθηκε πως καλύτερη απόδοση, και από άποψη ταχύτητας, αλλά και ακρίβειας, είχε ο linear kernel, έτσι χρησιμοποιήθηκε αυτός. Παρακάτω φαίνονται τα ROC curves της κάθε κλάσης



3. Random Forest:

Βρέθηκε πως αυτός ο αλγόριθμος είναι ο ταχύτερος και με τη μεγαλύτερη ακρίβεια σε σχέση με τους υπόλοιπους αλγορίθμους. Είναι ο αλγόριθμος που χρησιμοποιήθηκε για την πρόβλεψη του testing dataset. Παρακάτω φαίνονται τα ROC curves των κλάσεων.



4. K-Nearest Neighbours:

Σε αυτό τον αλγόριθμο, για κάθε κείμενο στο training set υπολογίζεται η απόσταση από όλα τα υπόλοιπα. Οι αποστάσεις αυτές ταξινομούνται κατά αύξουσα σειρά και επιλέγονται οι k μικρότερες (στα test μας, k = 20). Στην συνέχεια, ελέγχεται η κλάση των γειτόνων αυτών και επιλέγεται ως κλάση του κειμένου η πιο συχνή κλάση. Τα ROC curve και AUC δεν είναι διαθέσιμα για αυτό τον αλγόριθμο (βλ. 1.1)

2. Clustering:

Για το clustering χρησιμοποιήθηκε ο αλγόριθμος Kmeans (++ alternative για καλύτερη απόδοση) από την βιβλιοθήκη sklearn.cluster.KMeans. Το clustering έγινε στο training set με προεπεξεργασία αφαίρεσης stop words, Tfidf vectorizer, TruncatedSVD για την επιλογή χαρακτηριστικών και μείωση διαστάσεων αναπαράστασης καθώς και την κανονικοποίηση των διανυσμάτων για μεγαλύτερη ταχύτητα υπολογισμού. Παρακάτω παρουσιάζεται ένα αποτέλεσμα των ποσοστών των άρθρων κάθε κατηγορίας που περιέχεται σε κάθε συστάδα.

Μέγιστος αριθμός features που επιλέχθηκαν: 200

KMeans class percentage per cluster

Cluster #	Politics	Business	Football	Film	Technology
Cluster 1	0.0422	0.0037	0.0037	0.942	0.008
Cluster 2	0.186	0.041	0.030	0.378	0.3625
Cluster 3	0.001	0.007	0.944	0.003	0.42
Cluster 4	0.946	0.002	0.006	0.041	0.002
Cluster 5	0.001	0.982	0.002	0.001	0.012

3. Word Clouds:

Η βιβλιοθήκη που χρησιμοποιήθηκε για την παραγωγή των word clouds βρίσκεται εδώ [https://libraries.io/pypi/wordcloud]. Για την δημιουργία του εκάστοτε word cloud, δημιουργείται ένα string από όλα τα κείμενα της κατηγορίας το οποίο παρέχεται στην συνάρτηση wordcloud().generate(<str object>) και στην συνέχεια εμφανίζεται. Τα παραγμένα word clouds φαίνονται παρακάτω, καθώς και στον ξεχωριστό φάκελο word_clouds. Παρατηρούμε πως κάποιες από τις συχνά χρησιμοποιούμενες λέξεις, δεν είναι χαρακτηριστικές της κλάσης στην οποία αναφέρονται (π.χ. ‘said’, ‘also’, κλπ). Αυτό σημαίνει πως θα χρειαστεί να τις αφαρέσουμε από τους αλγορίθμους classification (όπως και έγινε).

Image 1 : Business

Image 2 : Politics

Image 3 : Film

Image 4 : Football

Image 5 : Technology

4. Αποτελέσματα:

Σύμφωνα με τα στατιστικά, ο αλγόριθμος Random Forest μας έδωσε τα καλύτερα αποτελέσματα. Έτσι, χρησιμοποιήθηκε αυτός για την πρόβλεψη κατηγορίας για το testing data set (data_sets/test_set.csv). Το σχετικό αρχείο βρίσκεται στο output/testSet_categories.csv

5. Βιβλιοθήκες:

Οι βιβλιοθήκες που χρησιμοποιήθηκαν:

1. WordCloud [https://libraries.io/pypi/wordcloud]
2. SciKitLearn [http://scikit-learn.org/stable/index.html]
3. SciPy [https://www.scipy.org/]
4. Nltk [http://www.nltk.org/]