



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Αναγνώριση Προτύπων

3^η εργαστηριακή άσκηση

Αναγνώριση Είδους και Εξαγωγή Συναισθήματος από Μουσική

Κωνσταντίνος Τσιγγέλης: 03117149

Χρήστος Σιαφαρίκας: 03117097

Σκοπός του εργαστηρίου

Σκοπός της άσκησης είναι η αναγνώριση του είδους και η εξαγωγή συναισθηματικών διαστάσεων από φασματογραφήματα (spectrograms) μουσικών κομματιών. Μας δίνονται 2 σύνολα δεδομένων, το Free Music Archive (FMA) genre με 3834 δείγματα χωρισμένα σε 20 κλάσεις (είδη μουσικής) και τη βάση δεδομένων (dataset) multitask music με 1497 δείγματα με επισημειώσεις (labels) για τις τιμές συναισθηματικών διαστάσεων όπως valence, energy και danceability. Τα δείγματα είναι φασματογραφήματα, τα οποία έχουν εξαχθεί από clips 30 δευτερολέπτων από διαφορετικά τραγούδια.

Εκτέλεση

Βήμα 0

Στο πρώτο βήμα της συγκεκριμένης εργαστηριακής άσκησης φορτώνουμε σε ένα kaggle_kernel το ζητούμενο dataset *Multitask affective music Classification2022*. Εκτελούμε την εντολή `os.listdir` για να εξερευνήσουμε σε μεγαλύτερο βάθος το περιεχόμενο του dataset αλλά και τους υποφακέλους του.



```
os.listdir("/kaggle/input/patreco3-multitask-affective-music/data")
```

```
[2]: ['multitask_dataset_beat',  
      'fma_genre_spectrograms',  
      'fma_genre_spectrograms_beat',  
      'multitask_dataset']
```

Βλέπουμε λοιπόν ότι υπάρχουν τέσσερα sub-datasets τα οποία αφορούν 2 μεγάλα σύνολα δεδομένων. Το πρώτο είναι το `fma_genre_spectrograms` το οποίο περιέχει 3834 δείγματα χωρισμένα σε 20 κλάσεις και τη βάση δεδομένων multitask_music με 1497 δείγματα με επισημειώσεις για τις τιμές συναισθηματικών διαστάσεων όπως valence, energy και danceability. Τα δείγματα είναι φασματογραφήματα, τα οποία έχουν εξαχθεί από clips 30 δευτερολέπτων από διαφορετικά τραγούδια.

Η επιπρόσθετη επισημείωση beat αναφέρεται στον συγχρονισμό των spectrograms με το beat του τραγουδιού ούτως ώστε να μειώσουμε την διάσταση των δεδομένων χωρίς όμως να χάσουμε σημαντική πληροφορία. Οι τιμές προκύπτουν ως η διάμεσος μεταξύ των beat.

Τέλος, καθένας από τους παραπάνω υποφακέλους περιλαμβάνει train και test δείγματα αλλά και txt.files για τα labels.



```
os.listdir("/kaggle/input/patreco3-multitask-affective-music/data/fma_genre_spectrograms")
```

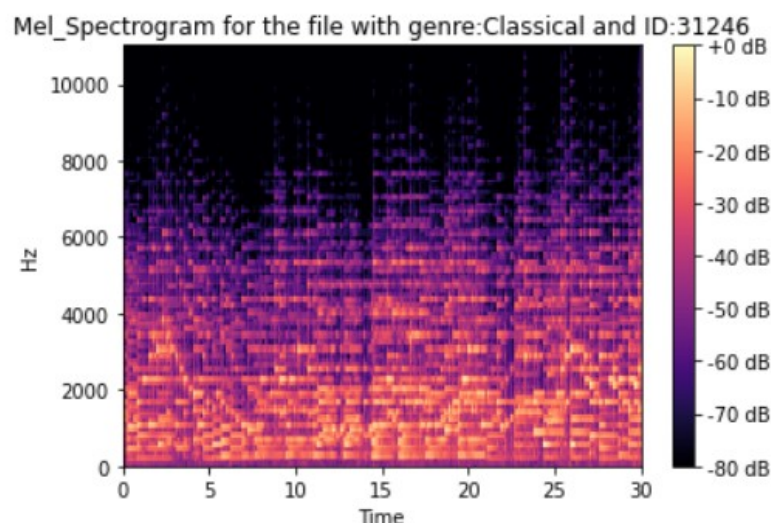
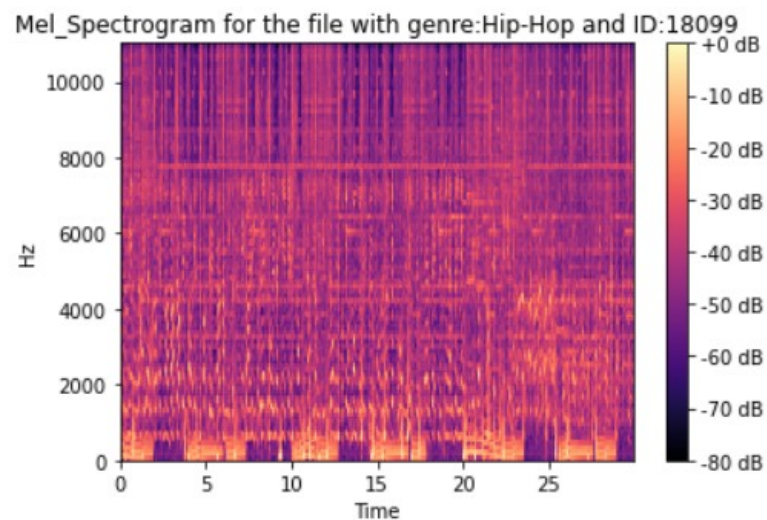
```
[3]: ['train_labels.txt', 'test', 'test_labels.txt', 'train']
```

Βήμα 1

Στο συγκεκριμένο βήμα διαβάζω 2 τυχαία δείγματα ,που ανήκουν σε διαφορετικές κατηγορίες, από το path ‘.../fma_genre_spectrograms/train/'. Η τυχαία επιλογή γίνεται με την εντολή choice της βιβλιοθήκης random και τα shape τους είναι τα ακόλουθα :

```
the shape of the fist sample is: (140, 1291)
the shape of the secondsample is: (140, 1292)
The shape is (mel + chroma frequencies, timesteps)
```

Στην συνέχεια απεικονίζω τα φασματογραφήματά τους σε κλίμακα mel , τα οποία περιγράφονται από τις πρώτες 128 γραμμές του πίνακα κάθε δείγματος.



- Το φασματογράφημα είναι μία αναπαράσταση των συχνοτήτων του σήματος συναρτήσει του χρόνου και η εξαγόμενη εικόνα αναπαριστά την ενέργεια του σήματος για διαφορετικές ζώνες συχνοτήτων και χρονικά

παράθυρα.

- Όσον αφορά τα φασματογραφήματα που απεικονίσαμε εύκολα παρατηρούμε ότι στην κλασική μουσική η ενέργεια σε υψηλές συχνότητες άνω των 7kHz είναι μηδενική . Αντίθετα η hip-hop μουσική παρουσιάζει υψηλή ενέργεια σε όλο το φασματικό εύρος και ταυτόχρονα οι συχνотικές μεταβολές κατά μήκος του χρόνου φαίνονται να είναι πιο αργές και ομαλές. Τέλος ιδιαίτερα στην hip-hop μουσική βλέπουμε σε χαμηλές συχνότητες μια μεταβολή που προσομοιάζει σε μια περιοδική παλμοειδή συνάρτηση που προφανώς αντιπροσωπεύει το μπάσο/beat του τραγουδιού.

Βήμα 2

Παρουσιάζω πάλι τα shape των samples:

```
the shape of the first sample is: (140, 1291)
the shape of the second sample is: (140, 1292)
The shape is (mel + chroma frequencies, timesteps)
```

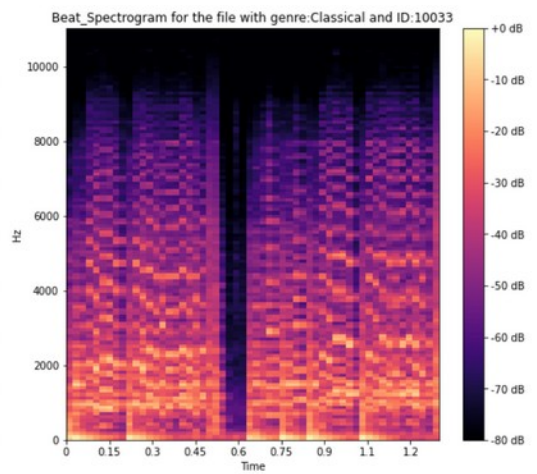
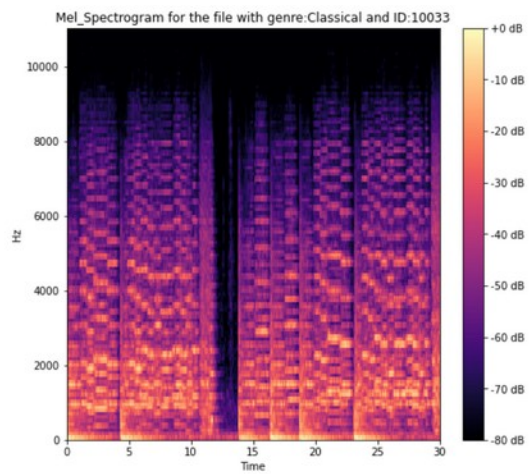
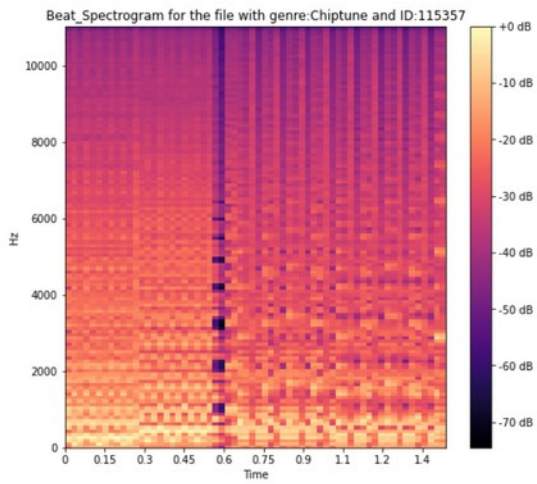
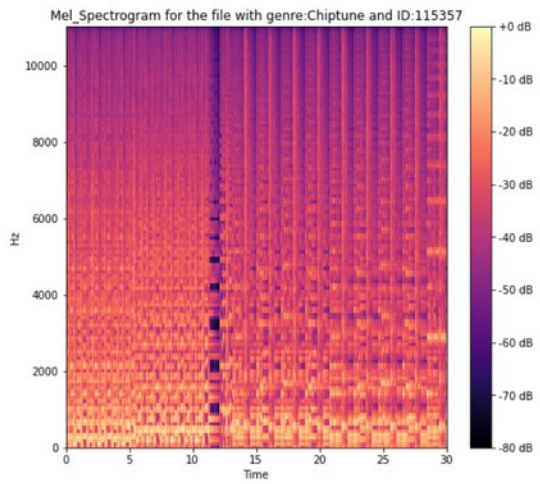
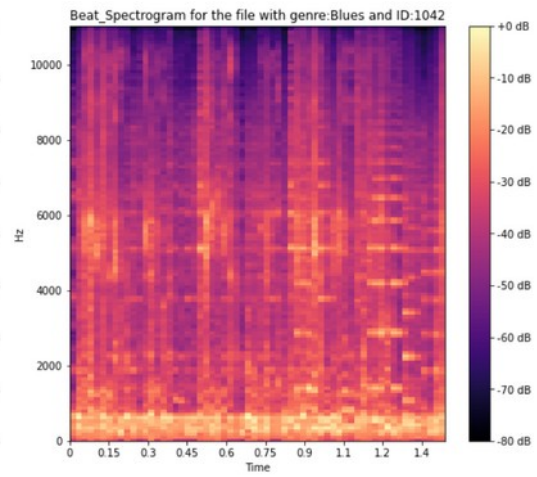
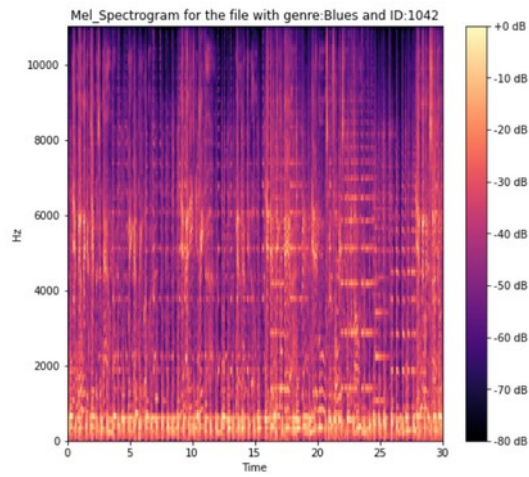
Παρατηρούμε ότι το πρώτο δείγμα αποτελείται από 1291 timesteps ενώ το δεύτερο από 1292 timesteps. Για τόσο μεγάλες χρονικές ακολουθίες υπάρχει πολύ μεγάλη καθυστέρηση στην εκτέλεση κάθε εποχής των recurrent networks (LSTM) Έτσι, δεν είναι τόσο αποδοτικό να φτιάχνουμε LSTM, γιατί έχουμε πάρα πολλά timesteps και άρα θα αργήσει πάρα πολύ η εκπαίδευσή του. Για αυτό τον λόγο χρησιμοποιούμε τα beat_synchronized samples.

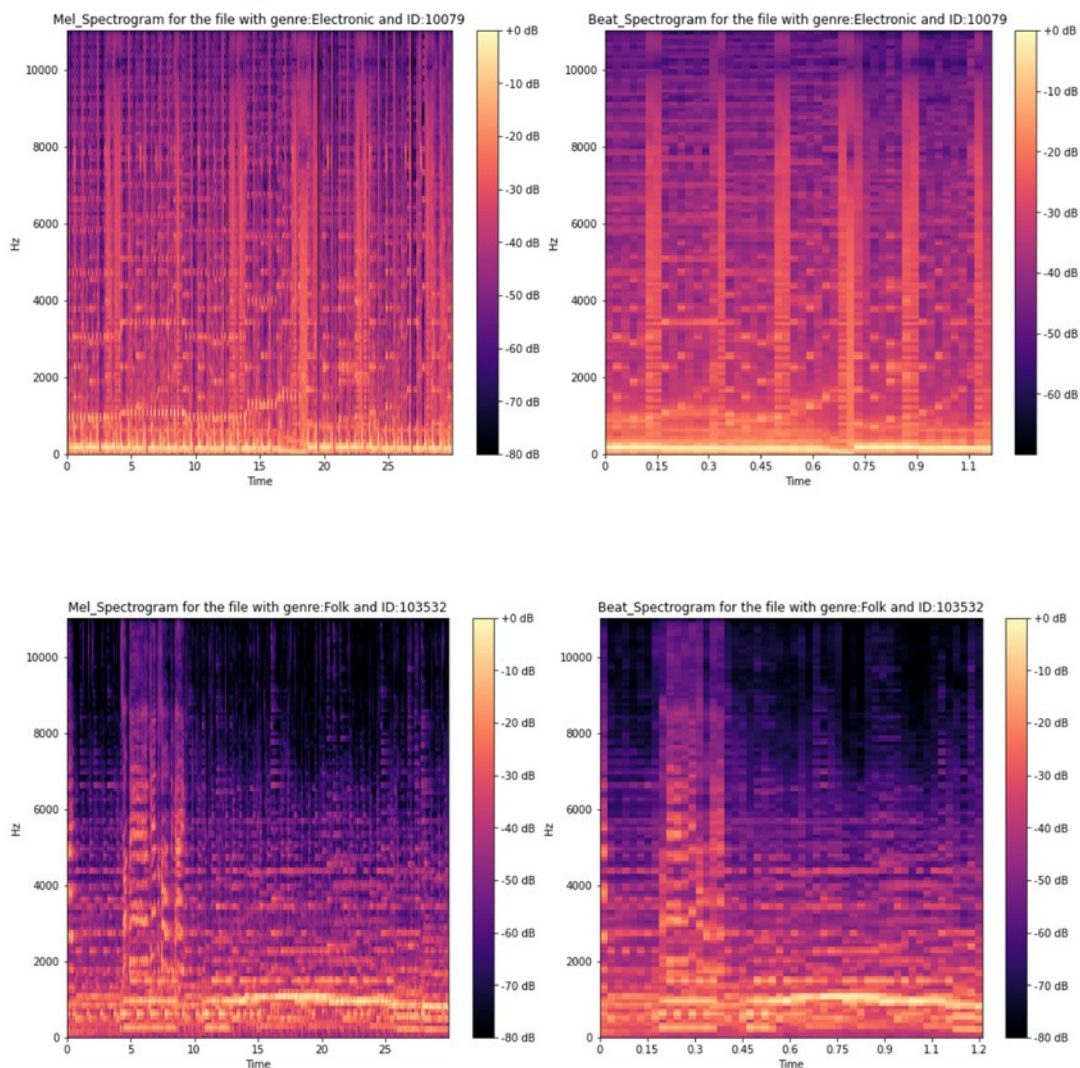
Εκτυπώνω τις διαστάσεις για τα αντίστοιχα beat_sync samples :

```
the shape of the first beat_sync sample is : (140, 44)
the shape of the second beat_sync sample is : (140, 77)
```

Πράγματι βλέπουμε σημαντική μείωση των διαστάσεων . Παρόλα αυτά η ελάττωση αυτή μας επιβαρύνει με απώλεια πληροφορίας?

Για να απαντήσουμε στην ερώτηση αυτή παραθέτουμε τα φασματογραφήματα τυχαίων δειγμάτων για κάθε κλάση τόσο για τα συνολικά όσο και για τα beat_sync αρχεία.





...Τα ζεύγη φασματογραφημάτων για τις υπόλοιπες κλάσεις φαίνονται στον κώδικα (ακολουθούν το ίδιο μοτίβο οπότε δεν προσφέρουν κάτι στην αναφορά).

Βλέπουμε λοιπόν πως εφόσον μειώσαμε κατα πολύ τα timesteps έχουμε περιορίσει την ακρίβεια καθώς και την ευκρίνεια των φασματογραφημάτων. Ωστόσο, ανάμεσα στα διαδοχικά beats έχουμε παρόμοιο συχνοτικό περιεχόμενο και άρα μια πυκνή απεικόνιση τους δεν προσφέρει κάποια σημαντική επιπλέον πληροφορία.

Ως επακόλουθο με αυτό τον τρόπο μπορούμε να εκπαιδεύσουμε τα αναδρομικά δίκτυα πολύ πιο γρήγορα και ενδεχομένως το ίδιο ή/και πιο αποτελεσματικά.

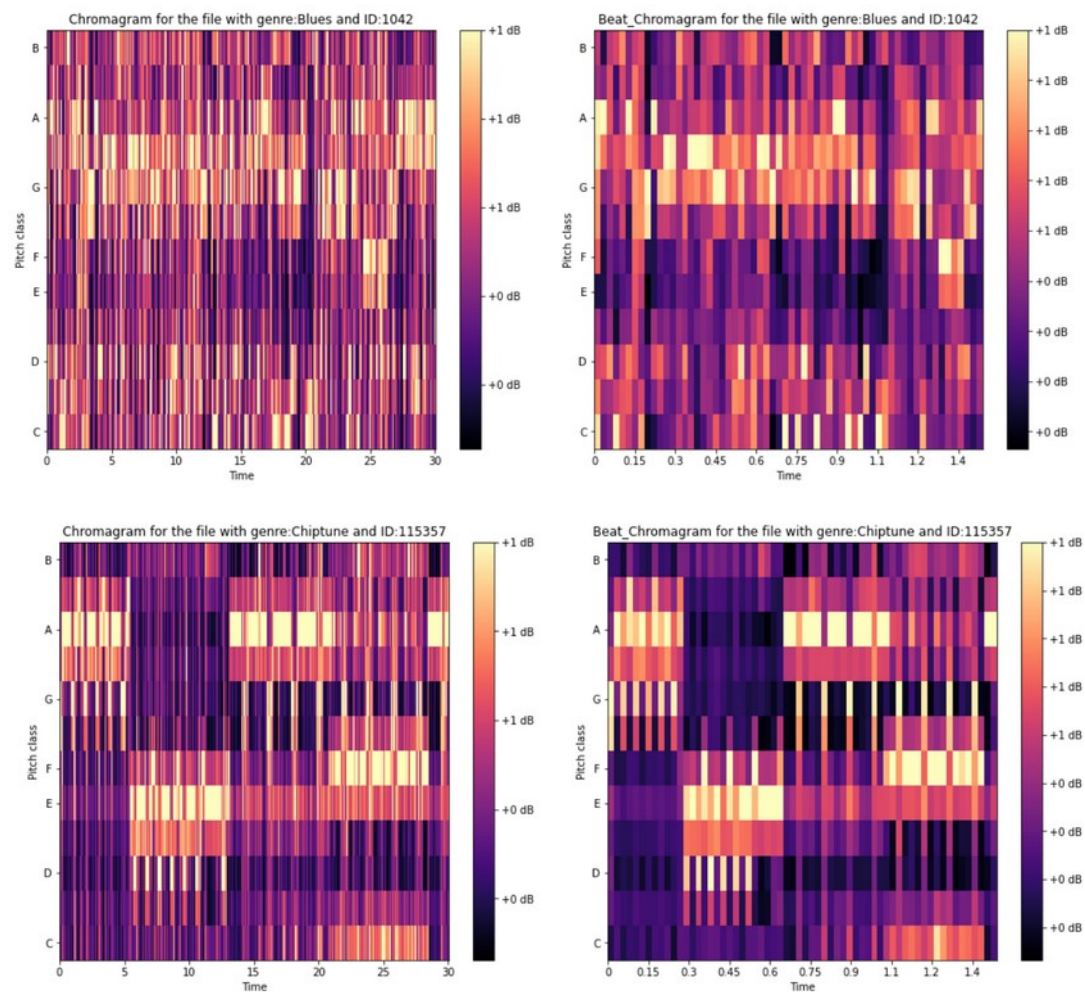
Βήμα 3

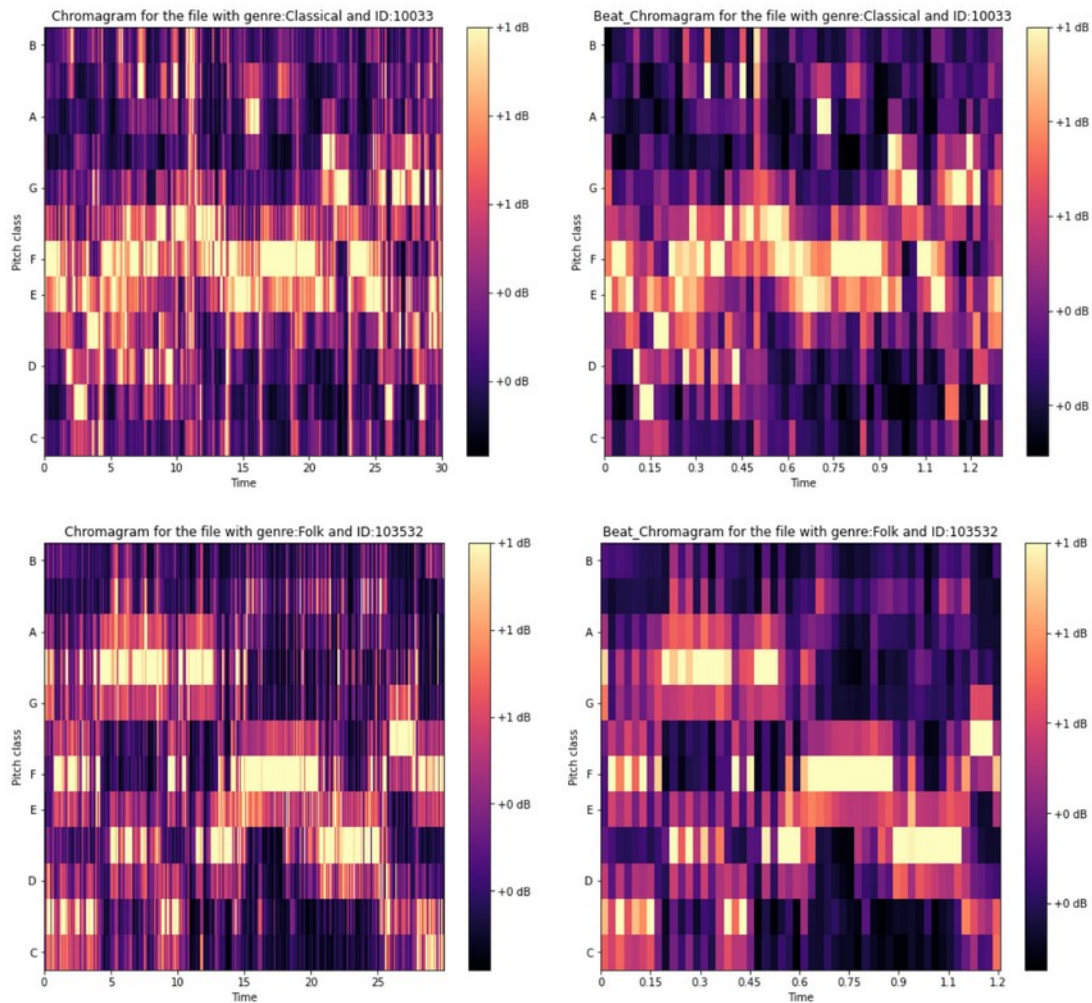
Όπως αναφέραμε προηγουμένως το shape των samples προκύπτουν ως (mel+chroma,timesteps). Εφόσον λοιπόν τα mel_frequencies αντιστοιχούν στις

πρωτες 128 γραμμές οι υπόλοιπες 12 αντιστοιχούν στις chroma frequencies.

Τα χρωμογράφημα (chromagrams) απεικονίζουν την ενέργεια του σήματος μουσικής για τις ζώνες συχνότητας που αντιστοιχούν στις δώδεκα διαφορετικές νότες της κλίμακας κλασικής μουσικής {C, C#, D, D#, E, F, F#, G, G#, A, A#, B} και μπορούν να χρησιμοποιηθούν ως εργαλείο για την ανάλυση της μουσικής αναφορικά με τα αρμονικά και μελωδικά χαρακτηριστικά της ενώ επίσης είναι αρκετά εύρωστα και στην αναγνώριση των αλλαγών του ηχοχρώματος και των οργάνων.

Επαναλαμβάνουμε λοιπόν τα προηγούμενα ζητούμενα για τα χρωματογράφημα:



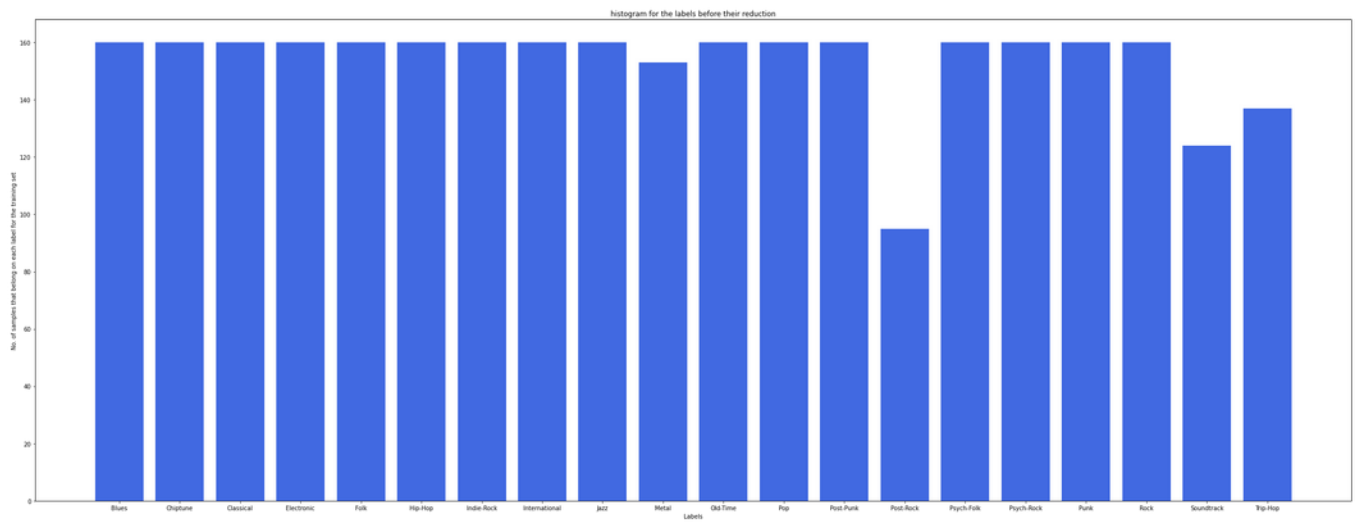
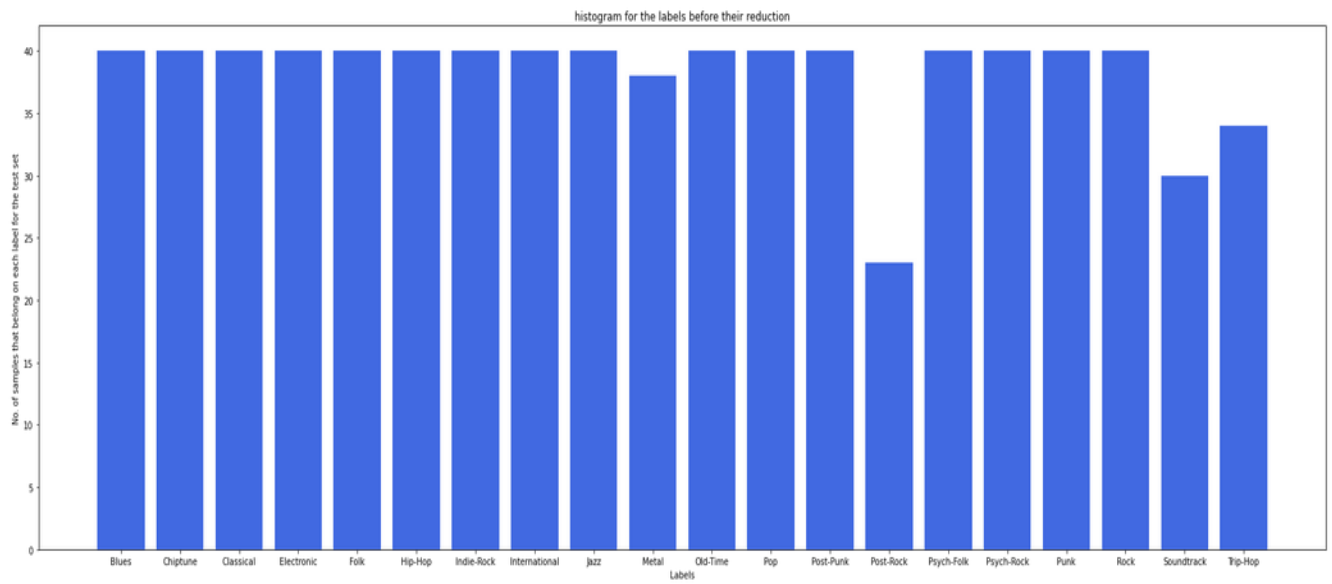


.....Τα χρωματογραφήματα για τις υπόλοιπες κλάσεις υπάρχουν στον κώδικα (ακολουθούν το ίδιο μοτίβο οπότε δεν προσφέρουν κάτι στην αναφορά).

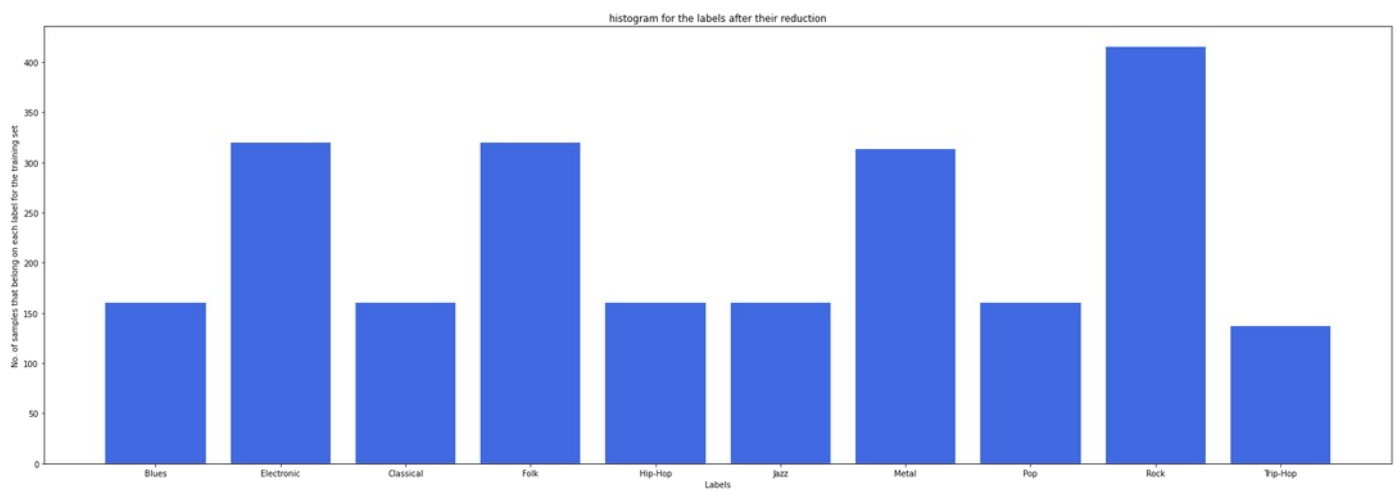
Είναι εμφανής πάλι η διάκριση μεταξύ των διαφορετικών ειδών μουσικής χρησιμοποιώντας την ανάλυση των χρωματογραφημάτων. Επίσης όσον αφορά την σύγκριση μεταξύ των beat_sync samples και των συνολικών καταλήγουμε σε παρόμοια συμπεράσματα με προηγουμένως , δηλαδή παρόλο που η ακρίβεια μειώνεται η γενική μορφή/πληροφορία παραμένει ίδια.

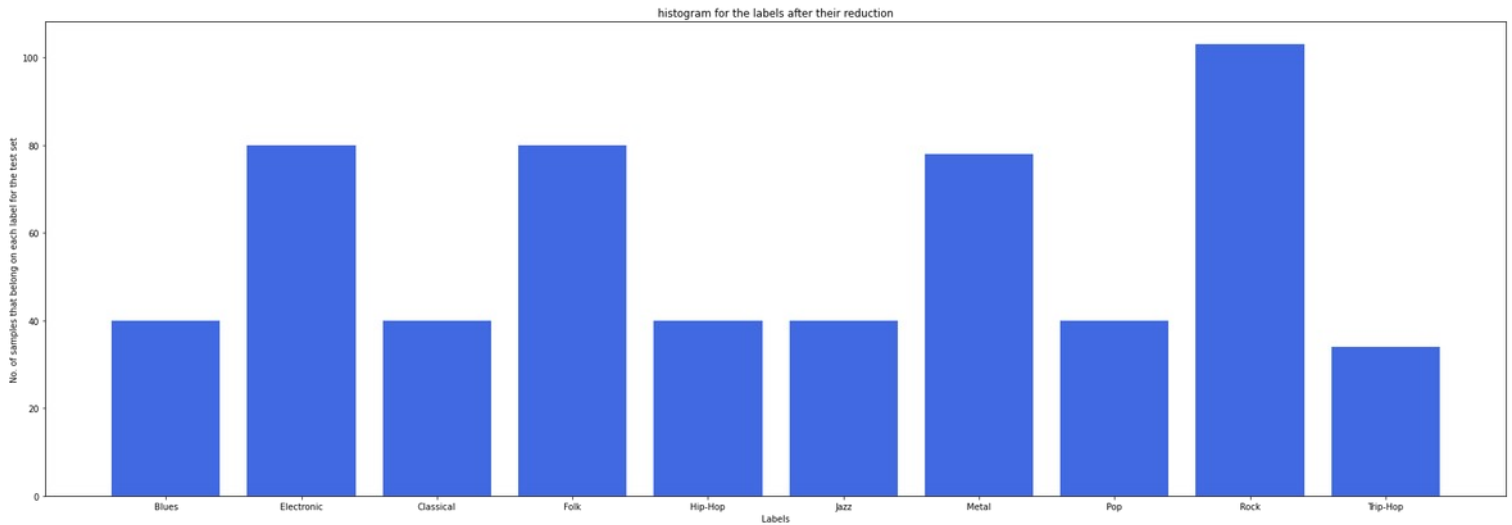
Βήμα 4

Before mapping



After mapping





Βλέπουμε πως με το mapping, το train και το test έχουν την ίδια κατανομή ως προς τις συχνότητες των αντίστοιχων, συγχωνευμένων και μη, labels.

Βήμα 5 και 6

Κάποια γενικά στοιχεία για τις μετρικές που θα χρησιμοποιήσουμε.

Accuracy: με το accuracy βλέπουμε το ποσοστό των δειγμάτων που ταξινομήθηκαν σε ορθή κλάση σε σχέση με το συνολικό πλήθος των δειγμάτων. Με την μετρική αυτή μπορούμε να εκτιμήσουμε σχετικά αποτελεσματικά την επίδοση του μοντέλου μας.

Precision : είναι τα ορθώς ταξινομημένα δείγματα σε μια κλάση A σε σχέση με το σύνολο των δειγμάτων που ταξινομήθηκαν σε αυτή την κλάση. Μας δίνει μια εικόνα για τα δείγματα που απορρίπτονται από το μοντέλο ως προς την κλάση αυτή.

Recall: είναι τα ορθώς ταξινομημένα δείγματα σε μια κλάση A σε σχέση με το σύνολο των δειγμάτων που ανήκουν κανονικά σε αυτή την κλάση. Μας δίνει μια εικόνα για τα δείγματα που το μοντέλο διακρίνει ότι ανήκουν στη κλάση αυτή.

F1-score: είναι ο σταθμισμένος αρμονικός μέσος όρος των precision και recall. Μας δίνει μια γενική αξιολόγηση του μοντέλου για μια συγκεκριμένη κλάση.

Macro: είναι ο μέσος όρος των αποτελεσμάτων αποφεύγοντας έτσι την πιθανότητα να έχουμε imbalanced dataset.

Micro: ενοποιεί τα σωστά ταξινομημένα στη κλάση A, τα λάθος ταξινομημένα στην κλάση αυτή και όσα θα έπρεπε να έχουν ταξινομηθεί στην κλάση αυτή αλλά δεν

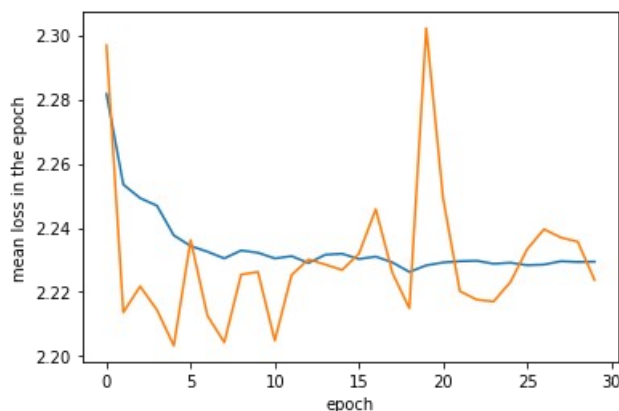
έγινε. Έτσι, λαμβάνουν υπόψη το class imbalance, και αυτή είναι και η πιο συχνή αιτία όταν αποκλίνουν τα micro-macro.

Προτιμούμε λοιπόν το accuracy και το micro. Η μεγάλη διαφορά ανάμεσα στα accuracy και F1 εμφανίζεται όταν το μοντέλο τείνει να συγκεντρώνει τα δείγματα σε μια ή δύο κλάσεις από τις πολλές που έχουμε, πράγμα που μπορεί να οφείλεται σε imbalanced dataset ή underfitting. Έτσι, το accuracy θα επηρεάζεται μόνο από το μέγεθος των samples αυτής της μεγάλης κλάσης και δεν θα λαμβάνει σε μεγάλο βαθμό υπόψη τις υπόλοιπες.

Τα precision και recall έχουν μεγάλη ισχύ σε περιπτώσεις που τα false positives/negatives έχουν σημαντικές συνέπειες στην μελέτη δηλαδή αν γίνει κάποιο λάθος θα έχουμε έως και fatal αποτελέσματα.

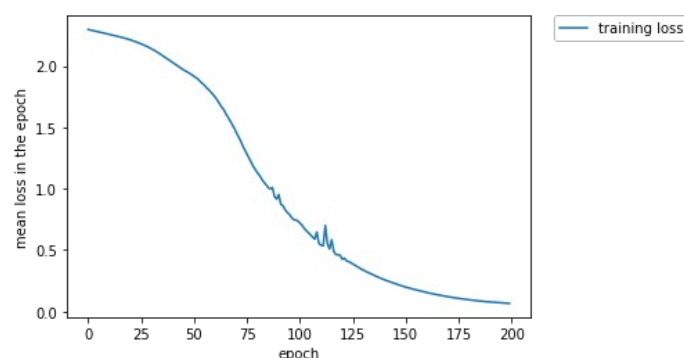
Για το ερώτημα αυτό χρησιμοποιήσαμε έναν LSTM όπως αυτός της πρώτης εργαστηριακής με ελάχιστες αλλαγές. Πλέον με τα δεδομένα που δέχεται για εισόδο να είναι τα φασματογραφήματα. Με την boolean μεταβλητή obover δημιουργούμε ένα "παράθυρο" στο train ώστε όταν είναι αληθής να γίνεται υπερεκπαίδευση του δικτύου. Τρέχουμε το LSTM για όλα οσα ζητούνται και τα αποτελέσματα είναι. Για τα ιδιαίτερα μεγάλα dataset χρησιμοποιήσαμε ένα subset για να επιταχύνουμε τη διαδικασία.

Φασματογραφήματα από το non_beat_synchronized dataset



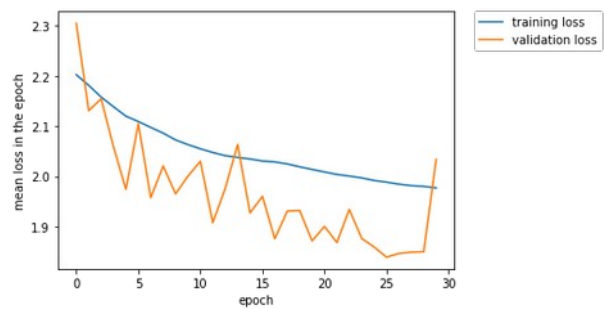
	precision	recall	f1-score	support
0	0.00	0.00	0.00	40
1	0.00	0.00	0.00	40
2	0.28	0.10	0.15	80
3	0.17	0.44	0.24	80
4	0.00	0.00	0.00	40
5	0.00	0.00	0.00	40
6	0.00	0.00	0.00	78
7	0.00	0.00	0.00	40
8	0.19	0.62	0.29	103
9	0.00	0.00	0.00	34
accuracy			0.19	575
macro avg	0.06	0.12	0.07	575
weighted avg	0.10	0.19	0.11	575

και για την υπερεκπαίδευση βλέπουμε το παρακάτω διάγραμμα :

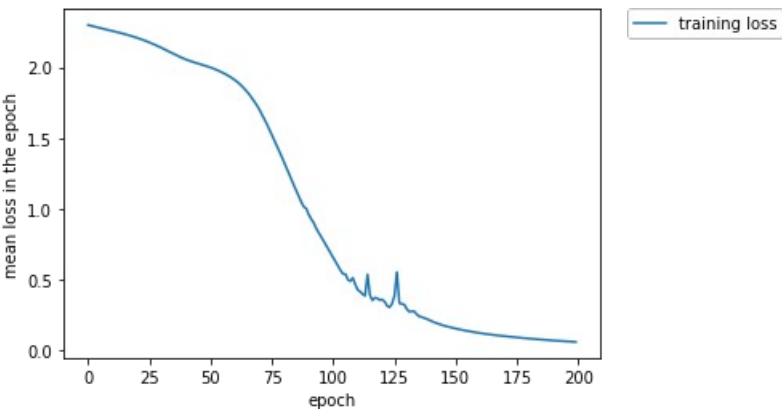


Η οποία μας δείχνει ότι μετά την πάροδο αρκετών εποχών το training loss πάει στο μηδέν.

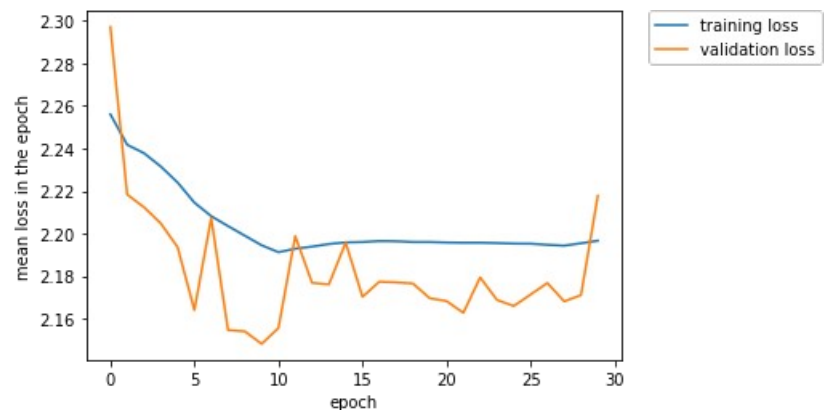
Φασματογραφήματα από το *beat_synchronized dataset*



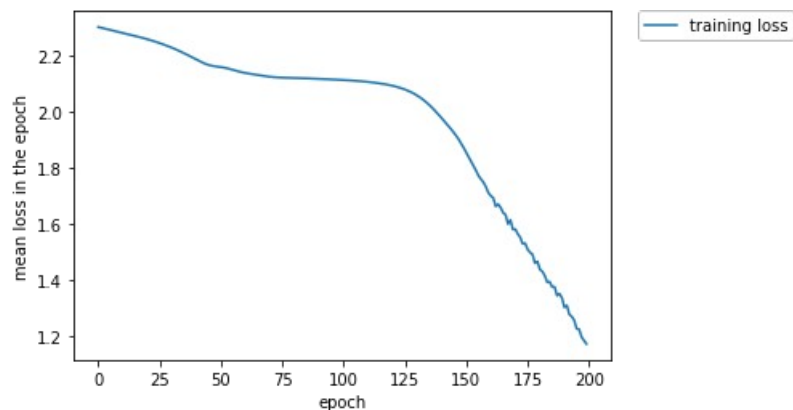
	precision	recall	f1-score	support
0	0.00	0.00	0.00	40
1	0.25	0.68	0.36	40
2	0.24	0.93	0.38	80
3	0.28	0.17	0.22	80
4	0.00	0.00	0.00	40
5	0.00	0.00	0.00	40
6	0.47	0.26	0.33	78
7	0.00	0.00	0.00	40
8	0.48	0.30	0.37	103
9	0.00	0.00	0.00	34
accuracy			0.29	575
macro avg	0.17	0.23	0.17	575
weighted avg	0.24	0.29	0.22	575



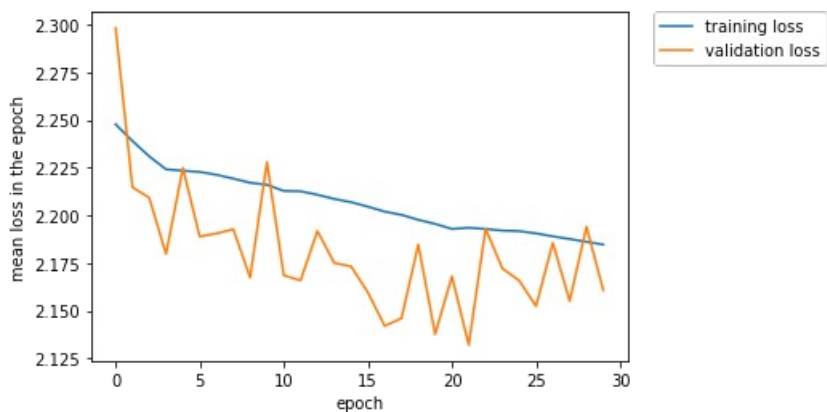
Χρωματογραφήματα από το *beat_synchronized dataset*



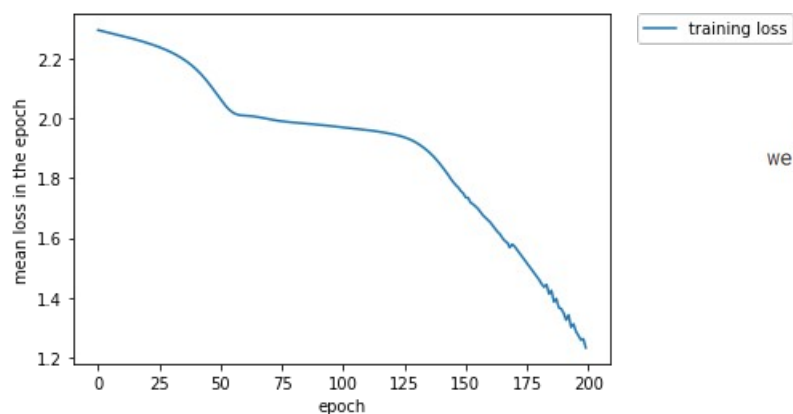
	precision	recall	f1-score	support
0	0.00	0.00	0.00	40
1	0.00	0.00	0.00	40
2	0.00	0.00	0.00	80
3	0.00	0.00	0.00	80
4	0.00	0.00	0.00	40
5	0.00	0.00	0.00	40
6	0.00	0.00	0.00	78
7	0.00	0.00	0.00	40
8	0.18	1.00	0.30	103
9	0.00	0.00	0.00	34
accuracy			0.18	575
macro avg	0.02	0.10	0.03	575
weighted avg	0.03	0.18	0.05	575



Χρωματογραφήματα από το non_beat_synchronized dataset

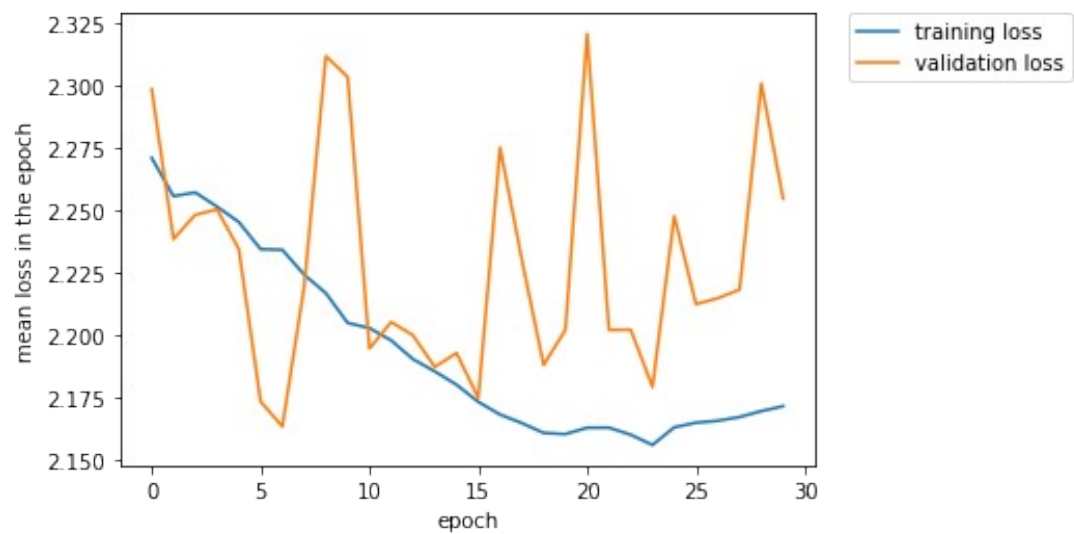


	precision	recall	f1-score	support
0	0.00	0.00	0.00	40
1	0.00	0.00	0.00	40
2	0.19	0.14	0.16	80
3	0.22	0.42	0.29	80
4	0.50	0.03	0.05	40
5	0.00	0.00	0.00	40
6	0.30	0.24	0.27	78
7	0.00	0.00	0.00	40
8	0.20	0.56	0.30	103
9	0.14	0.03	0.05	34

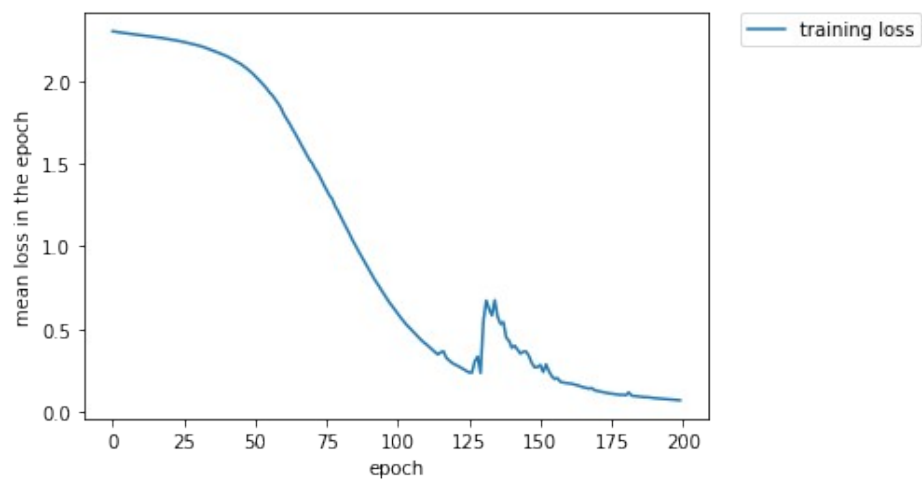


accuracy			0.22	575
macro avg	0.16	0.14	0.11	575
weighted avg	0.18	0.22	0.16	575

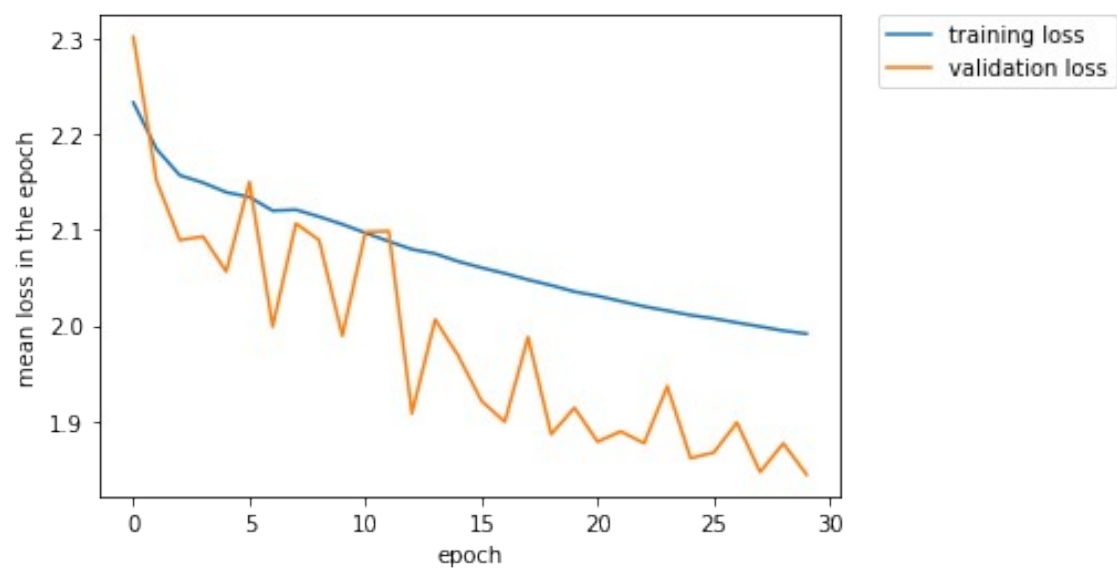
Ολόκληρο dataset (φασματογραφήματα+χρωματογραφήματα) από το non_beat_synchronized dataset



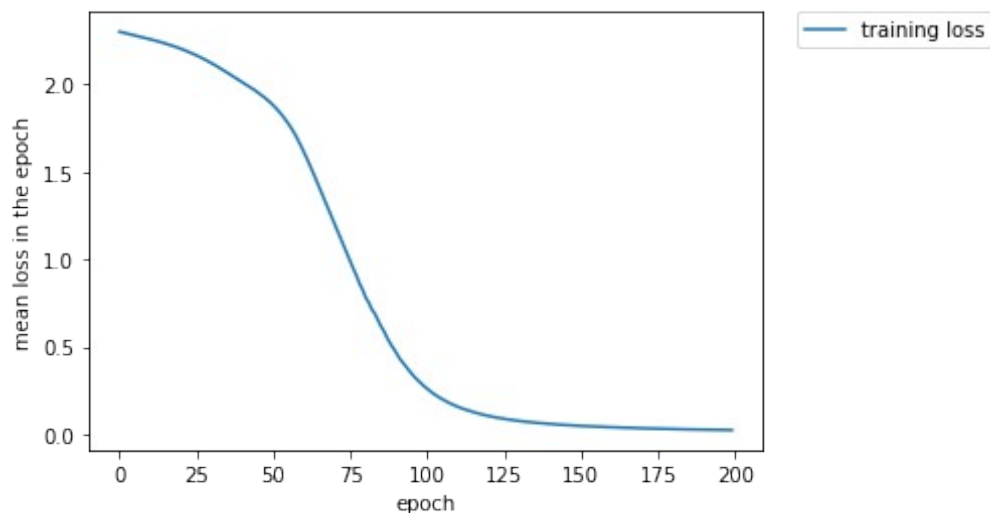
	precision	recall	f1-score	support
0	0.00	0.00	0.00	40
1	0.00	0.00	0.00	40
2	0.00	0.00	0.00	80
3	0.14	0.59	0.23	80
4	0.00	0.00	0.00	40
5	0.00	0.00	0.00	40
6	0.00	0.00	0.00	78
7	0.00	0.00	0.00	40
8	0.19	0.45	0.26	103
9	0.00	0.00	0.00	34
accuracy			0.16	575
macro avg	0.03	0.10	0.05	575
weighted avg	0.05	0.16	0.08	575



Ολόκληρο dataset (φασματογραφήματα+χρωματογραφήματα) από το *beat_synchronized dataset*



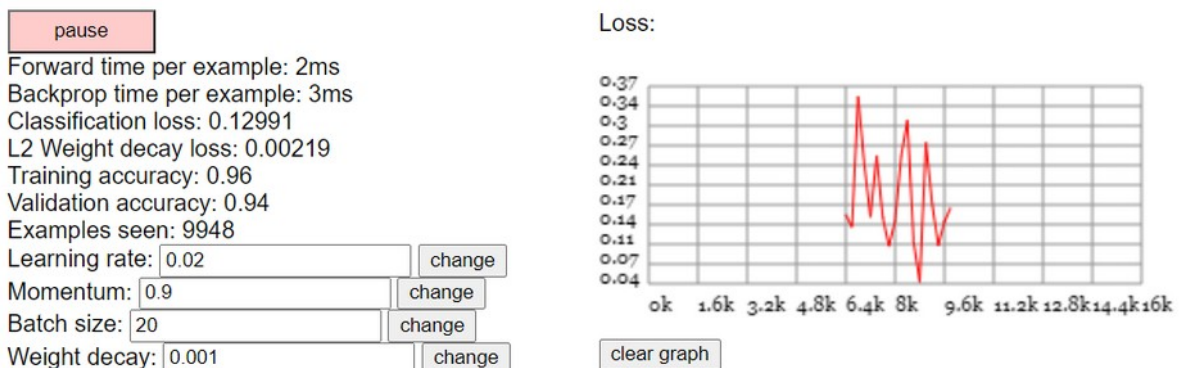
	precision	recall	f1-score	support
0	0.00	0.00	0.00	40
1	0.31	0.45	0.36	40
2	0.31	0.51	0.38	80
3	0.26	0.81	0.39	80
4	0.00	0.00	0.00	40
5	0.00	0.00	0.00	40
6	0.49	0.45	0.47	78
7	0.00	0.00	0.00	40
8	0.31	0.17	0.22	103
9	0.00	0.00	0.00	34
accuracy			0.31	575
macro avg	0.17	0.24	0.18	575
weighted avg	0.22	0.31	0.24	575



Βήμα 7

a)

Το προγραμμα αυτό του MIT μας δίνει τη δυνατότητα να εκπαιδεύσουμε ένα CNN με στόχο την κατηγοριοποίηση των χειρόγραφων νούμερων στα αντίστοιχα με όσο το δυνατόν καλύτερα αποτελέσματα.



Με αυτά τα δεδομένα δημιουργούμε το δίκτυο μας και ακολουθούν τα screenshot που περιέχουν τα διάφορα layers του νευρωνικού μας.

Input:

input (24x24x1)
 max activation: 1, min: 0
 max gradient: 0.0007, min: -0.00029

Activations:



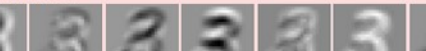
Activation Gradients:



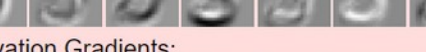
Εδώ έχουμε τη φωτογραφία της εισόδου όπως την βάζουμε στο σύστημα.

Convolution:


Activations:




Activation Gradients:



Weights:




Weight Gradients:




Relu και Pool:

Activations:



Activation Gradients:



Activations:



Activation Gradients:



Αυτα τα στάδια επαναλαμβάνονται ξανά ενώ στη συνέχεια έχουμε:

fc (1x1x10)
max activation: 9.70852, min: -16.31245
max gradient: 0.00034, min: -0.00036
parameters: $10 \times 256 + 10 = 2570$

Activations:
■ ■ ■ ■ ■ ■ ■ ■ ■ ■
Activation Gradients:
■ ■ ■ ■ ■ ■ ■ ■ ■ ■

softmax (1x1x10)
max activation: 0.99964, min: 0
max gradient: 0, min: 0

Activations:
■ ■ ■ ■ ■ ■ ■ ■ ■ ■

που όπως βλέπουμε κάνουμε ένα flatten και τελικά με τη softmax παίρνουμε 1x1x10 πράγμα που περιμέναμε αφού σαν κλασσικός classifier καταλήγει να έχει έξοδο με το πλήθος των κατηγοριών που έχουμε (0...9 = 10 κατηγορίες). Τελικά το δίκτυο μαθαίνει να κατηγοριοποιεί τις εικόνες που του δίνουμε στα νούμερα που αυτές δείχνουν.

Κάποια γενικά στοιχεία για τα διάφορα layers του CNN μας είναι:

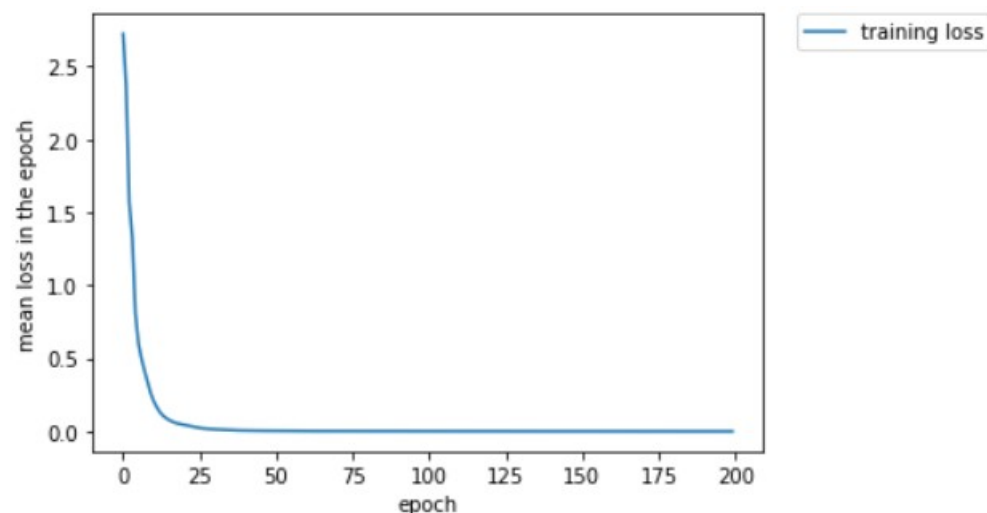
συνελίξεις: είναι φίλτρα εικόνων, όσων αφορά δυσδιάστατες ποσότητες. Στα νευρωνικά δίκτυα τα χρησιμοποιήσαμε καθώς μας δίνουν τη δυνατότητα εύρεσης ενός χαρακτηριστικού σε όλη την εικόνα.

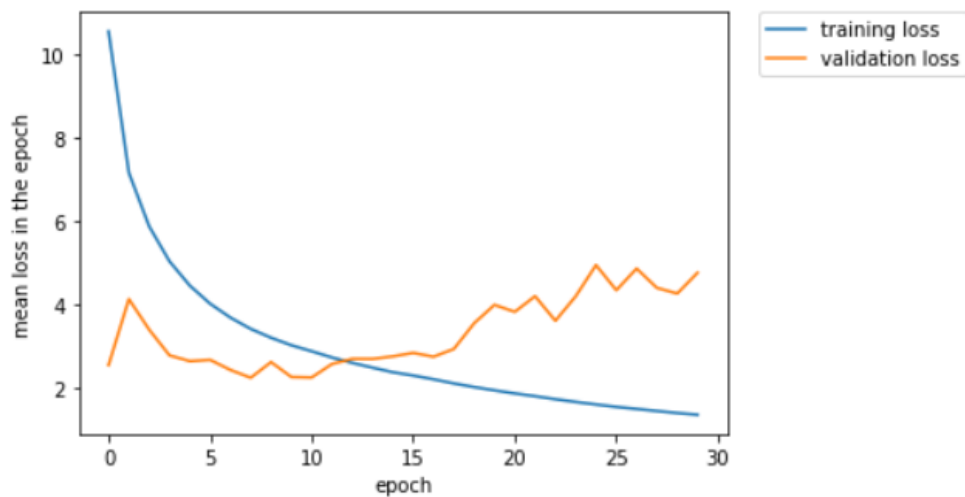
max pooling : μειώνουμε τις διαστάσεις της εικόνας επιλέγοντας από ένα προκαθορισμένου μεγέθους τετράγωνο κάποιων pixel αυτό με τη μεγαλύτερη τιμή.

ReLU : είναι μια activation μη-γραμμική συνάρτηση, η οποία επιτρέπει την μοντελοποίηση μη-γραμμικών εξαρτήσεων μεταξύ των data. Μέσω αυτής πετυχαίνουμε μια “άμυνα” απέναντι στα προβλήματα της vanishing/exploding gradient αν και το LSTM είναι καλύτερη και πιο αποτελεσματική λύση απέναντι σε αυτά τα ζητήματα.

batch normalization : βοηθά στην ομαλή μετάβαση μεταξύ των layers επιτρέποντας μεγαλύτερο learning rate, ενώ επιφέρει και το regularization του μοντέλου μέσα από την κανονικοποίηση του batch που κάθε φορά το δίκτυο δέχεται στην είσοδο κάθε επιπέδου, που εφαρμόζεται αυτή η τεχνική.

Τα αποτελέσματα του CNN είναι:



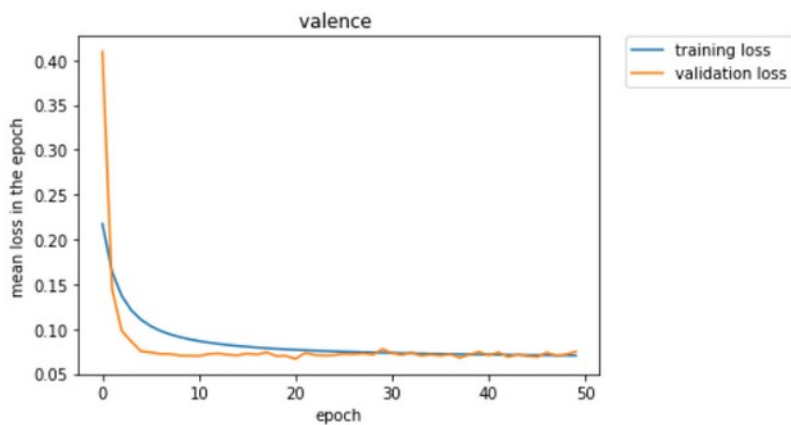


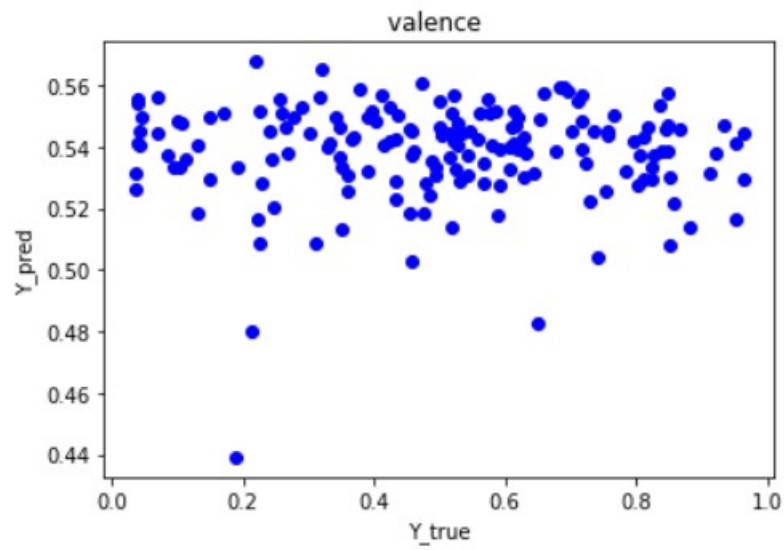
	precision	recall	f1-score	support
0	0.06	0.07	0.07	40
1	0.51	0.53	0.52	40
2	0.46	0.30	0.36	80
3	0.30	0.39	0.34	80
4	0.32	0.62	0.42	40
5	0.18	0.17	0.18	40
6	0.39	0.56	0.46	78
7	0.09	0.03	0.04	40
8	0.33	0.18	0.24	103
9	0.15	0.12	0.13	34
accuracy			0.31	575
macro avg	0.28	0.30	0.28	575
weighted avg	0.31	0.31	0.30	575

Η διάφορα σε σχέση με το 5α είναι ότι γενικά το CNN έχει καλύτερα αποτελέσματα όπως βλέπουμε από το classification report και στις 3 μετρικές.

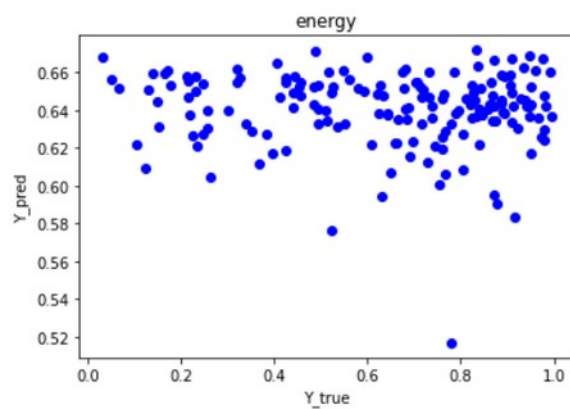
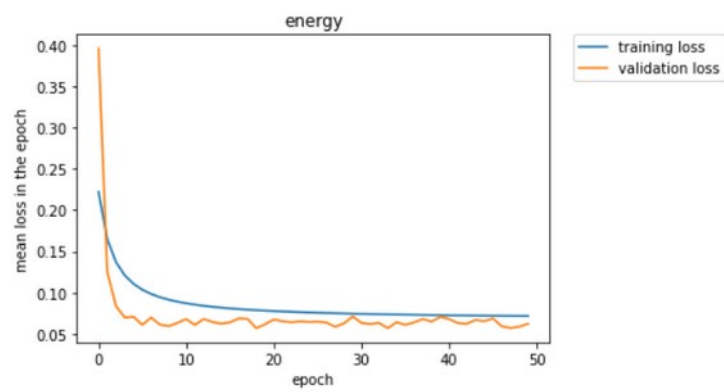
Βήμα 8

Για το βήμα αυτό κάνουμε regression και το αποτέλεσμα είναι:
Regression_Lstm for the feeling of **Valence**:

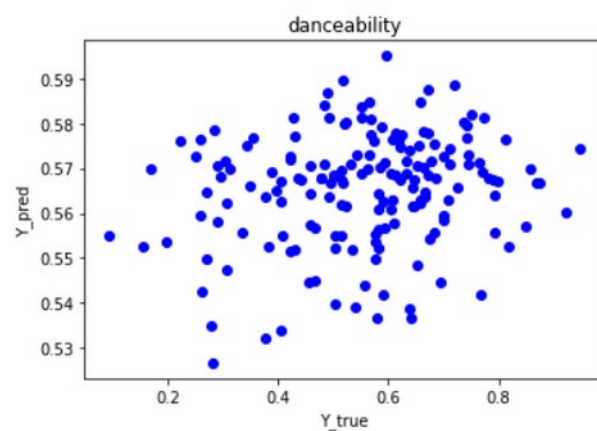
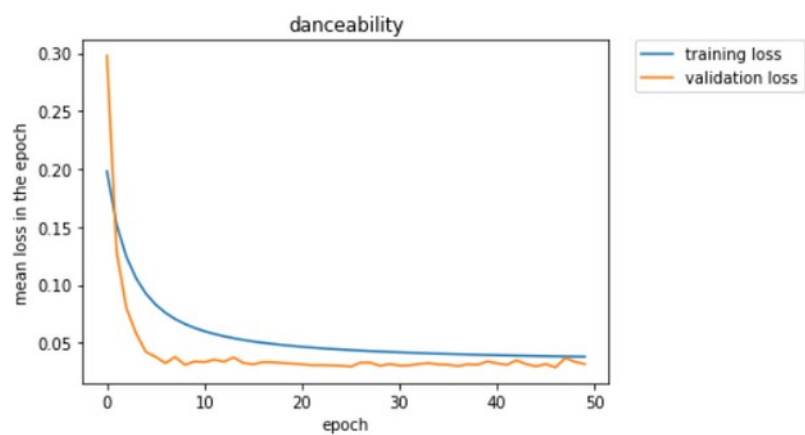




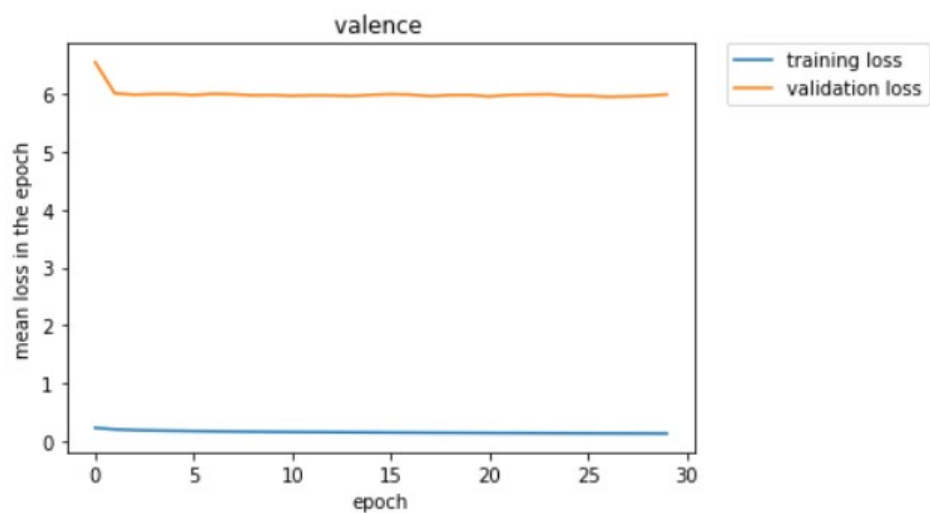
Regression_Lstm for the feeling of **Energy**



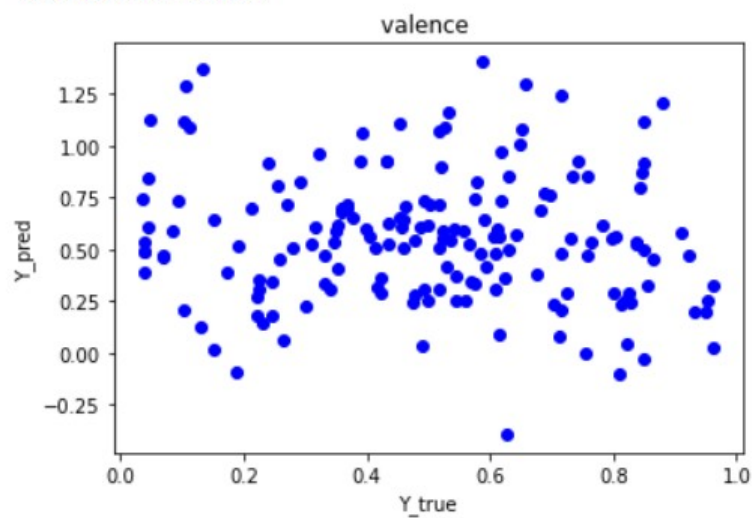
Regression_Lstm for the feeling of **DanceAbility**



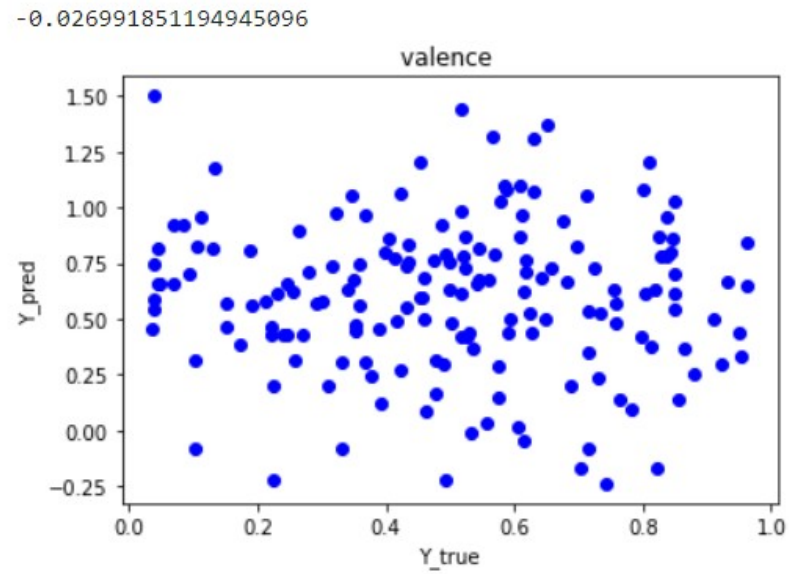
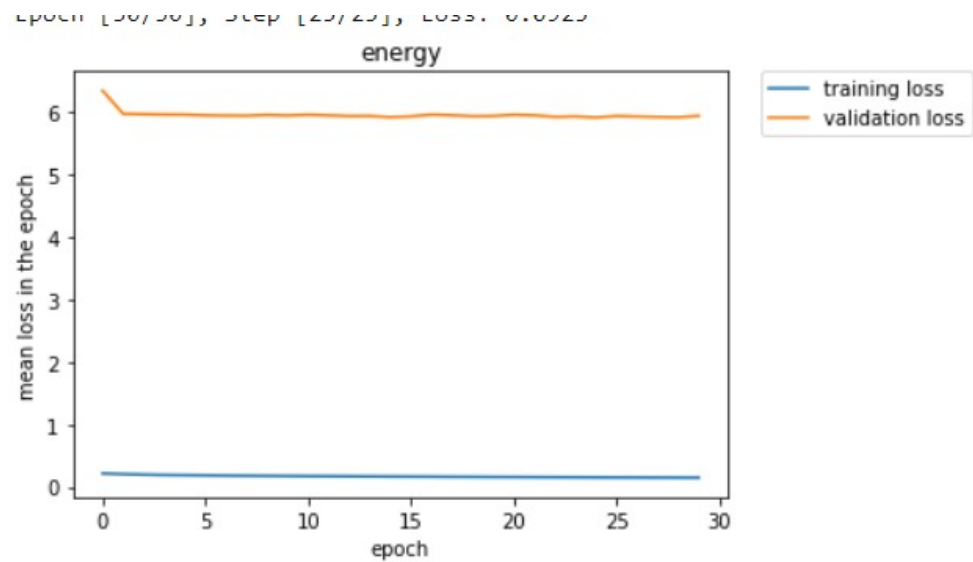
Regression_CNN for the feeling of **Valence**:



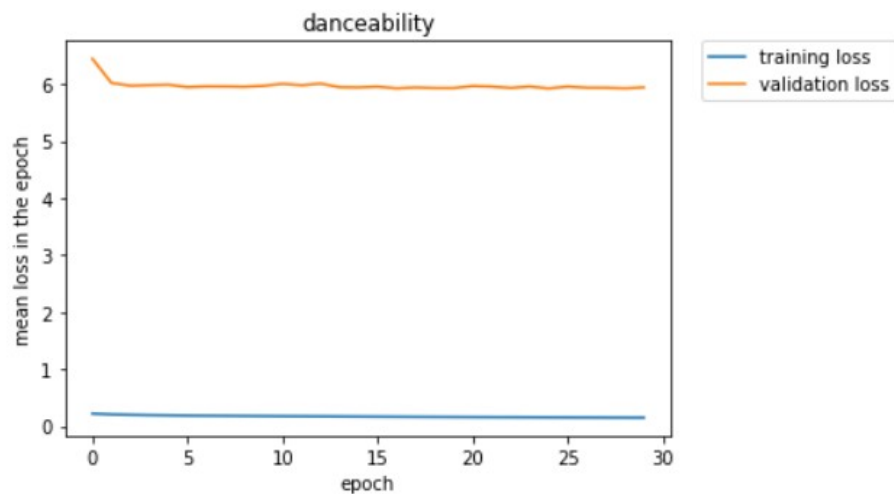
-0.0970390584638148



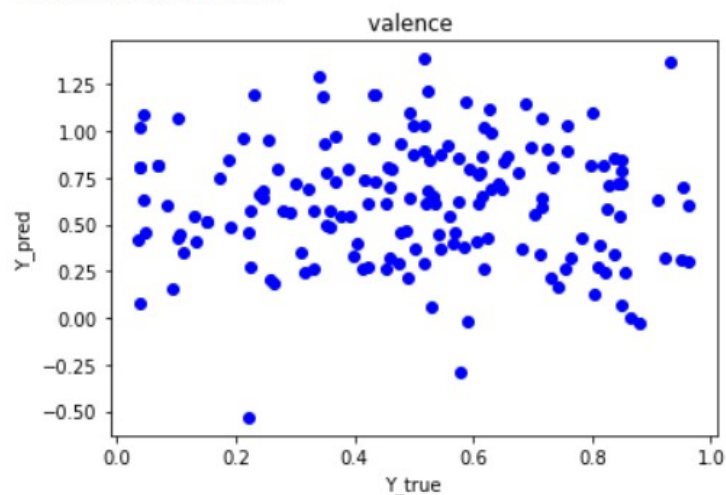
Regression_CNN for the feeling of **Energy**.



Regression_CNN for the feeling of **DanceAbility**



-0.039386843340021856



Στα 9 και 10 ο κώδικας ενώ δείχνει σωστός δε μας έδωσε αποτελέσματα.

Βήμα 9

Από τα papers βλέπουμε πως η μεταφορά γνώσης από παραπλήσιο task, μαζί με κατάλληλο fine-tuning, δίνει καλύτερα αποτελέσματα από την εκπαίδευση αποκλειστικά στο συγκεκριμένο για το task dataset καθώς ο συνδυασμός από τα βαρη οδηγεί σε πιο σωστες προβλέψεις. Ακόμα, τονίζεται πως όσο περισσότερα επίπεδα χρησιμοποιούμε, τόσο καλύτερη απόδοση μπορούμε να πετύχουμε τελικά. Αφού τελειώσαμε με την εκπαίδευση του πρώτου CNN δηλαδή του βήματος 7, το οποίο είχε 10 εξόδους (classification), τώρα θα κάνουμε το transfer learning στο νέο CNN που έχει μία έξοδο (regression). Το τελευταίο θα έχει αρχικοποιημένα τα βάρη του CNN από την εκπαίδευση του πρώτου, με στόχο μια καλύτερη επίδοση στο regression.

Βήμα 10

Μέσα από το paper τονίζεται πως ένα πολυδίκτυο χρησιμοποιήθηκε για να μάθει πολλά ετερόκλητα tasks. Η απόδοση αυτού αν και ήταν χειρότερη των βέλτιστων για κάθε task χωριστά, παρείχε αρκετά καλή επίδοση σε όλα, ειδικά συγκρινόμενο με άλλα σύγχρονα συστήματα. Τονίζει ακόμα ότι μέρη του υποδικτύου που δεν ήταν αρκετά χρήσιμα σε κάποια tasks, όταν τα προσθέταμε ξανά, παρείχαν ίσα η καλύτερα αποτελέσματα.