Επεξεργασία Φυσικής Γλώσσας – 2022/20223 - Ο.Π.Α

**Εργασία 3**

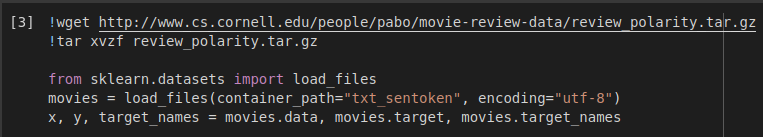
Φοίβος-Αστέριος Νταντάμης (f3312204 ) ([phivos93@yahoo.com](mailto:phivos93@yahoo.com))

**Άσκηση 9 notebook url:** https://colab.research.google.com/drive/1\_qvSOYCJyMEG4u0OF5JtmVKuTPbjpIiY?usp=sharing

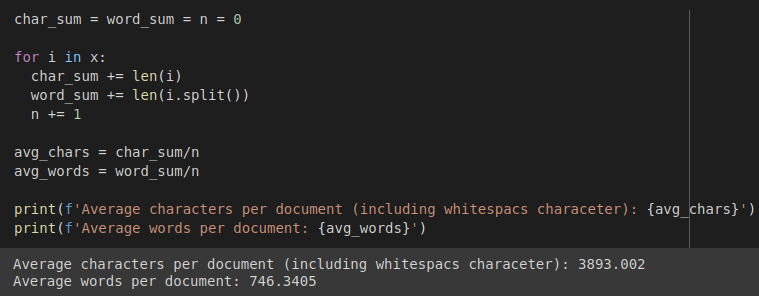
**Περιγραφή Dataset**

Το dataset όπως και η προεπεξεργασία του είναι τα ίδια με την προηγούμενη άσκηση. Για λόγους πληρότητας ωστόσο περιγράφουμε συνοπτικά ξανά τη διαδικασία.

Χρησιμοποιούμε το Cornell Movie Review Data από το ομώνυμο πανεπιστήμιο. Το dataset αυτό περιέχει κριτικές ταινιών από το αρχείο του IMDb κατηγοριοποιημένες ως θετικές(pos) ή αρνητικές(neg). Το μέγεθος του συνόλου όλων των κριτικών είναι 2000, στις οποίες οι 1000 ανήκουν στην κατηγορία pos και οι υπόλοιπες 1000 στην κατηγορία neg.



Ο μέσος όρος χαρακτήρων ανά κριτική είναι 3893.002, ενώ ο μέσος όρο των λέξεων είναι 746.3405.



**Προεπεξεργασία**

Το πρώτο βήμα που εφαρμόζουμε είναι να αφαιρέσουμε τα stopwords της αγγλικής γλώσσας, φιλτράροντας όλα τα κείμενα για τις λέξεις που θεωρεί το nltk ως stopwords.

Στη συνέχεια με τη βοήθεια του WordNetLemmatizer αντικαθιστούμε όλες τις λέξεις με τα λήμματά τους.

Συνεχίζοντας, μετατρέπουμε όλους τους χαρακτήρες σε μικρούς.

Σε αυτήν την φάση έχουν προκύψει λέξεις οι οποίες ανήκουν στα stopwords, οπότε θα ξαναφιλτράρουμε τα κείμενα.

Έπειτα αφαιρούμε τα σημεία στίξης και τα σύμβολα που ουσιαστικά δεν αποτελούν κάποια λέξη.

Αφαιρούμε όλες τις λέξεις που αποτελούνται από έναν χαρακτήρα μόνο, μιας και δεν προσφέρουν κάποια ιδιαίτερη πληροφορία για την κατηγοριοποίηση.

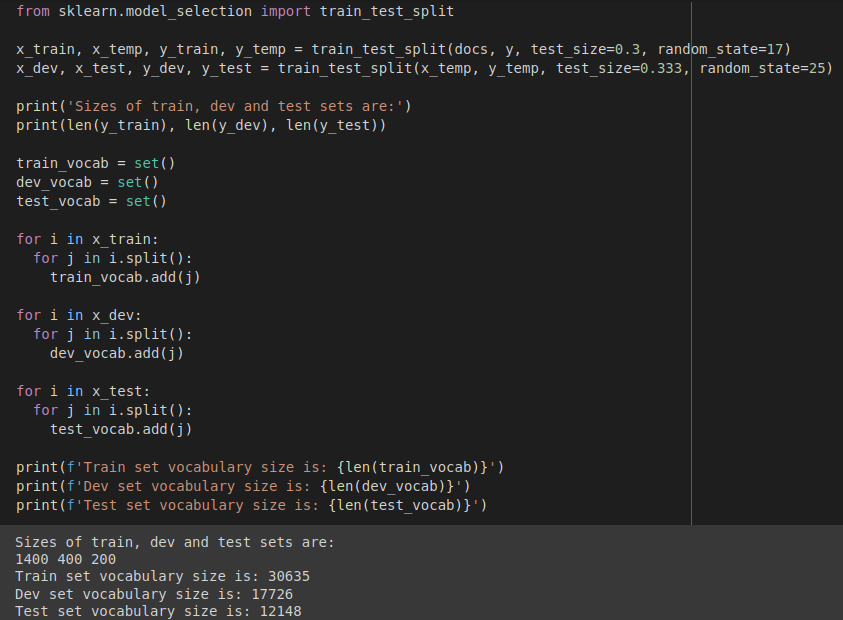
Σε αυτό το σημείο και πάλι έχουν προκύψει κάποιες λέξεις που είναι stopwords οπότε φιλτράρουμε εκ νέου.

Αφαιρούμε τώρα όλους τους αριθμούς.

Αφαιρούμε τέλος τα honorifics

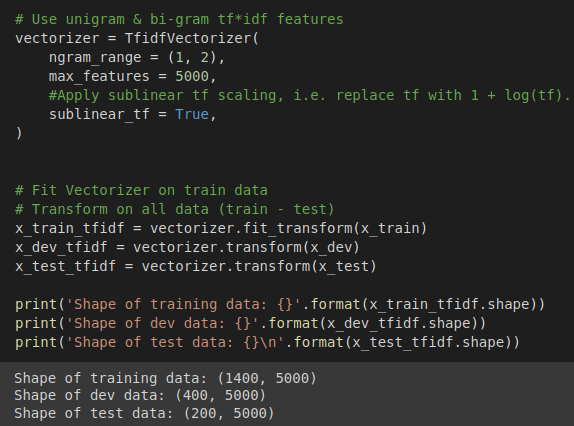
**Διαχωρισμός Dataset**

Σπάσαμε το Dataset σε Train, Development και Test sets με ποσοστά 70%, 20%, 10% αντίστοιχα. Προκύπτουν έτσι το Train Dataset με 1400 κριτικές και μέγεθος λεξιλογίου 30635. Το Development Dataset με 400 κριτικές και μέγεθος λεξιλογίου 17726. Το Testing Dataset με 200 κριτικές και μέγεθος λεξιλογίου 12148.

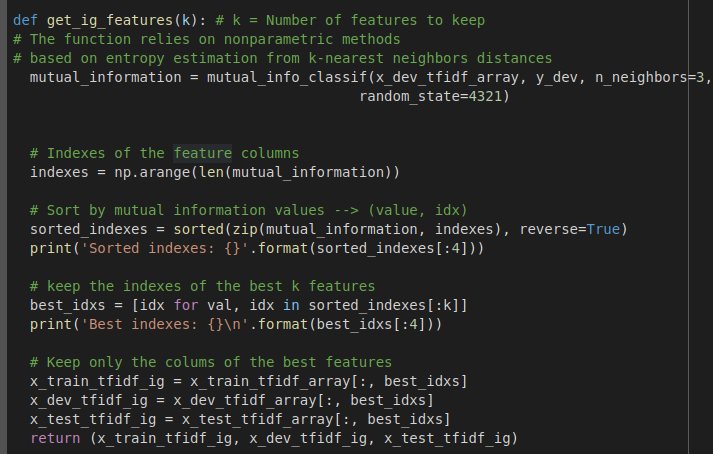


**Feature Exctraction**

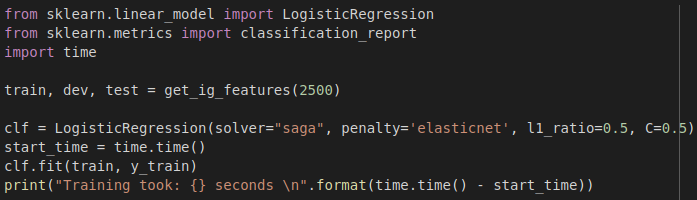
Προκειμένου να μετατρέψουμε τα κείμενα σε διανύσματα, χρησιμοποιούμε TF-IDF features.

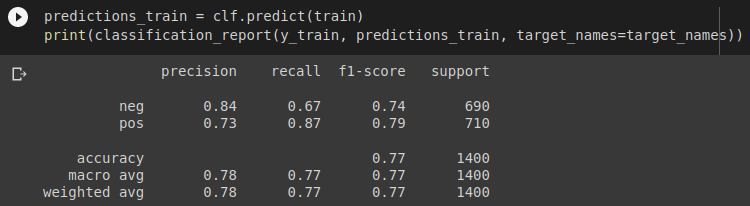


Ορίζουμε τώρα τη συνάρτηση get\_ig\_features, η οποία δέχεται ως όρισμα τον αριθμό των features που θέλουμε να κρατήσουμε και επιλέγει εκείνα για τα οποία μεγιστοποιείται το information gain.



Ως σημείο αναφοράς παίρνουμε τα αποτελέσματα του LogisticRegression με τις καλύτερες παραμέτρους όπως προέκυψαν στην προηγούμενη άσκηση.

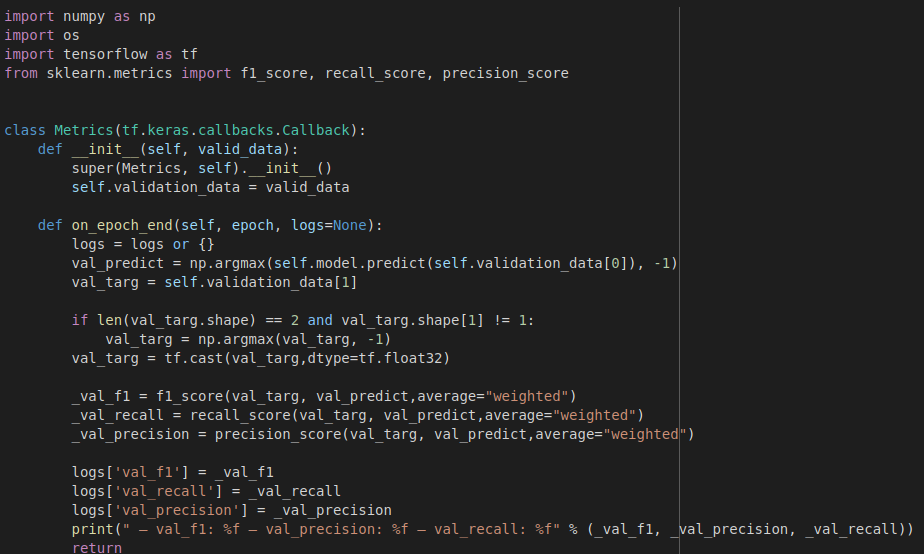




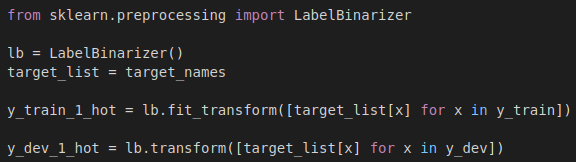
**MLP**

**Κλάση callback**

Αρχίζοντας να χτίζουμε το MLP μοντέλο μας, ορίζουμε αρχικά μία callback συνάρτηση, η οποία θα καλείται στο τέλος της κάθε εποχής του νευρωνικού μας δικτύου και θα υπολογίζει τα f1, precision και recall.



Συνεχίζοντας, οδηγούμενοι από τον τρόπο που ζητάνε τις κατηγορίες οι συναρτήσεις του tensorflow, δημιουργούμε 1-hot vectors για τις δύο κλάσεις του προβλήματός μας.



Στη συνεχεία ορίζουμε μία παραμετροποιήσιμη συνάρτηση στην οποία θα υλοποιείται το βασικό pipeline του αλγορίθμου που σχεδιάζουμε. Η συνάρτηση αυτή δέχεται τα εξής ορίσματα

**number\_of\_features\_ig**: ο αριθμός των features που θέλουμε να κρατήσουμε σύμφωνα με το information gain

**activation:** το είδος της συνάρτησης ενεργοποίησης για τα κρυφά επίπεδα του νευρωνικού μας δικτύου

**input\_size:** ο αριθμός των νευρώνων στο επίπεδο εισόδου του νευρωνικού μας δικτύου

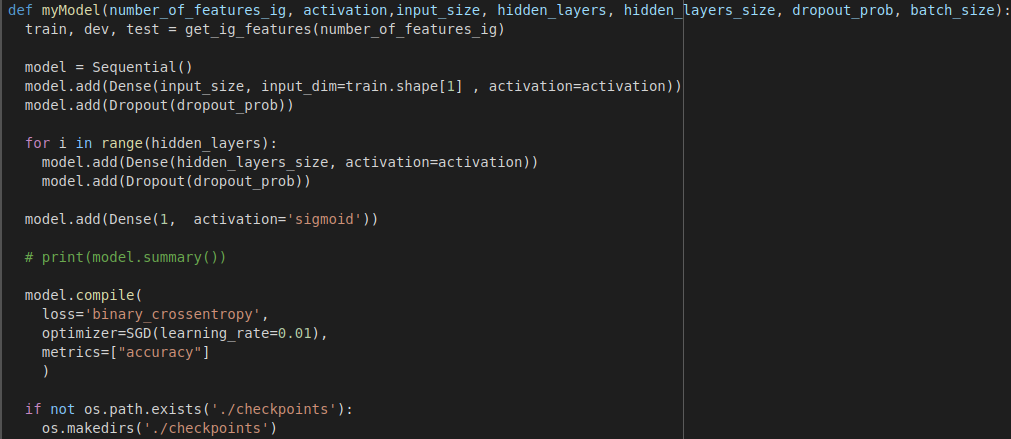
**hidden\_layers:** το πλήθος των κρυφών επιπέδων που θα περιέχεται στο νευρωνικό μας δίκτυο

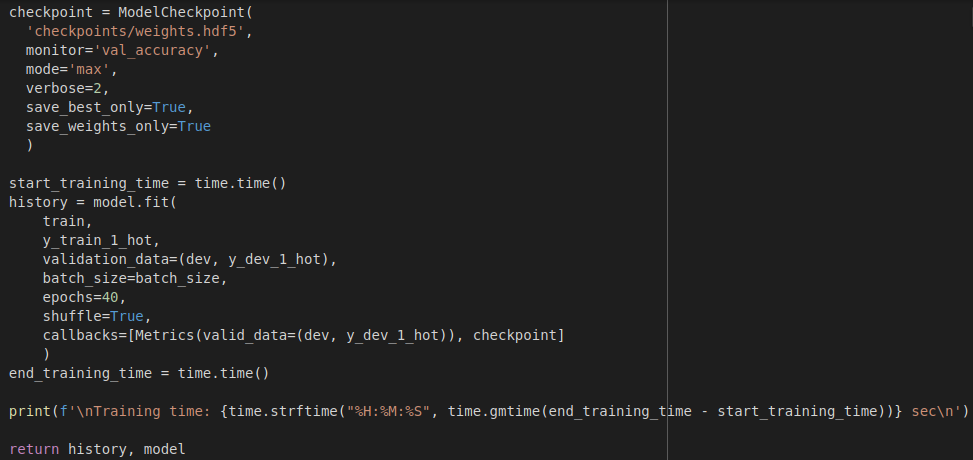
**hidden\_layers\_size:** ο αριθμός των νευρώνων στα κρυφά επίπεδα του νευρωνικού μας δικτύου

**dropout\_prob:** η πιθανότητα με την οποία κάθε νευρώνας κάνει drop out

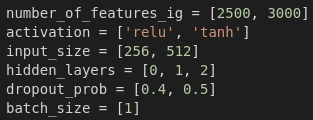
**batch\_size:** το μέγεθος του batch που χρησιμοποιείται για την εκπαίδευση

Μέσα στη συνάρτηση χτίζουμε το νευρωνικό μας δίκτυο ανάλογα με τις παραμέτρους. Λόγω του ότι το πρόβλημα είναι δυαδικό, το επίπεδο εξόδου χρησιμοποιεί sigmoid συνάρτηση ενεργοποίησης , όπως επίσης για loss χρησιμοποιείται η binary\_crossentropy.

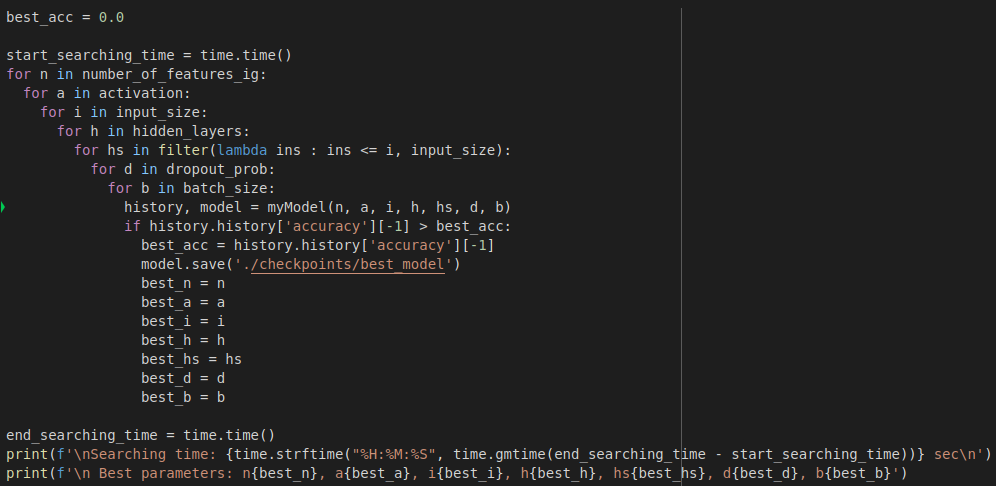




Στη συνέχεια ορίζουμε τον χώρο αναζήτησης στον οποίο θα ψάξουμε τις βέλτιστες τιμές των υπερπαραμέτρων. Ο χώρος είναι λίγο περιορισμένος διότι έτσι είναι και ο χρόνος :) . Το collab από ό,τι φαίνεται έχει περιορισμούς στην χρήση της gpu, οπότε η διαδικασία αυτή επαναλήφθηκε αρκετές φορές με αποτέλεσμα να μειώσουμε το εύρος αναζήτησης ώστε να βγει κάποιο αποτέλεσμα.

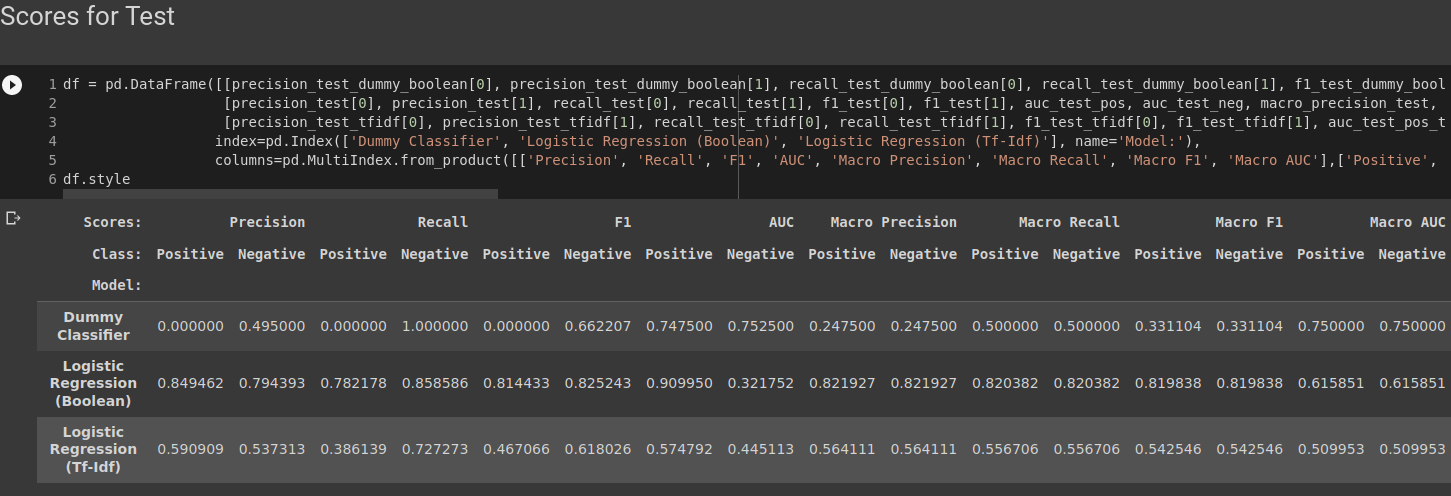


Συνεχίζοντας, με brute force θα ψάξουμε τον καλύτερο συνδυασμό υπερπαραμέτρων. Όλες οι παράμετροι επιτρέπεται να πάρουν όλες τις διαθέσιμες τιμές, εκτός από μέγεθος των κρυφών επιπέδων που στη βιβλιογραφία αναφέρεται ότι καλό είναι να κυμαίνεται μεταξύ του μεγέθους του επιπέδου εισόδου και το μέγεθος το επιπέδου εξόδου. Μετά το τέλος κάθε επανάληψης αποθηκεύουμε το καλύτερο μοντέλο με κριτήριο το accuracy.

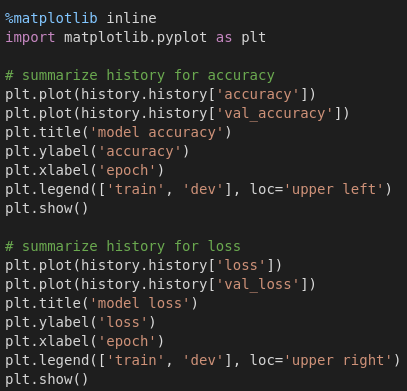


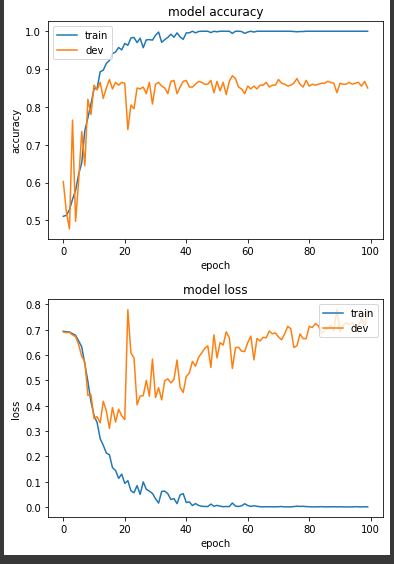
Από ό,τι φαίνεται το brute force δεν μπορεί να χρησιμοποιηθεί στο collab, μιας και πάλι σταμάτησε την εκτέλεση λόγω χρησιμοποίησης πόρων. Θα χρησιμοποιήσουμε για τον λόγο αυτόν ένα από ενδιάμεσα αποτελέσματα και όχι το πραγματικά καλύτερο, ώστε να μπορούμε να συνεχίσουμε.

Παραθέτουμε τα αποτελέσματα των μετρικών των προβλέψεων ώστε να συγκριθούν με τα αντίστοιχα του logisticRegression. Παραθέτουμε εδώ τόσο αυτά τα αποτελέσματα όσο και του Dummy Classifier που ζητούνται

****

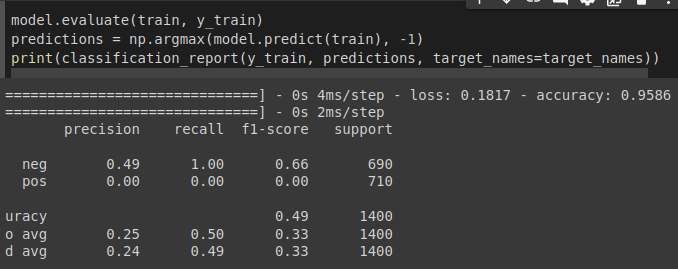
Τέλος σχεδιάζουμε τις καμπύλες για το accuracy και loss συναρτήσει του αριθμού των εποχών για τα training και development δεδομένα.

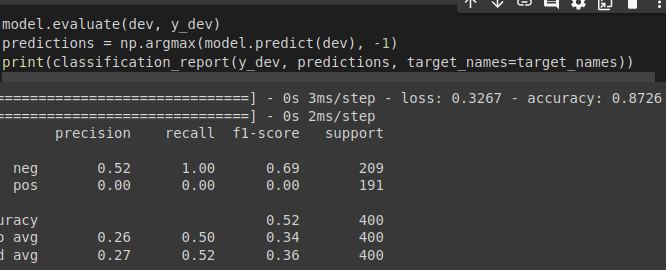


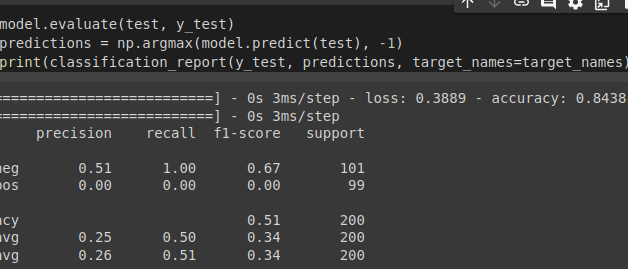
****

Παρατηρούμε ότι περίπου στην εποχή 12 είναι το βέλτιστο σημείο, όπου ναι μεν τα loss μειώνονται και τα accuracies αυξάνονται, αλλά δεν συμβαίνει ακόμη overfitting. Η μη ομαλή αύξηση ή μείωση κατά την αύξηση των εποχών, ίσως να οφείλεται στο ότι χρειαζόταν κάποιο μικρότερο learning rate για τα συγκεκριμένα δεδομένα. Οπότε θα ξανατρέξουμε τον αλγόριθμο εκπαίδευσης για τόσες εποχές.

Παραθέτουμε τα αποτελέσματα για κάθε κλάση







Τα νούμερα του classification report είναι προφανώς λάθος, είναι σαν προβλέπεται πάντα μόνο η αρνητική κλάση. Βλέπουμε από το evaluate όμως ότι το accuracy στα testing δεδομένα για το μοντέλο που δημιουργήθηκε με οδηγό τα development δεδομένα είναι 84.38. Δεν έχουμε υπολογίσει τα auc διότι δεν έχουν νόημα αν τα παραπάνω αποτελέσματα δεν είναι σωστά.