Logistic Regression Project 1

Γεώργιος Κωνσταντίνος Ζαχαρόπουλος sdi1900061

Νοέμβριος 29, 2022

0.1 Introduction

Στόχος της εργασίας είναι η υλοποίηση ενός sentiment classifier για ένα data set από imdb movie reviews με την χρήση ενός logistic regression classifier.

0.2 Imports

Κάνουμε import ότι είναι απαραίτητο από τις βιβλιοθήκες: pandas, numpy, sklearn, matplotlib, nltk, re

0.3 Loading Data Set

Διαβάζουμε το data set και δημιουργούμε μια στήλη ¨αποτελέσματος' στην οποία βάζουμε 0 αν η κριτική είναι αρνητική και 1 αν είναι θετική. Αυτές θα είναι και οι κλάσεις μας.

0.4 Data Pre-processing

Χρειάζεται να καθαρίσουμε το data set καθώς δεν εκφράζουν όλες οι λέξεις sentiment και αυτό μπορεί να μπερδέψει στο training του μοντέλου.
Για παράδειγμα στα reviews υπάρχουν html tags, αριθμοί που εκφράζουν σκορ και stop words που δεν προσφέρουν στην εκμάθηση του μοντέλου. Όλα αυτά θα τα αφαιρέσουμε από το data set.

0.4.1 Separate features from targets

Φτιαχνουμε δύο καινούργια sets, τα X , Y στα οποία θα εμπεριέχονται τα features και targets αντίστοιχα. Στο X ως features βάζουμε μόνο τα reviews, ενώ στο Y την στήλη με τα 0 και 1 που δημιουργήσαμε πριν.

0.4.2 Split Train and Test Sets

Μετά χωρίζουμε το X και το Y σε δύο σύνολα, train και test. Έτσι θα έχουμε ένα άγνωστο για το μοντέλο test set πάνω στο οποίο θα το βαθμολογήσουμε και ένα train set με το οποίο θα εκπαιδεύσουμε το μοντέλο μας. Είναι ένας τρόπος να δούμε την απόδοση του.

0.4.3 Vectorization (Feature Extraction)

Για να μπορέσουμε να εκπαιδεύσουμε το μοντέλο θα πρέπει να απλοποιήσουμε τα δεδομένα που θα εισάγουμε στο πρόγραμμα. Η μέθοδος bag of words μοντελοποιεί την φυσική γλώσσα και εξαγάγει τα features από text documents. Χρησιμοποιούμε το countVectorizer βάζοντας ως ορίσματα το lowercase=True καθώς έτσι ομαλοποιούμε τα δεδομένα, το stopwords='english' για να αφαιρέσουμε λέξεις χωρίς sentiment value (stop-words), df-max = 3000 το οποίο είναι και αυτό ένα μέτρο για να την αντιμετώπιση του overfit καθώς περιορίζουμε τον αριθμό των features (feature selection) και ngram-range=(1, 2) για να έχουμε unigrams και bigrams το οποίο με βάση δικά μου πειράματα δίνει καλύτερα αποτελέσματα μιας και θα εμπεριέχονται στα features λέξεις όπως not worth, not good κτλ.

Ο παρακάτω πίνακας δείχνει το accuracy των διαφορετικών solver χώρις κάποιο data pre-processing

	lbfgs	liblinear	newton-cg	saga	sag
Accuracy:	0.8828038213730283	0.8829149077982671	0.8829149077982671	0.8888024883359253	0.8874694512330593

Σημείωση:Οι solvers saga και sag χρειάζονται αρκετά περισσότερη ώρα για κάνουν fit χωρίς ουσιαστικά καλύτερα αποτελέσματα οπότε δεν θα τους χρησιμοποιήσουμε στη συνέχεια.

Ενώ στον επόμενο πίνακα έχουμε εφαρμόσει ότι έχουμε αναφέρει προηγουμένος για data pre-processing

	lbfgs	liblinear	newton-cg
Accuracy:	0.8760275494334593	0.8760275494334593	0.8760275494334593

Παρατηρούμε ότι οι solvers δίνουν τα ίδια αποτελέσματα οπότε η επιλογή του solver θα κριθεί από την αύξηση της

επίδοσης μετά την αλλαγή των υπερπαραμέτρων. Επίσης βλέπουμε ότι έχει πέσει το accuracy, αυτό μπορεί να οφείλεται στο ότι το μοντέλο είχε μπερδευτεί και έκανε classify με features που δεν είχαν sentiment value όπως το the, I, me κτλ.

0.5 Train the Model and Tune the Hyperparameters

Ετρεξα ένα Grid Search CV για να βρω τις καλύτερες υπερπαραμέτρους για τον καθένα με βάση το accuracy. Τα καλύτερα αποτελέσματα τα έδωσε ο liblinear. Αυτή η εξαντλητική αναζήτηση για τις πιθανές τιμές των υπερπαραμέτρων μέσα από τα grid-values έδωσε ως καλύτερες επιλογές τα: penalty=l2, tol=0.1, C=0.1 .
Επειτα κάνουμε Cross validation με kfold = 5 για να ελέγξουμε ότι το μοντέλο μας είναι αποδοτικό και σε διαφορετικά train και test case. Τα αποτελέσματα είναι τα εξής:

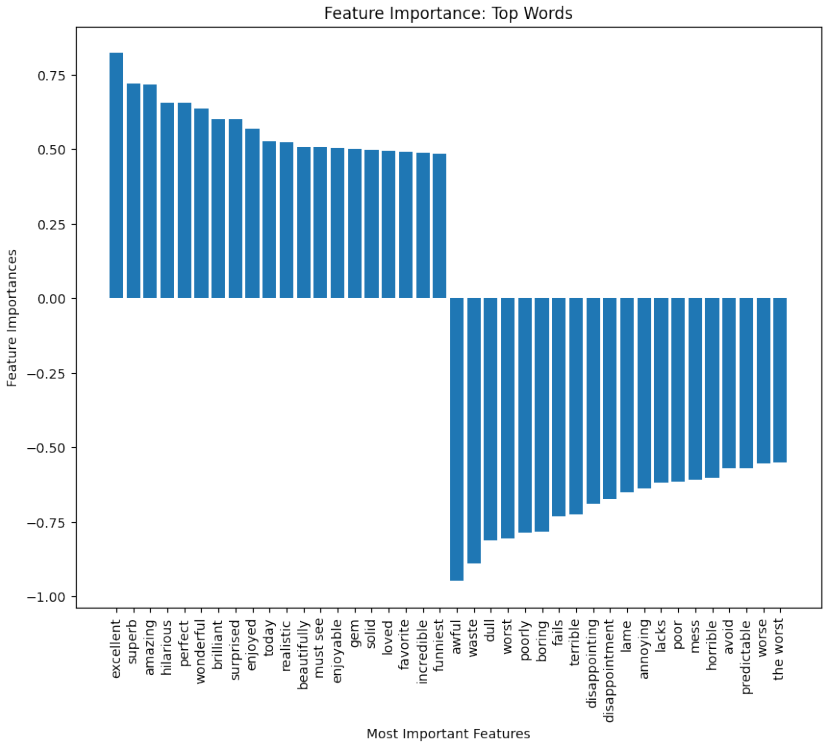
	Accuracy	Precision	Recall	f1
fold 1:	0.87725632	0.86502058	0.88942172	0.87705146
fold 2:	0.88001666	0.87609409	0.88627559	0.88115543
fold 3:	0.87737814	0.87031624	0.89237057	0.88120544
fold 4:	0.8784891	0.86951814	0.89098474	0.88012056
fold 5:	0.87932232	0.87909667	0.88128106	0.88018751

Οπότε φαίνεται μέχρι στιγμής ότι το μοντέλο μας δεν κάνει overfit αφού υπάρχει συνέπεια στα διαφορετικά σκορ.

0.6 Evaluate model's performance and Feature Importance

0.6.1 Feature selection

Είναι χρήσιμο να ελέγξουμε τα πιο σημαντικά features του μοντέλου γιατί βάση αυτών θα κάνει το classification. Εμφανίζουμε τα 100 πιο χρήσιμα features με αρνητικό και θετικό πρόσημο και ελέγχουμε αν έχουν sentiment value. Οτι θεωρούμε ότι δεν έχει αξία για το μοντέλο όπως (ονόματα ηθοποιών, χαρακτήρων και ταινιών) και πιστεύουμε ότι μπορεί να το μπερδέψει,τα προσθέσουμε στα my-stop-words ώστε όταν κάνουμε train το μοντέλο μας να τα αγνοήσει. Στο παρακάτω σχήμα φαίνονται τα 20 πιο σημαντικά θετικά και αρνητικά features. Βλέπουμε πως το μοντέλο μας έχει διαλέξει ως top features λέξεις με sentiment value.



	Before Feature Cleaning	After Feature Cleaning
Accuracy:	0.8760275494334593	0.8862475005554321

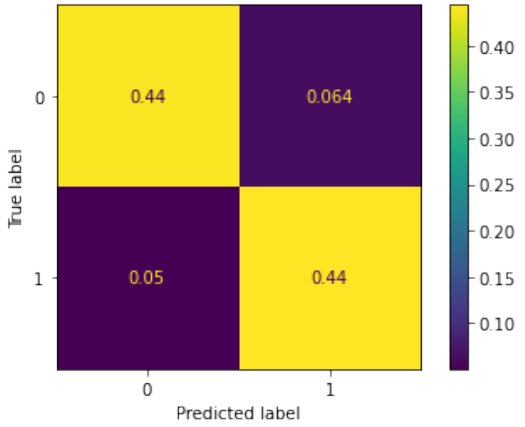
Βλέπουμε πως υπάρχει μια μικρή βελτίωση στην ακρίβεια του μοντέλου

0.6.2 Useful Metrics and Plots

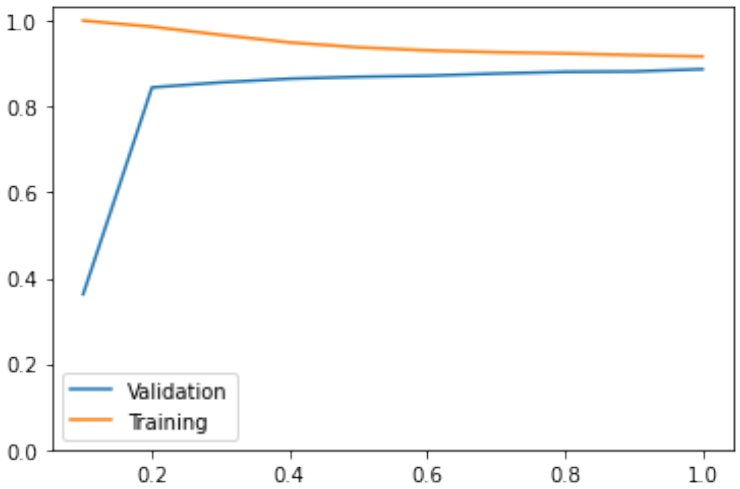
Παρακάτω φαίνεται η απόδοση του μοντέλου βάση των μετρικών Accuracy, f1, Precision, Recall πάνω στο test set:

	Accuracy	f1	Precision	Recall
classifier:	0.8862475005554321	0.8865248226950354	0.8748906386701663	0.8984725965858041

Παρουσιάζεται και το Confusion Matrix το οποίο δείχνει τα TN FP FN TP που βγάζει το μοντέλο μας στο train set.



Επίσης παρακάτω βρίσκεται και το γράφημα με τα learning curves του μοντέλου μας



Το γράφημα μας δίνει σημαντικές πληροφορίες για το μοντέλο μας. Καταρχάς βλέπουμε πως όλο και συγκλίνουν οι δύο καμπύλες όσο αυξάνουμε τα μεγέθη του train και του test set και δεν φτάνουν σε ένα σταθερό επίπεδο. Αυτό δείχνει ότι το μοντέλο μας δεν κάνει overfit ούτε undefit.

0.6.3 Overfit

Για να αποφύγουμε το overfit έχουμε βάλει πέναλτι l2 norm ώστε να πετύχουμε regularization. Επίσης κάνουμε και feature selection βάζοντας ως μέγιστο αριθμό features = 3000.

0.6.4 Underfit

Μπορούμε να παρατηρήσουμε ότι το μοντέλο μας δεν κάνει underfit καθώς έχει αρκετά καλές επιδόσεις στο training data, τάξης του 0.88

0.7 Comments

Μερικά κομμάτια στον κώδικα έχουν γίνει commented για να μην αργεί να τρέξει το παραδοτέο, όπως το κομμάτι του cross validation και της εύρεσης των υπερπαραμέτρων. Όμως τα αποτελέσματα αυτών τα έχω αναφέρει αναλυτικά στο report. Μπορείται άμα θέλετε να τρέξετε κάνοντάς τα uncomment.