Αναφορά 1ης Άσκησης Τεχνικές Εξόρυξης Δεδομένων

1)Υλοποίηση Word Cloud

Για την υλοποίηση του word cloud χρησιμοποιήσαμε μια έτοιμη βιβλιοθήκη από το github.

https://github.com/amueller/word_cloud

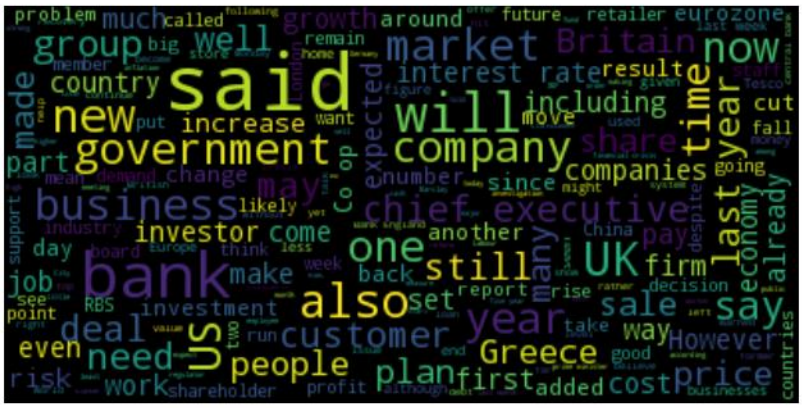
Προσθέσαμε και επιπλέον stopwords που παρατηρήσαμε στις εικόνες μετά τις πρώτες εκτελέσεις.

Ακολουθούν τα αποτελέσματα εικόνων που πήραμε:

Films:

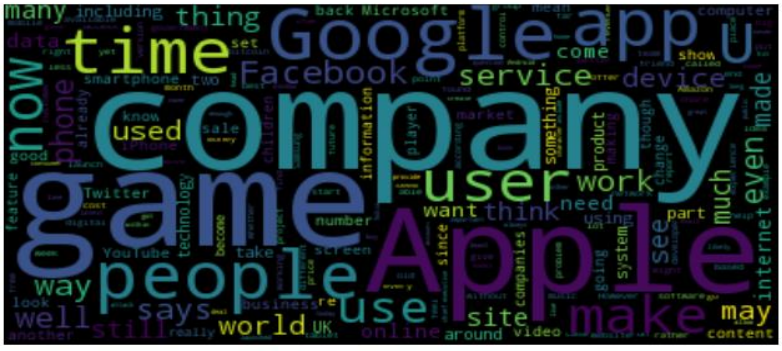
Football:

Politics:



Buisness:

Technology:



2)Υλοποίηση Συσταδοποίησης (Clustering)

Για τη συσταδοποίηση των δεδομένων δημιουργήσαμε ένα script σε Python ("cluster.py") στο οποίο υλοποιήσαμε τον αλγόριθμο K-Means. Για την υλοποίηση του K-Means ακολουθήσαμε τον αλγόριθμο που βρίσκεται στις σημειώσεις

του μαθήματος.

Πιο συγκεκριμένα τα βήματα της υλοποίησης είναι τα ακόλουθα:

1. Διαβάζει τα δεδομένα από το αρχείο train_set.csv.
2. Δημιουργεί ένα vocabulary με κάθε λέξη που υπάρχει στα άρθρα.
3. Για κάθε άρθρο δημιουργεί ένα vector με την απαρίθμηση του αριθμού εμφανίσεων της κάθε λέξης.
4. Επιλέγει 5 τυχαία άρθρα ως αρχικά centroids.
5. Σε κάθε επανάληψη του loop του K-Means τα άρθρα χωρίζονται σε clusters με βάση το μέγιστο cosine similarity τους από τα centroids.
6. Στη συνέχεια ο αλγόριθμος ελέγχει αν υπάρχουν αλλαγές στα clusters (αν δεν υπάρχουν αλλαγές ο αλγόριθμος τερματίζει). Για τον παραπάνω έλεγχο αρχικά συγκρίνεται το μέγεθος των clusters. Αν αυτό είναι ίδιο για κάθε cluster, τότε πραγματοποιείται έλεγχος και στα στοιχεία τους .
7. Εφόσον ο αλγόριθμος δεν τερματίσει υπολογίζονται καινούρια centroids (με βάση την μέση τιμή).
8. Επαλαμβάνει τα βήματα 5-7 μέχρι να μην υπάρχουν αλλαγές στα clusters.

Ο αλγόριθμος δεν τερματίζει σωστά και επομένως δεν εμφανίζει αποτελέσματα , πιθανόν λόγω επιλογής αρχικών σημείων. Δεν καταφέραμε να υλοποιήσουμε τον αλγόριθμο ουτε με ευκλήδεια απόσταση ωστε να έχει αποτελέσματα.

Γενικώς δοκιμάσαμε και άλλες υλοποιήσεις, αρχικά του scikit ο οποίος όμως δέχεται μόνο την ευκλήδεια απόσταση και άρα δεν μας κάλυπτε για την άσκηση. Δοκιμάσαμε επίσης μια υλοποίηση που βρήκαμε στο stack overflow και πάλι χωρίς ικανοποιητικά αποτελέσματα. Τέλος δοκιμάσαμε να εφαρμόσουμε την εκδοχή του nltk.cluster.kmeans

χωρίς και εκεί να έχουμε κάποιο αποτέλεσμα που να μας ικανοπποιεί ως προς την εργασία μας.

3)Υλοποίηση Κατηγοριοποίησης(Classification)

Για την κατηγοριοποίηση των δεδομένων δημιουργήσαμε ένα script σε Python (“class.py”).

Υλοποιήσαμε όλες τις μεθόδους τις εργασίας(SVM,Random Forests,Naïve Bayes, K-Nearest Neghbor), καθώς και όλες τις μετρικές(Precision / Recall / F-Measure, Accuracy, AUC, ROC Plot).

Υλοποίηση συναρτήσεων:

1)Συνάρτηση κατηγοριοποίησης(def Classification), η οποία χρησιμοποιεί pipeline για να κάνει την διαδικασία ευκολότερη, καθώς και gridsearch για το cross-validation.

2)Συνάρτηση πρόβλεψης(def predict_category), η οποία κάνει train όλο το δοσμένο dataset, προβλέπει κατηγορίες για το κάθε άρθρο και τέλος κάνει export τα αποτελέσματα σε ένα csv.

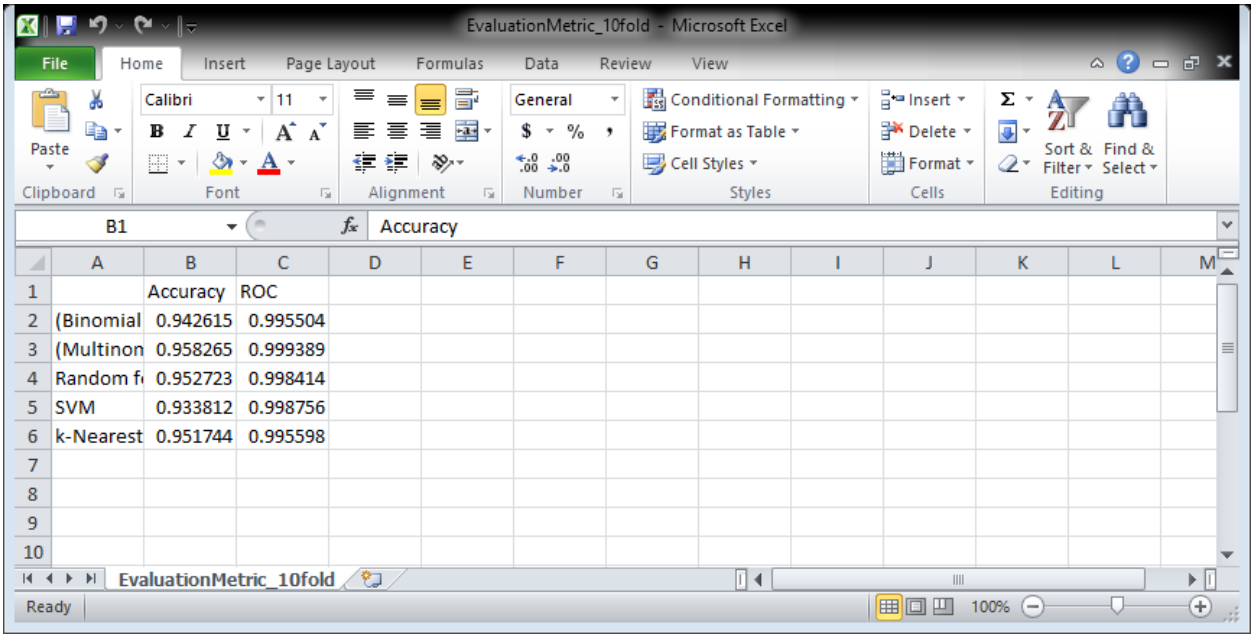
3)Συνάρτηση ROC(def roc_curve_estimator), η οποία μετατρέπει δοσμένο dataset σε binary μορφή και υπολογίζει το AUC.

Τέλος στη main κάνουμε merge τον τίτλο με το περιεχόμενο ώστε να αξιοποιηθεί και καλώντας τις συναρτήσεις υπολογίζουμε όλα τα ζητούμενα.

Είχαμε ένα θέμα με το πέρασμα μερικών απο τις μετρικές σε csv λόγω ενός τύπου(multiclass) κι έτσι δεν φαίνονται στο ζητούμενο csv. Κατα την εκτέλεση όμως εμφανίζονται κανονικά τα αποτελέσματα στην κονσόλα.

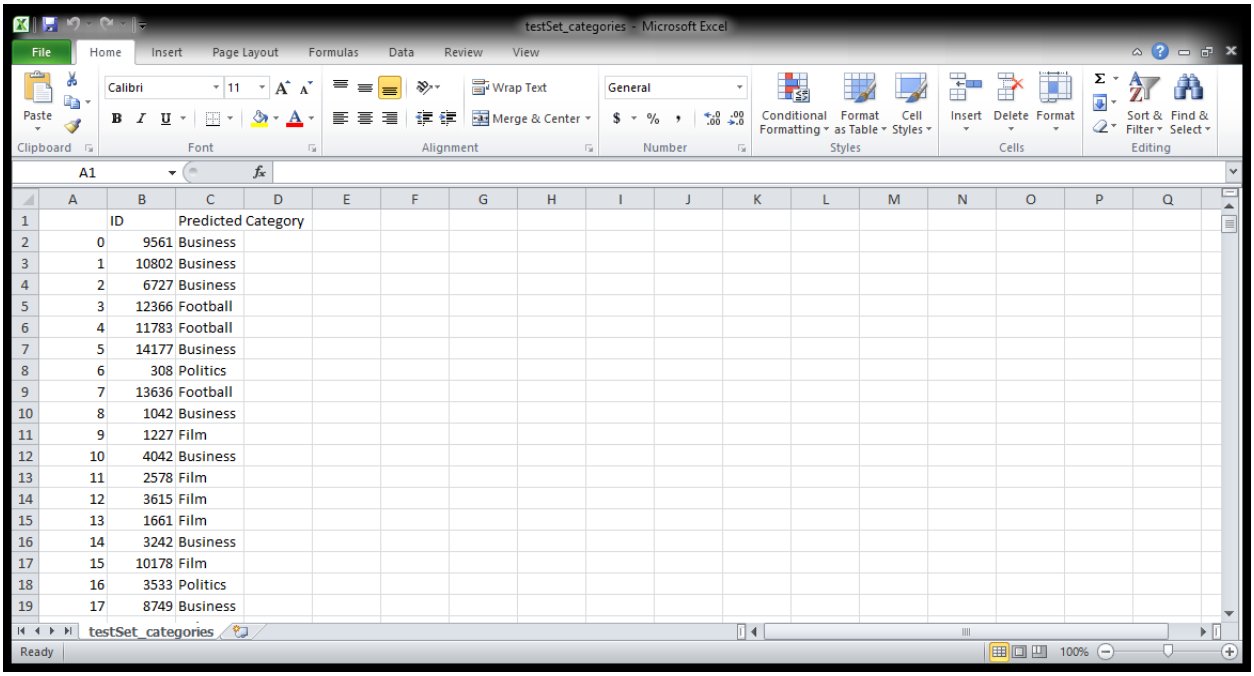
Ακολουθούν τα αρχεία εξόδου:

Evaluation metics-10 fold :



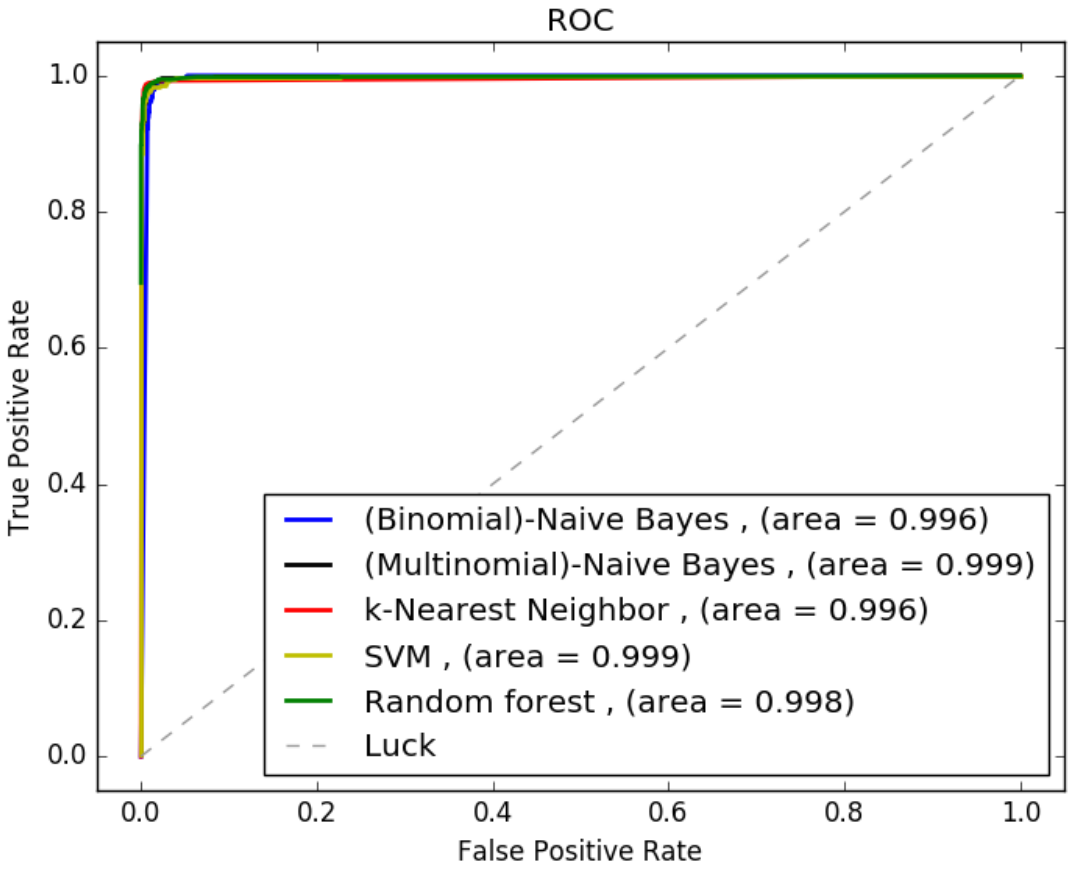
	A	B	C
1		Accuracy	ROC
2	(Binomial	0.942615	0.995504
3	(Multinon	0.958265	0.999389
4	Random f	0.952723	0.998414
5	SVM	0.933812	0.998756
6	k-Nearest	0.951744	0.995598
7			
8			
9			
10			

TestSet_categories: (ένα κομμάτι του)



	A	ID	Predicted Category
1			
2	0	9561	Business
3	1	10802	Business
4	2	6727	Business
5	3	12366	Football
6	4	11783	Football
7	5	14177	Business
8	6	308	Politics
9	7	13636	Football
10	8	1042	Business
11	9	1227	Film
12	10	4042	Business
13	11	2578	Film
14	12	3615	Film
15	13	1661	Film
16	14	3242	Business
17	15	10178	Film
18	16	3533	Politics
19	17	8749	Business

Roc Plot:


ROC
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
(Binomial)-Naive Bayes , (area = 0.996)
(Multinomial)-Naive Bayes , (area = 0.999)
k-Nearest Neighbor , (area = 0.996)
SVM , (area = 0.999)
Random forest , (area = 0.998)
Luck