ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

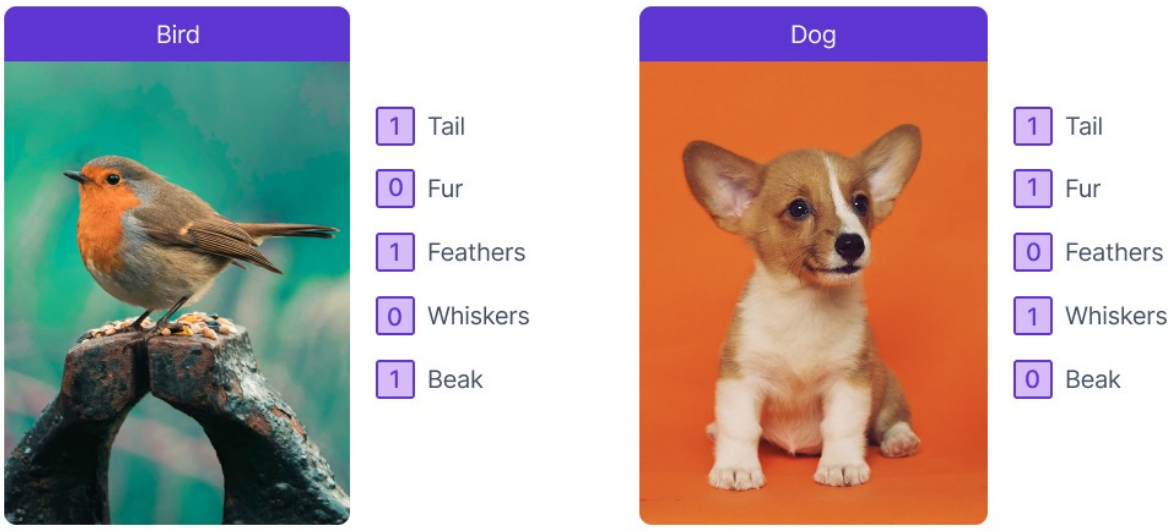
ΣΧΟΛΗ ΨΗΦΙΑΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

Συγκριτική αξιολόγηση μεθόδων γενικευμένης εκμάθησης χωρίς παραδείγματα για την κατηγοριοποίηση εικόνων

Πτυχιακή εργασία

Λευτέρης Δημητρίου



Πηγή Εικόνας: [53]

Αθήνα, Φεβρουάριος 2023

HAROKOPIO UNIVERSITY

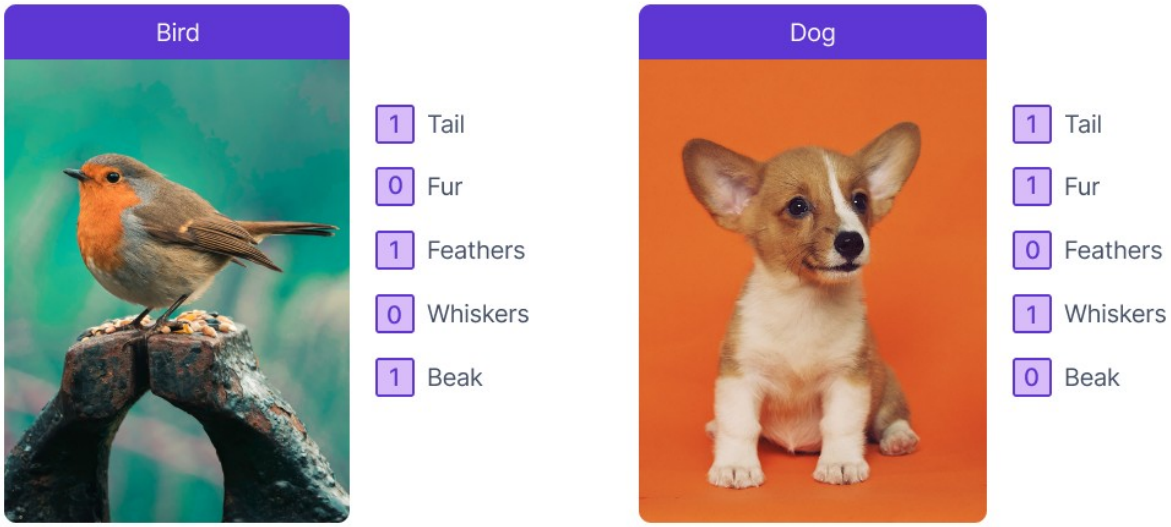
SCHOOL OF DIGITAL TECHNOLOGY

DEPARTMENT OF INFORMATICS AND TELEMATICS

Comparative evaluation of generalized zero-shot learning methods for image classification

Bachelor thesis

Lefteris Dimitriou



Athens, February 2023

ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΣΧΟΛΗ ΨΗΦΙΑΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

Τριμελής Εξεταστική Επιτροπή

Χρήστος Δίου
Επίκουρος Καθηγητής, Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Δημήτριος Μιχαήλ
Αναπληρωτής Καθηγητής, Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Κωνσταντίνος Τσερπές
Αναπληρωτής Καθηγητής, Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Ο Λευτέρης Δημητρίου

δηλώνω υπεύθυνα ότι:

1) Είμαι ο κάτοχος των πνευματικών δικαιωμάτων της πρωτότυπης αυτής εργασίας και από όσο γνωρίζω η εργασία μου δε συκοφαντεί πρόσωπα, ούτε προσβάλει τα πνευματικά δικαιώματα τρίτων.

2) Αποδέχομαι ότι η ΒΚΠ μπορεί, χωρίς να αλλάξει το περιεχόμενο της εργασίας μου, να τη διαθέσει σε ηλεκτρονική μορφή μέσα από τη ψηφιακή Βιβλιοθήκη της, να την αντιγράψει σε οποιοδήποτε μέσο ή/και σε οποιοδήποτε μορφότυπο καθώς και να κρατά περισσότερα από ένα αντίγραφα για λόγους συντήρησης και ασφάλειας.

3) Όπου υφίστανται δικαιώματα άλλων δημιουργών έχουν διασφαλιστεί όλες οι αναγκαίες άδειες χρήσης ενώ το αντίστοιχο υλικό είναι ευδιάκριτο στην υποβληθείσα εργασία.

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου, κ. Χρήστο Δίου, που με την καθοδήγησή του με έκανε να κατανοήσω καλύτερα το θέμα της πτυχιακής μου εργασίας και το τι έπρεπε να κάνω. Επίσης θα ήθελα να ευχαριστήσω τους γονείς μου, τους φίλους μου και όσους άλλους με στήριξαν για να τα καταφέρω.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Περίληψη στα Ελληνικά

Η μηχανική μάθηση πλέον βρίσκεται παντού στην καθημερινότητάς μας, βελτιώνοντας τον τρόπο ζωής μας χωρίς πολλές φορές να το καταλαβαίνουμε. Ένα προϊόν της μηχανικής μάθησης είναι η κατηγοριοποίηση εικόνων. Η κατηγοριοποίηση εικόνων δεν είναι κάτι καινούργιο, όμως τα τελευταία χρόνια έχουν αναπτυχθεί μέθοδοι που δεν χρειάζονται παραδείγματα για να το κάνουν αυτό. Στη παρούσα διπλωματική εργασία θα δούμε βασικούς ορισμούς, παραδείγματα που χρησιμοποιείται η κατηγοριοποίηση εικόνων, πως ξεκίνησαν τα πρώτα νευρωνικά δίκτυα και πως είναι τώρα, και διατυπώνεται το πρόβλημα του Zero Shot Learning. Για την καλύτερη κατανόηση αναλύονται τέσσερις μέθοδοι γενικευμένης εκμάθησης χωρίς καθόλου παραδείγματα, οι CADA-VAE, TF-VAEGAN, CE-GZSL και LFGAA, με κάθε μέθοδο να έχει διαφορετική προσέγγιση στο πρόβλημα κάνοντάς την ξεχωριστή από τις υπόλοιπες. Στη συνέχεια θα γίνουν πειράματα πάνω σε αυτές τις μεθόδους σε τρία διαδεδομένα σύνολα δεδομένων για τέτοια προβλήματα, τα AWA, CUB και SUN. Θα γίνει σύγκριση των επιδόσεων μεταξύ των μεθόδων που επιλέχθηκαν στα πειράματα που πραγματοποιήθηκαν και θα βγουν κάποια συμπεράσματα για αυτά. Τέλος, προτείνονται τρόποι για τη βελτίωση μελλοντικών μεθόδων πάνω σε αυτό το θέμα.

Λέξεις κλειδιά: Κατηγοριοποίηση Εικόνων, Γενικευμένη Εκμάθηση, Αξιολόγηση

Abstract ή Περίληψη στα Αγγλικά

Machine Learning is everywhere in our daily lives, drastically improving it in many ways even when we do not even realise it. Image classification is a product of machine learning. Image classification is not a new concept, however in recent years new methods have been developed that do not require any samples to do that. In this bachelor thesis we will see basic terms of machine learning, how neural networks started and their condition now, and we give the problem definition of Zero Shot Learning. For our better understanding, we analyze four unique methods for Generalized Zero Shot Learning. These are CADA-VAE, TF-VAEGAN, CE-GZSL and LFGAA. We conduct experiments on these four methods and evaluate them with three of the most frequently used datasets for ZSL/GZSL problems, AWA, CUB and SUN. We compare the results from these experiments and make conclusions based on them. Finally, we suggest ways for future GZSL methods to improve.

Keywords: Image Classification, Zero-Shot Learning

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ/ΑΚΡΩΝΥΜΙΑ

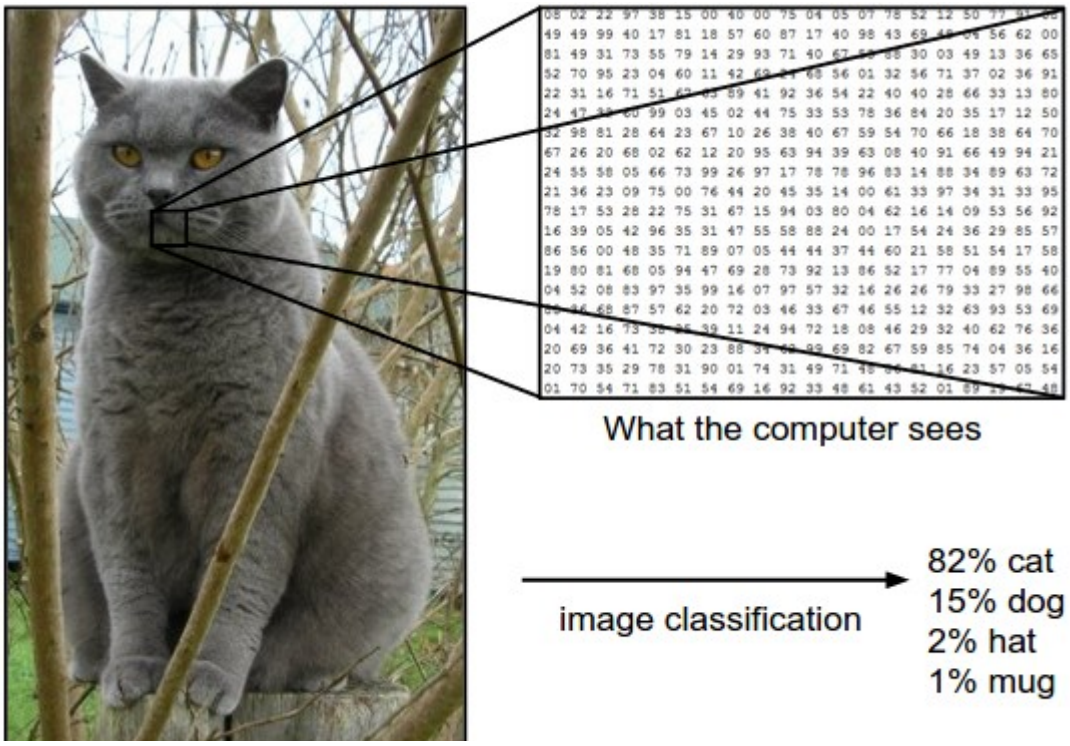
IoT	Internet of Things
AI	Artificial Intelligence
ZSL	Zero Shot Learning
GZSL	Generalized Zero-Shot Learning
NN	Neural Network
ANN	Artificial Neural Network
ADALINE	Adaptive Linear Neuron
MLP	Multilayer Perceptron
CNN	Convolutional Neural Network
ReLU	Rectified Linear Unit
GPU	Graphics Processing Unit
GAN	Generated Adversarial Network
VAE	Variational Autoencoder
VI	Variational Inference
SED	Semantic Embedding Decoder
LFGAA	Latent Feature Guided Attention
LGA	Latent Guided Attention
PS	Proposed Split

Κεφάλαιο 1: Εισαγωγή

1.1 Τι είναι το Image Classification

Στη σημερινή εποχή σχεδόν τα πάντα βρίσκονται σε ψηφιακή μορφή και το Ίντερνετ των πραγμάτων (IoT) και η τεχνητή νοημοσύνη (AI) γίνονται ολοένα και πιο συνηθισμένα στις τεχνολογίες που αναπτύσσουμε. Αυτό έχει ως αποτέλεσμα να παράγεται ένα τεράστιο πλήθος από δεδομένα που πρέπει να διαχειριστούμε και να επεξεργαστούμε. Αυτά τα δεδομένα μπορεί να είναι σε διάφορες μορφές, όπως για παράδειγμα σε μορφή κειμένου, εικόνας, ομιλίας ή ακόμη μπορεί να είναι και συνδυασμός αυτών. [1] Στη παρούσα πτυχιακή εργασία θα εστιάσουμε στην αναγνώριση αντικειμένων σε εικόνες.

Η αναγνώριση αντικειμένων μπορεί σε εμάς τους ανθρώπους να φαίνεται κάτι απλό, για τις μηχανές όμως, όπως έχει αποδειχθεί, αποτελεί μια εξαιρετικά πολύπλοκη διαδικασία. Για αυτό το λόγο το Image Classification θεωρείται και ως ένα από τα κυριότερα προβλήματα του computer vision.



Εικόνα 1: Πως ο υπολογιστής "βλέπει" μια εικόνα
Πηγή Εικόνας: [2]

Σε πρώιμο στάδιο το Image Classification βασίστηκε στην ικανότητα των ηλεκτρονικών υπολογιστών να διασπούν τις εικόνες σε μεμονωμένα pixels. Το πρόβλημα με το παραπάνω είναι ότι δύο εικόνες που απεικονίζουν το ίδιο αντικείμενο μπορεί να φαίνονται τελείως διαφορετικές λόγω διαφορετικού φόντου, οπτικής γωνίας και άλλων παραγόντων. Αυτό

δημιουργεί μια επιπλέον πρόκληση στους ηλεκτρονικούς υπολογιστές στο να αναγνωρίσουν και να κατηγοριοποιήσουν σωστά μια εικόνα. Για την επίλυση αυτού του προβλήματος, τα computer vision μοντέλα πρόσθεσαν νέα χαρακτηριστικά στο pixel data, όπως την ανάλυση χρωμάτων, υφών και σχημάτων. Ακόμη και αυτά όμως έφεραν στην επιφάνεια νέα προβλήματα. [2]

Το Image Classification, συχνά αναφερόμενο και ως Image Recognition, είναι η διαδικασία με την οποία γίνεται η συσχέτιση ενός (single-label classification) ή παραπάνω (multi-label classification) labels σε μια εικόνα.

Στο single-label classification, όπως καταλαβαίνουμε και από την ονομασία , κάθε εικόνα αντιστοιχεί μόνο σε ένα label ή keyword. Συνεπώς η ανάλυση και η κατηγοριοποίηση κάθε εικόνας από το μοντέλο βασίζεται σε ένα μόνο κριτήριο. [3] Για παράδειγμα εικόνες που περιέχουν ακριβώς ένα χειρόγραφο αριθμό από το 0 έως το 9.

Από την άλλη, στο multi-label classification οι εικόνες μπορεί να έχουν πολλαπλά labels. Μπορεί να συναντήσουμε και περιπτώσεις όπου μια εικόνα αντιστοιχεί σε όλα τα labels που χρησιμοποιούμε ταυτόχρονα. Η χρήση του multi-label classification είναι αρκετά διαδεδομένη στον τομέα της ιατρικής, όπου ένας ασθενής μπορεί να διαγνωσθεί σε παραπάνω από μία ασθένειες από δεδομένα στη μορφή ακτινογραφιών. Επιπλέον, μπορούμε να θεωρήσουμε ως multi-label classification την αναγνώριση αντικειμένων που παρουσιάζονται στο φυσικό τους περιβάλλον στις εικόνες. [3]

Εκτός από τα παραπάνω παραδείγματα, το Image Classification έχει και άλλες εφαρμογές και όσο βελτιώνεται και γίνεται όλο και πιο αξιόπιστο, τόσο θα αυξάνονται και οι προοπτικές που θα έχουμε για να το αξιοποιήσουμε ακόμη περισσότερο στο μέλλον. Μερικά παραδείγματα είναι τα αυτόνομα αυτοκίνητα, η αναγνώριση προσώπου, ο αυτόματος έλεγχος ποιότητας προϊόντων και άλλα. [4]

1.2 Τι είναι το Zero Shot Learning

Ένας τρόπος για την κατηγοριοποίηση εικόνων είναι και το Zero Shot Learning (ZSL). Η διαφορά ενός ZSL μοντέλου σε σχέση με τα παραδοσιακά μοντέλα είναι ότι καλείται να αναγνωρίσει αντικείμενα για τα οποία δεν έχει εκπαιδευτεί από πριν. Κάτι τέτοιο είναι χρήσιμο και έχει αρκετές πρακτικές. [5]

Μια εφαρμογή του ZSL είναι στη περίπτωση που έχουμε την εμφάνιση νέων κλάσεων στα δεδομένα μας μετά την εκπαίδευση του μοντέλου. Αυτό μπορεί να συμβεί όταν δουλεύουμε με κλάσεις που συνεχώς αυξάνονται σε μέγεθος. Ένα τέτοιο παράδειγμα είναι αν δουλεύουμε πάνω στην καταγραφή των ζωντανών οργανισμών και εντοπίσουμε ένα νέο είδος. Υπάρχει όμως και η τελείως αντίθετη περίπτωση που ένα ZSL μοντέλο μας είναι χρήσιμο και αυτή είναι όταν ψάχνουμε κάτι πολύ συγκεκριμένο, σε σημείο που μας καθιστά αδύνατο να έχουμε κάποιο στοιχείο ή τα διαθέσιμα στοιχεία είναι ελάχιστα οπότε δεν επαρκούν για να εκπαιδεύσουμε το μοντέλο μας και να το αναγνωρίσει μετά με επιτυχία. [6] Ένα επίκαιρο παράδειγμα σε αυτό είναι η διάγνωση του COVID-19 από ακτινογραφίες.

Το παραπάνω αποτελεί και το βασικό πρόβλημα του ZSL. Η προσέγγιση για την επίλυση αυτού του προβλήματος είναι παρόμοια και με τη διαδικασία όπου ένας άνθρωπος μπορεί να αναγνωρίσει ένα αντικείμενο που δεν έχει ξαναδεί, απλά έχοντας διαβάσει πληροφορίες για αυτό. Δηλαδή, διαβάζει μια λεπτομερής περιγραφή με τα χαρακτηριστικά του νέου αντικειμένου και στη συνέχεια συγκρίνει τις ομοιότητες με αντικείμενα που έχει μάθει από πριν. [6]

Συνεπώς, το Zero Shot Learning είναι μια διαδικασία που αποτελείται από δύο στάδια. Το πρώτο στάδιο είναι η εκπαίδευση, όπου συλλέγεται η γνώση για τα γνωρίσματα του αντικειμένου και το δεύτερο στάδιο είναι η συμπερασματολογία, όπου γίνεται η κατηγοριοποίηση του αντικειμένου σε ένα νέο σετ από κλάσεις. Συνήθως οι περισσότερες προσπάθειες για βελτίωση γίνονται στο πρώτο στάδιο. [6]

1.3 Διατύπωση του προβλήματος

Στο ZSL, έχουμε δύο ανεξάρτητες κλάσεις: S, όπου συμβολίζουμε τις κλάσεις που μας είναι γνωστές στο Y_s και *U*, όπου συμβολίζουμε τις κλάσεις που μας είναι άγνωστες στο Y_u, και που ισχύει $Y_s \cap Y_u = \emptyset$. Αν υποθέσουμε ότι έχουμε διαθέσιμες *N* κατηγοριοποιημένες περιπτώσεις από τις γνωστές κλάσεις Ys για να εκπαιδεύσουμε το μοντέλο μας τότε: $D_{tr} = \{(x_1, y_2), \ldots, (x_N, y_N)\}$, όπου το $x_i \in X$ δηλώνει τη περίπτωση και το $y_i \in Y_s$ είναι η αντίστοιχη ετικέτα από τις γνωστές κλάσεις. Το test set $D_{te} = \{x_{N+1}, \ldots, x_{N+M}\}$ περιέχει M περιπτώσεις χωρίς ετικέτες. Στο συνηθισμένο ZSL, όλες οι περιπτώσεις στο D_{te} προέρχονται μόνο από τις άγνωστες κλάσεις.

Στο πιο απαιτητικό Generalized Zero-Shot Learning (GZSL), οι περιπτώσεις στο D_{te} προέρχονται και από τις γνωστές, αλλά και από τις άγνωστες κλάσεις. Ταυτόχρονα, η σημασιολογική περιγραφή σε επίπεδο κλάσης των γνωστών και άγνωστων κλάσεων δίνεται από το $A = \{a^1, \ldots, a^s, a^{S+1}, \ldots, a^{S+U}\}$, όπου η σημασιολογική περιγραφή του πρώτου *S* αντιστοιχεί στις γνωστές κλάσεις στο Y_s και η σημασιολογική περιγραφή του τελευταίου *U* αντιστοιχεί στις άγνωστες κλάσεις στο Y_u. Μπορούμε να συμπεράνουμε τη σημασιολογική περιγραφή α για μια κατηγοριοποιημένη περίπτωση x από τις ετικέτες για τις κλάσεις y. [7]

1.4 Δομή Πτυχιακής Εργασίας

Η παρούσα πτυχιακή εργασία με θέμα "Εκπαίδευση μοντέλων αναγνώρισης αντικειμένων σε εικόνες με λίγα ή καθόλου παραδείγματα" αποτελείται από πέντε κεφάλαια.

Στο πρώτο κεφάλαιο εξετάσαμε γενικότερα το πρόβλημα της κατηγοριοποίησης εικόνων και είδαμε γιατί μας είναι χρήσιμη, καθώς και εφαρμογές της στην καθημερινή μας ζωή με παραδείγματα. Επίσης, είδαμε ειδικότερα για το πρόβλημα της εκμάθησης χωρίς παραδείγματα και τη διατύπωση του προβλήματος του ZSL και του GZSL.

Στο δεύτερο κεφάλαιο θα δούμε γενικά τι είναι τα νευρωνικά δίκτυα και τα συνελικτικά νευρωνικά δίκτυα και θα εξετάσουμε κάποια από τα βασικά μοντέλα για την κατηγοριοποίηση

των εικόνων και κάποια συναφή σύνολα δεδομένων. Σε αυτό το κεφάλαιο θα γίνει και μια βιβλιογραφική επισκόπηση γύρω από το πρόβλημα του ZSL και του GZSL.

Στο τρίτο κεφάλαιο θα γίνει πιο αναλυτική παρουσίαση τεσσάρων επιλεγμένων μεθόδων από τη βιβλιογραφία.

Στο τέταρτο κεφάλαιο θα εκτελέσουμε τις μεθόδους που επιλέχθηκαν στο 3ο κεφάλαιο και θα αναλύσουμε και θα συγκρίνουμε τα αποτελέσματά τους.

Στο πέμπτο κεφάλαιο θα δούμε τι συμπεράσματα προκύπτουν από τα πειράματα και τι βήματα μπορούν να γίνουν στο μέλλον πάνω σε αυτό το θέμα.

Στο τέλος υπάρχει ένα παράρτημα με τις τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της πτυχιακής εργασίας και η βιβλιογραφία.

Κεφάλαιο 2: Θεωρητικό Υπόβαθρο και Βιβλιογραφική Επισκόπηση

2.1 Μηχανική Μάθηση

Τι είναι η μηχανική μάθηση:

Η μηχανική μάθηση (Machine Learning – ML) είναι ένας κλάδος της τεχνητής νοημοσύνης (Artificial Intelligence – AI) και του computer science που με τη χρήση δεδομένων και αλγορίθμων προσπαθεί να μιμηθεί τον τρόπο με τον οποίο μαθαίνουν οι άνθρωποι και σταδιακά να βελτιώνει την ακρίβειά της [8].

Πολλές πρωτοπόρες εταιρείες της εποχής μας, όπως η Google και το Facebook, έχουν κάνει τη μηχανική μάθηση ένα σημαντικό παράγοντα για τις επιχειρήσεις τους [9].

Οι κατηγορίες μηχανικής μάθησης:

Οι τέσσερις βασικές κατηγορίες της μηχανικής μάθησης είναι η μάθηση με επίβλεψη (supervised learning), η μάθηση χωρίς επίβλεψη (unsupervised learning), η μάθηση με μερική επίβλεψη (semi-supervised learning) και η ενισχυτική μάθηση (reinforcement learning)[9].

Μάθηση με επίβλεψη: Στη μάθηση με επίβλεψη τα δεδομένα είναι προεπισημασμένα. Το μοντέλο εκπαιδεύεται χρησιμοποιώντας αυτά τα γνωστά δεδομένα, οπότε στο τέλος κάθε επανάληψης γνωρίζει που έκανε λάθη και κάνει τις κατάλληλες ενέργειες ώστε να βελτιώνεται κάθε φορά.

Μάθηση χωρίς επίβλεψη: Στη μάθηση χωρίς επίβλεψη τα δεδομένα δεν έχουν ετικέτες. Το μοντέλο καλείται να αναγνωρίσει και να κατηγοριοποιήσει σωστά την εικόνα από μόνο του. Η επιτυχία του μοντέλου σε αυτό καθορίζεται από το αν το δίκτυο κατάφερε να μειώσει ή να αυξήσει την συνάρτηση κόστους (cost function).

Μάθηση με μερική επίβλεψη: Είναι ένας συνδυασμός των δύο παραπάνω κατηγοριών. Τα περισσότερα δεδομένα για την εκπαίδευση του μοντέλου είναι προεπισημασμένα, το μοντέλο όμως είναι ελεύθερο να εξερευνήσει τα δεδομένα και να σχηματίσει από μόνο του μια ιδέα για το σύνολο των δεδομένων.

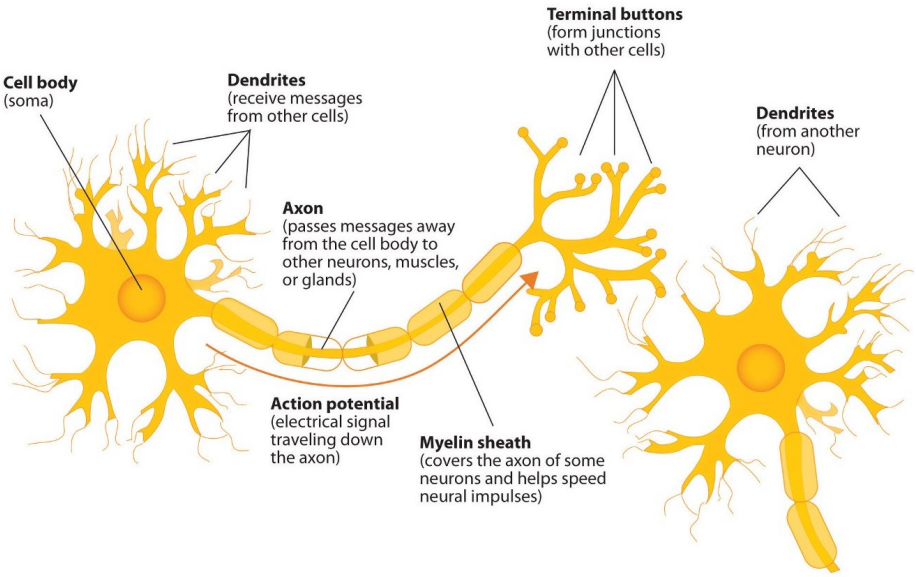
Ενισχυτική μάθηση: Με την ενισχυτική μάθηση μία μηχανή μαθαίνει να ολοκληρώνει μια διαδικασία πολλαπλών βημάτων για την οποία έχουν τεθεί σαφείς κανόνες. Κατά την διάρκεια αυτής της διαδικασίας, ο αλγόριθμος ανάλογα με τις ενέργειες που κάνει για να την ολοκληρώσει, δέχεται θετικά ή αρνητικά ερεθίσματα. Με αυτόν τον τρόπο ενισχύεται, γίνεται καλύτερος.

Παρόλα αυτά, είναι σημαντικό να τονίσουμε πως τις περισσότερες φορές τα μοντέλα που ασχολούνται με την αναγνώριση μοτίβων σε εικόνες συνήθως βασίζονται για την κατηγοριοποίηση στη χρήση της μάθησης με επίβλεψη.

2.2 Νευρωνικά Δίκτυα

<u>Τι είναι τα νευρωνικά δίκτυα</u>:

Τα νευρωνικά δίκτυα (Neural Networks – NN), ή αλλιώς Artificial Neural Networks (ANN), είναι ένα υποσύνολο της μηχανικής μάθησης (machine learning – ML) και αποτελούν το Α και το Ω για τους αλγόριθμους βαθιάς μάθησης (deep learning – DL). Όπως μπορούμε να καταλάβουμε και από την ονομασία τους, τα νευρωνικά δίκτυα είναι άμεσα συνδεδεμένα με τον ανθρώπινο εγκέφαλο και αυτό γιατί έχουν σχεδιαστεί με βάση αυτόν και λειτουργούν όπως ακριβώς επικοινωνούν μεταξύ τους οι νευρώνες ενός εγκεφάλου. [10–12]



Εικόνα 2: Νευρώνας ενός εγκεφάλου
Πηγή Εικόνας:[10]

Συνήθως ένα ANN δομείται από εκατοντάδες απλές υπολογιστικές μονάδες ή κόμβους οι οποίοι είναι συνδεδεμένοι μεταξύ τους δημιουργώντας έτσι ένα περίπλοκο επικοινωνιακό δίκτυο. Οι κόμβοι διαχωρίζονται σε επίπεδα. Αυτά τα επίπεδα είναι ένα input layer, ένα ή περισσότερα hidden layers και ένα output layer. Κάθε κόμβος αντιπροσωπεύει μια απλοποιημένη μορφή ενός πραγματικού νευρώνα που στέλνει ένα νέο σήμα ή ενεργοποιείται αν δεχτεί κάποιο σήμα από ένα άλλο κόμβο με τον οποίο είναι συνδεδεμένος, διαφορετικά δεν μεταφέρεται καμία πληροφορία.

Τα ANN εξαρτώνται από τα δεδομένα με τα οποία γίνεται η εκπαίδευσή τους για να μάθουν και βελτιώνουν την απόδοσή τους με τη πάροδο του χρόνου. Παρόλα αυτά όταν ολοκληρώσουν την εκπαίδευσή τους αποτελούν ισχυρά εργαλεία λόγω του ότι μπορούν να συλλέγουν και να κατηγοριοποιούν δεδομένα με πολύ γρήγορο ρυθμό, εξοικονομώντας μας έτσι πολύτιμο χρόνο.

Τα νευρωνικά δίκτυα είναι ευρέως διαδεδομένα στις επιχειρήσεις, στην εκπαίδευση, στην οικονομία, στην υγεία, αλλά χρησιμοποιούνται και γενικότερα για την επίλυση άλλων καθημερινών μας προβλημάτων. Τα NN διαπρέπουν στο να αναγνωρίζουν τάσεις και μοτίβα, για αυτό και χρησιμοποιούνται στο να κάνουν προβλέψεις. Πιο συγκεκριμένα χρησιμοποιούνται για την αναγνώριση προσώπων, για την αναγνώριση γραφικών χαρακτήρων, για την ανάλυση ιατρικών εξετάσεων και τη διάγνωση ασθενειών, για προβλέψεις στις τάσεις στις αγορές αλλά και στα social media, για τη πρόγνωση του καιρού και άλλα. [13,14]

Η ιστορία των νευρωνικών δικτύων:

Προτού αναλύσουμε περισσότερο πως λειτουργεί ένα νευρωνικό δίκτυο, νομίζω έχει ενδιαφέρον να δούμε συνοπτικά πως ξεκίνησαν, αλλά και κάποια κύρια γεγονότα που έπαιξαν σημαντικό ρόλο στη διαμόρφωσή τους για να φτάσουν στο σημείο που βρίσκονται σήμερα. [15]

Το 1943 ο Warren S. McCulloch και ο Walter Pitts δημοσίευσαν την έρευνά τους με τίτλο "A logical calculus of the ideas immanent in nervous activity". Σκοπός τους ήταν να κατανοήσουν πως ο ανθρώπινος εγκέφαλος είναι ικανός να παράγει πολύπλοκα μοτίβα χάρις τους νευρώνες του. Μια ιδέα που αναδείχθηκε ήταν να συγκρίνουν τη λειτουργία των νευρώνων σε λογική boolean, δηλαδή σε 0 ή 1 και true ή false statements.

Το 1957 ο Frank Rosenblatt στηρίχθηκε στην έρευνα του Warren S. McCulloch και του Walter Pitts και την εξέλιξε περισσότερο προσθέτοντας βάρη (weights) στην εξίσωση. Έτσι δημιούργησε το perceptron, όπου και κατέγραψε στη δική του έρευνα με τίτλο "The Perceptron: A Probabillistic Model for Information Storage and Organization in the Brain". Ο Frank Rosenblatt ήταν ο πρώτος που κατάφερε να εκπαιδεύσει έναν ηλεκτρονικό υπολογιστή να ξεχωρίζει με επιτυχία ποιες κάρτες είναι σημαδεμένες στα αριστερά και ποιες στα δεξιά.

Το 1960 ο Widrow B. και ο Hoff M. E. εφεύρεσαν ένα διαφορετικό νευρωνικό μοντέλο, το ADALINE (Adaptive Linear Neuron). Η διαφορά του ADALINE με ένα perceptron είναι ότι άλλαξε τον τρόπο με τον οποίο υπολογίζονται τα βάρη από το μοντέλο χρησιμοποιώντας λογική, κάνοντάς το πιο πρακτικό.

Το 1969 ο Minsky M. L. και ο Pappert S. A. στο βιβλίο τους με τίτλο "Perceptrons" κατέκριναν τα perceptrons και τα ADALINE μοντέλα διότι δεν μπορούν να λύσουν λογικά προβλήματα τύπου XOR. Αυτό είχε ως αποτέλεσμα να επηρεάσει αρκετό κόσμο και να τον κάνει να χάσει το ενδιαφέρον του στη διερεύνηση των NN, φέρνοντας έτσι τον πρώτο "AI Winter".

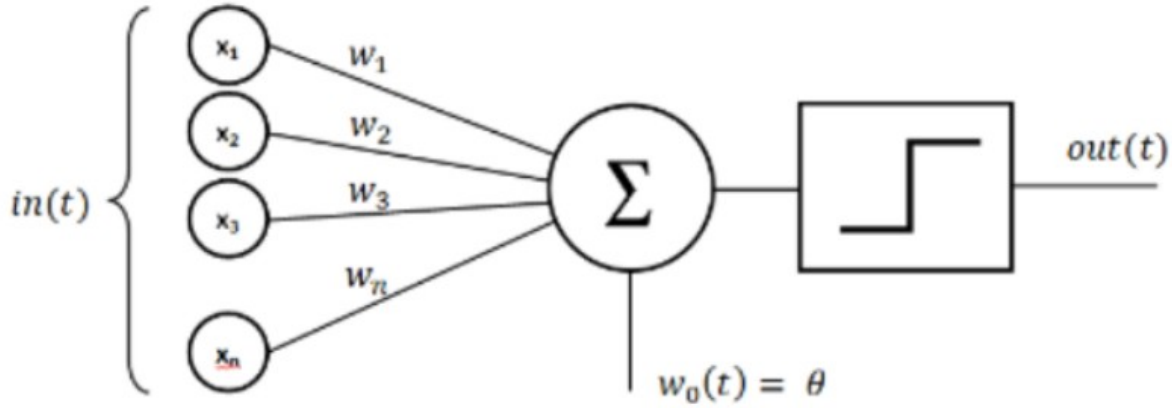
Το 1974 ο Paul Werbos ανέπτυξε τη μέθοδο μάθησης backpropagation. Παρόλα αυτά αυτή η μέθοδος δεν αξιοποιήθηκε μέχρι το 1986 όπου ο Rumelhart et al την ανέδειξε και πάλι με το βιβλίο του "Learning Internal Representation by Error Propagation". Το backpropagation είναι μια μορφή αλγορίθμου gradient descent που χρησιμοποιείται με τα ANN για την ελαχιστοποίηση των λαθών.

Perceptrons:

Στην αρχή του κεφαλαίου περιγράψαμε τα νευρωνικά δίκτυα ως ένα περίπλοκο επικοινωνιακό σύστημα κατασκευασμένο από εκατοντάδες κόμβους χωρισμένους σε επίπεδα και που συνδέονται μεταξύ τους. Κάθε ένας από αυτούς τους κόμβους αποτελεί και ένα perceptron. Η ονομασία του προέρχεται από την ικανότητα του ανθρώπου να αντιλαμβάνεται (perception), δηλαδή να βλέπει και να αναγνωρίζει εικόνες. Το perceptron είναι ένα πολύ απλό μοντέλο νευρωνικού δικτύου που χρησιμοποιείται για τη δυαδική ταξινόμηση (Binary

Classification). Αν και σήμερα το θεωρούμε ως έναν αλγόριθμο, στην εποχή του προοριζόταν ως μια μηχανή αναγνώρισης εικόνων. [12,16]

Ένα perceptron αποτελείται από τέσσερα μέρη. Τις τιμές των inputs, τα weights και το bias, το weighted sum και το activation function. Το weighted sum είναι και η κύρια διαφορά αυτού του μοντέλου σε σύγκριση με τα προηγούμενα.



Εικόνα 3: Perceptron
Πηγή Εικόνας: [16]

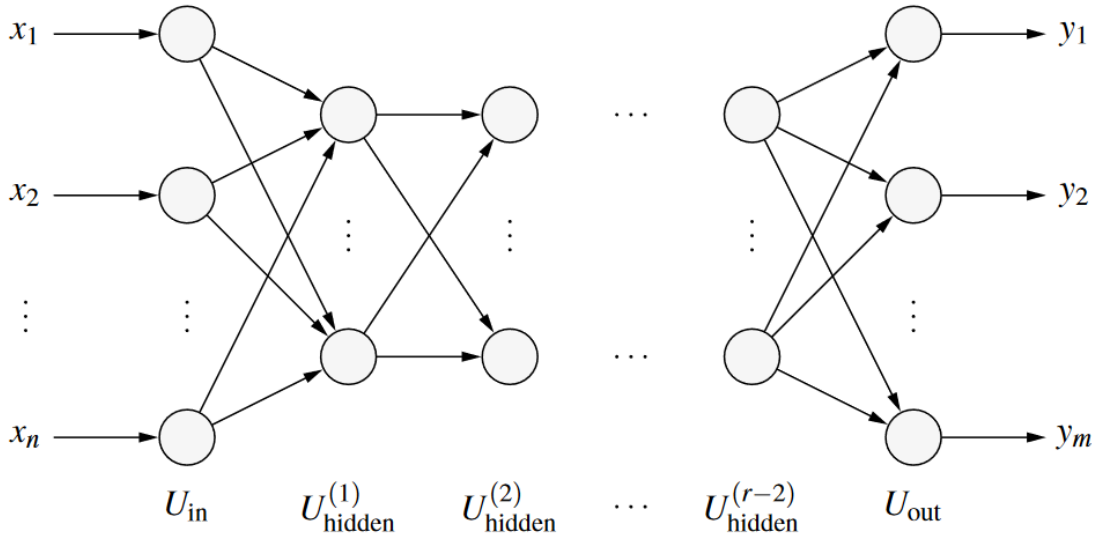
Αρχικά ένα perceptron δέχεται αριθμητικά δεδομένα (input values), το καθένα με το δικό του βάρος (weight), και το bias. Στη συνέχεια πολλαπλασιάζει τις τιμές των inputs με τα αντίστοιχα βάρη τους και προσθέτει τα γινόμενα μεταξύ τους, υπολογίζοντας έτσι το weighted sum. Έπειτα στο weighted sum προστίθεται το bias. Το άθροισμα αυτών των δύο πηγαίνει στο activation function όπου και υπολογίζεται το τελικό αποτέλεσμα. Για τον υπολογισμό των βαρών που θα δώσουν το βέλτιστο δυνατό αποτέλεσμα το perceptron χρησιμοποιεί Stochastic Gradient Descent.

Multilayer Perceptrons:

Όπως είδαμε και στη σύντομη αναφορά της ιστορίας των νευρωνικών δικτύων, τα perceptrons έχουν ένα σημαντικό περιορισμό και αυτός είναι ότι δεν μπορούν να λύσουν προβλήματα λογικού τύπου XOR. Για την αντιμετώπιση αυτού του περιορισμού δημιουργήθηκαν τα Multilayer Perceptrons (MLP). [10,17]

Το MLP θεωρείται ως ο πιο χρήσιμος τύπος νευρικών δικτύων. Απαρτίζεται από ένα input layer, ένα ή περισσότερα hidden layers και ένα output layer. Κάθε επίπεδο περιέχει ένα σύνολο από πολλούς νευρώνες. Το MLP ανήκει στην κατηγορία των feedforward αλγορίθμων, αφού κάθε επίπεδο, από το πρώτο hidden layer μέχρι και το output layer, διαδίδει πληροφορίες στο επόμενο και το αντίθετο δεν είναι εφικτό.

Οι υπολογισμοί που γίνονται σε ένα νευρώνα στο MLP μοντέλο είναι οι ίδιοι με αυτούς που γίνονται στο απλό perceptron, αλλά με μια σημαντική διαφορά. Κάθε γραμμικός συνδυασμός διαδίδεται στο επόμενο επίπεδο. Επειδή η πληροφορία προχωράει πάντα προς τα μπροστά, όταν φτάσει στο τελευταίο επίπεδο που είναι το output layer, θα σταματήσει εκεί. Αν ο αλγόριθμος τρέχει μόνο για μια επανάληψη, τότε δεν έχει τη δυνατότητα να υπολογίσει τις βέλτιστες τιμές, δηλαδή δεν μαθαίνει. Με το backpropagation ξεπερνούμε αυτό το εμπόδιο.



Εικόνα 4: Multi-layer Perceptron r-επιπέδων
Πηγή Εικόνας:[17]

Backpropagation:

Το backpropagation είναι ένας μηχανισμός μάθησης που επιτρέπει στο MLP να προσαρμόζει με κάθε επανάληψή του τα βάρη και το bias κάθε κόμβου με σκοπό την ελαχιστοποίηση της συνάρτησης κόστους (cost function). Ίσως είναι και ο θεμελιώδης αλγόριθμος για την κατασκευή ενός νευρωνικού δικτύου. [12,18]

Το backpropagation εκπαιδεύει αποτελεσματικά ένα νευρωνικό δίκτυο μέσω της μεθόδου της αλυσίδας (chain rule). Με αυτή τη μέθοδο, κάθε φορά που ολοκληρώνεται μια επανάληψη, γίνεται η διάσχιση του μοντέλου, αυτή τη φορά από το τελευταίο επίπεδο προς το αρχικό επίπεδο, αλλάζοντας σε κάθε κόμβο που περνά τις τιμές του βάρους του και του bias του, βελτιώνοντας με κάθε τέτοιο πέρασμα την απόδοσή του.

2.2 Συνελικτικά Νευρωνικά Δίκτυα

Τι είναι τα συνελικτικά νευρωνικά δίκτυα:

Τα συνελικτικά νευρωνικά δίκτυα (Convolutional Neural Networks – CNN) είναι παρόμοια με τα απλά νευρωνικά δίκτυα υπό την έννοια ότι και αυτά αποτελούνται από

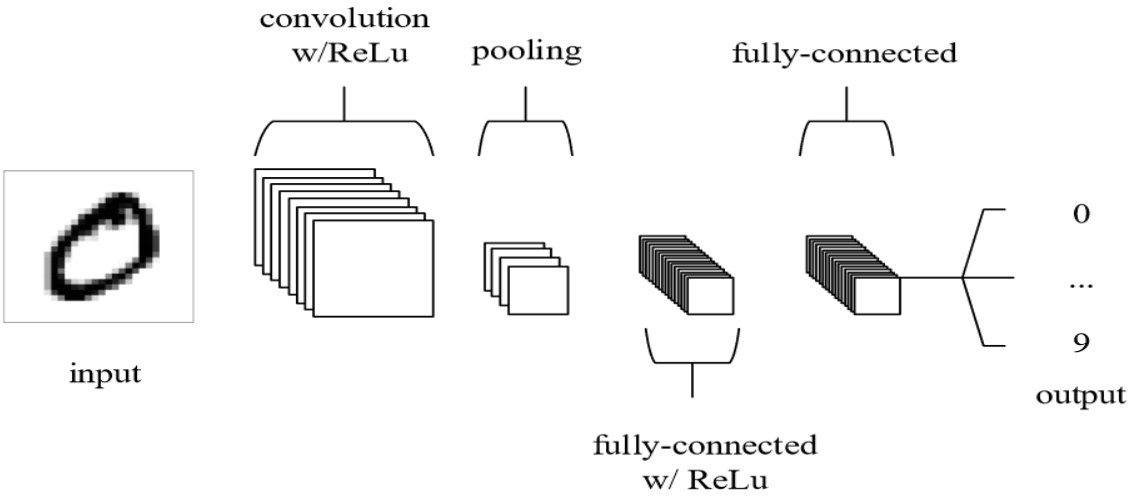
νευρώνες που βελτιώνονται μόνοι τους μέσω της εκπαίδευσής τους. Και τα CNN στηρίζονται στη βασική αρχή των περισσότερων ANN. [19]

Η μόνη διαφορά των CNN από τα ANN είναι ότι τα πρώτα χρησιμοποιούνται κυρίως για την αναγνώριση μοτίβων στις εικόνες. Αυτό μας επιτρέπει να έχουμε στην αρχιτεκτονική ενός συνελικτικού νευρωνικού δικτύου δυνατότητες αποκλειστικές για την επεξεργασία εικόνων, ενώ ταυτόχρονα μειώνονται οι παράμετροι που χρειάζονται για το στήσιμο του μοντέλου. Τα CNN μοντέλα εκπαιδεύονται με τη χρήση της μεθόδου του backpropagation, όπως γίνεται και στα ANN.

Το μεγαλύτερο μειονέκτημα ενός παραδοσιακού νευρωνικού δικτύου είναι ότι δυσκολεύεται να κάνει τους περίπλοκους υπολογισμούς που χρειάζονται για την ανάλυση των δεδομένων της εικόνας. Αυτό, σε βάσεις δεδομένων όπως το πολύ γνωστό MNIST database που περιέχει ασπρόμαυρες εικόνες χαμηλής ανάλυσης από χειρόγραφους αριθμούς δεν δημιουργεί κάποιο πρόβλημα στα ANN. Όταν όμως έχουμε να κάνουμε με πιο περίπλοκες εικόνες, τότε υπάρχει η ανάγκη για CNN.

<u>Η αρχιτεκτονική των συνελικτικών νευρωνικών δικτύων</u>:

Τα CNN αποτελούνται από τρεις τύπους επιπέδων και είναι το συνελικτικό επίπεδο (convolutional layer), το επίπεδο ομαδοποίησης (pooling layer) και το πλήρες συνδεδεμένο επίπεδο (fully connected layer). [20,21]



Εικόνα 5: Η αρχιτεκτονική ενός απλού CNN με πέντε στάδια
Πηγή Εικόνας:[20]

Όπως βλέπουμε και από την παραπάνω εικόνα, μπορούμε να ξεχωρίσουμε τις βασικές λειτουργίες ενός CNN στα εξής μέρη:

• Όπως συναντάμε και σε άλλες μορφές ANN, το input layer περιέχει τις τιμές των πίξελ της εικόνας.

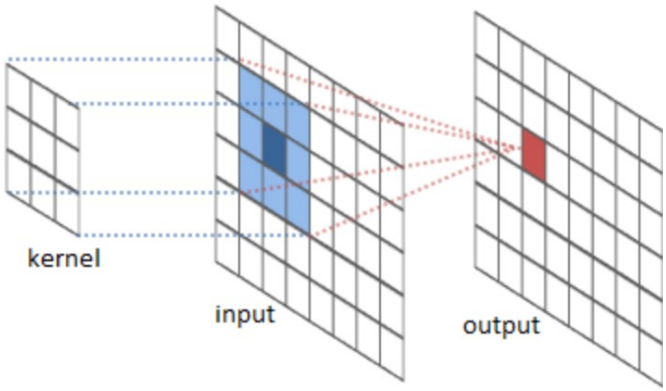
• Το συνελικτικό επίπεδο περιέχει ένα σετ από φίλτρα που εξάγουν τα βασικά χαρακτηριστικά της εικόνας. Συνήθως το μέγεθος αυτών των φίλτρων είναι μικρότερο από αυτό της εικόνας. Η ανορθωμένη γραμμική μονάδα (Rectified Linear Unit – ReLU) προσθέτει στο αποτέλεσμα που παράχθηκε από το προηγούμενο επίπεδο μια συνάρτηση που ενεργοποιείται για κάθε στοιχείο ξεχωριστά, όπως είναι η σιγμοειδής συνάρτηση (sigmoid function).

• Στο επίπεδο ομαδοποίησης η εικόνα θα απλουστευθεί ακόμα περισσότερο, μειώνοντας και άλλο το πλήθος των παραμέτρων στα πλαίσια αυτής της εκτέλεσης.

• Τέλος στο πλήρες συνδεδεμένο επίπεδο θα γίνει η ίδια διαδικασία που γίνεται και στα ANN για να παραχθούν τα αποτελέσματα των κλάσεων και να χρησιμοποιηθούν για την κατηγοριοποίηση. Συνιστάται να χρησιμοποιηθεί μια ReLU μεταξύ αυτών των επιπέδων για να βελτιωθεί η απόδοση.

<u>Το συνελικτικό επίπεδο</u>

Το συνελικτικό επίπεδο μετατρέπει την εικόνα με σκοπό να εξάγει τα χαρακτηριστικά της. Κατά τη διάρκεια αυτής της επεξεργασίας η εικόνα συνελίσεται με ένα kernel (ή φίλτρο). [22]



Εικόνα 6: Συνέλιξη Εικόνας
Πηγή Εικόνας:[22]

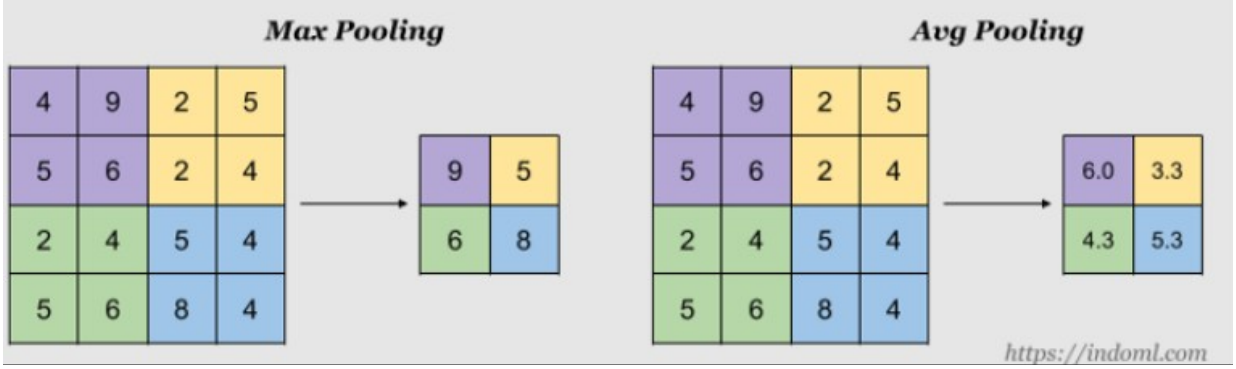
Το kernel είναι ένας πίνακας μικρότερος σε διαστάσεις (ύψος-πλάτος) από την εικόνα. Αυτό το φίλτρο ολισθαίνει κατά το μήκος και το πλάτος της εικόνας και για κάθε θέση υπολογίζει το εσωτερικό γινόμενο του φίλτρου και της εικόνας. Το μήκος με το οποίο κινείται μετατοπίζεται το φίλτρο ονομάζεται δρασκελισμός (stride).

Στη συνέλιξη ο αριθμός των καναλιών στο φίλτρο πρέπει να είναι ίδιος με τον αριθμό των καναλιών της εικόνας, δηλαδή για μια έγχρωμη εικόνα τριών καναλιών, το φίλτρο πρέπει να έχει επίσης τρία κανάλια. Μπορούμε να χρησιμοποιήσουμε πολλαπλά φίλτρα για να εξάγουμε παραπάνω από ένα χαρακτηριστικό από μια εικόνα. Σε αυτή τη περίπτωση όλα τα φίλτρα πρέπει να έχουν τις ίδιες διαστάσεις. Τα συνελιγμένα χαρακτηριστικά της εικόνας στοιβάζονται το ένα πάνω στο άλλο για να δημιουργήσουν ένα output με αριθμό καναλιών ίσο με των αριθμό των φίλτρων που χρησιμοποιήθηκαν.

Η συνάρτηση ενεργοποίησης (activation function) είναι το τελευταίο συστατικό του συνελικτικού επιπέδου για να κάνει πιο μη-γραμμικά τα αποτελέσματα. Συνήθως για αυτή τη διαδικασία χρησιμοποιείται η συνάρτηση ανορθωμένης γραμμικής μονάδας (ReLU function) ή η συνάρτηση tanh (tanh function). Το αποτέλεσμα που παράγεται από αυτό λέγεται πίνακας χαρακτηριστικών (feature map).

Το επίπεδο ομαδοποίησης

Σε αυτό το επίπεδο μειώνεται το μέγεθος της εικόνας. Σε ένα CNN, συνήθως, το επίπεδο ομαδοποίησης έρχεται μετά το συνελικτικό επίπεδο. Το επίπεδο ομαδοποίησης επιταχύνει τους υπολογισμούς και κάνει τα χαρακτηριστικά που έχουν εντοπιστεί πιο έντονα/ευδιάκριτα. (robust) Και το επίπεδο ομαδοποίησης, όπως το συνελικτικό επίπεδο, χρησιμοποιεί kernel και stride. Υπάρχουν πολλά διαφορετικά είδη ομαδοποίησης με πιο συνηθισμένα σε ένα CNN να είναι αυτά του Max Pooling και του Average Pooling. Στο Max Pooling επιλέγεται η μεγαλύτερη τιμή από κάθε κομμάτι του χάρτη χαρακτηριστικών (feature map) για να φτιαχτεί ο νέος μειωμένος χάρτης (reduced map), ενώ στο Average Pooling υπολογίζεται ο μέσος όρος κάθε κομματιού.[22]



Εικόνα 7: Παράδειγμα Max Pooling και Average Pooling
Πηγή Εικόνας: [22]

Το πλήρες συνδεδεμένο επίπεδο

Το πλήρες συνδεδεμένο επίπεδο είναι το τελευταίο επίπεδο ενός CNN. Ο χάρτης χαρακτηριστικών που έχει δημιουργηθεί από τα προηγούμενα επίπεδα γίνεται επίπεδος (flattened) για να πάρει τη μορφή διανύσματος (vector). Αυτό το διάνυσμα πηγαίνει στο πλήρες συνδεδεμένο επίπεδο όπου εκεί συλλέγονται οι περίπλοκες σχέσεις μεταξύ των χαρακτηριστικών υψηλού επιπέδου. Από αυτή τη διαδικασία παράγεται ένα μονοδιάστατο διάνυσμα χαρακτηριστικών. [22]

2.3 Μοντέλα Κατηγοριοποίησης Εικόνων

Υπάρχουν πολλά μοντέλα για την κατηγοριοποίηση εικόνων. Τα μοντέλα κατηγοριοποίησης εικόνων μπορεί να διαφέρουν μεταξύ τους ως προς την επίδοσή τους, την ακρίβεια και το μέγεθός τους. Το πως θα επιλέξουμε κάποιο από αυτά εξαρτάται από την περίπτωσή μας. Για παράδειγμα θα επιλέξουμε ένα γρήγορο μοντέλο για να φτιάξουμε ένα

barcode scanner, ενώ σε άλλες πιο σημαντικές περιπτώσεις μπορεί να προτιμήσουμε ένα πιο αργό αλλά υψηλότερης ακριβείας μοντέλο. [23]

Παρακάτω θα δούμε κάποια από τα πιο γνωστά μοντέλα που χρησιμοποιούνται για την κατηγοριοποίηση εικόνων με λίγα λόγια για το καθένα από αυτά.

AlexNet: Το AlexNet έφερε αρκετές καινοτομίες που αργότερα χρησιμοποιήθηκαν και από τα μοντέλα που το ακολούθησαν. Ήταν το πρώτο μοντέλο που χρησιμοποίησε ReLU αντί για τη σιγμοειδή συνάρτηση που χρησιμοποιούσαν τα υπόλοιπα μοντέλα της εποχής του. Επίσης, ήταν το πρώτο μοντέλο που επέτρεψε την χρήση δύο GPU για την εκπαίδευσή του, που σημαίνει ότι ένα πολύ μεγαλύτερο μοντέλο μπορεί να εκπαιδευτεί, ενώ ταυτόχρονα μειώνεται ο χρόνος που χρειάζεται για να γίνει αυτό. Το AlexNet είχε μεγάλο πρόβλημα με το overfitting. Για την αντιμετώπιση αυτού του προβλήματος χρησιμοποιεί τεχνικές όπως το data augmentation και το dropout. [24]

VGG: Είναι ένα από τα καλύτερα, αν όχι το καλύτερο, μοντέλο κατηγοριοποίησης εικόνων. Βασίζεται στην αρχιτεκτονική του AlexNet, όμως δίνει έμφαση και σε ένα άλλο σημαντικό στοιχείο των CNN που είναι το βάθος (depth). Το VGG υποστηρίζει μέχρι και 19 επίπεδα. Οι πιο γνωστές εκδόσεις του VGG είναι το VGG-16 και VGG-19, όπου ο αριθμός στην ονομασία τους υποδηλώνει των αριθμό των επιπέδων που έχουν. [25,26]

Inception: Το Inception είναι ένα μοντέλο της Google. Αρχικά ονομαζόταν GoogleNet. Είναι πολύ μικρότερο από τα μοντέλα VGG και AlexNet, παρόλα αυτά έχει μικρή πιθανότητα λάθους. Αυτό που το έκανε να ξεχωρίζει το Inception είναι το Inception Moule που πραγματοποιεί συνελίξεις με διαφορετικά μεγέθη φίλτρων στις εισαγωγές (στο input), πραγματοποιεί Max Pooling και συνενώνει το αποτέλεσμα για το επόμενο Inception Moule. Με αυτή τη διαδικασία οι παράμετροι μειώνονται δραστικά. [27]

ResNet: Η ονομασία του προέρχεται από τις λέξεις Residual Networks. Το ResNet μπορεί να εκπαιδεύει εκατοντάδες ή ακόμη και χιλιάδες επίπεδα χωρίς να χάνει την απόδοσή του. Το ξεχωριστό σε αυτό το μοντέλο είναι ότι αντιμετωπίζει το πρόβλημα του overfitting παραλείποντας ένα ή περισσότερα επίπεδα. Με αυτή την τεχνική εξασφαλίζεται ότι όσο προχωράνε τα επίπεδα θα αποδίδουν τουλάχιστον το ίδιο καλό με το προηγούμενο επίπεδο και όχι χειρότερα. [28]

MobileNet: Όπως καταλαβαίνουμε και από την ονομασία του, το MobileNet έχει σχεδιαστεί για να χρησιμοποιείται σε εφαρμογές για κινητά (mobile applications). Το βασικότερο πλεονέκτημά του είναι ότι μειώνει το πλήθος των παραμέτρων στο νευρωνικό δίκτυο. Μπορεί να κάνει πολλές εργασίες (tasks) ταυτόχρονα, γεγονός που το κάνει πολύ γρήγορο. Είναι το μικρότερο μοντέλο με μέγεθος μόλις 16MB. [26]

2.4 Βιβλιογραφική επισκόπηση ZSL και GZSL

Με το Zero Shot Learning μεταφέρεται το μοντέλο αναγνώρισης αντικειμένων από γνωστές (seen) και άγνωστες (unseen) κλάσεις μέσω του σημασιολογικού χώρου (semantic space) που μοιράζονται, στο οποίο και οι γνωστές και οι άγνωστες κλάσεις έχουν τις δικές τους σημασιολογικές περιγραφές (semantic descriptors)[7]. Η επίδοση της μεθόδου Zero Shot Learning μετριέται αποκλειστικά στο πόσο ακριβής είναι η κατηγοριοποίησή του στις άγνωστες κλάσεις.

Τα πρώτα μοντέλα ZSL προσέγγισαν το πρόβλημα με μια μέθοδο δύο βημάτων όπου χρησιμοποιούνται τα γνωρίσματα για να συμπεράνουν τις ετικέτες (labels) των εικόνων που ανήκουν σε μια από τις άγνωστες κλάσεις. Για να το θέσουμε πιο απλά, τα γνωρίσματα των εικόνων προβλέπονται στο πρώτο βήμα και έπειτα στο δεύτερο βήμα, ψάχνοντας την κλάση με τα περισσότερα κοινά γνωρίσματα, συνάγονται οι ετικέτες των κλάσεών τους. Αυτή η μέθοδος δύο βημάτων έχει επεκταθεί και σε περιπτώσεις όπου τα γνωρίσματα των εικόνων δεν είναι διαθέσιμα. Τα μοντέλα που χρησιμοποιούν αυτή τη μέθοδο αντιμετωπίζουν το πρόβλημα του domain shift [29].

Το domain shift είναι η αλλαγή στην κατανομή των δεδομένων με τα οποία γίνεται η εκπαίδευση στον αλγόριθμο και των δεδομένων που συναντά σε πραγματικές καταστάσεις. [30]

Τα νεότερα ZSL μοντέλα μαθαίνουν να κάνουν απευθείας την σύνδεση των χαρακτηριστικών της εικόνας με την σημασία της. Επίσης, υπάρχουν μοντέλα ZSL που μαθαίνουν από τις μη γραμμικές multi-modal ενσωματώσεις (non-linear multi-modal embeddings).

Μια άλλη προσέγγιση που προσαρμόζει το zero shot learning είναι η ενσωμάτωση της εικόνας και των σημασιολογικών χαρακτηριστικών σε έναν άλλο κοινό ενδιάμεσο χώρο. Τα υβριδικά μοντέλα, όπως το SSE, το CONSE και το SYNC, ενσωματώνουν πολλές αναπαραστάσεις κειμένου (text representasions) και πολλά οπτικά μέρη (visual parts) σε βασικά γνωρίσματα που βρίσκονται σε διάφορα σημεία της εικόνας.

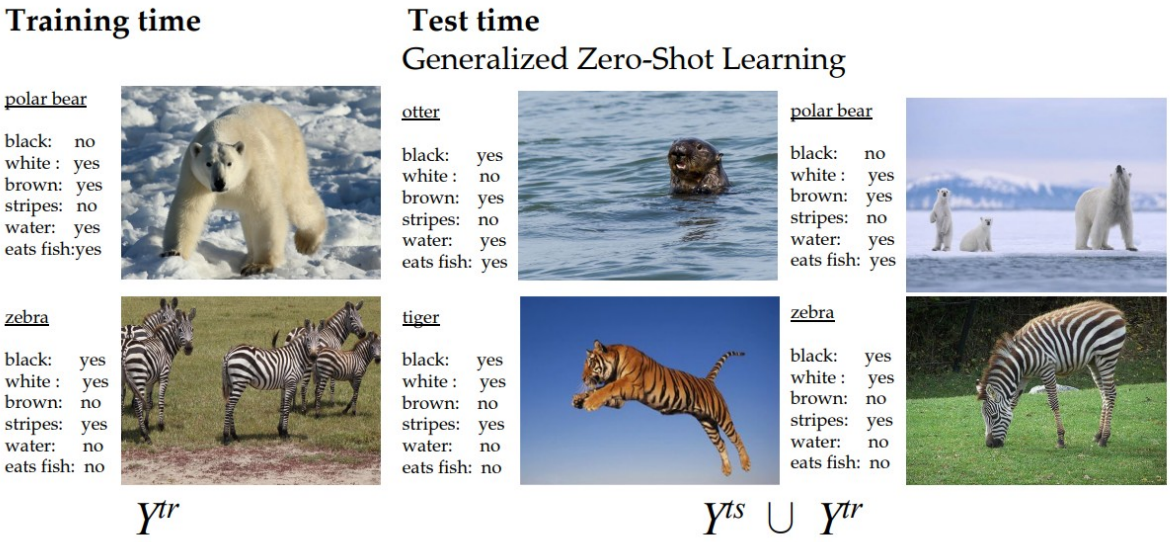
Το Generalized Zero Shot Learning είναι μια πιο ρεαλιστική εκδοχή του ZSL. Σκοπός του GZSL είναι να βελτιώσει την απόδοση της κατηγοριοποίησης και στις γνωστές, αλλά και στις άγνωστες κλάσεις. Η επίδοση αυτής της μεθόδου υπολογίζεται από την αρμονική μέση τιμή της ακρίβειας στην κατηγοριοποίηση των γνωστών και άγνωστων κλάσεων. [31]

Για την επίτευξη αυτού του σκοπού, πρέπει να βρούμε πως θα μεταφέρουμε τη γνώση από τις γνωστές στις άγνωστες κλάσεις και πως το μοντέλο θα μάθει να αναγνωρίζει εικόνες και από τις δύο κλάσεις χωρίς να έχει πρόσβαση στις περιγραφές των εικόνων των άγνωστων κλάσεων. Για το λόγο αυτό, έχουν προταθεί πολλές μέθοδοι που μπορούν να χωριστούν σε δύο γενικές κατηγορίες. [32]

Στην πρώτη κατηγορία ανήκουν οι μέθοδοι που βασίζονται στην ενσωμάτωση (Embedding-base methods). Αυτές οι μέθοδοι μαθαίνουν ένα χώρο ενσωμάτωσης να συσχετίζει τα χαμηλού επιπέδου οπτικά χαρακτηριστικά των γνωστών κλάσεων με τα

αντίστοιχα σημασιολογικά διανύσματα. Οι embedding-based μέθοδοι αντιμετωπίζουν το πρόβλημα του overfitting για τις γνωστές κλάσεις. Για την αντιμετώπιση αυτού του προβλήματος, κάποιες από τις μεθόδους έχουν αναπτύξει νέες συναρτήσεις (loss functions) για να ισορροπήσουν τις προβλέψεις του μοντέλου στις γνωστές και στις άγνωστες κλάσεις. Επιπλέον υπάρχει και το γράφημα γνώσης (knowledge graph) μέσω του οποίου διαδίδονται οι γνώσεις από τις γνωστές στις άγνωστες κλάσεις.

Στην δεύτερη κατηγορία ανήκουν οι μέθοδοι που βασίζονται στη παραγωγή (Generative-based methods). Οι μέθοδοι που ανήκουν σε αυτή την κατηγορία μαθαίνουν σε ένα μοντέλο να παράγει εικόνες ή οπτικά χαρακτηριστικά για τις άγνωστες κλάσεις. Για να παράξει αυτές τις εικόνες ή τα οπτικά χαρακτηριστικά, το μοντέλο στηρίζεται σε δείγματα από τις γνωστές κλάσεις και στη σημασιολογική αναπαράσταση των γνωστών και των άγνωστων κλάσεων.



Εικόνα 8: Zero-Shot Learning vs Generalized Zero-Shot Learning
Πηγή Εικόνας:[29]

Κατά τη διάρκεια της εκπαίδευσης και στις δύο περιπτώσεις οι εικόνες και τα γνωρίσματα των γνωστών κλάσεων είναι διαθέσιμα. Κατά τη διάρκεια της αξιολόγησης, στη περίπτωση του ZSL το εκπαιδευμένο μοντέλο αξιολογείται μόνο στις άγνωστες κλάσεις, ενώ στη περίπτωση του GZSL, ο χώρος αναζήτησης περιέχει γνωστές και άγνωστες κλάσεις μαζί. Για τη διευκόλυνση της κατηγοριοποίησης χωρίς ετικέτες, τα ZSL και GZSL χρησιμοποιούν επιπλέον πληροφορίες σε διάφορες μορφές, όπως για παράδειγμα τα γνωρίσματα της κάθε κλάσης. [29]

2.5 Συναφή Σύνολα Δεδομένων

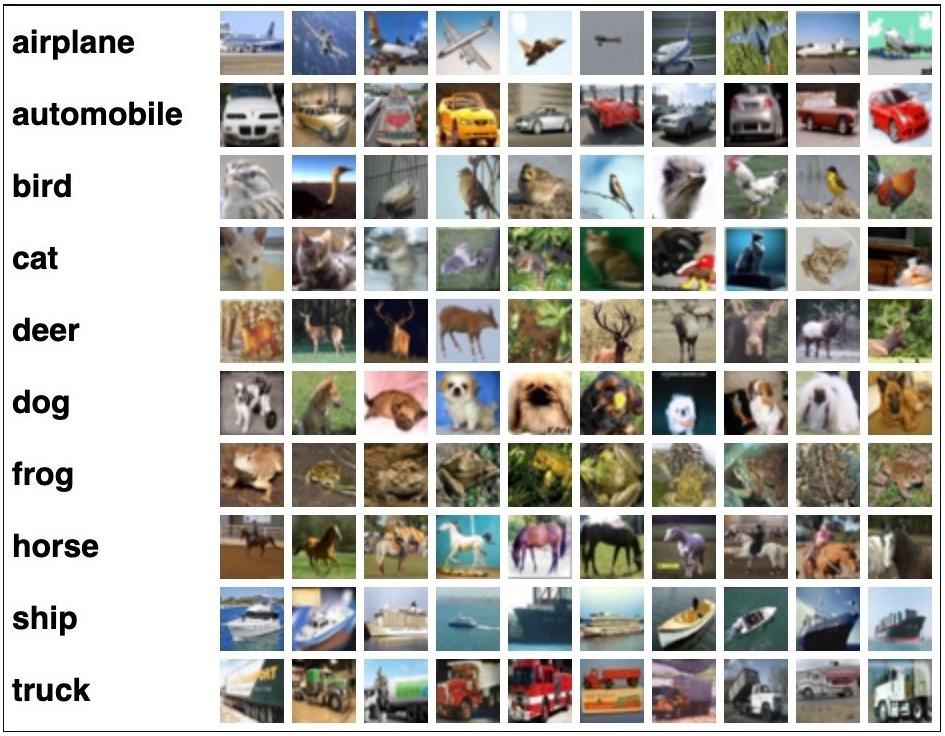
Για την εκπαίδευση των μοντέλων κατηγοριοποίησης εικόνων χρειαζόμαστε δεδομένα. Υπάρχουν πολλά έτοιμα σύνολα δεδομένων (datasets) διαθέσιμα για αυτό τον σκοπό. Ανάλογα τις εικόνες που θέλουμε να αναγνωρίζει το μοντέλο που εκπαιδεύουμε υπάρχουν και τα αντίστοιχα datasets. Για παράδειγμα υπάρχουν datasets με εικόνες από τέχνη, γεωργία, αθλήματα, φαγητά, ζώα, τοπία και άλλα. [33]

Παρακάτω θα δούμε κάποια από τα πιο γνωστά σύνολα δεδομένων που χρησιμοποιούνται για την κατηγοριοποίηση εικόνων.

MNIST: Το MNIST dataset (Modified National Institute of Standards and Technology) είναι μια μεγάλη συλλογή 70 χιλιάδων εικόνων από χειρόγραφους αριθμούς. Οι διαστάσεις αυτών των εικόνων είναι 28x28 και είναι ασπρόμαυρες. Οι αριθμοί που απεικονίζονται σε αυτές τις εικόνες κυμαίνονται από το 0 έως το 9. Το MNIST dataset είναι υποσύνολο δύο άλλων μεγαλύτερων συνόλων δεδομένων, του NIST Special Database 3 και του Special Database 1. Το MNIST dataset έχει και μια άλλη εκδοχή του, το Fashion-MNIST, όπου αποτελείται από 70 χιλιάδες ασπρόμαυρες εικόνες προϊόντων μόδας, χωρισμένα σε 10 κατηγορίες, με διαστάσεις 28x28. [34,35]

ImageNet: Το ImageNet dataset περιέχει 14.197.122 επισημασμένες εικόνες. Από το 2010 χρησιμοποιείται στο ImageNet Large Visual Recognition Challenge (ILSVRC), ως σημείο αναφοράς (benchmark) στην κατηγοριοποίηση εικόνων και τον εντοπισμό αντικειμένων. Υπάρχει και μια άλλη εκδοχή του όπου οι εικόνες δεν είναι επισημασμένες. [36]

CIFAR-100: Το CIFAR-100 dataset (Canadian Institute for Advanced Research, 100 classes) είναι ένα υποσύνολο του Tiny Images dataset. Αποτελείται από 60 χιλιάδες 32x32 έγχρωμες εικόνες. Οι 100 κλάσεις που έχει χωρίζονται σε 20 υπερκλάσεις. Κάθε εικόνα έχει 2 ετικέτες, μία για την κλάση και μία για την υπερκλάση που ανήκει. [37]



Εικόνα 9: Δείγματα από εικόνες του συνόλου δεδομένων CIFAR-100
Πηγή Εικόνας:[37]

Για τα μοντέλα ZSL και GZSL χρησιμοποιούνται πολύ συχνά τα σύνολα δεδομένων aPY, AWA/AWA2, CUB και SUN.

aPY: Το aPY dataset (Attribute Pascal and Yahoo) αποτελείται από 15339 εικόνες από τρεις ευρύς κατηγορίες, ζώα, αντικείμενα και οχήματα. Αυτές οι τρεις κατηγορίες χωρίζονται σε 32 υποκατηγορίες, πχ. αεροπλάνο, ζέβρα, κλπ.. [38]

AWA/AWA2: Το AWA (Animals with Attributes) ήταν ένα σύνολο δεδομένων που χρησιμοποιούταν για την εκπαίδευση αλγορίθμων μεταφοράς μάθησης (transfer learning algorithms). Αποτελούνταν από 30475 εικόνες 50 διαφορετικών τύπων ζώων. Δεν χρησιμοποιείται πια λόγω του ότι δεν του ανήκουν τα πνευματικά δικαιώματα των εικόνων που είχε. Για αυτό το λόγο αντικαταστάθηκε από το AWA2. Το AWA2 αποτελείται από 37322 εικόνες και έχει 50 διαφορετικούς τύπους ζώων. [39,40]

CUB: Το CUB-200-2011 (Caltech-UCSD Birds-200-2011) dataset είναι το πιο διαδεδομένο σύνολο δεδομένων για τη λεπτομερή οπτική κατηγοριοποίηση. Περιέχει 11788 εικόνες χωρισμένες σε 200 υποκατηγορίες πτηνών. Κάθε εικόνα έχει λεπτομερή σχόλια. [41]

SUN: Το SUN dataset αποτελείται από 14340 εικόνες που απεικονίζουν 717 διαφορετικές περιπτώσεις τοπίων. Το συγκεκριμένο σύνολο δεδομένων μπορεί να χρησιμοποιηθεί για υψηλού επιπέδου κατανόησης τοπίων (high level scene understanding) και λεπτομερή αναγνώριση τοπίων (fine grained scene recognition). [42,43]

Transfer Learning

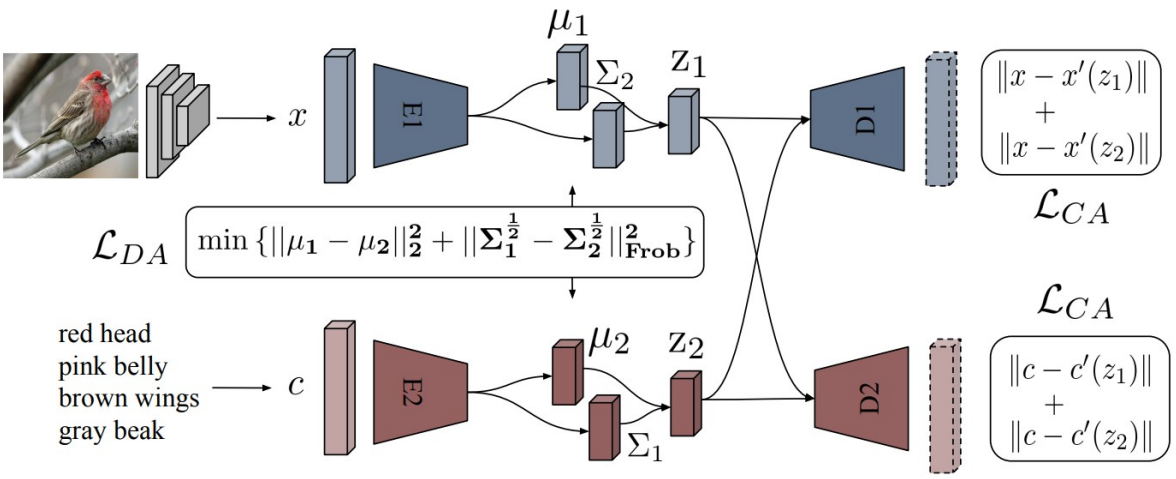
Όπως είδαμε υπάρχουν πολλά έτοιμα σύνολα δεδομένων για διάφορες περιπτώσεις, είναι όμως σχεδόν απίθανο να υπάρχει κάποιο σύνολο δεδομένων που να περιλαμβάνει όλες τις περιπτώσεις που θα συναντήσει το μοντέλο μας. Επίσης απίθανο είναι να εκπαιδεύσουμε το μοντέλο μας να έχει υψηλή ακρίβεια χρησιμοποιώντας ένα πολύ μικρό σύνολο δεδομένων. Για την αντιμετώπιση αυτών των προβλημάτων υπάρχει η μεταφορά μάθησης (transfer learning).

Η μεταφορά μάθησης είναι μια μέθοδος με την οποία χρησιμοποιούμε πληροφορίες που έχουμε από ένα προ εκπαιδευμένο μοντέλο ως την αρχή ενός άλλου μοντέλου για μια καινούργια δουλειά. [44] Για παράδειγμα η γνώση ενός μοντέλου που έχει μάθει να αναγνωρίζει γάτες μπορεί να μεταφερθεί σε ένα νέο μοντέλο για να αναγνωρίζει σκύλους. Με αυτό τον τρόπο τα μοντέλα κατηγοριοποίησης εικόνων μειώνουν τον χρόνο που χρειάζονται για την εκπαίδευσή τους, βελτιώνουν τις επιδόσεις τους, ενώ παράλληλα μειώνουν και το πλήθος των δεδομένων που απαιτούνται. [45]

Τα περισσότερα μοντέλα για την κατηγοριοποίηση εικόνων, αν όχι όλα πλέον, χρησιμοποιούν αυτή τη μέθοδο για την εκπαίδευσή τους. Μοντέλα όπως το AlexNet, το ImageNet και το Inception έχουν τη βασική ιδέα της μεταφοράς μάθησης. [44]

Κεφάλαιο 3: Μέθοδοι ZSL και GZSL

3.1 Μέθοδος CADA-VAE



Εικόνα 10: Αρχιτεκτονική μοντέλου CADA-VAE
Πηγή Εικόνας: [31]

<u>Το μοντέλο CADA-VAE</u>:

Από τις κατηγορίες των μεθόδων GZSL αυτές που βασίζονται στη παραγωγή δεδομένων επιτυγχάνουν καλύτερες επιδόσεις από άλλες μεθόδους. Η κατηγοριοποίηση των χαρακτηριστικών των παραγόμενων εικόνων με τη χρήση GANs (Generated Adversarial Networks) ή περιστασιακών VAEs (Variational Autoencoders) συχνά συναντά το πρόβλημα της διαστασιμότητας (dimensionality). Αντίθετα, το μοντέλο CADA-VAE έχει τον έλεγχο στις διαστάσεις και στη δομή των χαρακτηριστικών που θα κατηγοριοποιηθούν. Η βασική ιδέα του CADA-VAE είναι να παράγει χαμηλών διαστάσεων κρυφά (latent) χαρακτηριστικά έτσι ώστε να είναι σταθερό κατά την εκπαίδευσή του, αλλά να πετυχαίνει και state of the art επιδόσεις. Συνεπώς, για να γίνει αυτό θα πρέπει να επιλέξουμε ένα VAE κρυφού χώρου (latent space), ένα κριτήριο για την ανοικοδόμηση και την διασταυρωτή ανοικοδόμηση (cross reconstruction) για να διατηρηθούν οι πληροφορίες των κλάσεων σε χαμηλές διαστάσεις, καθώς και αποκλειστικό διαμοιρασμό ευθυγράμμισης (explicit distribution alignment) για να ενθαρρύνει αναπαραστάσεις αγνωστικού domain. [31]

<u>Η λειτουργία του VAE στο μοντέλο</u>:

Η λειτουργία του VAE είναι να χτίζει τα κομμάτια του μοντέλου. Με τη μέθοδο του Variational Inference (VI) γίνεται γνωστή η σχέση μεταξύ της πραγματικής υπό συνθήκης πιθανότητας διανομής (true conditional probability distribution) με τις κρυφές μεταβλητές, $p_\varphi(z|x)$. Λόγω της αλληλεπίδρασης αυτής της διανομής, μπορεί να εκτιμηθεί βρίσκοντας το κοντινότερο proxy posterior, $q_\vartheta(z|x)$, ελαχιστοποιώντας την απόστασή τους με τη χρήση ενός variational lower bound limit. Τα φ και θ είναι οι παράμετροι των $p_\varphi(z|x)$ και $q_\vartheta(z|x)$

αντίστοιχα, το x η μεταβλητή που έχει παρατηρηθεί και z η κρυφή μεταβλητή. Η συνάρτηση ενός VAE δίνεται από τον τύπο:

$$L = \mathrm{E}_{q_\phi(z|x)}\left[\log p_\theta\left(x \mid z\right)\right] - D_{KL}\left(q_\phi(z \mid x) \,||\, p_\theta\left(z\right)\right)$$

Στον παραπάνω τύπο ο πρώτος όρος είναι το σφάλμα της ανοικοδόμησης και ο δεύτερος όρος είναι η απόκλιση Kullback-Leibler.

<u>Ανάλυση μοντέλου CADA-VAE</u>:

Ο στόχος του μοντέλου CADA-VAE είναι να μάθει πως να αναπαριστά συνδυασμούς των M τρόπων λειτουργίας των δεδομένων μέσα σε ένα κοινό χώρο. Για κάθε τρόπο λειτουργίας υπάρχει στο μοντέλο και από ένας κωδικοποιητής. Για να περιοριστούν οι απώλειες πληροφορίας, τα αρχικά δεδομένα πρέπει να ανοικοδομούνται μέσω των δικτύων των αποκωδικοποιητών. Οι βασικές απώλειες των VAE του μοντέλου δίνονται από τον τύπο:

$$L_{VAE} = \sum_{i}^{M} \mathrm{E}_{q_\phi(z|x)}\left[\log p_\theta\left(x^{(i)} \mid z\right)\right] - \beta D_{KL}\left(q_\phi\left(z \mid x^{(i)}\right) \,||\, p_\theta\left(z\right)\right)$$

Το *β* συμβολίζει το βάρος της απόκλισης Kullback-Leibler.

Για να μάθουν οι αυτοματοποιημένοι κωδικοποιητές (autoencoders) παρόμοιες αναπαραστάσεις για κάθε τρόπο λειτουργίας χρειάζονται επιπλέον ορισμοί κανονικοποίησης. Για αυτό το λόγο το μοντέλο CADA-VAE ευθυγραμμίζει με σαφή τρόπο τις κρυφές διανομές και επιβάλλει ένα κριτήριο για τη διασταυρωτή ανοικοδόμηση. Το μοντέλο αποτελείται από δύο μορφές κρυφών διανομών, την διασταυρωτή ευθυγράμμιση (Cross Alignment - CA) και την ευθυγράμμιση διανομών (Distribution Alignment – DA).

<u>Cross-Alignment (CA) Loss</u>:

Σε αυτό το σημείο γίνονται ανοικοδομήσεις από την αποκωδικοποίηση της κρυφής κωδικοποίησης ενός δείγματος από έναν άλλο τρόπο λειτουργίας, από την ίδια κλάση όμως. Επομένως, κάθε αποκωδικοποιητής που προορίζεται για κάποιο συγκεκριμένο τρόπο λειτουργίας εκπαιδεύεται σε ένα σύνολο κρυφών διανυσμάτων που έχουν παραχθεί από άλλους τρόπους λειτουργίας. Αυτό δίνεται από τον τύπο:

$$L_{CA} = \sum_{i}^{M} \sum_{j \neq i}^{M} | \, x^{(j)} - D_j\left(E_i\left(x^{(i)}\right)\right)|$$

όπου E_i είναι ο κωδικοποιητής ενός χαρακτηριστικού από τον τρόπο λειτουργίας i και D_j είναι ο κωδικοποιητής ενός χαρακτηριστικού από την ίδια κλάση αλλά από τον τρόπο λειτουργίας j.

<u>Distribition-Alignment (DA) Loss</u>:

Το ταίριασμα της παραγόμενης εικόνας με τις αναπαριστάμενες κλάσεις μπορεί να γίνει και με την ελαχιστοποίηση της απόστασής τους. Στο μοντέλο CADA-VAE ελαχιστοποιείται η απόσταση Wassertein μεταξύ των κρυφών κανονικών (ή Gaussian) κατανομών με πολλές μεταβλητές. Σε αυτή τη περίπτωση η απόσταση δίνεται από τον τύπο:

$$W_{ij} = \left[\| \mu_i - \mu_j \|_2^2 + Tr\left(\Sigma_j\right) - 2\left(\Sigma_i^{\frac{1}{2}} \Sigma_i \Sigma_j^{\frac{1}{2}} \right)^{\frac{1}{2}} \right]^{\frac{1}{2}}$$

Αυτός ο τύπος μπορεί να γραφτεί πιο απλά, αφού ο κωδικοποιητής κάνει προβλέψεις σε διαγώνιους πίνακες συν διακύμανσης, ως εξής:

$$W_{ij} = \left(\| \mu_i - \mu_j \|_2^2 + \| \Sigma_i^{\frac{1}{2}} - \Sigma_j^{\frac{1}{2}} \|_{Frobenius}^2 \right)^{\frac{1}{2}}$$

Ο τύπος για την απώλεια της ευθυγράμμισης των διανομών για ένα σύνολο M γράφεται ως:

$$L_{DA} = \sum_{i}^{M} \sum_{j \neq i}^{M} W_{ij}$$

<u>Cross- and Distribution Alignment (CADA-VAE) Loss</u>:

Τέλος, ο υπολογισμός της απώλειας CADA-VAE αποτελεί ένα συνδυασμό των παραπάνω τύπων απώλειας:

$$L_{CADA-VAE} = L_{VAE} + \gamma L_{CA} + \delta L_{DA}$$

όπου γ και δ συμβολίζουν τα βάρη των αντίστοιχων τύπων απώλειας.

Το μοντέλο CADA-VAE μπορεί να μάθει παραπάνω από δύο τρόπους λειτουργίας, χωρίς να χρειάζεται παραδείγματα από όλους τους τρόπους λειτουργίας σε όλες τις κλάσεις.

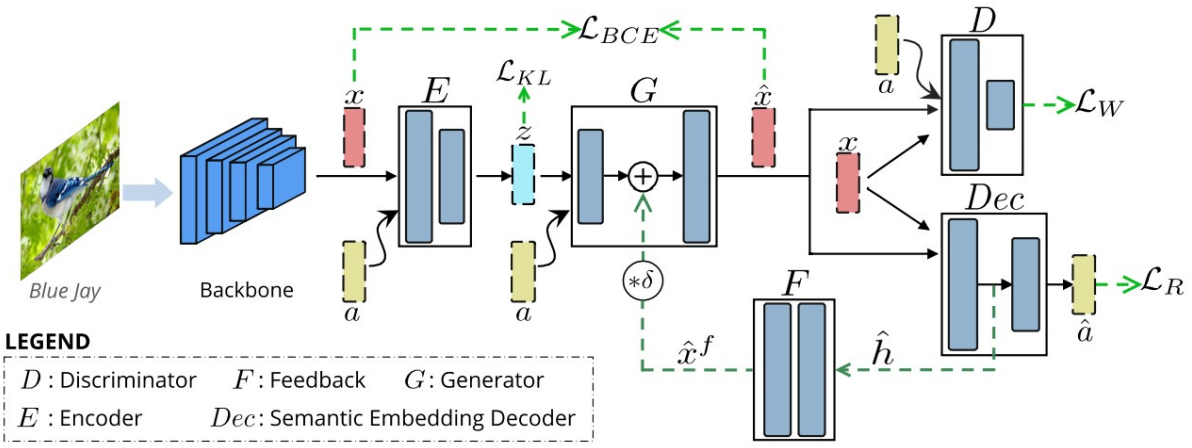
3.2 Μέθοδος TF-VAEGAN

<u>Που βασίζεται το μοντέλο TF-VAEGAN</u>:

Το μοντέλο TF-VAEGAN βασίζεται σε ένα πρόσφατο μοντέλο ZSL, το f-VAEGAN, το οποίο συνδυάζοντας τα δυνατά στοιχεία των VAE και των GAN επιτυγχάνει αξιοθαύμαστα αποτελέσματα σε κατηγοριοποιήσεις ZSL. Το μοντέλο f-VAEGAN παράγει σημασιολογικά σταθερά χαρακτηριστικά σε σύγκριση με άλλα μοντέλα που βασίζονται σε GAN. Αυτό το πετυχαίνει μοιράζοντας τον αποκωδικοποιητή και την γεννήτρια του VAE και του GAN. Στο μοντέλο f-VAEGAN, το VAE που παράγει τα χαρακτηριστικά (f-VAE) περιλαμβάνει έναν κωδικοποιητή E(x, α) , ο οποίος κωδικοποιεί το εισαγόμενο χαρακτηριστικό x σε έναν κρυφό κώδικα z, και έναν αποκωδικοποιητή G(z, α) ο οποίος ανοικοδομεί το x από το z. Και ο κωδικοποιητής, αλλά και ο αποκωδικοποιητής, προσαρμόζονται στην ενσωμάτωση α. Επομένως προκύπτει ο παρακάτω τύπος:

$$L_V = KL\left(\mathrm{E}\left(x,\alpha\right) \| \, p\left(z \mid \alpha\right)\right) - \mathrm{E}_{\mathrm{E}(x,\alpha)}\left[\log G\left(z,\alpha\right)\right]$$

όπου KL είναι η απόκλιση Kullback-Leibler, p(z|α) είναι η εκ των προτέρων κατανομή πιθανότητας, υποθέτοντας ότι N(0, 1) και log G(Z, α) είναι η απώλεια κατά την ανοικοδόμηση.



Εικόνα 11: Αρχιτεκτονική μοντέλου TF-VAEGAN
Πηγή Εικόνας: [46]

Το δίκτυο που παράγει τα χαρακτηριστικά συμπεριλαμβάνει μια γεννήτρια G(z, α) και ένα διευκρινιστή (discriminator) D(x, α). Η γεννήτρια συνθέτει ένα χαρακτηριστικό x̂ ε X από έναν τυχαίο εισαγόμενο θόρυβο z, εφόσον ο διευκρινιστής πάρει ένα εισαγόμενο χαρακτηριστικό x και εξάγει τη πραγματική τιμή υποδεικνύοντας το πόσο αληθινά ή ψεύτικά είναι τα εισαγόμενα χαρακτηριστικά. Και η γεννήτρια και ο διευκρινιστής είναι προσαρμοσμένα στην ενσωμάτωση α, βελτιώνοντας έτσι τον τύπο για την απώλεια του Wassertein GAN:

$$L_W = \mathrm{E}\left[D(x,\alpha)\right] - \mathrm{E}\left[D(\hat{x},\alpha)\right] - \lambda \mathrm{E}\left[\left(\| \nabla D(\tilde{x},\alpha) \|_2 - 1\right)^2\right]$$

Στον παραπάνω τύπο το x̂ είναι το συντιθέμενο χαρακτηριστικό, λ είναι ο συντελεστής της ποινής και το x είναι μια τυχαία τιμή από τη συνδετική γραμμή του x και του x̂.

Επομένως ο τύπος του f-VAEGAN γράφεται ως:

$$L_{vaegan} = L_V + \alpha L_W$$

όπου το α είναι μια υπερπαράμετρος. [46]

Ο τύπος για την απώλεια στην εκπαίδευση του f-VAEGAN περιέχει ένα περιορισμό με τον οποίο διασφαλίζει ότι για όλες τις επαναλήψεις κατά την διάρκεια της εκπαίδευσης τα παραγόμενα οπτικά χαρακτηριστικά παραμένουν σταθερά με τα αυθεντικά οπτικά χαρακτηριστικά. Παρόλα αυτά δεν υπάρχει παρόμοιος περιορισμός για τις σημασιολογικές ενσωματώσεις. Άλλα μοντέλα ZSL που βασίζονται στο GAN χρησιμοποιούν βοηθητικές μονάδες, πέρα από τη γεννήτρια, για να πετύχουν αυτή την επαναληπτική σταθερότητα στις ενσωματώσεις. Αυτές οι βοηθητικές μονάδες χρησιμοποιούνται μόνο για την εκπαίδευση του μοντέλου και όχι για τη σύνθεση των χαρακτηριστικών και στην ZSL κατηγοριοποίηση.

Το μοντέλο TF-VAEGAN χρησιμοποιεί έναν αποκωδικοποιητή σημασιολογικής ενσωμάτωσης (semantic embedding decoder – SED) για να πετύχει αυτή την επαναληπτική σταθερότητα στις σημασιολογικές ενσωματώσεις κατά την διάρκεια της εκπαίδευσης, της σύνθεσης χαρακτηριστικών και στην ZSL κατηγοριοποίηση. Η γεννήτρια και ο SED πραγματοποιούν αντίστροφες μεταμορφώσεις (transformations) συγκριτικά μεταξύ τους. Η γεννήτρια μεταμορφώνει τις σημασιολογικές ενσωματώσεις σε στιγμιότυπα χαρακτηριστικών, ενώ ο SED μεταμορφώνει τα στιγμιότυπα χαρακτηριστικών σε σημασιολογικές ενσωματώσεις. Το μοντέλο TF-VAEGAN εκμεταλλεύεται τις πληροφορίες από αυτές τις δύο συμπληρωματικές μονάδες για τη βελτίωση της σύνθεσης των χαρακτηριστικών και για τη μείωση ασαφειών μεταξύ των κλάσεων κατά τη διάρκεια της ZSL κατηγοριοποίησης.

<u>Αρχιτεκτονική μοντέλου TF-VAEGAN</u>:

Η αρχιτεκτονική του μοντέλου απαρτίζεται από έναν κωδικοποιητή (E), μια γεννήτρια (G) και έναν διευκρινιστή (D). Ο κωδικοποιητής δέχεται πραγματικά χαρακτηριστικά από τις γνωστές κλάσεις (x) και τις σημασιολογικές ενσωματώσεις (α) και εξάγει τις παραμέτρους μια κατανομής θορύβου. Αυτές οι παράμετροι ταιριάζουν με τις παραμέτρους μιας zero-mean unit-variance Gaussian prior distribution χρησιμοποιώντας την απόκλιση Kullback-Leibler (L_{KL}). Η γεννήτρια δέχεται με τη σειρά της τον θόρυβο (z) και τις ενσωματώσεις (α) με τα οποία συνθέτει τα χαρακτηριστικά (x̂). Τα χαρακτηριστικά που συντέθηκαν συγκρίνονται με τα αυθεντικά χαρακτηριστικά (x) με τη χρήση της απώλειας δυαδικής διασταυρωμένης εντροπίας (binary cross-entropy loss) (L_{BCE}). Ο διευκρινιστής (D) παίρνει ως εισαγωγές είτε τα αυθεντικά χαρακτηριστικά (x), είτε τα χαρακτηριστικά που συντέθηκαν (x̂) μαζί με τις ενσωματώσεις (α)

και υπολογίζει έναν πραγματικό αριθμό με τον οποίο καθορίζει αν η εισαγωγή είναι αληθινή ή ψεύτικη. Στο αποτέλεσμα του διευκρινιστή (D) εφαρμόζεται η απώλεια WGAN (L_W) με την οποία μαθαίνει να ξεχωρίζει ανάμεσα στα αληθινά και τα ψεύτικα χαρακτηριστικά.

Το μοντέλο TF-VAEGAN έχει επιπλέον έναν ακόμα αποκωδικοποιητή, τον SED (Dec), ο οποίος χρησιμοποιείται και στο στάδιο της σύνθεσης χαρακτηριστικών, αλλά και στο στάδιο της ZSL κατηγοριοποίησης. Επιπλέον, έχει μια μονάδα ανατροφοδότησης (F) που χρησιμοποιείται παράλληλα με τον SED στην εκπαίδευση και στη σύνθεση χαρακτηριστικών. Με αυτές τις δύο μονάδες μειώνονται οι ασάφειες στην κατηγοριοποίηση. Ο SED παίρνει ως εισαγωγές είτε τα αυθεντικά χαρακτηριστικά (x), είτε τα χαρακτηριστικά που συντέθηκαν ($\hat{x}$) και με ανοικοδομεί τις ενσωματώσεις (α). Αφού εκπαιδευτεί με την απώλεια επαναληπτικής σταθερότητας (L_R) χρησιμοποιείται μεταγενέστερα σε κατηγοριοποιητές ZSL/GZSL. Η μονάδα ανατροφοδότησης (F) μετατρέπει την κρυφή ενσωμάτωση του SED και τη στέλνει πίσω στην κρυφή αναπαράσταση της γεννήτριας (G) με σκοπό να βελτιώσει τη σύνθεση χαρακτηριστικών.

3.3 Μέθοδος CE-GZSL

<u>Υβριδικό μοντέλο GZSL</u>

Ο σκοπός της σημασιολογικής ενσωμάτωσης στο συμβατικό Zero Shot Learning είναι να μάθει μια λειτουργία ενσωμάτωσης E η οποία χαρτογραφεί ένα οπτικό χαρακτηριστικό x μέσα στο χώρο σημασιολογικού περιγραφέα (semantic descriptor space) που συμβολίζεται με E(x). Οι πιο διαδεδομένες μέθοδοι σημασιολογικής ενσωμάτωσης βασίζονται σε μια δομημένη συνάρτηση απώλειας (structured loss function). Η δομημένη συνάρτηση απώλειας χρειάζεται την ενσωμάτωση του x να είναι πιο κοντά στο σημασιολογικό περιγραφέα α της πραγματικής του κλάσης, παρά στους σημασιολογικούς περιγραφείς άλλων κλάσεων. Ο τύπος για τη δομημένη συνάρτηση απώλειας είναι ο εξής: [7]

$$L_{se}^{real}\left(\mathrm{E}\right)=\mathrm{E}_{p(x,\alpha)}\left[\max\left(0,\Delta-\alpha^{\mathrm{T}}\mathrm{E}\left(x\right)+\left(\alpha'\right)^{\mathrm{T}}\mathrm{E}\left(x\right)\right)\right]$$

όπου p(x, α) είναι η εμπειρική κατανομή των πραγματικών δειγμάτων των γνωστών κλάσεων, α’ ≠ α είναι ένας τυχαία επιλεγμένος σημασιολογικός περιγραφέας των άλλων κλάσεων και Δ>0 είναι μια παράμετρος για το περιθώριο που έχει η λειτουργία ενσωμάτωσης E ώστε να γίνει πιο ολοκληρωμένη.

Επειδή οι μέθοδοι σημασιολογικής ενσωμάτωσης είναι λιγότερο αποδοτικοί στο Generalized Zero Shot Learning, λόγω του μεγάλου bias ως προς τις γνωστές κλάσεις, έχουν προταθεί πολλές μέθοδοι παραγωγής χαρακτηριστικών (feature generation methods) για τη σύνθεση των ελλιπών δειγμάτων εκπαίδευσης των άγνωστων κλάσεων. Η δουλειά των μεθόδων παραγωγής χαρακτηριστικών είναι να μάθουν σε ένα δίκτυο γεννήτριας υπό συνθήκες (conditional generator network) G να παράγει δείγματα $\bar{x} = G(\alpha, \varepsilon)$ σύμφωνα με το θόρυβο Gauss (Gaussian noise) $\varepsilon \sim N(0, I)$ και το σημασιολογικό περιγραφέα α. Παράλληλα ένα

δίκτυο διευκρινιστής D (discriminator network) εκπαιδεύεται μαζί με το δίκτυο G ώστε να ξεχωρίζουν το πραγματικό ζευγάρι (x, α) από το συνθετικό ($\tilde{x}$, α). Η γεννήτρια χαρακτηριστικών G προσπαθεί να εξαπατήσει το διευκρινιστή D παράγοντας πανομοιότυπα συνθετικά χαρακτηριστικά. Οι μέθοδοι παραγωγής χαρακτηριστικών προσπαθούν να ταιριάξουν με επιτυχία την κατανομή συνθετικών χαρακτηριστικών με την κατανομή των πραγματικών χαρακτηριστικών. Η παραπάνω διαδικασία περιγράφεται από τον τύπο:

$$V\left(G,D\right)=\mathrm{E}_{p(x,\alpha)}\left[\log D\left(x,\alpha\right)\right]+\mathrm{E}_{p_G(\tilde{x},\alpha)}\left[\log\left(1-D\left(\tilde{x},\alpha\right)\right)\right]$$

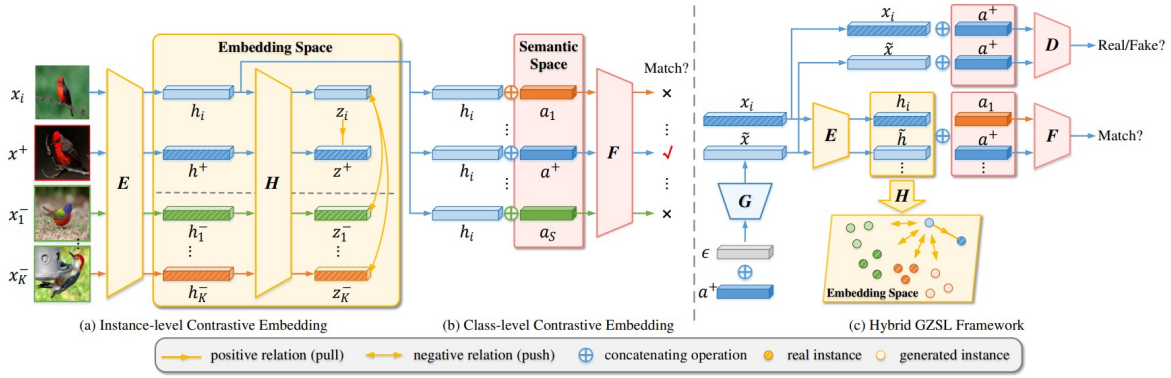
όπου $p_G(\tilde{x}, \alpha) = p_G(\tilde{x}|\alpha)p(\alpha)$ είναι η κοινή κατανομή (joint distribution) ενός συνθετικού χαρακτηριστικού και του αντίστοιχου σημασιολογικού περιγραφέα του.

Οι μέθοδοι παραγωγής χαρακτηριστικών μαθαίνουν να συνθέτουν τα οπτικά χαρακτηριστικά στον αυθεντικό χώρο χαρακτηριστικών (original feature space), το οποίο δεν είναι ιδανικό για την κατηγοριοποίηση με GZSL γιατί σε αυτό το χώρο τα οπτικά χαρακτηριστικά συνήθως δεν είναι καλά δομημένα. Για αυτό το λόγο στη μέθοδο που αναλύουμε τώρα προτείνεται ένα υβριδικό GZSL μοντέλο συνδυάζοντας το μοντέλο ενσωμάτωσης (embedding model) και το μοντέλο παραγωγής χαρακτηριστικών (feature generation model). Σε αυτή τη προσέγγιση τα πραγματικά χαρακτηριστικά και τα συνθετικά χαρακτηριστικά συγκεντρώνονται σε ένα χώρο ενσωμάτωσης, όπου γίνεται και η τελική GZSL κατηγοριοποίηση. Ο τύπος για τις απώλειες ενσωμάτωσης (embedding loss) των συνθετικών χαρακτηριστικών είναι ο εξής:

$$L_{se}^{sync}\left(G,E\right)=\mathrm{E}_{\alpha}\left[\max\left(0,\Delta-\alpha^{\mathrm{T}}\mathrm{E}\left(G\left(\alpha,\varepsilon\right)\right)+\left(\alpha'\right)^{\mathrm{T}}\mathrm{E}\left(G\left(\alpha,\varepsilon\right)\right)\right)\right]$$

Το αποτέλεσμα στον παραπάνω τύπο παράγεται χρησιμοποιώντας μόνο τους σημασιολογικούς περιγραφείς των γνωστών κλάσεων. Επομένως, οι συνολικές απώλειες του υβριδικού GZSL μοντέλου είναι της μορφής:

$$\max_{D}\min_{G,E}V\left(G,D\right)+L_{se}^{real}\left(E\right)+L_{se}^{sync}\left(G,E\right)$$



Εικόνα 12: Αρχιτεκτονική μοντέλου CE-GZSL (hybrid)
Πηγή Εικόνας: [7]

Αντιθετική Ενσωμάτωση

Σε αυτό το paper προτείνεται άλλη μια μέθοδος GZSL, η οποία είναι παραλλαγή του υβριδικού μοντέλου που αναλύσαμε πριν, το μοντέλο αντιθετικής ενσωμάτωσης (Contrastive Embedding – CE). Σε αντίθεση με τη προηγούμενη εκδοχή που βασίζεται στο "παραδοσιακό" μοντέλο σημασιολογικής ενσωμάτωσης, η αντιθετική ενσωμάτωση αποτελείται από την αντιθετική ενσωμάτωση σε επίπεδο παραδείγματος (Instance-level CE) που βασίζεται στην επίβλεψη παραδειγμάτων, και την αντιθετική ενσωμάτωση σε επίπεδο κλάσης (Class-level CE) που βασίζεται στην επίβλεψη των κλάσεων.

Στην αντιθετική ενσωμάτωση, σε επίπεδο παραδείγματος το οπτικό δείγμα x συμβολίζεται στον χώρο ενσωμάτωσης ως h = E(x). Για κάθε δεδομένο h_i που ενσωματώνεται είτε από ένα πραγματικό, είτε από ένα συνθετικό χαρακτηριστικό που έχει δει, υπάρχει ένα (K+1)-διαστάσεων υποπρόβλημα κατηγοριοποίησης για να διακρίνει το μοναδικό θετικό παράδειγμα h^+ από το σύνολο των K αρνητικών παραδειγμάτων $\{h_1, \ldots, h_K\}$. Το θετικό παράδειγμα h^+ που επιλέγεται τυχαία έχει την ίδια ετικέτα κλάσης με το h_i, ενώ οι ετικέτες κλάσεων των αρνητικών παραδειγμάτων είναι διαφορετικές από αυτή του h_i. Επιπλέον, προστίθεται μια μη-γραμμική προβολή H στο χώρο ενσωμάτωσης: $z_i = H(h_i) = H(E(x_i))$, όπου πάνω στο z_i πραγματοποιείται η (K+1)-διαστάσεων κατηγοριοποίηση ώστε να μάθει την ενσωμάτωση h_i. Η απώλεια διασταυρωμένης εντροπίας για το πρόβλημα της (K+1)-διαστάσεων κατηγοριοποίησης μπορεί να υπολογιστεί με τον τύπο:

$$\ell_{ce}^{ins}\left(z_i, z^+\right) = -\log \frac{\exp\left(z_i^{\mathrm{T}} z^+ / \mathrm{T}_e\right)}{\exp\left(z_i^{\mathrm{T}} z^+ / \mathrm{T}_e\right) + \Sigma_{k=1}^{K} \exp\exp\left(z_i^{\mathrm{T}} z_k^- / \mathrm{T}_e\right)}$$

όπου $T_e > 0$ είναι η παράμετρος θερμοκρασίας (temperature parameter) για την αντιθετική ενσωμάτωση σε επίπεδο παραδείγματος και K ο αριθμός των αρνητικών παραδειγμάτων. Μεγάλος αριθμός αρνητικών παραδειγμάτων κάνει το πρόβλημα πιο δύσκολο, γιατί έτσι

προτρέπει τη συνάρτηση ενσωμάτωσης E να αιχμαλωτίσει μικρές λεπτομέρειες που μοιράζονται μεταξύ τους τα πραγματικά και τα συνθετικά δείγματα από την ίδια κλάση στο χώρο ενσωμάτωσης.

Για να μάθουμε τη συνάρτηση ενσωμάτωσης E, τη μη-γραμμική προβολή H και το δίκτυο παραγωγής χαρακτηριστικών G, υπολογίζουμε τη συνάρτηση απώλειας για την αντιθετική ενσωμάτωση σε επίπεδο παραδειγμάτων ως την αναμενόμενη απώλεια που υπολογίζεται από το τυχαία επιλεγμένο ζευγάρι του πραγματικού παραδείγματος z_i και του συνθετικού παραδείγματος z^+, όπου τα z_i και z^+ είναι διαφορετικά μεταξύ τους, αλλά ανήκουν στην ίδια γνωστή κλάση.

$$L_{ce}^{ins}\left(G,E,H\right)=\mathrm{E}_{z_i,z^+}\left[\ell_{ce}^{ins}\left(z_i,z^+\right)\right]$$

Με παρόμοιο τρόπο διατυπώνεται και η αντιθετική ενσωμάτωση σε επίπεδο κλάσης. Επειδή στο μοντέλο δεν υπάρχει περιορισμός ότι ο χώρος ενσωμάτωσης πρέπει να είναι ο χώρος του σημασιολογικού περιγραφέα, δεν μπορεί να υπολογιστεί απευθείας το εσωτερικό γινόμενο ομοιότητας (dot-product similarity) μεταξύ ενός ενσωματωμένου σημείου δεδομένων (embedded data point) και ενός σημασιολογικού περιγραφέα. Για αυτό το λόγο υπάρχει ένα δίκτυο συγκριτής F (comparator network) που μετράει πόσο συναφείς είναι μια ενσωμάτωση h και ένας σημασιολογικός περιγραφέας α. Η διατύπωση για την απώλεια της αντιθετικής ενσωμάτωσης σε επίπεδο κλάσης για ένα τυχαία επιλεγμένο σημείο h_i γίνεται με τη βοήθεια του δικτύου συγκριτή F στο χώρο ενσωμάτωσης και θεωρείται ως ένα υποπρόβλημα κατηγοριοποίησης S-διαστάσεων. Ο σκοπός αυτού του υποπροβλήματος είναι να επιλέξει το μόνο σωστό σημασιολογικό περιγραφέα από τους συνολικά S σημασιολογικούς περιγραφείς των γνωστών κλάσεων. Συνεπάγεται λοιπόν πως ο μόνος θετικός σημασιολογικός περιγραφέας θα είναι αυτός που έχει επιλεχθεί και θα αντιστοιχεί στη κλάση του h_i , ενώ οι υπόλοιποι που θα απορριφθούν από τις άλλες κλάσεις αντιμετωπίζονται ως τους αρνητικούς σημασιολογικούς περιγραφείς. Η απώλεια διασταυρωμένης εντροπίας για αυτό το υποπρόβλημα υπολογίζεται με το παρακάτω τύπο:

$$\ell_{ce}^{cls}\left(h_i,\alpha^+\right)=-\log\frac{\exp\left(F\left(h_i,\alpha^+\right)/\mathrm{T}_S\right)}{\Sigma_{s=1}^{S}\exp\left(F\left(h_i,\alpha_S\right)/\mathrm{T}_S\right)}$$

όπου $T_S > 0$ είναι η παράμετρος θερμοκρασίας για την αντιθετική ενσωμάτωση σε επίπεδο κλάσης και S είναι ο αριθμός των γνωστών κλάσεων. Αυτού του είδους αντιθετικής ενσωμάτωσης βασίζεται στην επίβλεψη των κλάσεων για την ενδυνάμωση της διακριτικής ικανότητας των δειγμάτων στο νέο χώρο ενσωμάτωσης.

Η συνάρτηση απώλειας για την αντιθετική ενσωμάτωση σε επίπεδο κλάσης είναι οι αναμενόμενες απώλειες των πραγματικών και των συνθετικών δειγμάτων στο νέο χώρο ενσωμάτωσης με τους σημασιολογικούς περιγραφείς που τους αντιστοιχούν (πχ ο θετικός περιγραφέας), και δίνεται από τον τύπο:

$$L_{ce}^{cls}\left(G,E,F\right)=\mathrm{E}_{h_i,\alpha^+}\left[\ell_{ce}^{cls}\left(h_i,\alpha^+\right)\right]$$

Συνολική απώλεια μοντέλου

Όπως είδαμε παραπάνω το μοντέλο αντιθετικής ενσωμάτωσης χωρίζεται σε δύο επίπεδα, επίπεδο παραδειγμάτων και επίπεδο κλάσης και περιγράψαμε τους τύπους για τις απώλειες για το καθένα από αυτά. Επομένως οι συνολικές απώλειες για το υβριδικό μοντέλο GZSL με την αντιθετική ενσωμάτωση διατυπώνεται ως:

$$\max_{D}\min_{G,E,H,F}V\left(G,D\right)+L_{ce}^{ins}\left(G,E,H\right)+L_{ce}^{cls}\left(G,E,F\right)$$

3.4 Μέθοδος LFGAA

Προσοχή γνωρίσματος βάση αντικειμένου

Η προσοχή γνωρίσματος (Attribute Attention) επικεντρώνεται σε ένα διακεκριμένο σετ γνωρισμάτων και αγνοεί λιγότερο σημαντικά. Η προσοχή γνωρίσματος βάση αντικειμένου (object-based attribute attention) είναι πολύ σημαντική για την αντιμετώπιση των σημασιολογικά διφορούμενων αντικειμένων (semantic-ambiguous objects). Με τον όρο σημασιολογικά διφορούμενα περιγράφονται τα αντικείμενα που έχουν κοινά χαρακτηριστικά άλλων κλάσεων με αποτέλεσμα το μοντέλο να τα κατηγοριοποιεί λάθος βάση αυτών. [47]

Αρχικά η αναζήτηση της κοντινότερης κλάσης λαμβάνεται ως πολλαπλές δυαδικές κατηγοριοποιήσεις ανάμεσα σε διάφορους συνδυασμούς κατηγοριών, όπως για παράδειγμα για να αποφασιστεί αν ένα αντικείμενο x είναι πιο κοντά στην κλάση y_1 ή στην κλάση y_2 με:

$$D\left(x,y_1,y_2;W\right)=F\left(x,y_1;W\right)-F\left(x,y_2;W\right)$$

όπου το D είναι το αποτέλεσμα της δυαδικής κατηγοριοποίησης και D=0 είναι το υπερεπίπεδο απόφασής του.

Η πιο απλή μορφή συμβατικής βαθμολογίας (compatibility score) μεταξύ της σημασιολογικής πρόβλεψης και του γνωρίσματος σε επίπεδο κλάσης γράφεται ως:

$$D_{ip}\left(x,y_1,y_2\right)=\varphi\left(x\right)^{\mathrm{T}}\Delta\left(\alpha,y_1,y_2\right)$$
$$\Delta\left(\alpha,y_1,y_2\right)=\alpha_{y_1}-\alpha_{y_2}$$

όπου το εσωτερικό γινόμενο (inner product) εφαρμόζεται απευθείας σε αυτά. Από τον παραπάνω τύπο μπορεί να διαπιστώσει κανείς πως το D_{ip} καθορίζεται από τη σημασιολογική πρόβλεψη φ(x) και τη διαφορά γνωρίσματος Δ(α, y_1, y_2). Στο μοντέλο γίνεται χρήση l^1-κανονικοποιημένης Δ(α, y_1, y_2) ως ποσό πληροφορίας (information amount) για να βρεθεί πόση διακριτική πληροφορία (discriminative information) υπάρχει σε κάθε διάσταση γνωρίσματος (attribute dimension) στη δυαδική κατηγοριοποίηση. Για παράδειγμα το γνώρισμα i είναι πιο διακριτικό από το γνώρισμα j στην κατηγοριοποίηση μεταξύ της κλάσης y_1 και της κλάσης y_2 αν το $|\Delta(\alpha, y_1, y_2)_i|$ είναι μεγαλύτερο από το $|\Delta(\alpha, y_1, y_2)_j|$. Η κατηγοριοποίηση εξαρτάται άμεσα από μικρές υποκατηγορίες γνωρισμάτων οι οποίες περιέχουν υψηλές διακριτικές πληροφορίες.

Ένας άλλος ευρέως διαδεδομένος τρόπος για να μετρηθεί το πόσο όμοια είναι τα αντικείμενα είναι η απόσταση συνημιτόνου (cosine distance):

$$D_{\cos}\left(x, y_1, y_2\right) = \frac{1}{\| \varphi\left(x\right) \| \| \alpha_{y_1} \|} \varphi\left(x\right)^{\mathrm{T}} \Delta'\left(\alpha, y_1, y_2\right)$$

$$\Delta'\left(\alpha, y_1, y_2\right) = \alpha_{y_1} - \frac{\| \alpha_{y_1} \|}{\| \alpha_{y_2} \|} \alpha_{y_2}$$

Η απόσταση συνημιτόνου, σε σύγκριση με το απλό εσωτερικό γινόμενο, λαμβάνει υπόψιν της επιπλέον l^2 - κανονικοποιήσεις (l^2-norm) , όπου κλάσεις με μεγάλο κανονικοποιημένο γνώρισμα (attribute norm) είναι δυσμενείς στη διακρισιμότητα (discrimination).

Ενώ και οι δύο παραπάνω τρόποι αποδίδουν καλά γενικά, αποτυγχάνουν στις περιπτώσεις των σημασιολογικά διφορούμενων αντικειμένων. Εκτός από το πρόβλημα αυτών των αντικειμένων να κατηγοριοποιούνται βάση λανθασμένων κριτηρίων, όπως ειπώθηκε στην αρχή, δημιουργείται και ένα μεγαλύτερο πρόβλημα λόγω της συσχέτισης που υπάρχει στις διαστάσεις των γνωρισμάτων που αυξάνουν και άλλο την ασάφεια (πχ. το γνώρισμα "βούλες" σχετίζεται συνήθως με το γνώρισμα "άσπρο" και "μαύρο"). Από αυτή τη παρατήρηση προκύπτει ότι το χάσμα των σημασιολογικών προβλέψεων ανάμεσα σε διάφορες διαστάσεις θα πρέπει να μειωθεί έτσι ώστε μία ή λίγες διακεκριμένες προβλέψεις γνωρισμάτων να μην κυριαρχούν στην κατηγοριοποίηση.

Ο τύπος για τη συμβατική βαθμολογία με τη προτεινόμενη προσοχή γνωρισμάτων βάση αντικειμένων μπορεί να γραφτεί ως:

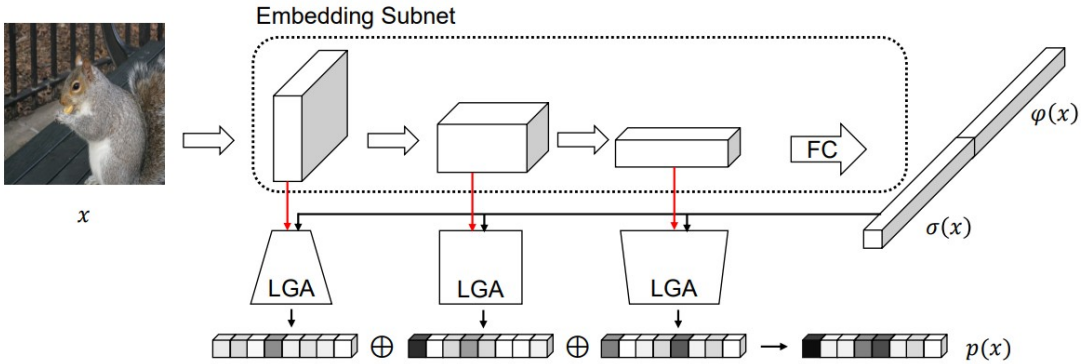
$$F'\left(x, y; W\right) = \theta\left(x\right)^{\mathrm{T}} W diag\left(p\left(x\right)\right) \alpha_y$$

όπου p(x) είναι η προτεινόμενη προσοχή και W είναι η παράμετρος για την οπτικό-σημασιολογική προβολή (visual-semantic projection). Η διαφορά αυτής της μεθόδου σε σχέση με το παραδοσιακό ZSL είναι ότι η προσοχή p(x) είναι με βάση το αντικείμενο και έτσι εκμεταλλεύεται ακόμη και τη χαμηλού επιπέδου οπτική πληροφορία. Επίσης σε αυτή τη

μέθοδο η προτεινόμενη προσοχή μαθαίνει ανεξάρτητη από το σημασιολογικό χώρο, γιατί η μάθηση που γίνεται μόνο στο σημασιολογικό χώρο κάνει τους πίνακες προβολής (projection matrices) άμεσα συνδεδεμένους με τα σημασιολογικά γνωρίσματα με αποτέλεσμα στις απαιτητικές περιπτώσεις των σημασιολογικά διφορούμενων αντικειμένων να δημιουργείται θέμα ασάφειας σε αυτό το χώρο.

Το μοντέλο LFGAA

Από το πρόβλημα σημασιολογικής ασάφειας προκύπτει ότι η προσοχή γνωρίσματος σχετίζεται άμεσα με τις καθολικές κατηγορίες γνωρισμάτων και την χαμηλού επιπέδου οπτική πληροφορία.



Εικόνα 13: Αρχιτεκτονική μοντέλου LFGAA
Πηγή Εικόνας:[47]

Στη παραπάνω εικόνα φαίνεται η δομή του μοντέλου Latent Feature Guided Attention (LFGAA). Το μοντέλο δέχεται την εικόνα και με τη χρήση του υποδικτύου ενσωμάτωσης (Embedding Subnet) γίνεται η εξαγωγή των οπτικών πληροφοριών. Το υποδίκτυο ενσωμάτωσης μαθαίνει προβολές από τον οπτικό χώρο, στον σημασιολογικό χώρο που έχει ορίσει ο χρήστης και στον χώρο κρυφού χαρακτηριστικού, ταυτόχρονα. Το υποδίκτυο ενσωμάτωσης χωρίζεται σε διάφορα τμήματα ανάλογα με τα πεδία που αντιστοιχούν. Σε κάθε τμήμα εφαρμόζεται ένα μόντουλο Κρυφής Καθοδηγούμενης Προσοχής (Latent Guided Attention - LGA) το οποίο ενώνει την οπτική πληροφορία με τις καθολικές κατηγορίες γνωρισμάτων. Για κάθε εικόνα που δέχεται το μοντέλο παράγει ταυτόχρονα τη σημασιολογική πρόβλεψη φ(x), τη πρόβλεψη κρυφού χαρακτηριστικού σ(x) και τη σημασιολογική προσοχή γνωρίσματος p(x).

Υποδίκτυο Ενσωμάτωσης Γνωρίσματος

Ενώ άλλοι μέθοδοι ZSL χρησιμοποιούν προ εκπαιδευμένα βαθιά CNN χαρακτηριστικά ως τις οπτικές τους αναπαραστάσεις, το μοντέλο LFGAA αξιοποιεί με τον καλύτερο τρόπο το βασικότερο (backbone) CNN, αλλά και άλλα μέρη του Υποδικτύου Ενσωμάτωσης. Τα χαρακτηριστικά της εικόνας που εξάγονται από το βασικότερο CNN τροφοδοτούνται σε διάφορα πλήρη συνδεδεμένα επίπεδα με ενεργοποίηση ReLU ώστε να προβληθούν μη-

γραμμικά στο σημασιολογικό και στον κρυφό χώρο χαρακτηριστικών αντίστοιχα. Στη συνέχεια τα κρυφά χαρακτηριστικά χρησιμοποιούνται στα LGA μόντουλα, αλλά και για να κάνουν προβλέψεις στον κρυφό χώρο. Παράλληλα τα σημασιολογικά χαρακτηριστικά συγκρίνονται με τις περιγραφές των γνωρισμάτων μέσω της προσοχής γνωρίσματος για να γίνουν οι σημασιολογικές προβλέψεις.

Μόντουλο Κρυφής Καθοδηγούμενης Προσοχής

Η προσοχή γνωρίσματος βάση αντικειμένου έχει ως κυριότερο στόχο την αντιμετώπιση των σημασιολογικά διφορούμενων αντικειμένων. Για αυτό το λόγο το μοντέλο μαθαίνει και στο σημασιολογικό χώρο, αλλά και στον κρυφό χώρο. Ακόμη και με αυτόν τον τρόπο όμως η απόδοσή του παραμένει καλή και σε γενικότερες περιπτώσεις. Το LGA μόντουλο προκύπτει από τη διαίσθηση ότι η προσοχή γνωρίσματος σχετίζεται με καθολικά χαρακτηριστικά επιπέδου κλάσης και με πληροφορία από άλλα οπτικά επίπεδα. Η προτεινόμενη προσοχή γνωρίσματος υπολογίζεται από το LGA με την εξής διαδικασία. Αρχικά ένας χάρτης οπτικού χαρακτηριστικού (visual feature map) $M_{i,j} \in R^{C \times H \times W}$, ο οποίος είναι για την ι-οστη εικόνα στο επίπεδο l του Υποδικτύου Ενσωμάτωσης, χαρτογραφείται μέσα από ένα σετ κανονικών συνελικτικών επιπέδων F για να παραχθεί το $M'_{i,j} \in R^{k \times H' \times W'}$ το οποίο μοιράζεται τις ίδιες διαστάσεις καναλιού (channel dimensions) με αυτές των κρυφών χαρακτηριστικών.

$$M'_{i,l} = F\left(M_{i,l}\right)$$

Ανεξάρτητα από το επίπεδο l όλα τα LGA μόντουλα έχουν τα ίδια H' και W'.

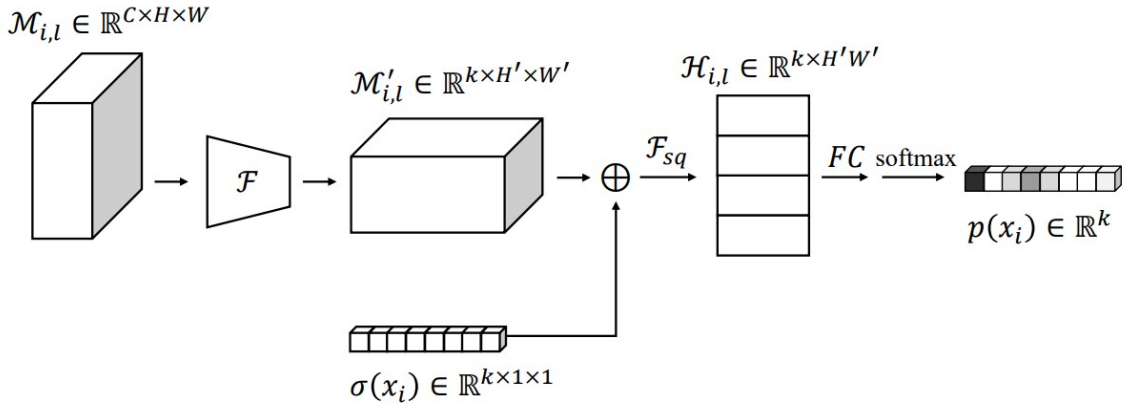
Στη συνέχεια το $M'_{i,j}$ συνδυάζεται με την ενσωμάτωση του κρυφού χαρακτηριστικού $\sigma(x_i)$ συμπληρώνω του $M'_{i,l}$ και προκύπτει η προσοχή γνωρίσματος:

$$H_{i,l} = F_{sq}\left(M'_{i,l} \oplus \sigma\left(x_i\right)\right)$$

$$p_l\left(x_i\right) = \text{softmax}\left(W_l H_{i,l} + b_l\right)$$

όπου $V \in R^{C \times HW}$ είναι η συνάρτηση συμπίεσης (squeeze function) που μετατρέπει τον χάρτη γνωρίσματος $M \in R^{C \times H \times W}$ σε διάνυσμα γνωρισμάτων $V \in R^{C \times HW}$ και $\oplus$ είναι η πρόσθεση από άποψη καναλιών (channel wise addition). W_l kai b_l είναι οι παράμετροι ενός πλήρους συνδεδεμένου επιπέδου για το συγκεκριμένο τμήμα του επιπέδου l που κατασκευάζει τοπική οπτική πληροφορία.

Κάθε επίπεδο έχει τη δική του προσοχή γνωρίσματος, οπότε για ένα πιο ολοκληρωμένο αποτέλεσμα ενώνονται πολλές προσοχές γνωρίσματος από διάφορα επίπεδα.



Εικόνα 14: Αναπαράσταση Κρυφής Καθοδηγούμενης Προσοχής (LGA)
Πηγή Εικόνας: [47]

Βελτιστοποίηση Μοντέλου

Στο συγκεκριμένο μοντέλο η οπτική-σημασιολογική προβολή και η οπτική-κρυφή προβολή βελτιστοποιούνται την ίδια στιγμή. Για την οπτική-κρυφή προβολή γίνεται χρήση τρίδυμης απώλειας (triplet loss) για τη μάθηση διακριτικής κρυφής κατηγορίας χαρακτηριστικών αυξάνοντας τη διαταξική απόσταση και ταυτόχρονα μειώνοντας την ενδοταξική απόσταση:

$$L_F = \frac{1}{N}\sum_{i}^{N}\left[\| \sigma(x_i) - \sigma(x_j) \|^2 - \| \sigma(x_i) - \sigma(x_k) \|^2 + \alpha\right]_+$$

όπου x_i είναι ο σταθεροποιητής, x_j το θετικό δείγμα και x_k το αρνητικό δείγμα. Το σύμβολο $[o]_+$ ισοδυναμεί με το max(o, 0). Το α είναι μια προκαθορισμένη παράμετρος για τον έλεγχο του επιθυμητού ορίου μεταξύ των ζευγαριών x_i-x_j και x_i-x_k.

Για την οπτική -σημασιολογική προβολή η μάθηση για τις παραμέτρους στο Υποδίκτυο Ενσωμάτωσης και στο LGA μόντουλο γίνεται από κοινού με τη χρήση της απώλειας softmax:

$$L_A = -\frac{1}{N}\sum_{i}^{N}\log\frac{\exp\left(\varphi(x_i)^{\mathrm{T}}\, diag\left(p(x_i)\right)\alpha_{y_i}\right)}{\sum_{y\in y^S}\exp\left(\varphi(x_i)^{\mathrm{T}}\, diag\left(p(x_i)\right)\alpha_{y}\right)}$$

Οι δύο παραπάνω τύποι συνδυάζονται με ένα παράγοντα εξισορρόπησης β δίνοντας τον τελικό στόχο βελτιστοποίησης του μοντέλου, ο οποίος γράφεται ως:

$$L = L_F + \beta L_A$$

Μοντέλο Αυτο-προσαρμογής (Self-Adaptation)

Στο μοντέλο LFGAA εξετάζονται και η επαγωγική και η μεταδοτική κατασκευή πρωτοτύπου για να αποδειχθεί η αποδοτικότητα της προσοχής γνωρίσματος. Στο υβριδικό LFGAA γίνεται η επαγωγική πρόβλεψη ZSL. Σε αυτήν όμως υπάρχει το πρόβλημα του domain shift.

Στο ίδιο paper προτείνεται και το LFGAA μοντέλο αυτο-προσαρμογής (Self-Adaptation) για το μεταδοτικό ZSL. Δύο είναι οι βασικές ιδέες για αυτο-προσαρμογή:

1)Τα δείγματα θα πρέπει να βρίσκονται κοντά στα πρωτότυπα που τους αντιστοιχούν.

2) Τα δείγματα που είναι κοντά στο σημασιολογικό χώρο τείνουν να είναι κοντά στο κρυφό χώρο.

Η απόσταση για την απόσταση μεταξύ των δειγμάτων υπολογίζεται με την ομοιότητα συνημιτόνου (cosine similarity).

Η εκδοχή του μοντέλου με αυτο-προσαρμογή κάνει τις παρακάτω ενέργειες ταυτόχρονα:

• Κατασκευάζει πρωτότυπα κρυφών χαρακτηριστικών
• Διορθώνει σημασιολογικά πρωτότυπα
• Κάνει προβλέψεις

Αυτές ήταν κάποιες από τις πολλές μεθόδους ZSL/GZSL που έχουν αναπτυχθεί τα τελευταία χρόνια. Για περισσότερες πληροφορίες για την κάθε μέθοδο μπορείτε να απευθυνθείτε στις αντίστοιχες πηγές.

Κεφάλαιο 4: Εκτέλεση των ZSL/GZSL μεθόδων

4.1 PyTorch

Οι μέθοδοι ZSL/GZSL που αναλύθηκαν στο προηγούμενο κεφάλαιο έχουν υλοποιηθεί με το PyTorch. Το PyTorch είναι ένα open source framework μηχανικής μάθησης που βασίζεται στη προγραμματιστική γλώσσα Python και στη βιβλιοθήκη Torch. Το Torch είναι μια open source βιβλιοθήκη μηχανικής μάθησης για τη δημιουργία βαθιών νευρωνικών δικτύων. [48,49]

Το PyTorch υποστηρίζει πάνω από 200 διαφορετικούς μαθηματικούς υπολογισμούς. Λόγω του ότι είναι απλό και εύκολο στη χρήση του, η δημοτικότητά του συνεχώς αυξάνεται. Χρησιμοποιείται κυρίως από επιστήμονες δεδομένων (data scientists) για έρευνα και εφαρμογές σε θέματα τεχνικής νοημοσύνης. Είναι ένα από τα πιο διαδεδομένα frameworks βαθιάς μάθησης λόγω της ευελιξίας του και της υπολογιστικής ισχύος του. Το PyTorch

αναδεικνύει τα μοναδικά χαρακτηριστικά της Python ώστε να γραφεί πιο ευανάγνωστος κώδικας και δίνει τη δυνατότητα για τη χρήση γραφημάτων δυναμικού υπολογισμού (dynamic computation graphs). Αυτό είναι που το κάνει ανταγωνιστικό σε σύγκριση με ένα άλλο διαδεδομένο framework βαθιάς μάθησης, το TensorFlow. Επίσης επιτρέπει στους χρήστες του να τρέξουν και να δοκιμάσουν μέρος του κώδικά τους προτού ολοκληρωθεί το πρόγραμμά τους. [48,50]

Τα δύο κυριότερα χαρακτηριστικά του PyTorch είναι:

• Υπολογισμοί Tensor με υποστήριξη επιτάχυνσης από κάρτα γραφικών.
• Αυτόματη διαφοροποίηση για τη δημιουργία και την εκπαίδευση βαθιών νευρωνικών δικτύων. [49]

4.2 Σύνολα δεδομένων που χρησιμοποιήθηκαν

Τα πειράματα έγιναν σε τρία σύνολα δεδομένων μεσαίου μεγέθους τα οποία συνηθίζονται να χρησιμοποιούνται σε θέματα που έχουν να κάνουν με το ZSL και το GZSL. Αυτά είναι το Animals with Attributes 2 (AWA2)[29], το Caltech-UCSD Birds-200-2011 (CUB)[51] και το SUN Attribute (SUN)[43]. Το AWA2 αποτελείται συνολικά από 37322 εικόνες από 50 κλάσεις ζώων, όπου η κάθε κλάση περιγράφεται από 85 γνωρίσματα. Το CUB περιέχει 11788 εικόνες από 200 διαφορετικά είδη πτηνών με 312 γνωρίσματα για το κάθε είδος. Το σύνολο δεδομένων SUN περιέχει 14340 εικόνες που απεικονίζουν διάφορα τοπία. Συνολικά το SUN περιλαμβάνει εικόνες από 717 διαφορετικά τοπία με 102 γνωρίσματα για το καθένα από αυτά.

Όσον αφορά το βαθμό λεπτομέρειας για το κάθε σύνολο δεδομένων, το AWA2 χαρακτηρίζεται ως coarse-grained, ενώ τα CUB και SUN ως fine-grained. Με τον όρο coarse-grained εννοούμε ότι μπορεί να μετατραπεί ολόκληρο το σύνολο δεδομένων, αλλά όχι ένα μεμονωμένο στοιχείο του. Αντίθετα, όταν ένα σύνολο δεδομένων είναι fine-grained υπάρχει η δυνατότητα να μετατραπούν μεμονωμένα τα στοιχεία του συνόλου. [52]

Η εξαγωγή των γνωρισμάτων 2048-διαστάσεων για όλα τα σύνολα δεδομένων έγινε με το ResNet-101 και για τα τέσσερα μοντέλα που αναλύθηκαν στο προηγούμενο κεφάλαιο και στα οποία έγιναν τα πειράματα. Ο διαχωρισμός των κλάσεων σε γνωστές και άγνωστες κλάσεις για κάθε σύνολο δεδομένων για τη διεξαγωγή των πειραμάτων έγινε με το διαχωρισμό που προτείνεται (Proposed Split – PS) σύμφωνα με την έρευνα των Yongqin Xian, Christoph H. Lampert, Bernt Schiele και Zeynep Akata, με τίτλο “Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly”. [29]

			Number of Classes				Number of Images								
								At Training Time					At Evaluation Time		
								SS		PS			SS		PS
Dataset	Size	Granularity	Att	$\mathcal{Y}$	$\mathcal{Y}^{tr}$	$\mathcal{Y}^{ts}$	Total	$\mathcal{Y}^{tr}$	$\mathcal{Y}^{ts}$	$\mathcal{Y}^{tr}$	$\mathcal{Y}^{ts}$	$\mathcal{Y}^{tr}$	$\mathcal{Y}^{ts}$	$\mathcal{Y}^{tr}$	$\mathcal{Y}^{ts}$
SUN [16]	medium	fine	102	717	580 + 65	72	14340	12900	0	10320	0	0	1440	2580	1440
CUB [17]	medium	fine	312	200	100 + 50	50	11788	8855	0	7057	0	0	2933	1764	2967
AWA1 [1]	medium	coarse	85	50	27 + 13	10	30475	24295	0	19832	0	0	6180	4958	5685
AWA2	medium	coarse	85	50	27 + 13	10	37322	30337	0	23527	0	0	6985	5882	7913
aPY [18]	small	coarse	64	32	15 + 5	12	15339	12695	0	5932	0	0	2644	1483	7924

Εικόνα 15: Χαρακτηριστικά και προτεινόμενος διαχωρισμός συνόλων δεδομένων
Εικόνα από [29] που δείχνει πίνακα με τα χαρακτηριστικά των συνόλων δεδομένων και το προτεινόμενο διαχωρισμό (PS) κατά τη διάρκεια της εκπαίδευσης και της αξιολόγησης των μοντέλων.

4.3 Πρωτόκολλο Αξιολόγησης

Οι μέθοδοι CADA-VAE, TF-VAEGAN, CE-GZSL και LFGAA βασίζονται στη στρατηγική αξιολόγησης που προτείνεται στην έρευνα "Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly"[33]. Για τη περίπτωση του συμβατικού Zero-Shot Learning αξιολογείται μόνο η top-1 ακρίβεια κάθε κλάσης στις άγνωστες κλάσεις, ενώ για τη περίπτωση του Generalized Zero-Shot Learning αξιολογείται η top-1 ακρίβεια στις γνωστές και στις άγνωστες κλάσεις. Οι γνωστές κλάσεις συμβολίζονται με *S* και οι άγνωστες κλάσεις με *U*. Η επίδοση του GZSL μετριέται από τον αρμονικό μέσο όρο (harmonic mean) μεταξύ των γνωστών και των άγνωστων κλάσεων. Ο αρμονικός μέσος όρος υπολογίζεται από τον τύπο:

$$H = \frac{2 * S * U}{(S + U)}$$

4.4 Μεθοδολογία για την εκτέλεση των πειραμάτων

Όπως αναφέρθηκε και στην αρχή του κεφαλαίου όλες οι μέθοδοι που επιλέχθηκαν έχουν υλοποιηθεί με PyTorch. Τα πειράματα στις μεθόδους CADA-VAE, CE-GZSL και LFGAA εκτελέστηκαν με Python 3.7 και στη μέθοδο TF-VAEGAN με Python 3.6. Στον κώδικα της κάθε μεθόδου μπορείτε να βρείτε συγκεκριμένα τα προαπαιτούμενα και τον τρόπο εκτέλεσής τους. Επίσης να σημειωθεί πως τα πειράματα για τη μέθοδο LFGAA έγιναν με το μοντέλο κατηγοριοποίησης εικόνων VGG19.

Και τα τέσσερα μοντέλα των μεθόδων που εξετάζουμε υποστηρίζουν cuda, δηλαδή υπάρχει η δυνατότητα να χρησιμοποιηθεί η κάρτα γραφικών για την επεξεργασία των δεδομένων, μειώνοντας έτσι το χρόνο που χρειάζονται για την εκτέλεσή τους. Βέβαια αν δεν μπορεί να υποστηριχθεί αυτό από το σύστημα που εκτελείται το μοντέλο ή αν επιθυμεί ο χρήστης να μην αξιοποιήσει αυτή τη δυνατότητα για κάποιο λόγο, τότε θα χρησιμοποιηθεί ο επεξεργαστής του συστήματος. Τα αποτελέσματα που αναγράφονται παρακάτω είναι από εκτελέσεις των μοντέλων με χρήση cuda.

4.5 Ρύθμιση Υπερπαραμέτρων για τις Μεθόδους

Η ρύθμιση των υπερπαραμέτρων είναι μια απαιτητική και χρονοβόρα διαδικασία, όμως είναι πολύ σημαντική για τις επιδόσεις των μοντέλων. Για να βρεθούν οι υπερπαράμετροι για κάποιο μοντέλο ώστε να έχει τα καλύτερα δυνατά αποτελέσματα μπορεί να χρειαστούν μέρες ή ακόμη και μήνες. Η κάθε μία μέθοδος που αναλύθηκε έχει τις δικές της υπερπαραμέτρους, όμως υπάρχουν και κοινές για όλες τις μεθόδους. Υπερπαράμετροι που έχουν όλες οι μέθοδοι και επηρεάζουν άμεσα τα αποτελέσματα είναι ο αριθμός των epochs, το batch size και το learning rate.

Epoch: Με τον όρο epoch (εποχή) αναφερόμαστε στο πόσες φορές "περνάει" από τον αλγόριθμο το σύνολο δεδομένων εκπαίδευσης.

Batch Size: Είναι ο αριθμός των δειγμάτων εκπαίδευσης που χρησιμοποιούνται σε μία επανάληψη.

Learning Rate: Ελέγχει το πόσο αλλάζουν τα βάρη του μοντέλου κατά τη διάρκεια της εκπαίδευσης. Συνήθως η τιμή αυτής της υπερπαραμέτρου κυμαίνεται από 0 έως 1. Είναι σημαντικό να βρεθεί μια τιμή για το ρυθμό μάθησης όπου με την οποία το μοντέλο θα εκπαιδεύεται σωστά.

	Epochs			Batch Size			Learning Rate		
	AWA2	CUB	SUN	AWA2	CUB	SUN	AWA2	CUB	SUN
CADA-VAE	100	100	100	50	50	50	0.001	0.001	0.001
TF-VAEGAN	120	300	400	64	64	64	0.001	0.001	0.001
CE-GZSL	130	450	1000	150	150	150	1e-4	1e-4	5e-5
LFGAA	12	12	12	32	32	32	1e-5	1e-5	1e-5

Πίνακας 1: Τιμές υπερπαραμέτρων που χρησιμοποιήθηκαν

Τιμές από κάποιες κοινές υπερπαραμέτρους των μεθόδων που χρησιμοποιήθηκαν για τα πειράματα

4.6 Σύγκριση των αποτελεσμάτων

Method	AWA2	CUB	SUN
CADA-VAE	58.56	57.18	60.97
CE-GZSL	69.27	**77.08**	61.36
TF-VAEGAN	69.91	63.27	**65.16**
LFGAA (Hybrid)	66.2	66.47	59.62
LFGAA (SA)	**73.91**	75.58	59.06

Πίνακας 2: Αποτελέσματα των μεθόδων σε top-1 ορθότητα στο ZSL

Στον Πίνακα 2 καταγράφονται τα αποτελέσματα των μεθόδων στα πειράματα ZSL που έγιναν. Στο σύνολο δεδομένων AWA2 το μοντέλο LFGAA (SA) έχει την καλύτερη επίδοση (73.9%) και ήταν το μόνο με επίδοση μεγαλύτερη από 70%. Παρόμοιες επιδόσεις έχουν τα

μοντέλα CE-GZSL και TF-VAEGAN, με το δεύτερο να είναι ελάχιστα καλύτερο. Κοντά σε αυτά τα δύο μοντέλα, με διαφορά μικρότερη από 4%, είναι η άλλη εκδοχή του LFGAA, το υβριδικό μοντέλο του. Το μοντέλο CADA-VAE έχει τη χειρότερη επίδοση και είναι το μόνο που δεν κατάφερε να ξεπεράσει το 60%. Το μοντέλο LFGAA (SA) συνέχισε να έχει πολύ καλή επίδοση και για το σύνολο δεδομένων CUB, όμως έρχεται δεύτερο, αφού το μοντέλο CE-GZSL έχει την καλύτερη επίδοση εδώ με 77%. Για άλλη μια φορά το μοντέλο CADA-VAE έχει τη χαμηλότερη επίδοση από τα υπόλοιπα μοντέλα (<60%). Για το σύνολο δεδομένων SUN τα αποτελέσματα όλων των μοντέλων είναι πολύ κοντά, με τις τιμές του να κυμαίνονται περίπου στο 60%. Το μοντέλο που ξεχώρισε και έχει την καλύτερη επίδοση σε αυτό το σύνολο δεδομένων είναι το TF-VAEGAN με 65%.

	AWA2			CUB			SUN		
Method	S	U	H	S	U	H	S	U	H
CADA-VAE	77.12	56.27	65.35	54.54	48.71	51.48	36.32	45.9	40.55
CE-GZSL	76.88	**62.75**	**69.1**	66.69	**62.02**	**64.27**	36.7	**47.12**	41.26
TF-VAEGAN	72.51	56.17	63.3	61.15	50.85	55.52	**39.62**	43.78	**41.59**
LFGAA (Hybrid)	**92.75**	25.53	40.03	**78.72**	34.91	48.36	38.55	16.31	22.92
LFGAA (SA)	88.96	48.61	62.86	77.9	41.02	53.74	32.37	18.28	23.36

Πίνακας 3: Αποτελέσματα των μεθόδων σε top-1 ορθότητα στο GZSL

Αποτελέσματα γνωστών (*S*) και άγνωστων (*U*) κλάσεων των μεθόδων σε top-1 ορθότητα στο Generalized Zero Shot Learning. Με *H* συμβολίζεται ο αρμονικός μέσος των *S* και *U*.

Στον Πίνακα 3 καταγράφονται τα αποτελέσματα των μεθόδων στα πειράματα για την πιο απαιτητική περίπτωση GZSL, όπου *S* τα αποτελέσματα στις γνωστές κλάσεις, *U* τα αποτελέσματα στις άγνωστες κλάσεις και *H* ο αρμονικός μέσος. Εντυπωσιακή είναι η επίδοση του LFGAA (Hybrid) στις γνωστές κλάσεις του AWA2 (92.7%) καθώς είναι και το μόνο αποτέλεσμα και για τα τρία σύνολα δεδομένων που ξεπέρασε το 90%. Αντίθετα, το ίδιο μοντέλο για τις άγνωστες κλάσεις του AWA2 έχει τη χειρότερη επίδοση με διαφορά, με μόλις 25.5%. Εξίσου αδύναμο σε αυτή την κατηγορία παρουσιάζεται και το LFGAA (SA) έχοντας τη δεύτερη χειρότερη επίδοση με 48.6%. Το μοντέλο με την καλύτερη επίδοση στις άγνωστες κλάσεις και το μόνο με αποτέλεσμα >60% είναι το CE-GZSL με 62.7%. Αυτό είναι και το μοντέλο που έχει και τον καλύτερο αρμονικό μέσο όρο σε αυτό το σύνολο δεδομένων με 69.1%. Ακριβώς το ίδιο μοτίβο παρατηρείται και στα αποτελέσματα των μεθόδων στο σύνολο δεδομένων CUB. Καλύτερη απόδοση στις γνωστές κλάσεις έχει το LFGAA (Hybrid) με 78.7%, ενώ στις άγνωστες κλάσεις και στον αρμονικό μέσο όρο το μοντέλο CE-GZSL είναι πρώτο με 62% και 64.2% αντίστοιχα. Στο τρίτο και τελευταίο σύνολο δεδομένων όπου δοκιμάστηκαν τα μοντέλα, υπάρχει μια εμφανής πτώση στα αποτελέσματα σε σύγκριση με αυτά των δύο προηγούμενων συνόλων δεδομένων. Για τις γνωστές κλάσεις του SUN οι επιδόσεις όλων των μοντέλων ήταν κοντά. Το ίδιο κοντά παρέμειναν και τα αποτελέσματα των CADA-VAE, CE-GZSL και TF-VAEGAN για τις άγνωστες κλάσεις και για τον αρμονικό μέσο όρο, ενώ οι δύο εκδοχές του LFGAA με παραπλήσια αποτελέσματα μεταξύ τους έχουν τις χειρότερες επιδόσεις σε αυτές τις δύο κατηγορίες. Συγκεκριμένα οι καλύτερες επιδόσεις στο SUN είναι το TF-VAEGAN στις γνωστές κλάσεις με 39.6%, το CE-GZSL στις άγνωστες κλάσεις με 47.1% και τέλος το μοντέλο TF-VAEGAN πάλι στον αρμονικό μέσο όρο με 41.5%.

Όσον αφορά το χρόνο εκπαίδευσης που χρειάζεται το κάθε μοντέλο, το πιο γρήγορο με μεγάλη διαφορά είναι το μοντέλο CADA-VAE, ακολουθούν τα μοντέλα TF-VAEGAN και CE-GZSL με παρόμοιους χρόνους και τέλος το πιο αργό με διαφορά είναι το μοντέλο LFGAA.

Επομένως από την ανάλυση των αποτελεσμάτων στα πειράματα προκύπτει ότι η απόδοση των μοντέλων επηρεάζεται άμεσα από τις υπερπαραμέτρους τους και από την πολυπλοκότητα των δεδομένων που έχουν να επεξεργαστούν. Επίσης διαπιστώνουμε ότι οι επιδόσεις των μοντέλων από τις μεθόδους στα σύνολα δεδομένων που έγιναν τα πειράματα είναι παρόμοιες σε αρκετές περιπτώσεις, εκτός από μερικές εξαιρέσεις. Άρα ο χρήστης θα πρέπει να κρίνει ανάλογα με τις επιθυμίες του και τον σκοπό που θέλει να επιτύχει αν επιλέξει ένα μοντέλο που εκτελείται γρήγορα αλλά με χαμηλότερη απόδοση από ένα πιο αργό αλλά που να έχει μεγαλύτερη ακρίβεια, ή το ανάποδο.

Κεφάλαιο 5: Συμπεράσματα και Μελλοντικές Προεκτάσεις

Όπως είδαμε, η ιδέα για κατηγοριοποίηση εικόνων υπάρχει εδώ και πολλά χρόνια. Πιο συγκεκριμένα, η παρούσα πτυχιακή εργασία εστίασε στην κατανόηση του προβλήματος του Zero Shot Learning και της πιο απαιτητικής περίπτωσής του, το Generalized Zero Shot Learning. Υπάρχουν πολλές διαφορετικές προσεγγίσεις για αυτό το πρόβλημα από τις οποίες αναλύσαμε τέσσερις που επιλέχθηκαν. Κάθε μία μέθοδος αποτελεί και μία περίπλοκη αρχιτεκτονική. Επίσης ασχοληθήκαμε με τρία από τα πιο διαδεδομένα σύνολα δεδομένων που χρησιμοποιούνται στις μεθόδους GZSL. Στη συνέχεια εκτελέσαμε τις μεθόδους που αναλύθηκαν με αυτά τα τρία σύνολα δεδομένων και έγινε μια ανάλυση των αποτελεσμάτων τους.

Από αυτές τις εκτελέσεις συμπεράναμε ότι τα αποτελέσματα των μεθόδων εξαρτώνται άμεσα από τις παραμέτρους και τις υπερπαραμέτρους τους. Συνεπώς η πιο απλή πρόταση για τη βελτίωση αυτών των μεθόδων είναι η καλύτερη ρύθμιση των παραμέτρων και υπερπαραμέτρων τους έτσι ώστε να παράγουν τα καλύτερα δυνατά αποτελέσματα με αυτές.

Είδαμε ότι στις περιπτώσεις ZSL και GZSL τα δεδομένα εκπαίδευσης είναι ξεχωριστά από τα δεδομένα αξιολόγησης και οι προβλέψεις γίνονται μόνο με βάση των ετικετών των κλάσεων. Οπότε μια άλλη πρόταση για τη βελτίωση μελλοντικών μεθόδων ZSL/GZSL είναι η χρήση ετικετών με μοναδικά χαρακτηριστικά για τις κλάσεις που περιγράφουν, ώστε να μην υπάρχουν αντιθέσεις κατά την κατηγοριοποίησή τους.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Boesch, G. (2022, June 5). *A Complete Guide to Image Classification in 2022*. Viso.Ai. https://viso.ai/computer-vision/image-classification/
2. Reed, N. (2020, July 28). ELI5: What is image classification in deep learning? *ThinkAutomation*. https://www.thinkautomation.com/eli5/eli5-what-is-image-classification-in-deep-learning/
3. *Image Classification in Machine Learning [Intro + Tutorial]*. (n.d.). Retrieved August 20, 2022, from https://www.v7labs.com/blog/image-classification-guide, https://www.v7labs.com/blog/image-classification-guide
4. *Basics of Machine Learning Image Classification Techniques*. (2019, August 15). OpenGenus IQ: Computing Expertise & Legacy. https://iq.opengenus.org/basics-of-machine-learning-image-classification-techniques/
5. Qiao, R., Liu, L., Shen, C., & Van Den Hengel, A. (2016). Less is More: Zero-Shot Learning from Online Textual Documents with Noise Suppression. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2249–2257. https://doi.org/10.1109/CVPR.2016.247
6. Romera-Paredes, B., & Torr, P. H. S. (2017). An Embarrassingly Simple Approach to Zero-Shot Learning. In R. S. Feris, C. Lampert, & D. Parikh (Eds.), *Visual Attributes* (pp. 11–30). Springer International Publishing. https://doi.org/10.1007/978-3-319-50077-5_2
7. Han, Z., Fu, Z., Chen, S., & Yang, J. (2021). Contrastive Embedding for Generalized Zero-Shot Learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2371–2381. https://doi.org/10.1109/CVPR46437.2021.00240
8. *What is Machine Learning?* (2022, July 6). https://www.ibm.com/cloud/learn/machine-learning
9. *What Is Machine Learning and Why Is It Important?* (n.d.). SearchEnterpriseAI. Retrieved August 26, 2022, from https://www.techtarget.com/searchenterpriseai/definition/machine-learning-ML

10. Bento, C. (2021, September 30). *Multilayer Perceptron Explained with a Real-Life Example and Python Code: Sentiment Analysis*. Medium. https://towardsdatascience.com/multilayer-perceptron-explained-with-a-real-life-example-and-python-code-sentiment-analysis-cb408ee93141
11. Wilamowski, B. (2009). Neural network architectures and learning algorithms. *IEEE Industrial Electronics Magazine*, *3*(4), 56–63. https://doi.org/10.1109/MIE.2009.934790
12. Abdi, H. (1994). *A Neural Network Primer*.
13. *Neural Network Definition*. (n.d.). Investopedia. Retrieved August 22, 2022, from https://www.investopedia.com/terms/n/neuralnetwork.asp
14. Shin, T. (2020, June 3). *A Beginner-Friendly Explanation of How Neural Networks Work*. Medium. https://towardsdatascience.com/a-beginner-friendly-explanation-of-how-neural-networks-work-55064db60df4
15. Raschka, S. (n.d.). *A Brief Summary of with Applications in Python the History of Neural Networks and Deep Learning*.
16. Bhardwaj, A. (2020, October 12). *What is a Perceptron? – Basics of Neural Networks*. Medium. https://towardsdatascience.com/what-is-a-perceptron-basics-of-neural-networks-c4cfea20c590
17. Kruse, R., Mostaghim, S., Borgelt, C., Braune, C., & Steinbrecher, M. (2022). Multi-layer Perceptrons. In R. Kruse, S. Mostaghim, C. Borgelt, C. Braune, & M. Steinbrecher (Eds.), *Computational Intelligence: A Methodological Introduction* (pp. 53–124). Springer International Publishing. https://doi.org/10.1007/978-3-030-42227-1_5
18. Riedmiller, M. (n.d.). *Advanced Supervised Learning in Multi-layer Perceptrons—From Backpropagation to Adaptive Learning Algorithms*. 10.
19. Albawi, S., Bayat, O., Al-Azawi, S., & Ucan, O. N. (2018). Social Touch Gesture Recognition Using Convolutional Neural Network. *Computational Intelligence and Neuroscience*, *2018*, 6973103. https://doi.org/10.1155/2018/6973103
20. O'Shea, K., & Nash, R. (2015). *An Introduction to Convolutional Neural Networks* (arXiv:1511.08458). arXiv. http://arxiv.org/abs/1511.08458

21. Wu, J. (n.d.). *Introduction to Convolutional Neural Networks*. 31.

22. Pokhrel, S. (2019, September 20). *Beginners Guide to Understanding Convolutional Neural Networks*. Medium. https://towardsdatascience.com/beginners-guide-to-understanding-convolutional-neural-networks-ae9ed58bb17d

23. *Image classification | TensorFlow Lite*. (n.d.). TensorFlow. Retrieved August 23, 2022, from https://www.tensorflow.org/lite/examples/image_classification/overview

24. Wei, J. (2020, September 25). *AlexNet: The Architecture that Challenged CNNs*. Medium. https://towardsdatascience.com/alexnet-the-architecture-that-challenged-cnns-e406d5297951

25. Wei, J. (2019, July 4). *VGG Neural Networks: The Next Step After AlexNet*. Medium. https://towardsdatascience.com/vgg-neural-networks-the-next-step-after-alexnet-3f91fa9ffe2c

26. Saket. (2018, November 16). 7 Best Models for Image Classification using Keras. *IT4nextgen*. https://www.it4nextgen.com/keras-image-classification-models/

27. Top 4 Pre-Trained Models for Image Classification | With Python Code. (2020, August 17). *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2020/08/top-4-pre-trained-models-for-image-classification-with-python-code/

28. Feng, V. (2017, July 17). *An Overview of ResNet and its Variants*. Medium. https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035

29. Xian, Y., Schiele, B., & Akata, Z. (2017). Zero-Shot Learning—The Good, the Bad and the Ugly. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3077–3086. https://doi.org/10.1109/CVPR.2017.328

30. Domain adaptation. (2022). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Domain_adaptation&oldid=1100497531

31. Schonfeld, E., Ebrahimi, S., Sinha, S., Darrell, T., & Akata, Z. (2019). Generalized Zero- and Few-Shot Learning via Aligned Variational Autoencoders. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8239–8247. https://doi.org/10.1109/CVPR.2019.00844

32. Pourpanah, F., Abdar, M., Luo, Y., Zhou, X., Wang, R., Lim, C. P., Wang, X.-Z., & Wu, Q. M. J. (2022). A Review of Generalized Zero-Shot Learning Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–20. https://doi.org/10.1109/TPAMI.2022.3191696

33. Team, T. A. (n.d.). *70+ Image Classification Datasets from different industry… – Towards AI*. Retrieved August 24, 2022, from https://towardsai.net/p/computer-vision/70-image-classification-datasets-from-different-industry-domains-part-2-cd1af6e48eda, https://towardsai.net/p/computer-vision/70-image-classification-datasets-from-different-industry-domains-part-2-cd1af6e48eda

34. *Papers with Code—MNIST Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/mnist

35. *Papers with Code—Fashion-MNIST Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/fashion-mnist

36. *Papers with Code—ImageNet Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/imagenet

37. *Papers with Code—CIFAR-100 Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/cifar-100

38. *Papers with Code—APY Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/apy

39. *Papers with Code—AwA Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/awa-1

40. *Papers with Code—AwA2 Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/awa2-1

41. *Papers with Code—CUB-200-2011 Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/cub-200-2011

42. *Papers with Code—SUN Attribute Dataset*. (n.d.). Retrieved August 24, 2022, from https://paperswithcode.com/dataset/sun-attribute

43. Patterson, G., Xu, C., Su, H., & Hays, J. (2014). The SUN Attribute Database: Beyond Categories for Deeper Scene Understanding. *International Journal of Computer Vision*, *108*(1–2), 59–81. https://doi.org/10.1007/s11263-013-0695-z

44. *What Is Transfer Learning? [Examples & Newbie-Friendly Guide]*. (n.d.). Retrieved August 24, 2022, from https://www.v7labs.com/blog/transfer-learning-guide, https://www.v7labs.com/blog/transfer-learning-guide

45. Transfer Learning | Understanding Transfer Learning for Deep Learning. (2021, October 30). *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2021/10/understanding-transfer-learning-for-deep-learning/

46. Narayan, S., Gupta, A., Khan, F. S., Snoek, C. G. M., & Shao, L. (2020). *Latent Embedding Feedback and Discriminative Features for Zero-Shot Classification* (arXiv:2003.07833). arXiv. http://arxiv.org/abs/2003.07833

47. Liu, Y., Guo, J., Cai, D., & He, X. (2019). Attribute Attention for Semantic Disambiguation in Zero-Shot Learning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 6697–6706. https://doi.org/10.1109/ICCV.2019.00680

48. *What is PyTorch?* (n.d.). Enterprise AI. Retrieved January 21, 2023, from https://www.techtarget.com/searchenterpriseai/definition/PyTorch

49. *What is PyTorch, and How Does It Work? | Simplilearn*. (2021, December 2). Simplilearn.Com. https://www.simplilearn.com/what-is-pytorch-article

50. *Pytorch Vs Tensorflow Vs Keras: Here are the Difference You Should Know*. (2020, July 27). Simplilearn.Com. https://www.simplilearn.com/keras-vs-tensorflow-vs-pytorch-article

51. Wah, C., Branson, S., Welinder, P., Perona, P., & Belongie, S. (n.d.). *The Caltech-UCSD Birds-200-2011 Dataset*.

52. Granularity. (2023). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Granularity&oldid=1133869380

53. *What Is Zero Shot Learning in Image Classification? [Examples]*. (n.d.). Retrieved February 5, 2023, from https://www.v7labs.com/blog/zero-shot-learning-guide, https://www.v7labs.com/blog/zero-shot-learning-guide