



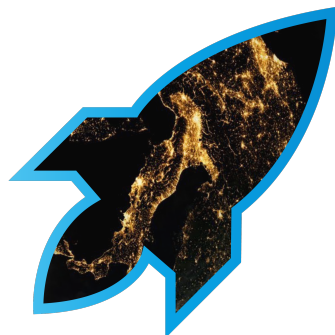
ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών
— ΙΔΡΥΘΕΝ ΤΟ 1837 —

ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΑΛΓΟΡΙΘΜΙΚΑ ΠΡΟΒΛΗΜΑΤΑ

ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2020

2Η ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ

ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΑΠΟΤΙΜΗΣΗ ΝΕΥΡΩΝΙΚΟΥ
ΔΙΚΤΥΟΥ ΑΥΤΟΚΩΔΙΚΟΠΟΙΗΣΗΣ ΕΙΚΟΝΩΝ
ΑΡΙΘΜΗΤΙΚΩΝ ΨΗΦΙΩΝ



Αριθμός Μητρώου(ΑΜ):

1115201700217

1115201700203

Ονοματεπώνυμο:

Ορέστης ΣΤΕΦΑΝΟΥ

Λεωνίδας ΕΦΡΑΙΜ

ΑΚΑΔΗΜΑΪΚΗ ΧΡΟΝΙΑ 2020-2021

ΠΕΡΙΕΧΟΜΕΝΑ

1	ΕΙΣΑΓΩΓΗ	3
2	ΜΕΡΟΣ Α' AUTOENCODER	4
3	ΜΕΡΟΣ Β' CLASSIFICATION	6

ΕΙΣΑΓΩΓΗ

Σε αυτή την εργασία κληθήκαμε να υλοποιήσουμε ένα νευρωνικό δίκτυο αυτοκωδικοποίησης ψηφιακών εικόνων. Για την υλοποίηση χρησιμοποιήσαμε τη γλώσσα Python με τη βοήθεια των βιβλιοθηκών Keras και Tensorflow. Εκτός από αυτά χρησιμοποιήσαμε και το Google Collab το οποίο μας παρείχε επεξεργαστική ισχύ για τους μεγάλους υπολογισμούς που χρειαστήκαμε μέσω των GPU που μας παρείχε. Στο πρώτο μέρος δημιουργήσαμε τον encoder και τον εκπαιδεύσαμε, ενώ στη συνέχεια στο δεύτερο μέρος υλοποιήσαμε και κατηγοριοποίηση στον encoder μας.

ΜΕΡΟΣ Α' AUTOENCODER

Για την υλοποίησή του χρησιμοποιήσαμε τις παρακάτω συναρτήσεις

- **read_data** Η συνάρτηση αυτή παίρνει ως όρισμα ένα συμπιεσμένο binary αρχείο με τη μορφή που ζητά η εκφώνηση, διαβάζει τα δεδομένα από το αρχείο και στη συνέχεια τα επιστρέφει. Αρχικά, αποσυμπίεζει το αρχείο και επιστρέφει ένα δείκτη στην αρχή του αρχείου. Στη συνέχεια διαβάζει ανά 4 bytes διαδοχικά, για να διαβάσει το magic number, τον αριθμό των εικόνων, το μήκος και το πλάτος της κάθε εικόνας. Τέλος, αφού γνωρίζει τον αριθμό των εικόνων που περιέχονται στο αρχείο διαβάζει όλες τις εικόνες και τις βάζει σε μια δομή με το όνομα data και τις επιστρέφει
- **encoder** Αυτή η συνάρτηση είναι υπεύθυνη για την κωδικοποίηση του input στα convolution layers. Εδώ χρησιμοποιήσαμε τη συνάρτηση από τις διάφανιες του φροντιστηρίου που μας δόθηκαν από τον Κ.Χαμόδρακα. Ο encoder δημιουργεί 4 layers, όπου το πρώτο είναι ένα convolution layer με 32 φίλτρα με μέγεθος 3x3. Το επόμενο layer έχει 64 φίλτρα με μέγεθος 3x3 και τα επόμενα δύο layers 128 και 256 φίλτρα με μέγεθος 3x3.
- **decoder** Η συνάρτηση αυτή κάνει την αντίστροφη διαδικασία από τον encoder, δηλαδή αποκωδικοποιεί το input. Πάλι εδώ χρησιμοποιήσαμε τη συνάρτηση από τις διάφανιες του φροντιστηρίου που μας δόθηκαν από τον Κ.Χαμόδρακα. Στο τέλος της αποκωδικοποίησης χρησιμοποιείται η σιγμοειδής συνάρτηση για να κανονικοποιήσουμε τις τιμές στο διάστημα 0 με 1

Το πρόγραμμα αρχικά ελέγχει αν έχουν δοθεί σωστά τα ορίσματα. Στη συνέχεια ζητά από τον χρήστη να δώσει το batch size και τον αριθμό των epoch. Ακολούθως διαβάζει το αρχείο που δόθηκε σαν όρισμα και χωρίζει τα δεδομένα σε train set και validation set για την εκπαίδευση και στη συνέχεια την επαλήθευση του δικτύου. Τέλος, δίνει τη δυνατότητα στον χρήστη να επιλέξει αν θέλει να τερματίσει το πρόγραμμα ή να αποθηκεύσει τα δεδομένα ή να παρουσιάσει τη γραφική παράσταση ή να επαναληφθεί ο αλγόριθμος εκμάθησης με νέες υπερπαραμέτρους

ΜΕΡΟΣ Β' CLASSIFICATION

Στο δεύτερο μέρος της εργασίας κληθήκαμε να υλοποιήσουμε ένα νευρωνικό δίκτυο κατηγοριοποίησης εικόνων βασιμμένοι στην κωδικοποίηση (encoder) που κάναμε στο πρώτο μέρος της εργασίας. Το πρόγραμμα `classification.py` δέχεται συμπιεσμένα αρχεία με τη μορφή που δόθηκαν στο πλαίσιο της εργασίας. Η συνάρτηση για την ανάγνωση των συμπιεσμένων αρχείων είναι η ίδια με το πρώτο μέρος της εργασίας (`read_data`). Αλλά χρειαστήκε ακόμα μια συνάρτηση για την ανάγνωση των αρχείων με τις ετικέτες, οπότε με λίγες τροποποιήσεις δημιουργήσαμε τη νέα συνάρτηση.

- **read_labels** Η συνάρτηση αυτή παίρνει ως όρισμα ένα συμπιεσμένο binary αρχείο με τη μορφή που ζητά η εκφώνηση, διαβάζει τα δεδομένα από το αρχείο και στη συνέχεια τα επιστρέφει. Αρχικά αποσυμπιέζει το αρχείο και επιστρέφει ένα δείκτη στην αρχή του αρχείου. Στη συνέχεια διαβάζει ανά 4 bytes διαδοχικά για να διαβάσει το magic number, τον αριθμό των ετικετών. Τέλος, αφού γνωρίζει τον αριθμό των ετικετών που περιέχονται στο αρχείο διαβάζει όλες τις ετικέτες και τις βάζει σε μια δομή με το όνομα `labels` και τις επιστρέφει

Επίσης χρησιμοποιούμε τις συναρτήσεις encoder από το πρώτο μέρος, επομένως η περιγραφή της είναι η ίδια. Άλλες συναρτήσεις που χρησιμοποιούνται είναι οι παρακάτω

- **printHelp** Η συνάρτηση αυτή τυπώνει ένα μήνυμα το οποίο υποδεικνύει τη λάθος καταχώρηση ορισμάτων στο πρόγραμμα

- **fc** Η συνάρτηση αυτή παίρνει σαν όρισμα τον `endocer` και προσθέτει ένα πλήρως συνδεδεμένο στρώμα για να ενωθεί με το στρώμα εξόδου

Το πρόγραμμα αρχικά ελέγχει αν έχουν δοθεί σωστά τα ορίσματα. Στη συνέχεια ζητά από τον χρήστη να δώσει το `batch size` και τον αριθμό των `epoch`. Ακολουθώ, διαβάζει τα αρχεία που δόθηκαν σαν ορίσματα και χωρίζει τα δεδομένα σε `train set` και `validation set` για την εκπαίδευση. Παράλληλα διαβάζει τις ετικέτες για τα αντίστοιχα `train` και `test` δεδομένα τα οποία τα αποθηκεύει στους αντίστοιχους τύπους δεδομένων. Στη συνέχεια δημιουργεί ένα νέο μοντέλο με βάση το μοντέλο που δόθηκε σαν όρισμα αντιγράφοντας όλα τα `layers` του στο καινούριο μοντέλο και προσθέτοντας ένα πλήρως συνδεδεμένο στρώμα για να υπάρξει σύνδεση με το στρώμα εξόδου. Το καινούριο μοντέλο που δημιουργήθηκε με τη κατηγοριοποίηση είναι έτοιμο για εκπαίδευση και επαλήθευση. Ανάλογα με την επιλογή του χρήστη μπορούμε να επαναλάβουμε τις μετρήσεις, να παρουσιάσουμε τις γραφικές παραστάσεις με τα `error` και το `accuracy` του δικτύου ή να προχωρήσουμε με κατηγοριοποίηση