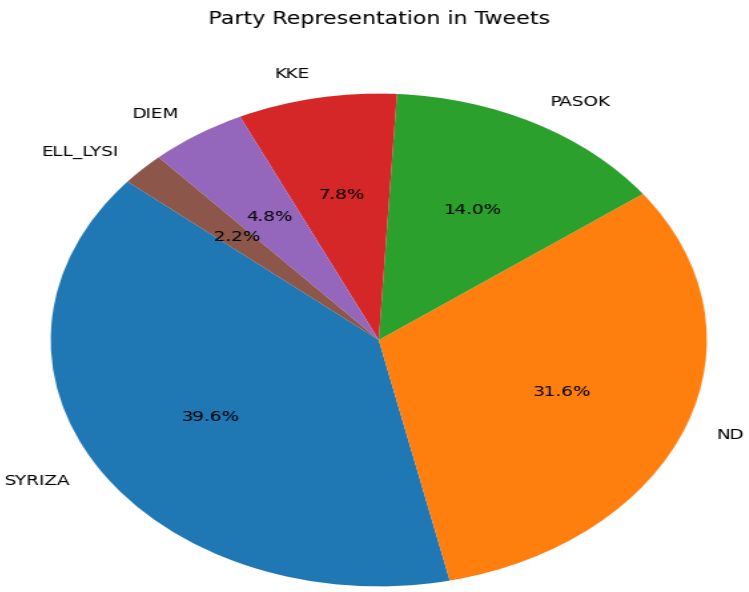
Ανάπτυξη Νευρωνικού Δικτύου για Πρόβλεψη Συναισθημάτων στο Twitter

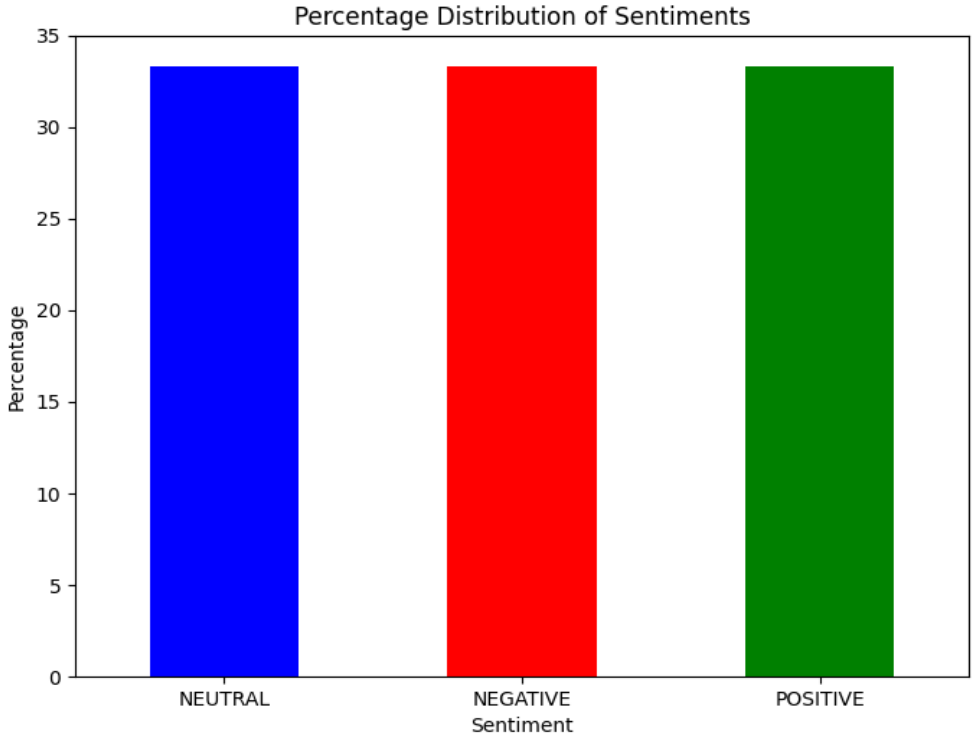
Ονοματεπώνυμο : Ευθύμιος Πατέλης

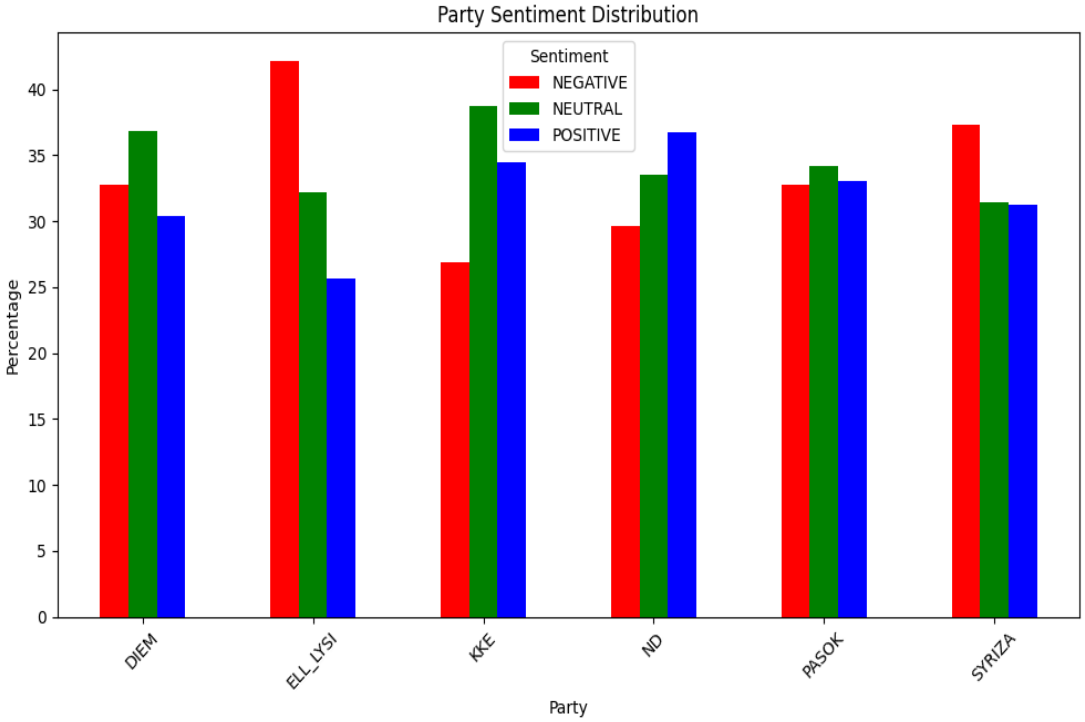
Αριθμός Μητρώου : 1115201300141

Οπτικοποίηση Δεδομένων

Στην παρούσα ανάλυση, επιλέχθηκαν τρία γραφήματα για την απεικόνιση ορισμένων βασικών πτυχών των δεδομένων. Αρχικά, χρησιμοποιήθηκε ένα ιστόγραμμα για την εύρεση της κατανομής των συναισθημάτων στο dataset. Έπειτα, υπολογίστηκε το γράφημα με τις ποσοστιαίες κατανομές των συναισθημάτων ανά κόμμα, αποκαλύπτοντας πώς τα διάφορα κόμματα εκλαμβάνονται συναισθηματικά στη δημόσια συζήτηση. Τρίτον, χρησιμοποιήθηκε ένα pie-chart για την αναπαράσταση της εκπροσώπησης των κομμάτων, δίνοντας μια οπτική αντίληψη της σχετικής παρουσίας κάθε κόμματος στις συζητήσεις πολιτικού περιεχομένου. Παρακάτω ακολουθούν τα διαγράμματα :




Percentage Distribution of Sentiments
Percentage
Sentiment
NEUTRAL
NEGATIVE
POSITIVE
0
5
10
15
20
25
30
35


Party Sentiment Distribution
Sentiment
NEGATIVE
NEUTRAL
POSITIVE
Percentage
Party
DIEM
ELL_LYSI
KKE
ND
PASOK
SYRIZA
0
5
10
15
20
25
30
35
40

Αρχικά παρατηρούμε ότι, όσον αφορά την κατανομή των κομμάτων στα δεδομένα μας, το κόμμα της ΣΥΡΙΖΑ κατέχει την πρώτη θέση με ποσοστό 39.6%, ακολουθούμενο από τη Νέα Δημοκρατία με 31.6%. Αυτή η συγκέντρωση στα δύο κύρια κόμματα ίσως ενδείκνυται ότι υπάρχει μια πιθανή προκατάληψη στα δεδομένα, καθώς τα μικρότερα κόμματα εκπροσωπούνται λιγότερο συχνά.

Έπειτα, παρατηρούμε ότι η κατανομή των συναισθημάτων είναι ομοιόμορφη, με κάθε κατηγορία να κατέχει περίπου το 33.3%. Αυτή η ισορροπία μπορεί να υποδηλώνει μια σχετική αντικειμενικότητα στη συζήτηση, όπου διάφορες απόψεις και στάσεις βρίσκουν έκφραση με (σχετικά) ίση συχνότητα.

Τέλος, από την ανάλυση των συναισθημάτων ανά κόμμα, παρατηρούμε διακυμάνσεις στην εκπροσώπηση των θετικών, αρνητικών και ουδέτερων απόψεων. Για παράδειγμα, το ELL_LYSI εμφανίζει το υψηλότερο ποσοστό αρνητικών συναισθημάτων και το χαμηλότερο θετικό, ενώ το ΚΚΕ έχει την υψηλότερη ουδέτερη στάση και την υψηλότερη θετική αξιολόγηση. Το ΣΥΡΙΖΑ και το ΠΑΣΟΚ διατηρούν πιο ισορροπημένες αναλογίες μεταξύ των τριών κατηγοριών. Αυτές οι διαφορές υποδεικνύουν πώς τα κόμματα ενδέχεται να εκλαμβάνονται διαφορετικά από το κοινό, αντικατοπτρίζοντας διάφορες δυναμικές και τις αντιλήψεις για καθένα από αυτά.

Προ-επεξεργασία Δεδομένων

Για τον καθαρισμό των δεδομένων πριν την εκπαίδευση του μοντέλου υλοποιήθηκαν οι παρακάτω τεχνικές :

1.) Tokenization
2.) Αφαίρεση αναφορών χρηστών (user mentions)
3.) Αφαίρεση συνδέσμων (URLs)
4.) Αφαίρεση τόνων
5.) Αφαίρεση ειδικών χαρακτήρων (τελειών, κομμάτων, κ.λπ.)
6.) Αφαίρεση προ-επιλεγμένων stopwords
7.) Μετατροπή των λέξεων σε uppercase
8.) **Custom** Lemmatization
9.) Αφαίρεση spaces εντός ή στα άκρα ενός token
10.) Αφαίρεση λέξεων μικρού μήκους

Το data cleaning έγινε με σκοπό την αφαίρεση στοιχείων που πιθανώς να μην φέρουν ιδιαίτερο συναισθηματικό νόημα, καθώς και για τη μεγαλύτερη δυνατή μείωση διάστασης του dataset ως προς το μέγεθος του vocabulary. Για τα stopwords χρησιμοποιήθηκε ένα custom set, το οποίο προέκυψε από την εύρεση κοινά χρησιμοποιουμένων λέξεων εντός του training set. Στη φάση του lemmatization, αφαιρέθηκαν τα γράμματα 'Σ' και 'Ν' από τις λέξεις με τέτοια κατάληξη, για περεταίρω pruning του vocabulary. Το παραπάνω data cleaning είναι όμοιο με της πρώτης άσκησης, με σκοπό την μετέπειτα, όσο το δυνατόν **δικαιότερη** σύγκριση των αποτελεσμάτων του τρέχοντος και του προηγουμένου μοντέλου ως προς τις μετρικές απόδοσης.

Word Embeddings

Μετά τον καθαρισμό του dataset, για την παραγωγή των word embeddings χρησιμοποιήθηκε το μοντέλο Word2Vec. Οι ελεύθερες μεταβλητές που επιλέχθηκαν για το μοντέλο ήταν το dimensionality των word embeddings, το μέγεθος του context window, η ελάχιστη επιτρεπτή συχνότητα λέξεων, καθώς και η επιλογή μεταξύ των μεθόδων CBOW και Skip-Gram. Για την αξιολόγηση των αποτελεσμάτων, χρησιμοποιήθηκε μια **custom** μετρική η οποία, δεδομένων των 4 δημοφιλέστερων λέξεων του corpus και των 8 ομοιότερων (ως προς το cosine similarity) λέξεων τους, προσμετράει τη σημασιολογική ομοιότητα (semantic similarity) μεταξύ τους ως :

$$score(experiment_k) = \sum_{i=1}^{4}\sum_{j=1}^{8} ss(word_i, most\_similar(word_i, j))$$

Παρακάτω ακολουθούν τα αποτέλεσμα των σκορ των word embeddings :

Vector Dimension	Context Window	Minimum Count	Method	Score
50	1	1	CBOW	9
50	2	2	CBOW	11
75	1	4	CBOW	19
75	2	8	CBOW	11
100	4	16	CBOW	16
50	1	1	Skip-Gram	12
50	2	2	Skip-Gram	12
75	1	4	Skip-Gram	14
75	2	8	Skip-Gram	7
100	4	16	Skip-Gram	6

Παρατηρούμε ότι το καλύτερο αποτέλεσμα το επέφερε η χρήση της μεθόδου CBOW με διάσταση διανύσματος 75, context window 1 και ελάχιστη συχνότητα 4, αποδίδοντας βαθμολογία 19. Αυτός ο συνδυασμός ήταν ιδιαίτερα αποτελεσματικός διότι η αυξημένη διάσταση διανύσματος 75 παρείχε μια πιο λεπτομερή αναπαράσταση των λέξεων, ενώ το στενό context window 1 εστίασε στις άμεσες συναφείς λέξεις. Η ελάχιστη συχνότητα 4 επέτρεψε την εκμάθηση βαθύτερων συνδέσεων μεταξύ λιγότερο συχνών αλλά σημαντικών λέξεων, αποφεύγοντας τον θόρυβο που δημιουργείται από τις πολύ σπάνιες λέξεις για τις οποίες δεν υπάρχει αρκετό context.

Feature Extraction

Για την αναπαράσταση των tweets ως διανύσματα, χρησιμοποιήθηκαν οι τεχνικές των sum και average aggregation, όπου στην πρώτη περίπτωση, ένα tweet ισούται με το άθροισμα των word embeddings που το αποτελούν, ενώ στη δεύτερη με το μέσο όρο αυτών. Η επιλογή του sum aggregation έγινε με τη λογική ότι τα tweets με μεγαλύτερο μήκος, περιέχουν περισσότερες λέξεις και συνεπώς περισσότερη πληροφορία, άρα πρέπει να έχουν και μεγαλύτερο βάρος / επιρροή στο μοντέλο. Αντίθετα, η χρήση της μεθόδου της μέσης τιμής βασίζεται στην εξισορρόπηση της επιρροής του μήκους των tweets, διασφαλίζοντας ότι μικρότερα tweets δεν υποβαθμίζονται.

Επιλογή Μοντέλου

Για την ανάπτυξη του μοντέλου, δόθηκε έμφαση στην επιλογή παραμέτρων που καθορίζουν την ακρίβειά του. Αυτές οι παράμετροι περιλαμβάνουν τον αριθμό των επιπέδων στο νευρωνικό δίκτυο, την ποσότητα των νευρώνων που απαρτίζει κάθε επίπεδο, το ποσοστό απόρριψης (dropout) που εφαρμόζεται σε κάθε επίπεδο για την αποφυγή του φαινομένου υπερπροσαρμογής, καθώς και την ειδική συνάρτηση ενεργοποίησης που χρησιμοποιείται σε κάθε επίπεδο, η οποία βοηθά στην καλύτερη εκπαίδευση του μοντέλου.

Εκπαίδευση Μοντέλου

Για την εκπαίδευση του μοντέλου , καθορίστηκαν διάφορες παράμετροι που επηρέασαν το τελικό αποτέλεσμα. Αυτές είναι ο αριθμός των epochs, η επιλογή συνάρτησης απώλειας, η ένταξη early stopping και elastic-net regularization για την αποφυγή υπερπροσαρμογής, ο τύπος του optimizer καθώς και ο τύπος scheduler με στόχο την αποφυγή της παγίδευσης της συνάρτησης απώλειας σε ένα περιορισμένο εύρος τιμών.

Στρατηγική Επιλογής Παραμέτρων

Για τον ορισμό ενός μέτρου σύγκρισης, αρχικά ορίστηκε ένα απλό νευρωνικό δίκτυο με μικρό αριθμό επιπέδων (layers) και νευρώνων (neurons). Έπειτα, το μοντέλο γίνεται προοδευτικά πολυπλοκότερο, αλλάζοντας σε κάθε πείραμα την τιμή μόνο μιας μεταβλητής, έτσι ώστε, σε περίπτωση αλλαγής των μετρικών, να γνωρίζουμε σε ποια μεταβλητή οφείλεται αυτή. Αν η αλλαγή αυτή επιφέρει καλυτέρα αποτελέσματα, τότε η τιμή της διατηρείται, αλλιώς γίνεται roll back στην προηγούμενη τιμή και επιλέγεται κάποια άλλη τιμή για το επόμενο πείραμα. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να εξαντληθούν όλες οι μεταβλητές που επηρεάζουν την απόδοση του μοντέλου.

Αποτελέσματα Πειραμάτων

Παρακάτω ακολουθούν τα scores του μοντέλου ως προς τις μετρικές Precision, Recall, F-1, και Accuracy. Το καλύτερο μοντέλο προέκυψε ως συνδυασμός των μετρικών αυτών **και** δεδομένων των learning curves (οι οποίες δεν παρουσιάζονται για κάθε πείραμα, για εξοικονόμηση χώρου) έτσι ώστε το μοντέλο να εμφανίζει όσο το λιγότερο δυνατό underfit ή overfit.

Layers	Neurons	Optimizer	Aggregation	Normalization	Batch	Reg. Strength	Precision	Recall	F-1	Accuracy
2	32,3	ASGD	Sum	STD	64	0.0	0.397	0.394	0.387	0.394
3	64,32,3	ASGD	Sum	STD	64	0.0	0.405	0.404	0.403	0.404
3	128,64.3	ASGD	Sum	STD	64	0.0	0.407	0.405	0.401	0.405
3	256,128,3	ASGD	Sum	STD	64	0.0	0.401	0.400	0.399	0.400
4	256,128,64,3	ASGD	Sum	STD	64	0.0	0.405	0.405	0.404	0.405
4	256,128,64,3	SGD	Sum	STD	64	0.0	0.408	0.407	0.403	0.407
4	256,128,64,3	Adam	Sum	STD	64	0.0	0.236	0.354	0.281	0.354
4	256,128,64,3	SGD	Sum	STD	128	0.0	0.400	0.399	0.398	0.399
4	**256,128,64,3**	**SGD**	**Sum**	**STD**	**32**	**0.0**	**0.409**	**0.408**	**0.403**	**0.408**
4	256,128,64,3	SGD	Average	STD	32	0.0	0.403	0.403	0.401	0.403
4	256,128,64,3	SGD	Average	MIN-MAX	32	0.0	0.384	0.384	0.383	0.384
4	256,128,64,3	SGD	Average	None	32	0.0	0.397	0.396	0.389	0.396
4	256,128,64,3	SGD	Average	None	32	0.1	0.376	0.377	0.376	0.377
4	256,128,64,3	SGD	Average	None	32	1	0.376	0.375	0.364	0.375
4	256,128,64,3	SGD	Average	None	32	0.01	0.376	0.377	0.374	0.377

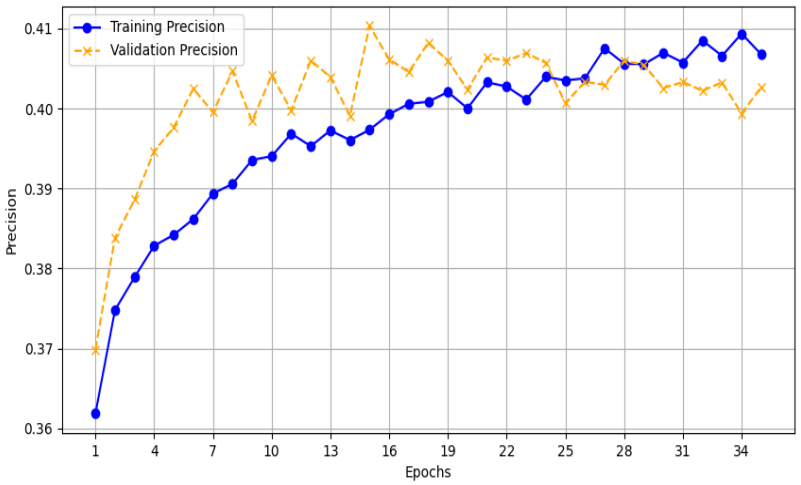
Fine Tuning

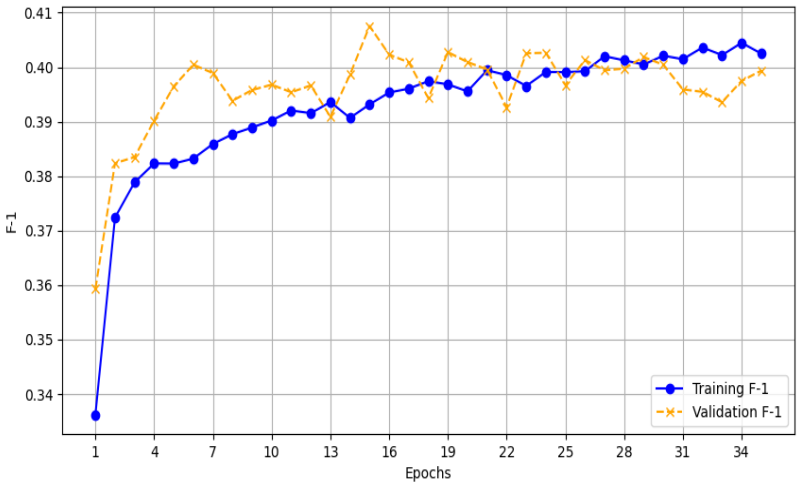
Για τη βελτιστοποίηση του καλύτερου, **βασικού** μοντέλου (το οποίο φαίνεται με bold στον παραπάνω πίνακα), χρησιμοποιήθηκε ο scheduler, με τη λογική ότι η προοδευτική μείωση του learning rate βοηθά το μοντέλο να εντοπίσει το τοπικό ελάχιστο της συνάρτησης απώλειας, αποτρέποντας το από το να "αναπηδά" στις υψηλότερες τιμές γύρω από αυτό το ελάχιστο, όπως και με όμοιο σκεπτικό για τις μετρικές απόδοσης. Παρακάτω, παρουσιάζονται τα αποτελέσματα :

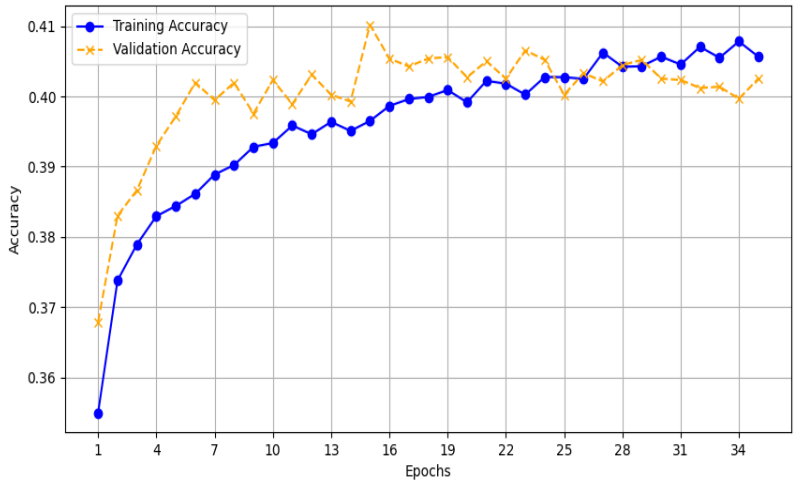
Scheduler	Gamma	Step	Precision	Recall	F-1	Accuracy
ExponentialLR	0.8	-	0.401	0.400	0.399	0.400
StepLR	0.4	5	0.403	0.402	0.400	0.402
StepLR	0.4	10	0.406	0.406	0.404	0.406
StepLR	**0.8**	**5**	**0.410**	**0.410**	**0.408**	**0.410**
StepLR	0.8	10	0.408	0.407	0.404	0.407

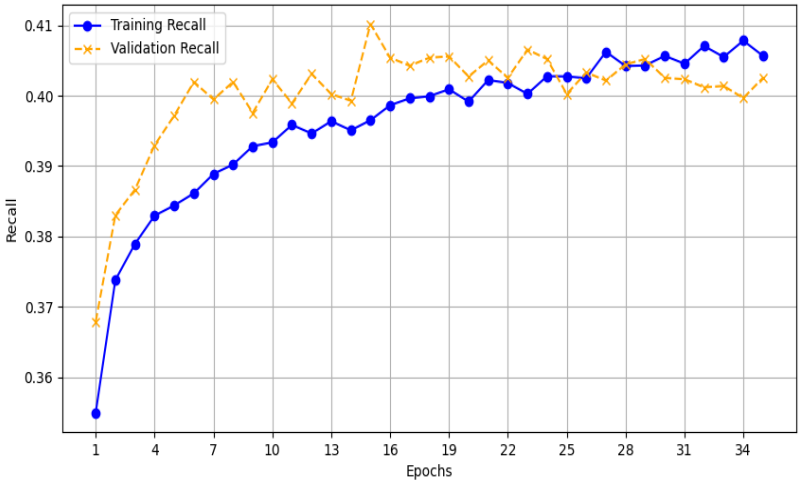
Learning Curves

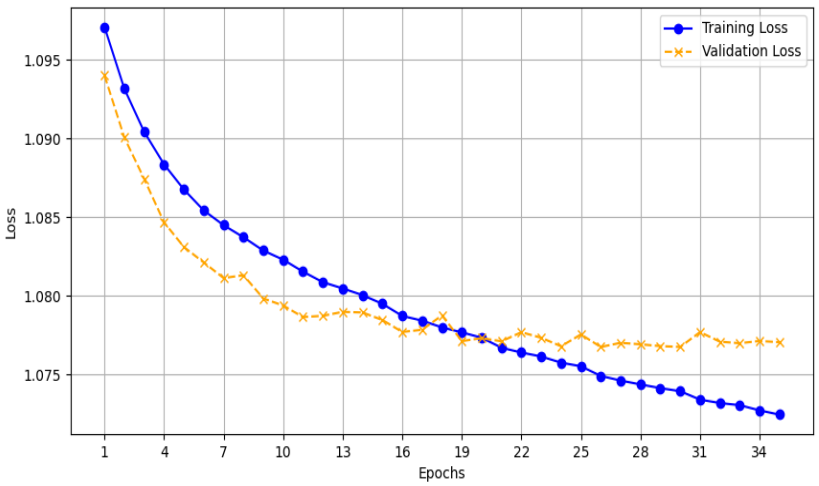
Παρακάτω παρατίθενται τα learning curves του καλύτερου μοντέλου που προέκυψε από το fine – tuning, στις μετρικές precision , accuracy, recall, f-1 και loss, καθώς και οι τιμές του καλύτερου instance του καλύτερου μοντέλου ως προς αυτές:

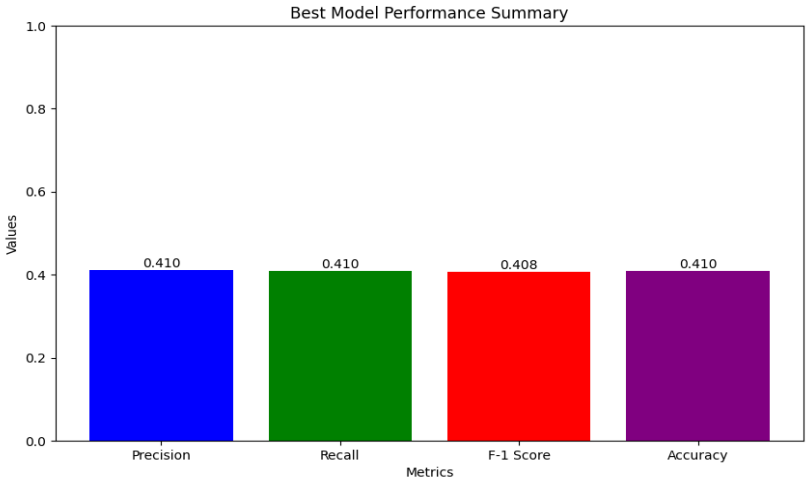





Training Accuracy
Validation Accuracy
Accuracy
Epochs


Training Recall
Validation Recall
Recall
Epochs


Training Loss
Validation Loss
Loss
Epochs

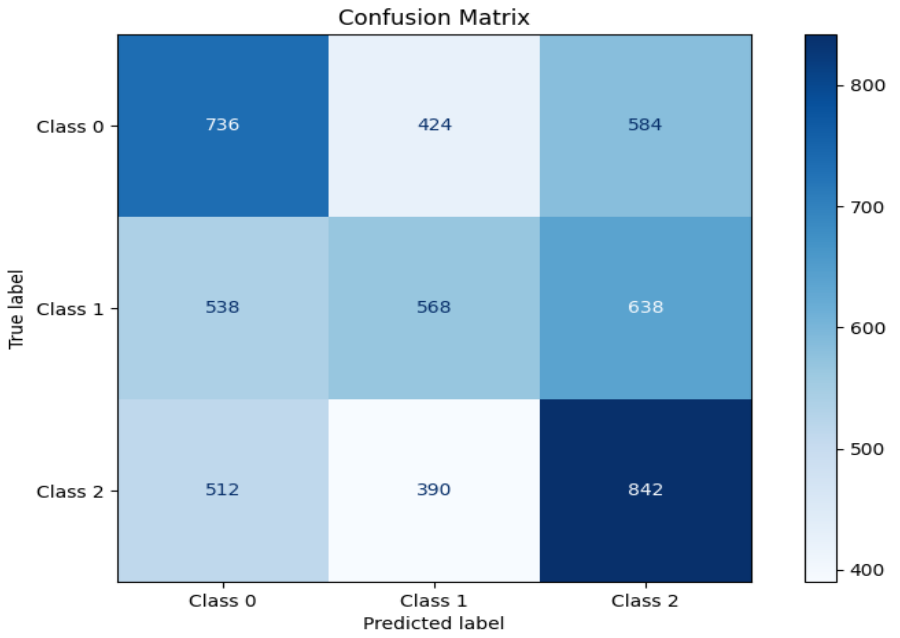


Εφόσον όλες οι καμπύλες (εκτός του loss), που σχετίζονται με το training set, είναι αύξουσες, το μοντέλο φαίνεται να μαθαίνει από τα δεδομένα, με τα scores να αυξάνουν επ' αόριστον, όσο αυξάνεται ο αριθμός των epochs.

Από τις καμπύλες του validation set, συμπεραίνουμε ότι το μοντέλο επίσης μαθαίνει να γενικεύει, αφού είναι αύξουσες, μέχρι ένα ολικό μέγιστο στην περιοχή του 0.40 , απ' το οποίο και έπειτα, αρχίζει να κάνει "bounce" στο εύρος [0.395,0.405].

Τέλος, στα πρώτα epochs το μοντέλο φαίνεται να κάνει underfit, αφού τα scores του validation set είναι υψηλοτέρα, το οποίο πρόβλημα όμως λύνεται αργότερα, αφού απ' το epoch 25 και έπειτα, το μοντέλο αρχίζει να σκοράρει συστηματικά υψηλοτέρα στο training set απ' ότι στο validation set.

Confusion Matrix



Από το confusion matrix παρατηρούμε ότι :

1.) Το μοντέλο έχει ακρίβεια περίπου 41%. Αυτό δείχνει ότι περίπου το 41% των προβλέψεων που έκανε το μοντέλο για **όλες** τις κλάσεις είναι σωστές.

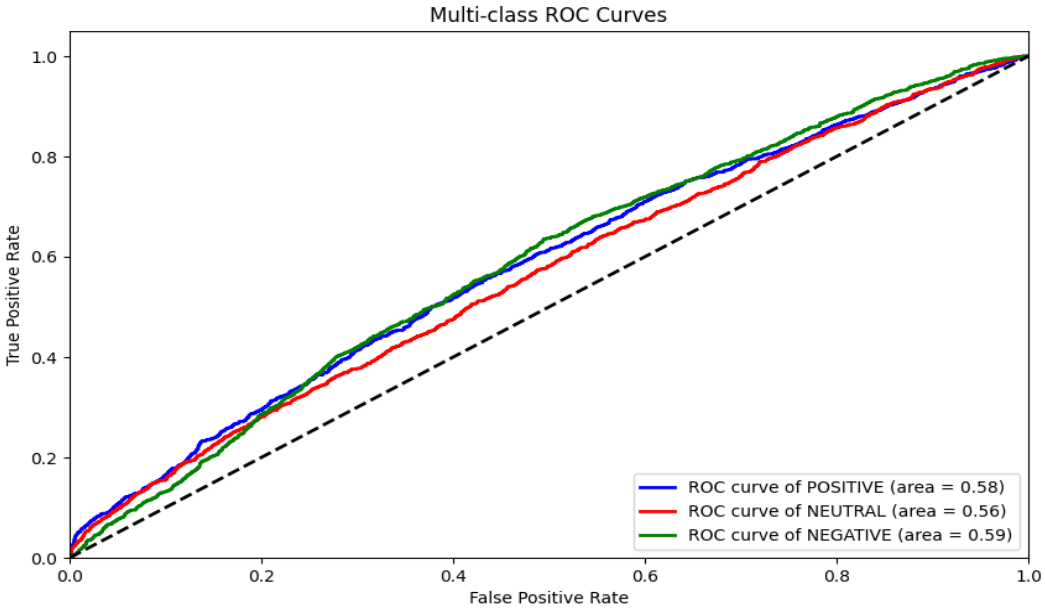
2.) Για το class 0 (Positive) , τα precision και recall είναι σχετικά ισορροπημένα (41.2% και 42.2% αντίστοιχα).

3.) Για το class 1 (Neutral), παρά το υψηλό precision (41.1%), το recall είναι χαμηλό (32.57%). Αυτό σημαίνει ότι αν και το μοντέλο είναι αρκετά ακριβές στο να **μην** κατατάσσει ένα non-neutral instance ως neutral, ταυτόχρονα κατατάσσει επίσης πολλά neutral instances ως non-neutral.

4.) Για το class 2 (Negative) , εμφανίζεται ο υψηλότερος δείκτης recall (48.2%), ωστόσο, παρατηρείται χαμηλό precision (40.7%). Αυτό σημαίνει ότι παρόλο που το μοντέλο εντοπίζει σχετικά αποτελεσματικά τις περιπτώσεις που ανήκουν πραγματικά στο class 2, ταυτόχρονα τείνει να κατατάσσει λανθασμένα και άλλες περιπτώσεις σε αυτό το class.

ROC curves

Αφού έχουμε τρία classes, χρησιμοποιήθηκαν τρία διαφορετικά ROC curves, με την τεχνική One vs Rest (OvR). Σε αυτήν, κάθε φορά ένα label ορίζεται ως το θετικό class και τα υπόλοιπα δύο συνθέτουν το αρνητικό class. Οι καμπύλες παρουσιάζονται παρακάτω:



Παρατηρούμε πως, οι τιμές AUC (**Area Under Curve**) του μοντέλου, που είναι 0.58, 0.56 και 0.59 για τις κλάσεις 0, 1 και 2 αντίστοιχα, και η τοποθέτηση των καμπυλών ROC πάνω από την κύρια διαγώνιο, υποδηλώνουν ότι το μοντέλο ξεπερνάει την απόδοση ενός τυχαίου μοντέλου στην ταξινόμηση των κλάσεων, στην οποία περίπτωση, τα ROC curves θα ταυτίζονταν με την κυρία διαγώνιο, ανεξαρτήτως τιμής threshold. Για όλα τα thresholds, το μοντέλο διακρίνει με μεγαλύτερη ακρίβεια τα instances της κλάσης 2 (Negative), ακολουθούμενα από τις κλάσεις 0 (Positive) και 1 (Neutral), δείχνοντας μια σχετική συνέπεια στην απόδοση μεταξύ των κλάσεων και ελαφρώς καλύτερη απόδοση στην αναγνώριση αρνητικών συναισθημάτων.

Σχολιασμός Τελικού Μοντέλου

Συγκρίνοντας τα μοντέλα των δύο ασκήσεων, παρατηρούμε ότι :

1.) Στην πρώτη άσκηση με την Λογιστική Παλινδρόμηση, οι μετρικές επίδοσης μειώνονταν κατά τη διάρκεια της εκπαίδευσης, δείχνοντας έλλειψη εκμάθησης. Αντιθέτως, στη δεύτερη άσκηση με το Νευρωνικό Δίκτυο, οι μετρικές βελτιώνονταν, υποδηλώνοντας αποτελεσματική (σχετικά) εκμάθηση.

2.) Στο πρώτο μοντέλο, η σημαντική απόκλιση ανάμεσα στις μετρικές πάνω στο training και validation set υποδείκνυαν υπερπροσαρμογή, αφού οι τιμές του training set ήταν αρκετά μεγαλύτερες αυτών του validation set. Αντίθετα, στο δεύτερο μοντέλο, οι τιμές κυμαίνονται περίπου στο ίδιο επίπεδο.

3.) Στην πρώτη άσκηση, τα ROC curves για τα positive και negative sentiments, ήταν κάτω από την κύρια διαγώνιο, το οποίο υποδείκνυε ότι το μοντέλο Λογιστικής Παλινδρόμησης απέδιδε χειρότερα από το αν τα sentiments επιλέγονταν τυχαία. Στη δεύτερη άσκηση, τα ROC curves του Νευρωνικό Δικτύου είναι πάνω από την κυρία διαγώνιο, επιδεικνύοντας καλύτερη διάκριση των κλάσεων σε σχέση με το προηγούμενο μοντέλο, ειδικά στην κατηγοριοποίηση των αρνητικών συναισθημάτων.

4.) Με τη χρήση νευρωνικό δικτύου, η γενική απόδοση του μοντέλου (δηλαδή ως προς τη μετρική f-1) αυξήθηκε κατά 6.5%.