

ΔΔΠΜΣ “Προηγμένα Συστήματα Υπολογιστών και Επικοινωνιών”, 2019-2020
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης

Υπολογιστική Νοημοσύνη- Συστήματα Εμπνευσμένα από τη Βιολογία


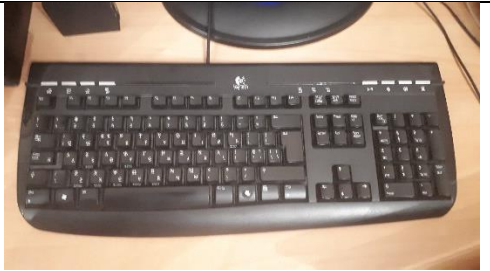




Ομάδα Ε

Μέλη της ομάδας:

Τσαλίδης Γεώργιος	AM:472	gtsalidis@ece.auth.gr
Λοΐας Ιωάννης	AM:453	loiasioann@ece.auth.gr

Πρώτο Μέρος

Στο πρώτο μέρος της εργασίας ζητήθηκε η λήψη δέκα φωτογραφιών και η ταξινόμησή τους χρησιμοποιώντας ένα ήδη εκπαιδευμένο συνελκτικό δίκτυο. Το δίκτυο που χρησιμοποιήθηκε είναι το Xception της βιβλιοθήκης tensorflow.keras ενώ τα βάρη που χρησιμοποιήθηκαν είναι αυτά του imagenet. Για κάθε φωτογραφία που δίνεται στον ταξινομητή λαμβάνουμε τις 5 κλάσεις με τη μεγαλύτερη πιθανότητα. Παρακάτω παρατίθενται οι φωτογραφίες με τις κλάσεις και τις αντίστοιχες πιθανότητες.

	sunglasses	0.4444469	
	sunglass	0.38734454	
	pitcher	0.035715956	
	brassiere	0.0067659887	
	teapot	0.0053804624	
	computer_keyboard	0.631242	
	space_bar	0.13870513	
	mouse	0.11468444	
	notebook	0.024200445	
	laptop	0.017712915	
	remote_control	0.9872277	
	hand-held_computer	0.0012697286	
	cellular_telephone	0.0010586645	
	radio	0.00081861176	
	modem	0.00040902998	
	lemon	0.9682217	
	dough	0.012192495	
	tennis_ball	0.0015300948	
	orange	0.0014811925	
	Granny_Smith	0.0011130435	
	tabby	0.45746297	
	Egyptian_cat	0.3457148	
	tiger_cat	0.1122225	
	carton	0.0020730142	
	printer	0.0020078006	
	mouse	0.9811023	
	crash_helmet	0.0035420621	
	ocarina	0.0023371703	
	pencil_sharpener	0.0015836121	
	pick	0.00093522883	

	water_bottle	0.95482934
	nipple	0.010367025
	water_jug	0.00636419
	spotlight	0.0031330136
	perfume	0.001443871
	monitor	0.26886553
	desktop_computer	0.24354598
	screen	0.14409539
	notebook	0.05795409
	television	0.05063977
	banana	0.90933925
	slug	0.03450569
	orange	0.0030577474
	flatworm	0.0020179055
	lemon	0.0015173818
	folding_chair	0.29586083
	bannister	0.065793015
	matchstick	0.06361558
	balance_beam	0.031011758
	hook	0.028088957

Παρατηρούμε ότι για τις περισσότερες φωτογραφίες η επικρατέστερη κλάση έχει πολύ μεγάλη πιθανότητα ($>0,9$) σε σχέση με την επόμενη κλάση. Αυτό οφείλεται στο γεγονός ότι σε κάθε φωτογραφία υπάρχει μόνο ένα αντικείμενο για αναγνώριση.

Σε κάποιες φωτογραφίες, όπως συμβαίνει στην τηλεόραση, οι πιθανότητες των 4 επικρατέστερων κλάσεων είναι αρκετά κοντά. Αυτό συμβαίνει διότι τα αντικείμενα των κλάσεων αυτών μοιάζουν αρκετά και στον φυσικό κόσμο και επομένως είναι δύσκολο να γίνει η κατάλληλη αναγνώριση από τη φωτογραφία.

Δεύτερο μέρος

Στο δεύτερο μέρος της εργασίας, με τη χρήση του [Detectron2](#) του Facebook AI, κάναμε segmentation σε 10 φωτογραφίες. Πιο συγκεκριμένα, χρησιμοποιήσαμε ένα προ-εκπαιδευμένο μοντέλο του Detectron2, το δοκιμάσαμε σε 10 φωτογραφίες και 1 βίντεο και παρατηρήσαμε τα αποτελέσματα και τα συμπεράσματα του.

Ανάλυση της διαδικασίας

Για ταχύτερα αποτελέσματα τρέξαμε τον κώδικα στο [Colab](#) (Cloud υπηρεσία της Google) που παρέχει runtime με accelerator GPU.

Αρχικά κάναμε την εγκατάσταση των απαραίτητων dependencies που απαιτούνται και μετέπειτα εγκαταστήσαμε το detectron2. Μετέπειτα φορτώσαμε τις βιβλιοθήκες και modules που χρειάζονται στο script μας.

Με το config (*COCO-PanopticSegmentation/panoptic_fpn_R_101_3x.yaml*) ,χρησιμοποιήσαμε μοντέλο για πανοραμική ανίχνευση αντικειμένων(**Panoptic segmentation**) το οποίο ενοποιεί τις τυπικά διακριτές εργασίες του **semantic segmentation** (εγκωφεί μια ετικέτα κλάσης σε κάθε pixel) και του **instance segmentation** (ανίχνευση και τμηματοποίηση/οριοθέτηση κάθε στιγμιότυπου αντικειμένου).

Ενδεικτικά αποτελέσματα:







Για περισσότερα αποτελέσματα: https://github.com/GiorgosTsal/Computational-Intelligence-Neural-Networks/blob/master/src/part2/panoptic_segmentation.ipynb

Επιπλέον κάνουμε **Instance Segmentation** για ανίχνευση και οριοθέτηση κάθε διακριτού αντικειμένου. Το επιτυγχάνουμε με τη χρήση του μοντέλου `model_zoo`, δημιουργώντας ένα `config` (πιο συγκεκριμένα το: `COCO-InstanceSegmentation/mask_rcnn_R_50_FPN_3x.yaml`) και έναν `detectron2 DefaultPredictor` για να βγάλουμε συμπεράσματα στις εικόνες μας. Φορτώνουμε προ-εκπαιδευμένα βάρη και καλούμε την `predictor()` στις εικόνες μας. Μετά κάνουμε οπτικοποίηση των αποτελεσμάτων στις εικόνες μας με τη χρήση του `Visualizer()`.

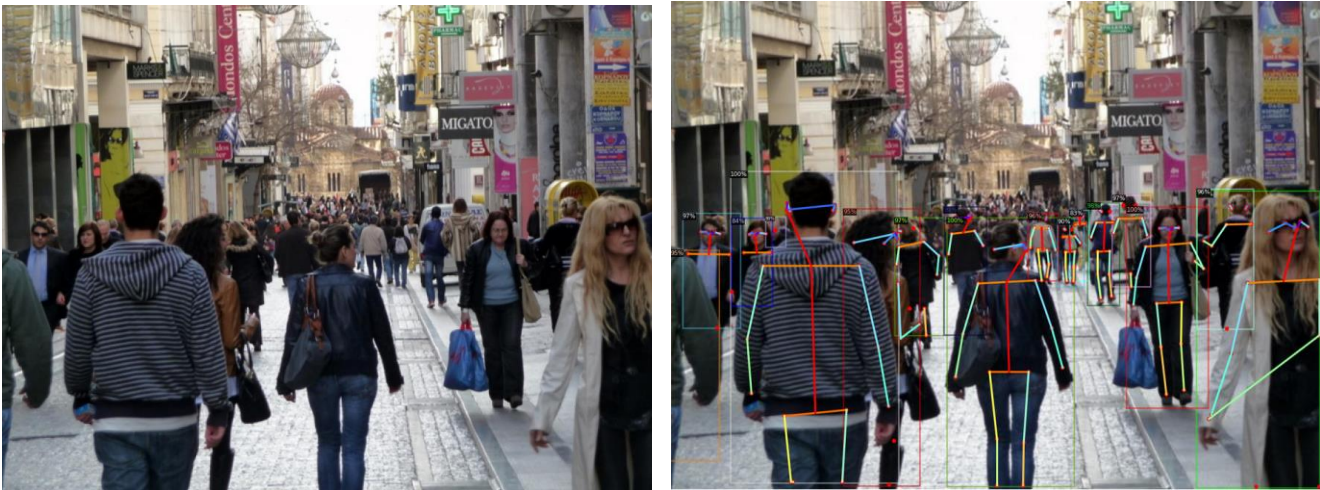
Ενδεικτικά αποτελέσματα:



Για περισσότερα αποτελέσματα : https://github.com/GiorgosTsal/Computational-Intelligence-Neural-Networks/blob/master/src/part2/instance_segmentation_and_video.ipynb

Επίσης με το config (*COCO-Keypoints/keypoint_rcnn_R_50_FPN_3x.yaml*) ,χρησιμοποιήσαμε μοντέλο για ανίχνευση βασικών σημείων (**Keypoint Detection**) που περιλαμβάνει την ταυτόχρονη ανίχνευση ατόμων και εντοπισμό των βασικών σημείων τους. Τα βασικά σημεία είναι τα ίδια με τα σημεία ενδιαφέροντος. Είναι χωρικές τοποθεσίες ή σημεία στην εικόνα που ορίζουν τι είναι ενδιαφέρον ή τι ξεχωρίζει στην εικόνα.

Ενδεικτικά αποτελέσματα:



Για περισσότερα αποτελέσματα: https://github.com/GiorgosTsal/Computational-Intelligence-Neural-Networks/blob/master/src/part2/keypoint_detection.ipynb

Τέλος, πειραματικά δοκιμάσαμε το μοντέλο για πανοραμική ανίχνευση αντικειμένων (**Panoptic segmentation**) σε βίντεο. Τα αποτελέσματα του μπορείτε να δείτε εδώ: <https://www.youtube.com/watch?v=kMTt5vgXQ4k> . Παρατηρήσαμε πως το μοντέλο δίνει πολύ καλά αποτελέσματα για σχεδόν όλα τα frames του βίντεο, ταξινομώντας και οριοθετώντας τα αντικείμενα εύστοχα.

Τρίτο Μέρος

Στο τρίτο μέρος της εργασίας ζητήθηκε η εκπαίδευση ενός συνελκτικού νευρωνικού δικτύου με σκοπό την αναγνώριση συγκεκριμένων προσώπων σε φωτογραφίες. Το σύνολο δεδομένων που χρησιμοποιήθηκε αποτελεί ένα υποσύνολο του VGG Face2. Συγκεκριμένα, περιλαμβάνει τις κλάσεις no000021 έως και no000025. Σε δεύτερη φάση προστέθηκε και μια επιπλέον κλάση ώστε να ταξινομηθεί κάποιο πρόσωπο που δεν ανήκει σε κάποια από τις προηγούμενες. Ως μετρική αξιολόγησης του μοντέλου χρησιμοποιήθηκε η ακρίβεια (accuracy) ενώ παράλληλα μετράτε και το σφάλμα LOSS.

Ανάλυση της διαδικασίας

Αρχικά, γίνεται ο διαχωρισμός του συνόλου δεδομένων σε σύνολο εκπαίδευσης, αξιολόγησης και δοκιμής. Το σύνολο εκπαίδευσης περιέχει το 70% των φωτογραφιών του αρχικού συνόλου δεδομένων ενώ τα άλλα δύο περιέχουν το 15%. Το σύνολο εκπαίδευσης και αξιολόγησης χρησιμοποιούνται για την κατάλληλη εκπαίδευση του νευρωνικού δικτύου ενώ το σύνολο δοκιμής ελέγχει την ορθότητα της ταξινόμησης που εκτελεί το νευρωνικό δίκτυο.

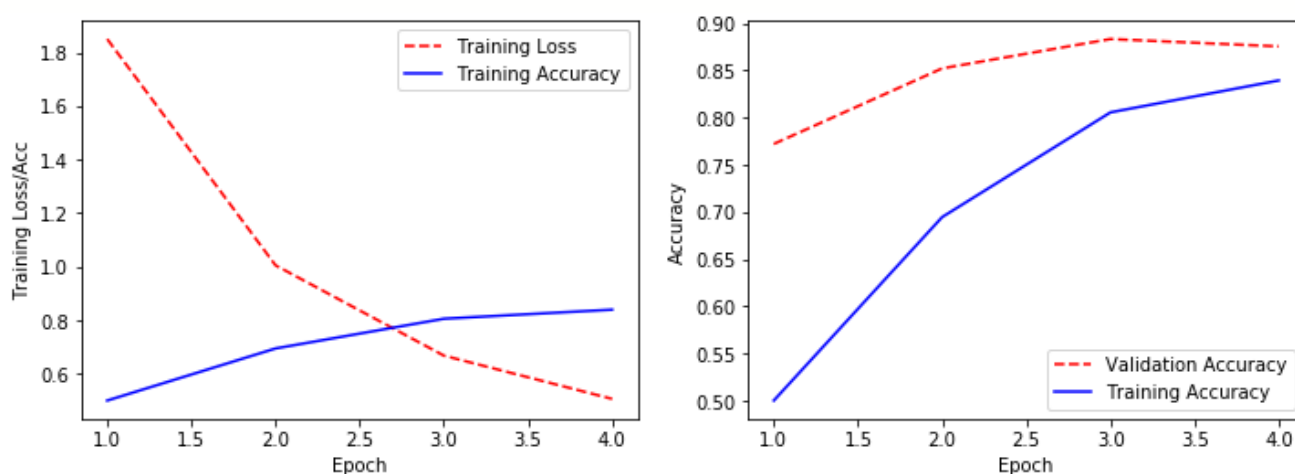
Επειδή το σύνολο δεδομένων που χρησιμοποιείται για την εκπαίδευση θεωρείται μικρό θα πρέπει να γίνει «αύξηση» των φωτογραφιών. Για το λόγο αυτό χρησιμοποιείται η κλάση ImageDataGenerator της βιβλιοθήκης keras.preprocessing.image. Η κλάση αυτή αφορά το σύνολο εκπαίδευσης και αξιολόγησης και ρυθμίστηκε έτσι ώστε να πραγματοποιείται μετατόπιση (κάθετη και πλάγια), περιστροφή αλλά και οριζόντια μετακίνηση. Επίσης, στις εν λόγω φωτογραφίες γίνεται και κλιμάκωση. Για το σύνολο δοκιμής, χρησιμοποιείται η κλάση ImageDataGenerator μόνο για κλιμάκωση των φωτογραφιών.

Στη συνέχεια, γίνεται η δημιουργία του βασικού μοντέλου(base model) του συνελκτικού νευρωνικού δικτύου, χρησιμοποιώντας προεκπαιδευμένα βάρη. Το base model διαθέτει όλες τις ενδιάμεσες βαθμίδες εκτός από τη βαθμίδα εξόδου. Ως base model

χρησιμοποιήθηκε το νευρωνικό δίκτυο Resnet50 της βιβλιοθήκης Keras ενώ τα προεκπαιδευμένα βάρη είναι εκείνα του imagenet. Οι ενδιάμεσες βαθμίδες ενός συνελκτικού νευρωνικού δικτύου συμβάλλουν στην αναγνώριση πιο αφηρημένων χαρακτηριστικών μιας φωτογραφίας. Έτσι, με την χρήση προεκπαιδευμένων βαρών, όλες οι παράμετροι του νευρωνικού δικτύου αποκτούν τις επιθυμητές τιμές σε λιγότερες εποχές, καθώς για πολλά βάρη οι αρχικές τιμές ήταν αρκετά κοντά στις τελικές.

Παρόλο που χρησιμοποιούμε προεκπαιδευμένα βάρη, οι τελικές κλάσεις στις οποίες ταξινομεί τις φωτογραφίες το νευρωνικό μας δίκτυο διαφέρουν από τις κλάσεις του imagenet. Έτσι, κατασκευάζουμε το μοντέλο κορυφής (top model), το οποίο περιέχει των αριθμό των κλάσεων που ορίσαμε αρχικά και αποτελεί την έξοδο του νευρωνικού δικτύου. Το μοντέλο αυτό χρησιμοποιεί ένα επίπεδο για να μετατρέψει όλα τα χαρακτηριστικά που αναγνωρίστηκαν σε προηγούμενο επίπεδο σε μια ενιαία πρόβλεψη ανά εικόνα.

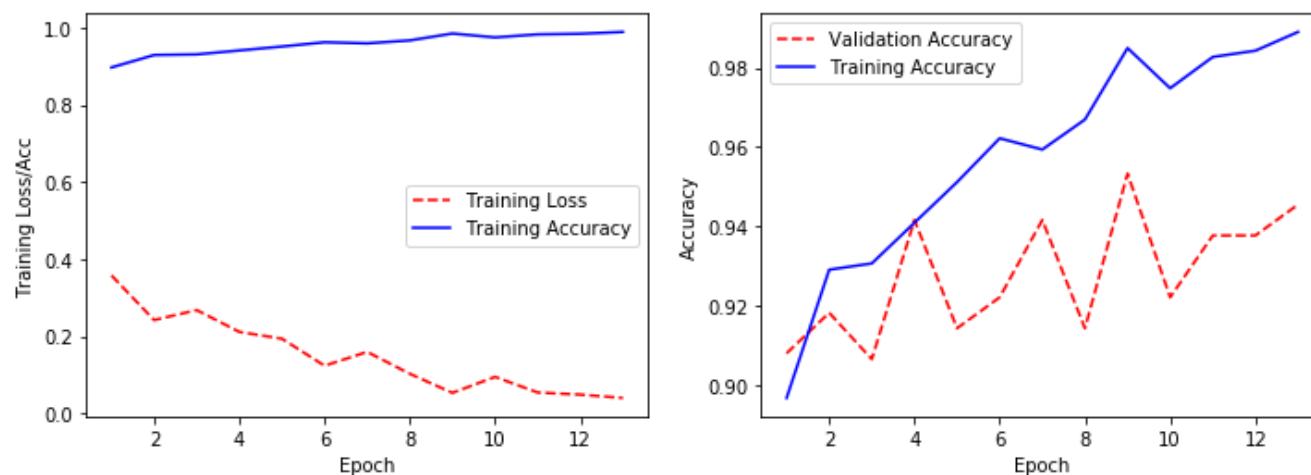
Αν πραγματοποιηθεί η εκπαίδευση βάσει του μοντέλου που περιγράψαμε, κατά την πρώτη εποχή θα γίνει μια λανθασμένη εκτίμηση των βαρών του τελευταίου επιπέδου του μοντέλου (μεγάλο σφάλμα LOSS). Αυτή η λανθασμένη εκτίμηση, θα σταλεί ως ανάδραση στα προηγούμενα επίπεδα με αποτέλεσμα να αντικατασταθούν τα προεκπαιδευμένα βάρη με χειρότερες τιμές. Για να αποφευχθεί αυτό, αρχικά για έναν μικρό αριθμό εποχών εκπαιδεύουμε μόνο το τελευταίο επίπεδο του μοντέλου μας, «παγώνοντας» όλα τα προηγούμενα επίπεδα. Στην δική μας υλοποίηση, αυτό συμβαίνει θέτοντας τη μεταβλητή trainable του base_model false πριν την εκπαίδευση. Η εκπαίδευση αυτή συμβαίνει για τις 4 πρώτες εποχές.



Εικόνα 1. Διαγράμματα με freeze τα βάρη για 5 κλάσεις

Στην Εικόνα 1 παρουσιάζονται οι γραφικές παραστάσεις του σφάλματος εκπαίδευσης και της ακρίβειας (αριστερά) καθώς και η ακρίβεια για το σύνολο εκπαίδευσης και δοκιμής για τις τέσσερις πρώτες εποχές που είναι «παγωμένα» όλα τα επίπεδα. Παρατηρούμε ότι με τη χρήση των προεκπαιδευμένων βαρών πετυχαίνετε από την πρώτη εποχή ακρίβεια στο σετ αξιολόγησης ίση με 0.7721 ενώ το σφάλμα loss ισούται με 0.6235. Μετά την τέταρτη εποχή οι τιμές αυτές είναι αντίστοιχα 0.8755 και 0.3794. Η ακρίβεια και το σφάλμα LOSS στο σύνολο δοκιμής μετά τις 4 αυτές εποχές είναι 0.39 και 0.91.

Στη συνέχεια, επαναλαμβάνουμε τη διαδικασία της εκπαίδευσης για άλλες 12 εποχές με τη διαφορά ότι συμμετέχουν όλα τα επίπεδα του νευρωνικού δικτύου. Η διαδικασία αυτή ονομάζεται fine tuning και έχει ως στόχο τη βελτίωση της ακρίβειας και τη μείωση του σφάλματος.

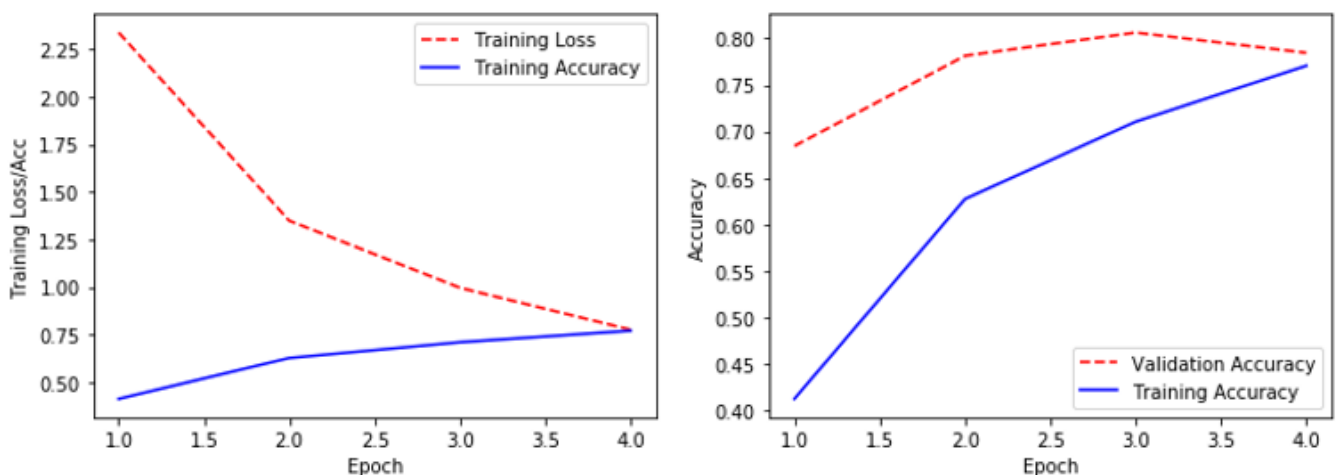


Εικόνα 2. Διαγράμματα Fine Tuning για 5 κλάσεις

Στην Εικόνα 2 παρουσιάζονται οι γραφικές παραστάσεις του σφάλματος εκπαίδευσης και της ακρίβειας (αριστερά) καθώς και η ακρίβεια για το σύνολο εκπαίδευσης και δοκιμής για τις εποχές που εφαρμόζεται η διαδικασία του fine tuning. Μετά το πέρας της εκπαίδευσης η ακρίβεια αξιολόγησης (validation accuracy) ισούται με 0.9377 ενώ το σφάλμα μειώνεται στο 0.3449. Παρατηρούμε ότι η ακρίβεια είναι αρκετά υψηλή και η διαφορά της από την τέταρτη εποχή είναι μικρή. Αυτό δικαιολογείται από την ανομοιογένεια των προσώπων του συνόλου δεδομένων. Η ακρίβεια και το σφάλμα LOSS στο σύνολο δοκιμής μετά το fine tuning είναι 0.28 και 0.95.

Στη συνέχεια προστίθεται μια νέα κλάση. Στην κλάση αυτή ταξινομούνται όσες φωτογραφίες προσώπων δεν ανήκουν στις άλλες πέντε. Για τις ανάγκες δημιουργίας αυτής της κλάσης χρησιμοποιήθηκαν φωτογραφίες τόσο από άντρες όσο και από γυναίκες με διαφορετικά χαρακτηριστικά. Οι φωτογραφίες αυτές προέκυψαν από τις υπόλοιπες κλάσεις του VGG Face2. Όλες οι κλάσεις έχουν περίπου τον ίδιο αριθμό φωτογραφιών.

Η διαδικασία που ακολουθήθηκε για την εκπαίδευση του νευρωνικού δικτύου είναι ίδια με πριν. Από την εκπαίδευση αυτή προκύπτουν, όπως παρουσιάζονται στην Εικόνα 3, τα διαγράμματα ακρίβειας και σφάλματος για το σύνολο εκπαίδευσης (αριστερά) και τα διαγράμματα ακρίβειας των συνόλων εκπαίδευσης και αξιολόγησης (δεξιά) κατά τις 4 πρώτες εποχές. Η ακρίβεια και το σφάλμα στο σύνολο δοκιμής είναι 0.81 και 0.76. Τα ίδια διαγράμματα, αφού εφαρμοστεί η διαδικασία του fine tuning παρουσιάζονται στην Εικόνα 4. Εκεί, η ακρίβεια και το σφάλμα στο σύνολο δοκιμής είναι 0.92 και 0.49.



Εικόνα 3. Διαγράμματα με freeze τα βάρη για 6 κλάσεις

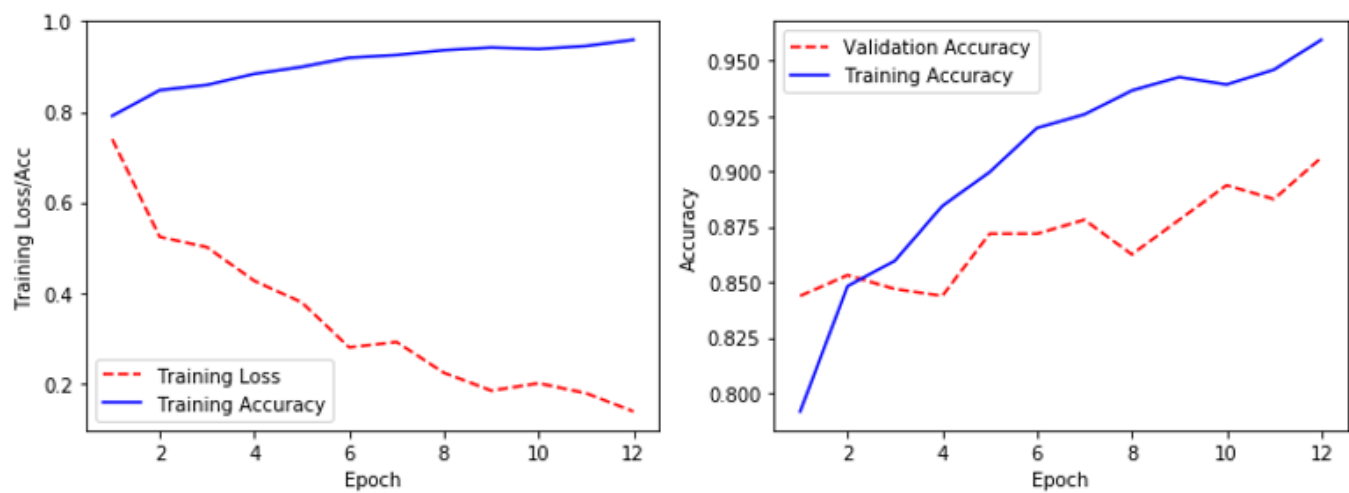
Classification Report:

	precision	recall	f1-score	support
aaron_diaz	0.90	0.89	0.89	61
aaron_tippin	0.80	0.93	0.86	44
aarti_chabria	0.89	0.90	0.89	60
abbey_clancy	0.75	0.91	0.82	69
abby_elliott	0.82	0.70	0.76	47
unkown	0.79	0.55	0.65	49
accuracy			0.82	330
macro avg	0.83	0.81	0.81	330
weighted avg	0.83	0.82	0.82	330

Confusion Matrix:

```
[[54  3  0  0  0  4]
 [ 3 41  0  0  0  0]
 [ 0  0 54  5  0  1]
 [ 0  2  0 63  3  1]
 [ 0  0  2 11 33  1]
 [ 3  5  5  5  4 27]]
```

Στη συνέχεια, επαναλαμβάνουμε τη διαδικασία της εκπαίδευσης για άλλες 12 εποχές:



Εικόνα 4. Διαγράμματα fine για 6 κλάσεις

Επιπλέον για την αξιολόγηση του μοντέλου χρησιμοποιήσαμε πίνακα σύγχυσης (confusion matrix), καθώς και Ακρίβεια, Ανάκληση και f1-score.

Classification Report:

	precision	recall	f1-score	support
aaron_diaz	0.91	0.97	0.94	61
aaron_tippin	0.98	0.95	0.97	44
aarti_chabria	0.92	0.95	0.93	60
abbey_clancy	0.87	0.96	0.91	69
abby_elliott	0.86	0.79	0.82	47
unkown	0.90	0.76	0.82	49
accuracy			0.90	330
macro avg	0.91	0.90	0.90	330
weighted avg	0.90	0.90	0.90	330

Confusion matrix:

```
[[59  0  0  0  0  2]
 [ 2 42  0  0  0  0]
 [ 0  0 57  1  1  1]
 [ 0  0  1 66  2  0]
 [ 0  0  1  8 37  1]
 [ 4  1  3  1  3 37]]
```