National Technical University of Athens

School of Electrical and Computer Engineering
Division of Signals, Control & Robotics
Speech & Language Processing Lab

Cognitively Motivated Machine Learning for Dimensionality Reduction and Domain Adaptation of Speech and Language Models in Resource-Constrained Settings

Doctoral Dissertation
of
Georgios Paraskevopoulos

Athens
January 2024

This work has been partially supported by the following European Commision and National projects: Babyrobot (EU Horizon 2020, grant number: 687831), Safety4All (RESEARCH — CREATE — INNOVATE, project code: T2EDK-04248), AI4EDU (Erasmus+, contract number: 101087451). The author has also worked for the following private entities during the writing of this dissertation: Behavioral Signals Technologies Inc., Amazon Inc. No confidential material under NDA has been used for the writing of this document. The author declares no known conflicts of interest.

Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Σηματων, Ελεγχου και Ρομποτικης
Εργαστηριο Επεξεργασιας Φωνης και Λογου

Μέθοδοι Μηχανικής Μάθησης Βασισμένες στη Γνωσιακή Επιστήμη για Μείωση Διαστατικότητας και Προσαρμογή μεταξύ Πεδίων Μοντέλων Φωνής και Γλώσσας σε Περιβάλλοντα με Περιορισμένους Πόρους

Διδακτορικη Διατριβη
του
Γεωργιου Παρασκευοπουλου

Αθηνα
Ιανουαριος 2024

Η εργασία έχει χρηματοδοτηθεί μερικώς από τα παρακάτω Ευρωπαϊκά και Εθνικά έργα: Babyrobot (EU Horizon 2020, αριθμός έργου: 687831), Safety4All (ΕΡΕΥΝΩ — ΔΗΜΙΟΥΡΓΩ — ΚΑΙΝΟΤΟΜΩ, κωδικός έργου: T2EDK-04248), AI4EDU (Erasmus+, αριθμός έργου: 101087451). Ο συγγραφέας έχει εργαστεί για τις παρακάτω εταιρείες κατά τη διάρκεια της συγγραφής: Behavioral Signals Technologies Inc., Amazon Inc. Δεν έχει χρησιμοποιηθεί υλικό που υπόκειται σε συμφωνία εμπιστευτικότητας για τη συγγραφή αυτής της εργασίας. Ο συγγραφέας δηλώνει ότι δεν υπάρχουν αντικρουόμενα συμφέροντα που σχετίζονται με οποιαδήποτε από της αναφερόμενες πηγές χρηματοδότησης.

(this page is left intentionally blank)

National Technical University of Athens
School of Electrical and Computer Engineering
Division of Signals, Control & Robotics
Speech & Language Processing Lab

Cognitively Motivated Machine Learning for Dimensionality Reduction and Domain Adaptation of Speech and Language Models in Resource-Constrained Settings

Doctoral Dissertation

of

Georgios Paraskevopoulos

Advisory Committee: Alexandros Potamianos, Associate Professor (Supervisor)
Petros Maragos, Full Professor
Kostantinos Tzafestas, Associate Professor

Approved by the seven-member examination committee at 29/1/2024.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Alexandros Potamianos Associate Professor NTUA	Petros Maragos Full Professor NTUA	Kostantinos Tzafestas Associate Professor NTUA

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....	.....
Athanasios Katsamanis Principal Researcher Athena R.C.	Gerasimos Potamianos Associate Professor University of Thessaly	Athanasios Rontogiannis Associate Professor NTUA	Dimitrios Fotakis Full Professor NTUA

Αθηνα, Ιανουάριος 2024.

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Σηματων, Ελεγχου και Ρομποτικης
Εργαστηριο Επεξεργασιας Φωνης και Λογου

Μέθοδοι Μηχανικής Μάθησης Βασισμένες στη Γνωσιακή Επιστήμη για Μείωση Διαστατικότητας και Προσαρμογή μεταξύ Πεδίων Μοντέλων Φωνής και Γλώσσας σε Περιβάλλοντα με Περιορισμένους Πόρους

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

ΤΟΥ

ΓΕΩΡΓΙΟΥ ΠΑΡΑΣΚΕΥΟΠΟΥΛΟΥ

Συμβουλευτική Επιτροπή: Αλέξανδρος Ποταμιάνος, Αναπληρωτής Καθηγητής (Επιβλέπων)
Πέτρος Μαραγκός, Καθηγητής
Κωσταντίνος Τζαφέστας, Αναπληρωτής Καθηγητής

Εγκρίθηκε από την επταμελή εξεταστική επιτροπή την 29/1/2024.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Αλέξανδρος Ποταμιάνος Αναπληρωτής Καθηγητής Ε.Μ.Π.	Πέτρος Μαραγκός Καθηγητής Ε.Μ.Π.	Κωσταντίνος Τζαφέστας Αναπληρωτής Καθηγητής Ε.Μ.Π.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....	.....
Αθανάσιος Κατσαμάνης Ερευνητής Β' Ε.Κ. Αθηνά	Γεράσιμος Ποταμιάνος Αναπληρωτής Καθηγητής Πανεπιστήμιο Θεσσαλίας	Αθανάσιος Ροντογιάννης Αναπληρωτής Καθηγητής Ε.Μ.Π.	Δημήτριος Φωτάκης Καθηγητής Ε.Μ.Π.

ΑΘΗΝΑ, ΙΑΝΟΥΑΡΙΟΣ 2024.

Cognitively Motivated Machine Learning for Dimensionality Reduction and Domain Adaptation of Speech and Language Models in Resource-Constrained Settings

Abstract

In the recent years, a dominant strategy has arised in machine learning, i.e., scaling-up model capacity and training data, with impressive results. However, the development of techniques for resource-limited settings can have a great economic, environmental, and research impact, especially for digitally under-represented communities. In this thesis, which is split into two major parts, we draw motivation from insights in the fields of cognitive sciences and neurosciences to design efficient and effective machine learning algorithms for data representation and model adaptation. First, we propose a novel algorithm for dimensionality reduction via multi-dimensional scaling based on the global geometry of the input data. The proposed algorithm, Pattern Search MDS is based on derivative-free direct search, and is able to capture the geometry of complex "pseudo"-metric spaces. Reduction of the algorithm to the General Pattern Search algorithmic family provides theoretical convergence guarantees, and an optimized implementation is provided to the research community. The performance and convergence of Pattern Search MDS is demonstrated on diverse tasks, i.e., manifold geometry, semantic similarity, and speech emotion recognition. In the second part we shift our focus to the problem of Unsupervised Domain Adaptation of speech and language models. To address the inherent stability-plasticity dilemma in this problem we propose mixed self-supervision, a robust and effective fine-tuning strategy, where the task is learned using annotated out-of-domain data, while relevant in-domain knowledge from pretraining is maintained via self-supervision on unlabeled in-domain data. We evaluate mixed self-supervision for text sentiment analysis based on product reviews, and the adaptation of speech recognition systems to new domains for Modern Greek. Particular emphasis is placed on the sample-efficiency of the proposed fine-tuning strategy in our ablations, where we demonstrate that 500 in-domain reviews, or 3 hours of in-domain speech, are enough for successful adaptation.

Keywords: Unsupervised Domain Adaptation, Dimensionality Reduction, Multi-dimensional Scaling, Self-Supervised Learning, Deep Learning, Text Sentiment Analysis, Speech Emotion Recognition, Speech Recognition

Μέθοδοι Μηχανικής Μάθησης Βασισμένες στη Γνωσιακή Επιστήμη για Μείωση Διαστατικότητας και Προσαρμογή μεταξύ Πεδίων Μοντέλων Φωνής και Γλώσσας σε Περιβάλλοντα με Περιορισμένους Πόρους

Περιληψη

Τα τελευταία χρόνια, μια κυρίαρχη στρατηγική έχει προκύψει στη μηχανική μάθηση, δηλαδή η κλιμάκωση της χωρητικότητας του μοντέλου και των δεδομένων εκπαίδευσης, με εντυπωσιακά αποτελέσματα. Ωστόσο, η ανάπτυξη τεχνικών για περιβάλλοντα με περιορισμένους πόρους μπορεί να έχει μεγάλο οικονομικό, περιβαλλοντικό και ερευνητικό αντίκτυπο, ειδικά για ψηφιακά υποεκπροσωπούμενες κοινότητες. Σε αυτή τη διατριβή, η οποία χωρίζεται σε δύο κύρια μέρη, αντλούμε κίνητρα από τους τομείς των γνωσιακών επιστημών και των νευροεπιστημών για να σχεδιάσουμε αποδοτικούς και αποτελεσματικούς αλγόριθμους μηχανικής μάθησης για αναπαράσταση δεδομένων και προσαρμογή μοντέλων. Πρώτον, προτείνουμε έναν νέο αλγόριθμο για τη μείωση διαστάσεων μέσω πολυδιάστατης κλιμάκωσης με βάση τη συνολική γεωμετρία των δεδομένων εισόδου. Ο προτεινόμενος αλγόριθμος, Pattern Search MDS βασίζεται σε άμεση αναζήτηση χωρίς παραγώγους και είναι σε θέση να συλλάβει τη γεωμετρία σύνθετων “ψευδομετρικών” χώρων. Η αναγωγή του αλγορίθμου στην οικογένεια αλγορίθμων General Pattern Search παρέχει θεωρητικές εγγυήσεις σύγκλισης, ενώ παρέχεται μια βελτιστοποιημένη υλοποίηση στην ερευνητική κοινότητα. Η απόδοση και η σύγκλιση του Pattern Search MDS επιδεικνύεται σε διάφορες εργασίες, π.χ., γεωμετρία πολλαπλοτήτων, σημασιολογική ομοιότητα και αναγνώριση συναισθημάτων από φωνή. Στο δεύτερο μέρος στρέφουμε την εστίασή μας στο πρόβλημα της μη επιβλεπώμενης προσαρμογής μοντέλων λόγου και γλώσσας σε νέους τομείς. Για να αντιμετωπίσουμε το εγγενές δίλημμα σταθερότητας-πλαστικότητας σε αυτό το πρόβλημα, προτείνουμε μικτή αυτο-επίβλεψη, μια ισχυρή και αποτελεσματική στρατηγική προσαρμογής, όπου η εργασία μαθαίνεται χρησιμοποιώντας επισημειωμένα δεδομένα εκτός τομέα, ενώ σχετική γνώση εντός τομέα από την προεκπαίδευση διατηρείται μέσω αυτο-επίβλεψης σε δεδομένα εντός τομέα χωρίς ετικέτες. Αξιολογούμε τη μικτή αυτο-επίβλεψη για την ανάλυση συναισθήματος από κείμενο με βάση κριτικές προϊόντων και την προσαρμογή συστημάτων αναγνώρισης ομιλίας σε νέους τομείς για τα Νέα Ελληνικά. Ιδιαίτερη έμφαση δίνεται στην αποτελεσματικότητα της προτεινόμενης στρατηγικής προσαρμογής για λίγα δείγματα, όπου δείχνουμε ότι 500 κριτικές ή 3 ώρες ήχου εντός τομέα είναι αρκετές για επιτυχημένη προσαρμογή.

Λεξεις-Κλειδια: Μη επιβλεπόμενη Προσαρμογή Τομέα, Μείωση Διαστάσεων, Πολυδιάστατη Κλιμάκωση, Αυτο-επιβλεπόμενη Μάθηση, Βαθιά Μάθηση, Κειμενική Ανάλυση Συναισθημάτων, Αναγνώριση Συναισθημάτων Φωνής, Αναγνώριση Ομιλίας

Στους γονείς και στην αδερφή μου

Ευχαριστίες

Με το παρόν κείμενο ολοκληρώνεται μια πολυετής πορεία ως υποψήφιος διδάκτορας στο Ε.Μ.Π., και κλείνει ένα σημαντικό κεφάλαιο της ζωής μου. Αυτά τα χρόνια είχα την τύχη να γνωρίσω πολλούς αξιόλογους ανθρώπους, στους οποίους θέλω να εκφράσω την ευγνωμοσύνη μου για τις εμπειρίες που μοιραστήκαμε και το χρόνο που περάσαμε μαζί.

Αρχικά, η συνεργασία με τον Αλέξανδρο Ποταμιάνο με ωρίμασε τόσο σαν ερευνητή όσο και σαν άνθρωπο. Ο Αλέξανδρος συστηματικά λείανε πτυχές μου χρειάζονταν λείανση, αλλά και άμβλυνε πτυχές μου που χρειάζονταν άμβλυνση. Τον ευχαριστώ για την την καθοδήγησή και την ενθάρρυνση να γίνομαι συνεχώς καλύτερος όλα αυτά τα χρόνια. Επιπλέον, ευχαριστώ θερμά τον καθηγητή Πέτρο Μαραγκό για τις σύντομες αλλά πολύτιμες αλληλεπιδράσεις μας και το στωικό παράδειγμά του, καθώς και τα υπόλοιπα μέλη της επταμελούς επιτροπής μου για τις συμβουλές τους. Είμαι τέλος περήφανος να κατατάσσω ανάμεσα στους ανθρώπους που με έχουν εμπνεύσει, και συνεχίζουν να με εμπνέουν, τους ερευνητές Βασίλη Κατσούρο, Θοδωρή Γιαννακόπουλο και Νάσο Κατσαμάνη, και είμαι ευγνώμων για την εμπιστοσύνη τους και την άψογη συνεργασία μας.

Από το 2017 έως και σήμερα, είχα τη χαρά να συνεργαστώ και με πολλούς συναδέλφους (και φίλους πλέον) και σε διαφορετικά περιβάλλοντα. Ιδιαιτέρως θα ευχαριστήσω τον Ευθύμη Γ., με τον οποίο περάσαμε τα τελευταία 5 χρόνια μαζί στο γραφείο 2.1.2 (κατά κόσμο Μόρντορ), το Θύμιο Τ., τον Κωσταντίνο Κ. και τον Θοδωρή Κ. για τις δημιουργικές συνεργασίες μας, τη Νάνσυ Ζλ. για τις συζητήσεις μας, καθώς και τον Κοσμά Κ. για τη σπάνια θετική ενέργεια που έχει φέρει. Επιπλέον θα ευχαριστήσω παλιούς και νέους φίλους και συνεργάτες από το Ε.Μ.Π., το Ε.Κ. Αθηνά, την Behavioral Signals και την Amazon για τις πολλές και ευχάριστες (εργάσιμες και μεταμεσονύκτιες) ώρες που έχουμε περάσει μαζί. Ειδικά τέλος θα ευχαριστήσω όλους τους φοιτητές και τις φοιτήτριες οι οποίοι συνεργάστηκαν μαζί μου. Μάθατε από εμένα, αλλά έμαθα και εγώ από εσάς.

Συνήθως στα ακαδημαϊκά κείμενα μνημονεύονται κυρίως ακαδημαϊκοί και οι πλησίοι τους. Νιώθω την ανάγκη να ευχαριστήσω και το διοικητικό προσωπικό, και ειδικά τη Δέσποινα Κ. και την Ελένη Τ., οι οποίες έχουν κάνει επανειλημμένα τα δύσκολα εύκολα.

Τα χρόνια αυτά δεν σημαδεύονται μόνο από νέες, αλλά και από διαχρονικές φιλίες και σχέσεις. Ευχαριστώ τους Σπύρο Τ., Άγγελο Μ., Λεωνίδα Μ. και Δημήτρη Σ. γιατί όλα τα αστεία, ανησυχίες, και φιλοδοξίες που έχουμε μοιραστεί. Ξεχωριστά ευχαριστώ τη Βίκυ για τη ανεκτίμητη στήριξη της.

Τέλος, όλα όσα έχω καταφέρει ως σήμερα δε θα ήταν δυνατά χωρίς τη διαρκή στήριξη και ενθάρρυνση από την οικογένεια μου. Χρωστάω πολλά στους γονείς μου, Πέτρο και Ευγενία, καθώς και στην αδερφή μου Γιώτα, στους οποίους αφιερώνεται αυτό το κείμενο.

Γιώργος Παρασκευόπουλος,
Αθήνα, Ιανουάριος 2024

Contents

List of Figures

List of Tables

1
Εκτεταμένη Περίληψη

1.1 Μερος Ι: Ο αλγοριθμος pattern search MDS

Στο πρώτο μέρος της διατριβής παρουσιάζουμε μια νέα οπτική για τη μη γραμμική μάθηση πολλαπλοτήτων (manifold learning) χρησιμοποιώντας τεχνικές βελτιστοποίησης χωρίς παραγώγους. Συγκεκριμένα, προτείνουμε μια επέκταση της κλασικής πολυδιάστατης μεθόδου κλιμάκωσης (MDS), όπου αντί να κάνουμε gradient descent, κάνουμε δειγματοληψία και αξιολογούμε πιθανές "κινήσεις" σε μια σφαίρα σταθερής ακτίνας για κάθε σημείο του ενσωματωμένου χώρου. Μια εγγύηση σύγκλισης σταθερού σημείου μπορεί να παρουσιαστεί διατυπώνοντας τον προτεινόμενο αλγόριθμο ως παράδειγμα του πλαισίου γενικής αναζήτησης προτύπων (GPS). Η αξιολόγηση τόσο σε καθαρά όσο και σε θορυβώδη συνθετικά σύνολα δεδομένων δείχνει ότι ο προτινώμενος αλγόριθμος pattern search MDS μπορεί να συμπεράνει με ακρίβεια την εγγενή γεωμετρία πολλαπλοτήτων, ενσωματωμένων σε χώρους υψηλών διαστάσεων. Επιπλέον, πειράματα σε πραγματικά δεδομένα, για προβλήματα ταξινόμησης εικόνων, σημασιολογικής ομοιότητας, και αναγνώρισης συναισθημάτων από φωνή, δείχνουν ότι ο προτεινόμενος αλγόριθμος αποδίδει αποτελέσματα τελευταίας τεχνολογίας, ακόμη και κάτω από θορυβώδεις συνθήκες.

1.1.1 Βασικος Αλγοριθμος

Η κύρια ιδέα πίσω από τον προτεινόμενο αλγόριθμο είναι να αντιμετωπίσουμε το MDS ως ένα πρόβλημα βελτιστοποίησης χωρίς παραγώγους, χρησιμοποιώντας μια παραλλαγή της γενικής αναζήτησης μοτίβων (GPS) για την ελαχιστοποίηση μιας συνάρτησης απώλειας. Η είσοδος στην αναζήτηση μοτίβων MDS είναι ένας πίνακας αποστάσεων στόχος $N \times N$ $\mathbf{T}$ και η επιθυμητή διάσταση L του ενσωματωμένου χώρου. Μια επισκόπηση του αλγορίθμου παρουσιάζεται στον Αλγ. 1.

Η διαδικασία αρχικοποίησης του αλγορίθμου αποτελείται από: 1) τυχαία αρχικοποίηση N σημείων στον ενσωματωμένο χώρο και κατασκευή του πίνακα $\mathbf{X}^{(0)} = [\mathbf{x}_1^{(0)}, \mathbf{x}_2^{(0)}, \ldots, \mathbf{x}_N^{(0)}] \in \mathbb{R}^{N \times L}$, 2) υπολογισμός του πίνακα αποστάσεων του ενσωματωμένου χώρου $\mathbf{D}^{(0)}$, όπου το στοιχείο $d_{ij}^{(0)}$ είναι η Ευκλείδεια απόσταση μεταξύ των διανυσμάτων $\mathbf{x}_i^{(0)}$ και $\mathbf{x}_j^{(0)}$ του $\mathbf{X}^{(0)}$, και 3) υπολογισμός του αρχικού σφάλματος προσέγγισης $e^{(0)} = f(\mathbf{T}, \mathbf{D}^{(0)})$, όπου e είναι το τετραγωνικό σφάλμα MSE μεταξύ των δύο πινάκων. Η συνάρτηση f που επιδιώκουμε να ελαχιστοποιήσουμε είναι το κανονικοποιημένο τετράγωνο της νόρμα Frobenius του πίνακα $\mathbf{T} - \mathbf{D}$, δηλαδή, $f(\mathbf{T}, \mathbf{D}) = (1/N^2)||\mathbf{T} - \mathbf{D}||_F^2$.

Ομοίως, μπορεί κανείς να εκφράσει το f στοιχειωδώς ως εξής:

$$f(\mathbf{T}, \mathbf{D}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (t_{ij} - d_{ij})^2, \quad \text{where } \mathbf{T}, \mathbf{D} \in \mathbb{R}^{N \times N} \tag{1.1}$$

Algorithm 1 Pattern Search MDS

procedure MDS($\mathbf{T}$, L, $r^{(0)}$)
  $k \leftarrow 0$ ▷ k is the number of epochs
  $\mathbf{X}^{(k)} \leftarrow$ UNIFORM($N \times L$)
  $\mathbf{D}^{(k)} \leftarrow$ DISTANCE_MATRIX($\mathbf{X}^{(k)}$)
  $e^{(k)} \leftarrow f(\mathbf{T}, \mathbf{D}^{(k)})$
  $e^{(k-1)} \leftarrow +\infty$
  $r^{(k)} \leftarrow r^{(0)}$
  while $r^{(k)} > \delta$ **do**
    if $e^{(k-1)} - e^{(k)} \leq \varepsilon \cdot e^{(k)}$ **then**
      $r^{(k)} \leftarrow \frac{r^{(k)}}{2}$
    $\mathrm{S} \leftarrow$ SEARCH_DIRECTIONS($r^{(k)}$, L)
    for all $x \in \mathbf{X}^{(k)}$ **do**
      $\mathbf{X}^*, e^* \leftarrow$ OPTIMAL_MOVE($\mathbf{X}^{(k)}$,x,S,$e^{(k)}$)
      $e^{(k-1)} \leftarrow e^{(k)}$
      $e^{(k)} \leftarrow e^*$
      $\mathbf{X}^{(k)} \leftarrow \mathbf{X}^*$
    $k = k + 1$

Μετά τα βήματα αρχικοποίησης, σε κάθε εποχή (επανάληψη), λαμβάνουμε υπόψη την επιφάνεια μιας υπερσφαίρας, ακτίνας r γύρω από κάθε σημείο $\mathbf{x}_i^{(k)}$. Οι πιθανές κατευθύνσεις αναζήτησης βρίσκονται στην επιφάνεια της υπερσφαίρας κατά μήκος της ορθογώνιας βάσης του χώρου. Αυτό δημιουργεί τον πίνακα κατευθύνσεων αναζήτησης S και συνοψίζεται στον Αλγ. 2.

Algorithm 2 Define search directions

function SEARCH_DIRECTIONS(r, L)
  $\mathrm{S}^+ \leftarrow r \cdot \mathbf{I}_L$
  $\mathrm{S}^- \leftarrow -r \cdot \mathbf{I}_L$
  $\mathrm{S} \leftarrow [\frac{\mathrm{S}^+}{\mathrm{S}^-}]$
  return S

Κάθε σημείο μετακινείται απληστα, χωρίς να λαμβάνει υπόψη άλλα σημεία, κατά τη διάσταση που παράγει το ελάχιστο σφάλμα. Σε αυτό το στάδιο, λαμβάνουμε υπόψη μόνο κινήσεις που οδηγούν σε μονότονη μείωση της συνάρτησης σφάλματος. Ο Αλγ. 3 βρίσκει την ιδανική κίνηση που

ελαχιστοποιεί το $e^{(k)} = f(\mathbf{T}, \mathbf{D}^{(k)})$ για κάθε νέο σημείο $\tilde{x}$ και μετακινεί το $\mathbf{X}$ προς αυτή την κατεύθυνση. Σημειώστε ότι όταν γράφουμε $s \in \mathrm{S}$, ο πίνακας S θεωρείται ως το σύνολο διανυσμάτων γραμμών.

Algorithm 3 Find optimal move for a point

```
1:  function OPTIMAL_MOVE(X^(k), x, S, e)
2:      e* ← e
3:      for all s ∈ S do
4:          x̃ ← x + s
5:          X ← UPDATE_POINT(X^(k), x, x̃)        ▷ Update x point of X^(k) with x̃
6:          D ← DISTANCE_MATRIX(X)
7:          ẽ ← f(T, D)
8:          if ẽ < e* then
9:              e* ← ẽ
10:             X* ← X
11:     return X*, e*
```

Το προκύπτον σφάλμα e^* υπολογίζεται μετά την εκτέλεση της βέλτιστης κίνησης για κάθε σημείο στο $\mathbf{X}^{(k)}$. Εάν η μείωση του σφάλματος φτάσει σε ένα πλατό, μειώνουμε την ακτίνα αναζήτησης στο μισό και προχωράμε στην επόμενη εποχή. Αυτό εκφράζεται ως $e^{(k)} - e^* < \varepsilon \cdot e^{(k)}$, όπου ε είναι μια μικρή θετική σταθερά, δηλαδή η μείωση του σφάλματος γίνεται πολύ μικρή σε σχέση με το $e^{(k)}$. Η διαδικασία σταματά όταν η ακτίνα αναζήτησης r γίνει πολύ μικρή, δηλαδή $r < \delta$, όπου δ είναι μια μικρή σταθερά, όπως φαίνεται στον Αλγ. 1.

ΒΕΛΤΙΣΤΟΠΟΙΗΣΕΙΣ

1. **"Κακές" Κινήσεις**: Τροποποιούμε τον αλγόριθμο ώστε να επιτρέπεται σε κάθε σημείο να κάνει την βέλτιστη κίνηση, ακόμη και αν αυτό προσωρινά αυξήσει το σφάλμα, γεγονός που μπορεί να οδηγήσει σε ταχύτερη σύγκλιση σε καλύτερες λύσεις, παρόμοια με τον αλγόριθμο simulated annealing.

2. **Online Υπολογισμός του Πίνακα Αποστάσεων**: Ενημερώνουμε μόνο την επηρεαζόμενη σειρά και στήλη στον πίνακα αποστάσεων για κάθε κίνηση, αντί να επανυπολογίζουμε ολόκληρο τον πίνακα, σύμφωνα με την Εξ. (1.2).

$$d_{i,j}^{(k+1)} = \sqrt{\left(d_{i,j}^{(k)}\right)^2 - \left(x_{i,l}^{(k)} - x_{j,l}^{(k)}\right)^2 + \left(x_{i,l}^{(k+1)} - x_{j,l}^{(k+1)}\right)^2} \tag{1.2}$$

3. **Επιλογή Βήματος και Κίνησης**: Υλοποιούμε τυχαία δειγματοληψία κατευθύνσεων στον χώρο ενσωμάτωσης για να επιλέξουμε μια "καλή" κατεύθυνση για μια κίνηση, αντί να αναζητάμε την ιδανική, με σκοπό τη μείωση της πολυπλοκότητας και του χρόνου εκτέλεσης.

4. **Παραλληλοποίηση**: Χρησιμοποιούμε παράλληλους υπολογισμούς, ειδικά το μοτίβο map-reduce και τη βιβλιοθήκη OpenMP, για να επιταχύνουμε την αναζήτηση για βέλτιστες κινήσεις, πετυχαίνοντας σημαντικές βελτιώσεις στον χρόνο εκτέλεσης.

Αλγοριθμικη πολυπλοκοτητα

Για κάθε εποχή αναζητούμε σε $2L$ διαστάσεις για N σημεία. Σε κάθε αναζήτηση χρειαζόμαστε επίσης $\mathcal{O}(N)$ πράξεις για να ενημερώσουμε τον πίνακα αποστάσεων, με χρήση του online υπολογισμού. Έτσι, η υπολογιστική πολυπλοκότητα του αλγορίθμου ανά εποχή είναι $\mathcal{O}(N^2L)$. Οι υπόλοιπες προτεινόμενες βελτιστοποιήσεις δεν αλλάζουν την πολυπλοκότητα του αλγορίθμου ανά εποχή με την αξιοσημείωτη εξαίρεση της βελτιστοποίησης επιλογής κίνησης: αν αντί για $2L$ κινήσεις ανά εποχή θα λάμβανε υπόψη μόνο $2K$ κινήσεις ($K < L$). Σε αυτή την περίπτωση, η συνολική πολυπλοκότητα ανά εποχή θα ήταν $\mathcal{O}(N^2K)$ αντί για $\mathcal{O}(N^2L)$. Ωστόσο, όπως θα δούμε στα πειράματα που ακολουθούν οι (υπόλοιπες) προτεινόμενες βελτιστοποιήσεις βελτιώνουν την ταχύτητα σύγκλισης, με αποτέλεσμα λιγότερες εποχές εκτέλεσης.

1.1.2 Πειραματα Γεωμετριας Πολλαπλοτητων

Η βασική υπόθεση στη μάθηση πολλαπλοτήτων είναι ότι τα δεδομένα εισόδου βρίσκονται σε μια χαμηλής διάστασης, μη γραμμική πολλαπλότητα, ενσωματωμένη σε έναν χώρο υψηλής διάστασης. Έτσι, οι μη γραμμικές τεχνικές μείωσης διαστατικότητας στοχεύουν στην εξαγωγή της χαμηλής διάστασης πολλαπλότητας από τον χώρο υψηλής διάστασης. Για να απεικονίσουμε αυτό δημιουργήσαμε μια ποικιλία γεωμετρικών σχημάτων πολλαπλοτήτων και συγκρίναμε την προτεινόμενη τεχνική MDS με άλλες, καθιερωμένες τεχνικές.

Πρέπει να σημειωθεί ότι οι αλγόριθμοι MDS με είσοδο πίνακες ευκλείδειας απόστασης δεν μπορούν να εντοπίσουν τη γεωμετρία των δεδομένων, έτσι χρειάζεται να παρέχουμε ως είσοδο έναν *πίνακα γεωδαιτικής απόστασης*. Αυτός ο πίνακας υπολογίζεται εκτελώντας τον αλγόριθμο του Djikstra για τη συντομότερη διαδρομή στον πίνακα γειτνίασης των εισαγόμενων δεδομένων. Για τα πειράματά μας δειγματοληπτούμε 3000 σημεία σε 11 $3D$ σχήματα και τα μειώνουμε σε 2 διαστάσεις χρησιμοποιώντας τον pattern search MDS, SMACOF, truncated SVD, ISOMAP, LLE, Hessian LLE, Modified LLE και LTSA.

Η Εικ. 1.1 δείχνει πειράματα για την μείωση διαστάσεων τριών σχημάτων πολλαπλοτήτων. Δίνονται γεωδαιτικοί πίνακες αποστάσεων στον pattern search MDS και SMACOF. Καταγράφουμε τους χρόνους που χρειάστηκε κάθε μέθοδος για να τρέξει.

Το πρώτο σχήμα που εξετάζουμε είναι το κλασικό swissroll, όπου ένα $2D$ επίπεδο είναι "τυλιγμένο" σε $3D$ χώρο και ο στόχος είναι να εξαχθεί το αρχικό $2D$ επίπεδο. Τα αποτελέσματα παρουσιάζονται στην Εικ. 1.1α′. Στη συνέχεια εξετάζουμε πώς οι αλγόριθμοι χειρίζονται αραιούς πίνακες αποστάσεων. Για αυτόν τον σκοπό, δημιουργούμε ένα σύνολο δεδομένων από $3D$ μη επικαλυπτόμενες ομάδες με μια γραμμή που συνδέει τα κεντροειδή, όπου η αραιότητα του πίνακα αποστάσεων προκύπτει διότι η πλειοψηφία των σημείων δειγματοληπτείται πολύ στενά μέσα στις ομάδες. Μια καλή απεικόνιση θα πρέπει να διατηρεί τη δομή της ομάδας σε χαμηλότερες διαστάσεις. Στο Σχήμα 1.1β′

βλέπουμε τα αποτελέσματα. Τέλος, παρουσιάζουμε πώς οι αλγόριθμοι αποδίδουν με μεταβάσεις από πυκνές σε αραιές περιοχές με ένα σχήμα ελικοειδούς τοροειδούς στο Σχήμα 1.1γ′.

1.1.3 Πειραματα σημασιολογικης ομοιοτητας

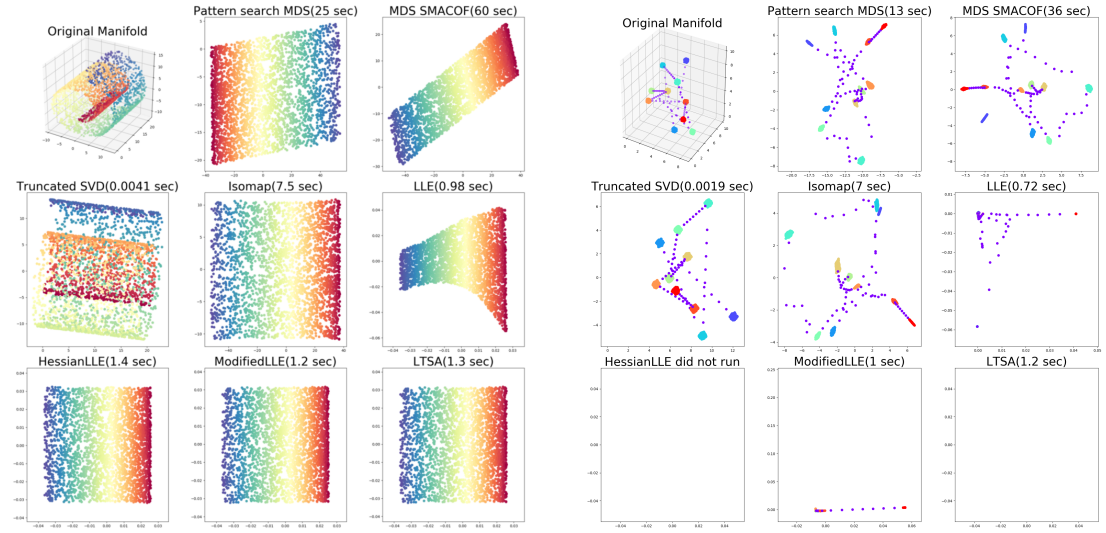
Η κατασκευή μοντέλων σημασιολογικών δικτύων συνίσταται από την αναπαράσταση εννοιών ως διανύσματα σε έναν, πιθανώς υψηλοδιάστατο, χώρο $\mathbb{R}^n$. Οι σχέσεις μεταξύ των εννοιών ποσοτικοποιούνται ως οι αποστάσεις, ή αντίστροφα οι ομοιότητες του ημιτόνου, μεταξύ των σημασιολογικών διανυσμάτων. Η εργασία της σημασιολογικής ομοιότητας στοχεύει στην αξιολόγηση της συσχέτισης των ομοιοτήτων μεταξύ εννοιών σε έναν δεδομένο σημασιολογικό χώρο έναντι ενός συνόλου τιμών ομοιότητας που παρέχονται από ανθρώπους.

Η μείωση της διαστατικότητας παρέχει έναν τρόπο για να μειωθεί το μέγεθος των διανυσμάτων που οδηγεί σε σημαντικά κέρδη σε χρόνο υπολογισμού και χώρο αποθήκευσης. Επιθυμητό είναι ένας τρόπος μείωσης της διαστατικότητας να διατηρεί τη δομή του αρχικού χώρου, και συγκεκριμένα τις αποστάσεις μεταξύ των εννοιών.

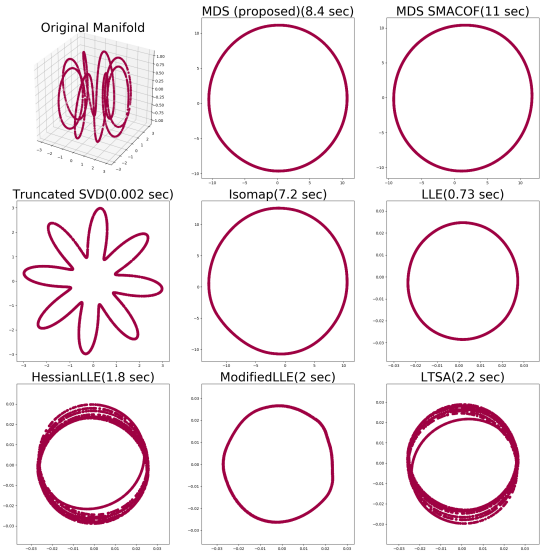
Αξιολογούμε την απόδοση των τεχνικών μείωσης διαστατικότητας που ερευνήθηκαν και στην προηγούμενη ενότητα για την εργασία σημασιολογικής ομοιότητας. Χρησιμοποιούμε τα σημασιολογικά σύνολα δεδομένων MEN και SimLex-999. Και τα δύο σύνολα δεδομένων παρέχονται στη μορφή λιστών ζευγών λέξεων, όπου κάθε ζεύγος συνδέεται με έναν βαθμό ομοιότητας. Αυτός ο βαθμός υπολογίστηκε μέσω του μέσου όρου των ομοιοτήτων που παρείχαν άνθρωποι. Ως σημασιολογικά διανύσματα λέξεων υψηλής διάστασης, χρησιμοποιούμε τα 300-διάστατα διανύσματα GloVe που κατασκευάστηκαν χρησιμοποιώντας ένα μεγάλο σώμα Twitter. Μειώνουμε τα διανύσματα στην επιθυμητή διάσταση L και υπολογίζουμε τον συντελεστή συσχέτισης Spearman μεταξύ των βαθμολογιών ομοιότητας που παρείχαν οι άνθρωποι και αυτών που υπολογίστηκαν αυτόματα. Τα αποτελέσματα περιλαμβάνονται στον Πίνακα 1.1 για $L = 10$. Παρατηρούμε ότι το LLE παρέχει τα καλύτερα αποτελέσματα για το MEN, ενώ το pattern search MDS είναι το καλύτερο για το SimLex-999. Επιπλέον, παρατηρούμε ότι οι μη γραμμικές τεχνικές μείωσης διαστατικότητας μπορούν να βελτιώσουν σημαντικά την απόδοση των σημασιολογικών διανυσμάτων σε κάποιες περιπτώσεις.

1.1.4 Συμπερασματα

Προτείνουμε το pattern Search MDS, έναν νέο αλγόριθμο για μη γραμμική μείωση διαστατικότητας, εμπνευσμένο από μεθόδους βελτιστοποίησης χωρίς χρήση κλίσης. Το Pattern Search MDS διατυπώνεται ως ένα παράδειγμα της ευρύτερης οικογένειας των μεθόδων GPS, προσφέροντας θεωρητικές εγγυήσεις σύγκλισης. Επιπλέον βελτιστοποιήσεις βελτιώνουν την απόδοση του αλγορίθμου μας ως προς την υπολογιστική αποδοτικότητα, την ανθεκτικότητα και την ποιότητα της λύσης. Η ποιοτική αξιολόγηση σε σύγκριση με άλλες δημοφιλείς τεχνικές μείωσης διαστατικότητας για καθαρές και θορυβώδεις γεωμετρίες διαμορφωμένων χώρων δείχνει ότι το pattern Search MDS μπορεί να ανακαλύψει με ακρίβεια την ενδογενή γεωμετρία των πολλαπλοτήτων που είναι ενσωματωμένες σε υψηλοδιάστατους χώρους. Επιπρόσθετα, η σύγκριση των χαρακτηριστικών σύγκλισης απέναντι στο SMACOF δείχνει ότι το pattern Search MDS συγκλίνει σε λιγότερες εποχές σε παρόμοιες ή καλύτερες λύσεις. Πειράματα σε πραγματικά δεδομένα παράγουν αποτελέσματα συγκρίσιμα με



(α′) **(β′)**



(γ′)

Εικόνα 1.1: Σύγκριση του αλγορίθμου pattern search MDS με άλλες τεχνικές μείωσης διαστατικότητας για μετατροπή: (α′) $3D$ swissroll σε $2D$ επίπεδο, (β′) $3D$ συστάδων σε $2D$ συστάδες, και (γ′) $3D$ τοροϊδή έλικα $2D$ κύκλο

Πίνακας 1.1: Σύγκριση τεχνικών μείωσης διαστατικότητας για τη σημασιολογική ομοιότητα στα MEN και Simlex-999.

Μέθοδος	Διαστάσεις	MEN	SimLex-999
-	300	0.635	0.177
Pattern search MDS	10	0.596	**0.242**
SMACOF	10	0.632	0.221
ISOMAP	10	0.625	0.132
Truncated SVD	10	0.562	0.140
LLE	10	**0.657**	0.172
Hessian LLE	10	0.157	0.004
Modified LLE	10	0.643	0.158
LTSA	10	0.154	0.004

τα κορυφαία για σημασιολογική ομοιότητα λεξιλογίου. Ανοιχτού κώδικα υλοποιήσεις του Pattern Search MDS και της διαδικασίας παραγωγής δεδομένων παρέχονται για να διευκολύνουν την αναπαραγωγή των αποτελεσμάτων μας.

1.1.5 Μεθοδοι μειωσης διαστατικοτητας για αναγνωριση συναισθηματων απο φωνη

Εξετάζουμε τη χρήση γραμμικών και μη γραμμικών αλγορίθμων μείωσης διαστατικότητας για την εξαγωγή low-rank χαρακτηριστικών για την αναγνώριση συναισθημάτων από φωνή. Χρησιμοποιούνται δύο σύνολα χαρακτηριστικών, ένα που βασίζεται σε χαρακτηριστικά χαμηλού επιπέδου και τις συναθροίσεις τους (IS10) και ένα μοντελοποιεί τα μη γραμμικά δυναμικά χαρακτηριστικά της ομιλίας (RQA), καθώς και τη σύντηξή τους. Αναφέρουμε αποτελέσματα αναγνώρισης συναισθημάτων ομιλίας (SER) για μαθημένες αναπαραστάσεις σε δύο βάσεις δεδομένων χρησιμοποιώντας διαφορετικές μεθόδους ταξινόμησης. Η ταξινόμηση με αναπαραστάσεις χαμηλών διαστάσεων αποφέρει βελτίωση της απόδοσης σε διάφορες ρυθμίσεις. Αυτό υποδηλώνει ότι η μείωση διαστάσεων είναι ένας αποτελεσματικός τρόπος για την καταπολέμηση της κατάρας της διαστατικότητας για την αναγνώριση συναισθημάτων από φωνή. Η οπτικοποίηση των χαρακτηριστικών σε δύο διαστάσεις παρέχει μια εικόνα για τις διακριτικές ικανότητες των μειωμένων συνόλων χαρακτηριστικών.

1.1.6 Χαρακτηριστικα

Χρησιμοποιούμε τα παρακάτω σύνολα χαρακτηριστικών:
IS10 σετ: Το σετ χαρακτηριστικών IS10 αποτελείται από 1582 χαρακτηριστικά. Το IS10 λαμβάνεται μετατρέποντας το σήμα στον χώρο Fourier. Τα χαρακτηριστικά αντιστοιχούν σε 21 στατιστικές λειτουργίες (π.χ. ποσοστά, συντελεστές γραμμικής παλινδρόμησης) που εφαρμόζονται σε 38 χαμηλού

επιπέδου περιγραφείς (MFCCs, PCM ένταση κ.λπ.) και τα δέλτα τους. Η εξαγωγή πραγματοποιείται με τη χρήση του εργαλείου openSMILE.

RQA σετ: Το σετ χαρακτηριστικών RQA αποτελείται από 432 χαρακτηριστικά. Αυτό το σετ χαρακτηριστικών λαμβάνεται αναλύοντας τη δυναμική της ομιλίας μέσω της αναπαράστασης του φάσματος χώρου. Ο χώρος φάσης ανακατασκευάζεται μέσω της χρήσης καθυστερημένων εκδόσεων του αρχικού σήματος και στη συνέχεια υπολογίζονται τα διαγράμματα επαναλήψεων ως κατωφλιωμένες αποστάσεις ζευγών σημείων στον χώρο φάσης. Τα χαρακτηριστικά εξάγονται ως συγκεντρωτικά μέτρα RQA από τα διαγράμματα επαναλήψεων.

Συγχωνευμένο σετ: Συγχωνεύουμε τα χαρακτηριστικά από IS10 και RQA σε μια αναπαράσταση των 2014 διαστάσεων, μοντελοποιώντας τόσο το περιεχόμενο συχνότητας των ηχητικών σημάτων όσο και την επαναληπτική δυναμική.

1.1.7 Σύνολα δεδομένων

Επίσης, χρησιμοποιούμε τα παρακάτω σύνολα δεδομένων:

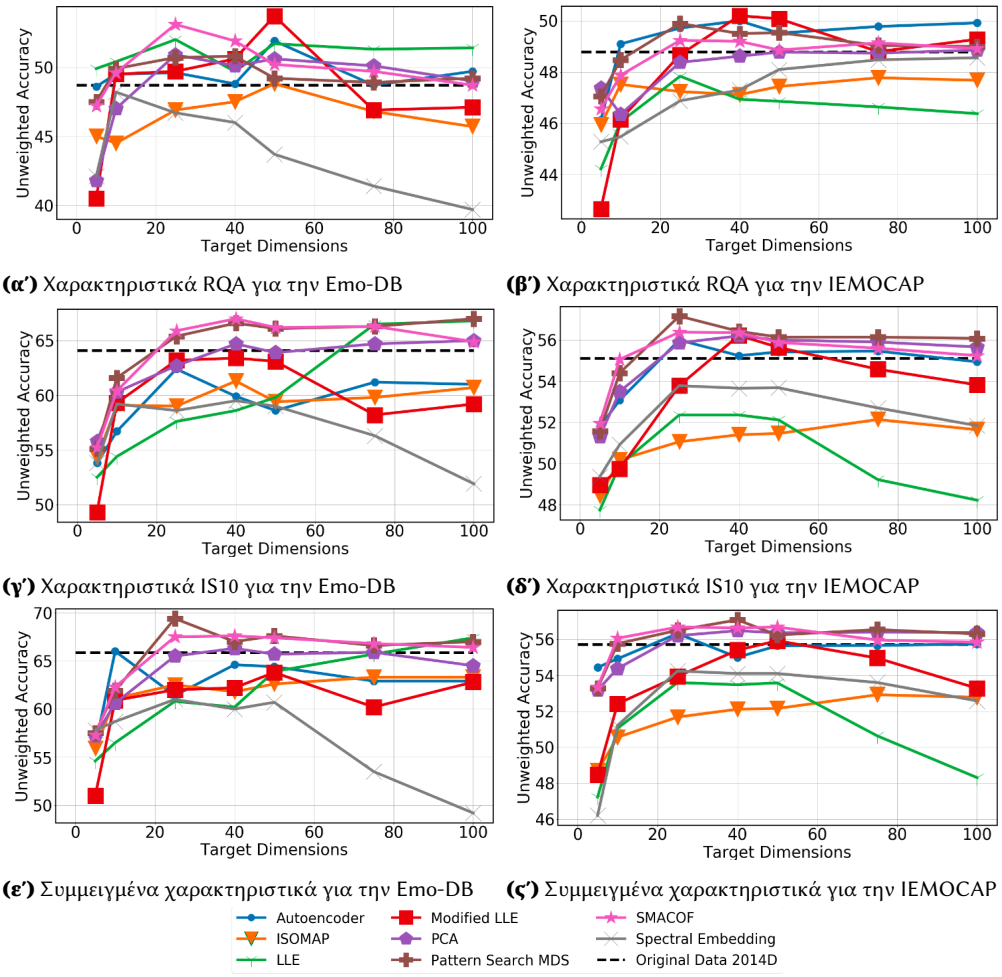
Emo-DB: Η βάση δεδομένων του Βερολίνου για την Συναισθηματική Ομιλία (Emo-DB) περιέχει 535 συναισθηματικές γερμανικές προτάσεις, εκφωνημένες από 10 ηθοποιούς (5 άνδρες και 5 γυναίκες). Συγκεκριμένα, περιλαμβάνονται 7 συναισθήματα, δηλαδή 127 θυμός, 45 αηδία, 70 φόβος, 71 χαρά, 60 λύπη, 81 βαρεμάρα και 70 ουδέτερα.

IEMOCAP: Η βάση δεδομένων IEMOCAP περιέχει 12 ώρες βιντεοσκοπημένων δεδομένων με διαλόγους που έχουν συνταχθεί και αυτοσχεδιαστεί, ηχογραφημένοι από 10 ηθοποιούς. Οι εκφωνήσεις οργανώνονται σε 5 συνεδρίες δυαδικών αλληλεπιδράσεων ανάμεσα σε ζευγάρια ηθοποιών. Για τα πειράματά μας λαμβάνουμε υπόψη 5531 εκφωνήσεις 4 συναισθημάτων (1103 θυμωμένες, 1636 χαρούμενες, 1708 ουδέτερες και 1084 λυπημένες), όπου συγχωνεύουμε την κατηγορία της ενθουσιώδους συναισθηματικής κατάστασης στην χαρά.

1.1.8 Αποτελέσματα

Στην Εικόνα 1.2 συνοψίζουμε τα αποτελέσματα ταξινόμησης συναισθήματος από φωνή μετά από μείωση διαστατικότητας μέσω του αλγορίθμου κοντινότερου γείτονα για τα διάφορα σετ χαρακτηριστικών και σύνολα δεδομένων.

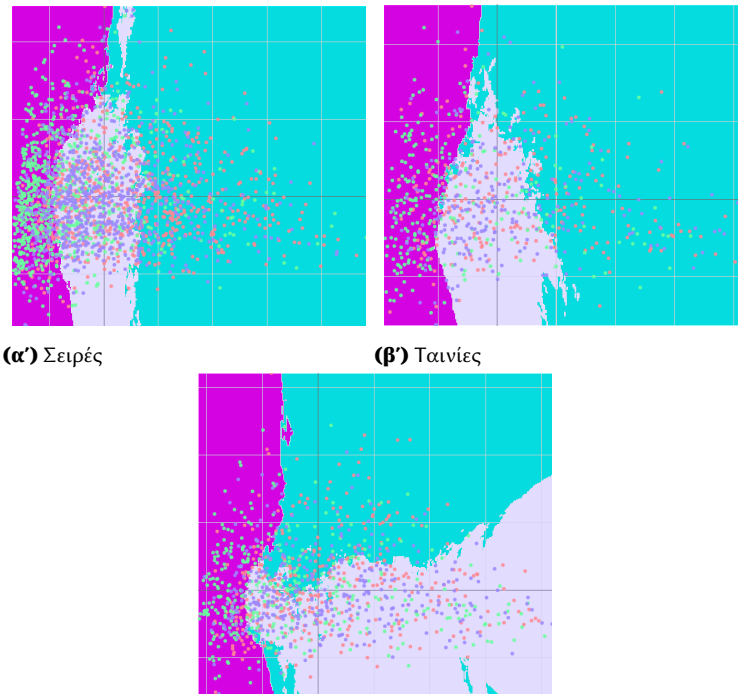
Στα αποτελέσματα αναφέρουμε την αβαρή ακρίβεια (unweighted accuracy)ένη LLE επιδεικνύει ξεχωριστή απόδοση στη μείωση των χαρακτηριστικών RQA, ειδικά στη διάσταση ενσωμάτωσης $L = 50$. Για τα χαρακτηριστικά IS10, οι αλγόριθμοι MDS υπερτερούν των υπολοίπων, δείχνοντας ότι αυτά τα χαρακτηριστικά μοιάζουν περισσότερο με υπερεπίπεδο παρά με μη-γραμμική πολλαπλότητα. Στην IEMOCAP, παρατηρούνται παρόμοιες τάσεις, με τη Τροποποιημένη LLE και τη Pattern Search MDS να έχουν καλή απόδοση για τα χαρακτηριστικά RQA και IS10 αντίστοιχα. Η συγχώνευση συνόλων χαρακτηριστικών δείχνει ότι οι MDS και PCA είναι οι πιο αποτελεσματικές, με τα χαρακτηριστικά IS10 να κυριαρχούν μετά τη συγχώνευση.



Εικόνα 1.2: Αποτελέσματα ταξινόμησης k-NN για τρία σετ χαρακτηριστικών στις IEMOCAP και Emo-DB

1.1.9 Οπτικοποιησεις

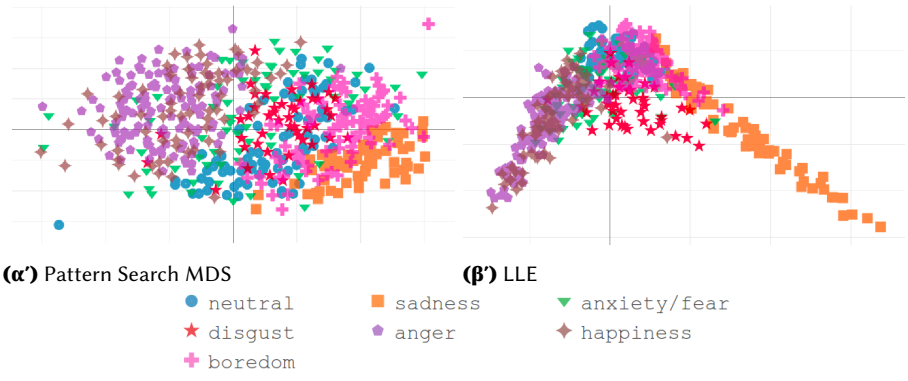
Οι οπτικοποιήσεις των χαρτών χαρακτηριστικών που έχουν μειωθεί σε δύο διαστάσεις αποκαλύπτουν κεντρικές παρατηρήσεις. Στο Σχήμα 1.3, η PCA εφαρμοσμένη σε ένα ποικίλο σύνολο δεδομένων ομιλίας αποκαλύπτει διακριτές κατανομές συναισθηματικών τάξεων. Ο θυμός (μπλε κατηγορία) εμφανίζει παρόμοια μοτίβα διανομής με τη λύπη και τη χαρά σε τομείς ταινιών και τηλεοπτικών σειρών, όπως φαίνεται στο Σχήμα 1.3α′ και στο Σχήμα 1.3β′. Ωστόσο, στις συνεντεύξεις (Σχήμα 1.3γ′), η πρωτεύουσα διάσταση PCA δεν διαχωρίζει σαφώς τα συναισθήματα, με τον θυμό και τη χαρά να διακρίνονται περισσότερο από τη δευτερεύουσα διάσταση. Αυτό το μοτίβο μοιάζει με την αντιπροσώπευση Valence-Arousal, δείχνοντας την ευαισθησία του τομέα στην αυτόματη μείωση διαστάσεων για το συναισθηματικό περιεχόμενο.



(α′) Σειρές **(β′)** Ταινίες

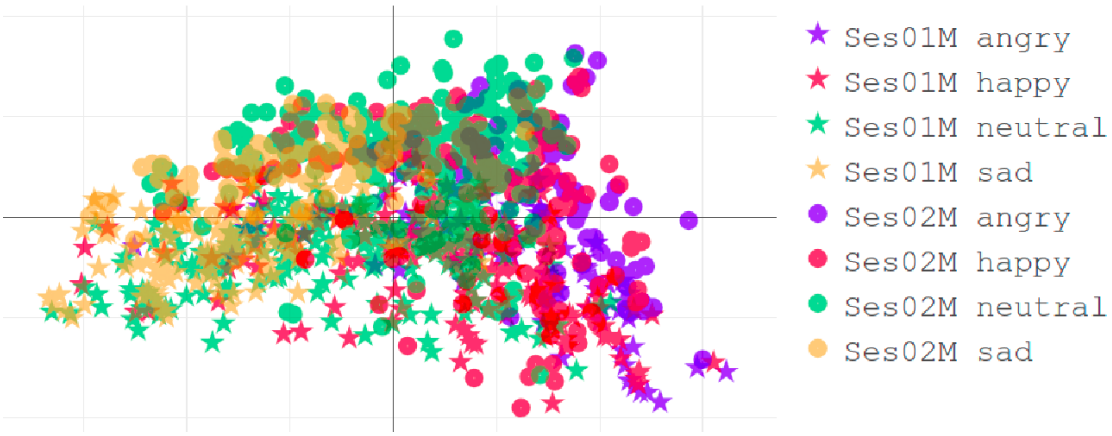
(γ′) Συνεντεύξεις

Εικόνα 1.3: Επιφάνειες αποφάσεων μεταξύ τομέων με μείωση διαστατικότητας σε 2D



(α′) Pattern Search MDS **(β′)** LLE

Εικόνα 1.4: Μείωση διάστασης σε 2D για το συγχωνευμένο σετ χαρακτηριστικών στην Emo-DB



Εικόνα 1.5: ISOMAP για τα συγχωνευμένα χαρακτηριστικά δύο ομιλητών της IEMOCAP

Το Σχήμα 1.4α′ δείχνει τη δημιουργία ενός σημαντικού δισδιάστατου χώρου από τη Pattern Search MDS, όπου συναισθήματα όπως ο θυμός και η λύπη καταλαμβάνουν διακριτές περιοχές, υποδηλώνοντας κρυφή κωδικοποίηση arousal. Η LLE, παρά τη χαμηλή ακρίβεια στα συνδυασμένα χαρακτηριστικά, διαχωρίζει τα συναισθήματα χαμηλού από αυτά υψηλού arousal στο Σχήμα 1.4β′. Το Σχήμα 1.5 παρουσιάζει τις ενσωματώσεις ISOMAP για δύο ομιλητές στην IEMOCAP, επιδεικνύοντας καλύτερη διάκριση ομιλητών, προσφέροντας υπόδειξη για την πιθανή χρήση τους σε διαχωρισμό ομιλητών βασισμένο σε γεωδαιτικές αποστάσεις.

1.1.10 Συμπερασματα

Σε αυτή την εργασία, εξερευνούμε τις επιδράσεις της μη επιβλεπόμενης γραμμικής και μη γραμμικής μείωσης διαστατικότητας σε κορυφαία χαρακτηριστικά ομιλίας για την αναγνώριση συναισθήματος από φωνή. Αξιολογούμε αυτούς τους αλγορίθμους για ανεξάρτητη από τον ομιλητή αναγνώριση συναισθήματος από φωνή στην IEMOCAP και την Emo-DB. Τα πειράματα δείχνουν ότι η απόδοση των αναπαραστάσεων χαμηλού βαθμού είναι ανταγωνιστική σε σχέση με τις αρχικές αναπαραστάσεις υψηλής διάστασης. Υποθέτουμε ότι αυτό το φαινόμενο προκαλείται από την κατάρα της διαστατικότητας, καθώς ο αριθμός των δειγμάτων στα σύνολα δεδομένων για την αναγνώριση συναισθήματος από φωνή δεν καλύπτει τον υψηλοδιάστατο χώρο. Η ερμηνεία των αποτελεσμάτων και η οπτικοποίηση των αναπαραστάσεων σε 2 διαστάσεις προσφέρει ενδιαφέροντα συμπεράσματα σχετικά με τις υψηλοδιάστατες δομές. Η πρώτη διαπίστωση είναι ότι τα χαρακτηριστικά IS10 μπορούν να αποδομηθούν με τη χρήση γραμμικής μείωσης διαστατικότητας, π.χ. με τη χρήση των αλγορίθμων PCA ή MDS. Δεύτερον, η μείωση διαστατικότητας με τεχνικές που διατηρούν αποστάσεις μπορεί να κωδικοποιήσει σημαντικές διαστάσεις, π.χ. arousal.

1.2 Μερος ΙΙ: Μη επιβλεπομενη προσαρμογη μεταξυ πεδιων για εφαρμογες γλωσσας και φωνης

Στο δεύτερο μέρος αυτής της διατριβής, σχεδιάζουμε στρατηγικές εκπαίδευσης που διευκολύνουν την μη επιβλεπόμενη προσαρμογή συστημάτων κειμένου και ομιλίας σε άγνωστους τομείς. Ειδική έμφαση δίνεται στην ποσότητα των δεδομένων που απαιτούνται για επιτυχημένη προσαρμογή, και επιβεβαιώνουμε την αποτελεσματικότητα των προτεινόμενων τεχνικών όσον αφορά τη χρήση δειγμάτων μέσω εκτεταμένων πειραμάτων. Επικυρώνουμε την προτεινόμενη μέθοδο για δύο διαφορετικές εφαρμογές, την ταξινόμηση κειμένου και την αναγνώριση ομιλίας. Οι προτεινόμενες μέθοδοι λειτουργούν σε συνδυασμό με δημοφιλή, κορυφαία μοντέλα, τα οποία εκπαιδεύονται με αυτοεπιβλεπόμενο τρόπο: για το κείμενο επιβεβαιώνουμε την προσέγγισή μας με το BERT και για την ομιλία με το XLSR-53.

1.2.1 Φορμαλισμος

Τυπικά, το πρόβλημα της μη-επιβλεπόμενης προσαρμογής μεταξύ πεδίων μπορεί να οριστεί ως εξής. Έστω $X \subseteq \mathbb{R}^n$ ένας πραγματικός χώρος τιμών που αποτελείται από n-διαστάσεις διανυσμάτων χαρακτηριστικών $x \in X$, και Y ένα πεπερασμένο σύνολο ετικετών $y \in Y$, δηλαδή, $Y = \{1, 2, \ldots, L\}$. Επιπλέον, υποθέτουμε δύο διαφορετικές κατανομές, δηλαδή, την κατανομή του πεδίου πηγής $\mathcal{S}(x, y)$ και την κατανομή του πεδίου στόχου $\mathcal{T}(x, y)$, οι οποίες ορίζονται στο καρτεσιανό γινόμενο $X \times Y$.

Ο στόχος είναι να εκπαιδευτεί ένα μοντέλο που μαθαίνει μια απεικόνιση μεταξύ των διανυσμάτων χαρακτηριστικών $x_{\mathcal{T}}$ και των αντίστοιχων ετικετών τους $y_{\mathcal{T}}$ για δείγματα που λαμβάνονται από την κατανομή στόχου $(x_{\mathcal{T}}, y_{\mathcal{T}}) \sim \mathcal{T}$.

Κατά την εκπαίδευση, έχουμε πρόσβαση σε δείγματα από την κατανομή πηγής $\mathcal{S}(x, y)$ και την περιθωριακή κατανομή στόχου $\mathcal{T}(x)$, δηλαδή, δεν παρέχονται ετικέτες στόχου. Ορίζουμε το σύνολο δεδομένων εκπαίδευσης D ως τη συγκέντρωση των συνόλων εκπαίδευσης πηγής και στόχου, $D = (D_S, D_T)$. Τα D_S και D_T ορίζονται ως ακολουθίες από πλειάδες, δηλαδή,

$$\begin{aligned} D_S &= \{(x_i, y_i) \,|\, (x_i, y_i) \sim \mathcal{S}(x, y),\ 1 \leq i \leq N\} \\ D_T &= \{(x_j, \emptyset) \,|\, x_j \sim \mathcal{T}(x),\ 1 \leq j \leq M\}, \end{aligned} \tag{1.3}$$

όπου λαμβάνουμε N δείγματα από το $\mathcal{S}(x, y)$ και M δείγματα από το $\mathcal{T}(x)$. Τέλος, ενισχύουμε τις πλειάδες στο D με μια συνάρτηση δείκτη πεδίου:

$$\begin{aligned} D &= \{(x_k, y'_k, 1_k) \,|\, 1 \leq k \leq N + M\} \\ 1_k &= \begin{cases} 0 & \text{if } x_k \sim \mathcal{S}(x), \\ 1 & \text{if } x_k \sim \mathcal{T}(x). \end{cases} \\ y'_k &= \begin{cases} y_k & \text{if } x_k \sim \mathcal{S}(x), \\ \emptyset & \text{if } x_k \sim \mathcal{T}(x). \end{cases} \end{aligned} \tag{1.4}$$

Μη-επιβλεπομενη ακουστικη προσαρμογη για αυτοματη αναγνωριση φωνης

Ο παραπάνω ορισμός μπορεί να επεκταθεί άμεσα στην περίπτωση της αναγνώρισης φωνής, με κάποιες τροποποιήσεις. Ειδικότερα, τροποποιούμε τον χώρο χαρακτηριστικών X, ώστε να είναι το σύνολο των πεπερασμένων ακολουθιών διανυσμάτων χαρακτηριστικών $(x_m)_{m\in\mathbb{N}\setminus\{\infty\}} \in X \subseteq (\mathbb{R}^n)^*$. Επιπλέον, ο χώρος ετικετών Y τροποποιείται ώστε να είναι το σύνολο των ακολουθιών $(y_n)_{n\in\mathbb{N}\setminus\{\infty\}}$, όπου το $Y = (\{1, 2, \ldots, L\})^*$ περιέχει πεπερασμένες ακολουθίες επάνω σε ένα πεπερασμένο λεξιλόγιο. Για την εκπαίδευση CTC υποθέτουμε ότι $m > n$ για κάθε δείγμα (x_m, y_n), δηλαδή οι ακολουθίες χαρακτηριστικών είναι μακρύτερες από τις αντίστοιχες ακολουθίες ετικετών. Οι υπόλοιποι ορισμοί δε χρειάζονται τροποποιήσεις.

Μη-επιβλεπομενη γλωσσικη προσαρμογη για αυτοματη αναγνωριση φωνης

Η προσαρμογή για συστήματα αυτόματης αναγνώρισης φωνής μπορεί επίσης να εκτελεστεί στο επίπεδο της γλώσσας, δηλαδή του χώρου των ετικετών. Σε αυτή τη ρύθμιση, υποθέτουμε ότι τα δείγματα του στοχευμένου πεδίου προέρχονται από την περιθωριοποιημένη κατανομή στόχου $\mathcal{T}(y)$. Τώρα το στοχευμένο σύνολο δεδομένων D_T αποτελείται από διατεταγμένα ζεύγη με τη μορφή $(\emptyset, y_j)$, όπου το y_j είναι η ακολουθία λέξεων ετικετών $(y_n)_{n\in\mathbb{N}\setminus\{\infty\}}$ για το j-οστό δείγμα.

Ασθενως επιβλεπομενη προσαρμογη για αυτοματη αναγνωριση φωνης

Η τελευταία ρύθμιση που εξετάζουμε είναι η περίπτωση όπου διατίθενται δείγματα από τον ηχητικό και γλωσσικό τομέα, αλλά η αντιστοίχιση μεταξύ τους είναι άγνωστη. Αυτή η κατάσταση μπορεί να συναντηθεί σε πραγματικές ρυθμίσεις, π.χ. στην περίπτωση που οι ηχητικές και κειμενικές πληροφορίες συλλέγονται ανεξάρτητα. Για παράδειγμα, σκεφτείτε την περίπτωση όπου συλλέγονται ηχητικά αποσπάσματα από ειδησεογραφικές εκπομπές μαζί με σύγχρονα εφημεριδικά άρθρα. Ένα άλλο παράδειγμα είναι η περίπτωση όπου διατίθενται μακροσκελή ηχητικά αποσπάσματα μαζί με τις μεταγραφές τους, αλλά χωρίς λεπτομερείς χρονικές αντιστοιχίσεις*. Σε αυτή την περίπτωση, τα δείγματα του στοχευμένου τομέα λαμβάνονται ανεξάρτητα από τις περιθωριοποιημένες κατανομές $\mathcal{T}(x)$ και $\mathcal{T}(y)$, και το στοχευμένο σύνολο δεδομένων D_T αποτελείται από διατεταγμένα ζεύγη με τη μορφή $(x_j, \emptyset)$ και $(\emptyset, y_j)$.

1.2.2 Προτεινομενη στρατηγικη προσαρμογης

Παρόλο που σύγχρονες τεχνικές προσαρμογής που βασίζονται σε συνεχή προεκπαίδευση (CPT) παράγουν σημαντικές βελτιώσεις σε ποικιλία εργασιών, ένα κοινό θέμα σε αυτές τις εργασίες είναι η υπόθεση εκατοντάδων ή χιλιάδων ωρών διαθέσιμων δεδομένων εντός τομέα, κυρίως από διαδικτυακές πηγές, π.χ., YouTube. Αυτό μπορεί να είναι ανέφικτο όταν εξετάζουμε πιο εξειδικευμένες

*Αν και μια πλήρως επιβλεπόμενη εσωτερική σειρά δεδομένων μπορεί να δημιουργηθεί σε αυτή την περίπτωση χρησιμοποιώντας μεθόδους μακροχρόνιας / εξαναγκασμένης αντιστοίχισης, αυτό δεν αποτελεί κεντρικό σημείο για το πειραματικό μέρος αυτής της εργασίας.

ρυθμίσεις προσαρμογής, ή πιθανές ανησυχίες απορρήτου, π.χ., πώς θα συλλέγαμε 1000 ώρες συνεδριών ψυχοθεραπείας στα Ελληνικά; Επιπλέον, οι προσεγγίσεις CPT βασίζονται στην προεκπαίδευση εντός τομέα, η οποία είναι ευαίσθητη στην καταστροφική λήθη. Μια τεχνική αντιμετώπισης της καταστροφικής λήθης, σχετική με την προσέγγισή μας, το Elastic Weight Consolidation (EWC) αντιμετωπίζει την καταστροφική λήθη κατά την εκμάθηση μιας νέας εργασίας B επιβραδύνοντας τη μάθηση σε νευρώνες που είναι σημαντικοί για μια προηγουμένη εργασία A. Το EWC στοχεύει να βρει *ρητά* μια ισορροπία στο δίλημμα σταθερότητας-πλαστικότητας, περιλαμβάνοντας έναν πρόσθετο όρο regularization. Ωστόσο, η άμεση εφαρμογή του EWC σε μεγάλα μοντέλα μπορεί να είναι υπολογιστικά ανέφικτη, καθώς ο προτεινόμενος όρος ρυθμιστικής απαιτεί τον υπολογισμό του πίνακα πληροφορίας Fisher και τον υπολογισμό μιας κανονικοποίησης πάνω σε όλες τις παραμέτρους του δικτύου θ_i, όπως φαίνεται από τον δεύτερο όρο στην Εξ. (8.3). Επιπλέον, πολλαπλά αντίγραφα των βαρών του δικτύου πρέπει να διατηρούνται στη μνήμη κατά την εκμάθηση πολλαπλών εργασιών.

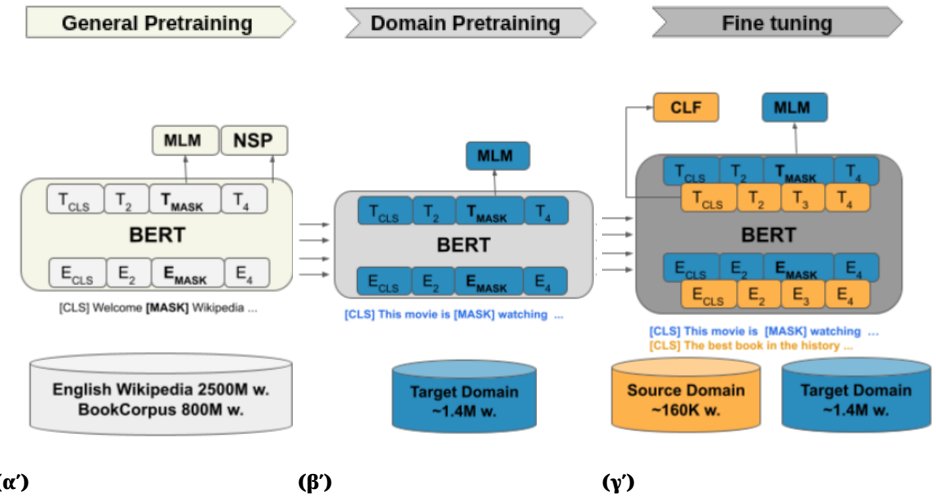
$$\mathcal{L}(\theta) = \mathcal{L}_B(\theta) + \sum_i \frac{\lambda}{2} F_i(\theta_i - \theta^*_{A,i}) \tag{1.5}$$

Ως εκ τούτου, επιλέγουμε να βρούμε *έμμεσα* μια ισορροπία μεταξύ σταθερότητας και πλαστικότητας, εκμεταλλευόμενοι την αυτοεπίβλεψη εντός τομέα για να αντιμετωπίσουμε την καταστροφική λήθη για τη μη-επιβλεπόμενη προσαρμογή μεταξύ πεδίων σε ένα περιβάλλον με περιορισμένα δεδομένα. Η βασική ιδέα είναι να ξεκινήσουμε από ένα προεκπαιδευμένο μοντέλο, εκπαιδευμένο με μια αυτοεπιβλεπόμενη απώλεια L_{SS}. Κατά τη διαδικασία μετεκπαίδευσης (fine-tuning), μαθαίνουμε το νέο έργο A χρησιμοποιώντας τα επισημειωμένα εκτός τομέα δεδομένα $x_{\mathcal{S}} \sim \mathcal{S}$, ενώ παράλληλα προσαρμόζουμε στον στόχο τομέα χρησιμοποιώντας τα μη επισημειωμένα, εντός τομέα δεδομένα $x_{\mathcal{T}} \sim \mathcal{T}$ με την αυτοεπιβλεπόμενη απώλεια. Η συνολική απώλεια μετεκπαίδευσης διατυπώνεται στην Εξ. (1.6). Διαπιστώνουμε ότι η προτεινόμενη στρατηγική λεπτής ρύθμισης με μεικτή αυτοεπίβλεψη είναι ανθεκτική, οδηγεί σε αποτελεσματική προσαρμογή, και μπορεί εύκολα να προσαρμοστεί για να λειτουργήσει σε διάφορες εφαρμογές.

$$L(x_{\mathcal{S}}, x_{\mathcal{T}}) = L_A(x_{\mathcal{S}}) + \lambda L_{SS}(x_{\mathcal{T}}) \tag{1.6}$$

1.2.3 Μη επιβλεπομενη προσαρμογη μεταξυ πεδιων μεσω γλωσσικης μοντελοποιησης

Σε αυτήν την εργασία εξερευνούμε τη μη επιβλεπόμενη προσαρμογή μεταξύ πεδίων (UDA) προεκπαιδευμένων μοντέλων γλώσσας για εργασίες κατάντη. Εισάγουμε το UDALM, μια διαδικασία μετεκπαίδευσης του μοντέλου, που χρησιμοποιεί μια μικτή συνάρτηση σφάλματος για την ταξινόμηση και τη γλωσσική μοντελοποίηση με μάσκα (MLM), που μπορεί να προσαρμοστεί στην κατανομή του τομέα-στόχου με ισχυρή επίδοση και αποδοτικά ως προς την ποσότητα δειγμάτων εισόδου. Τα πειράματά μας δείχνουν ότι η απόδοση των μοντέλων που έχουν εκπαιδευτεί με την προτεινόμενη στρατηγική κλιμακώνεται με την ποσότητα των διαθέσιμων δεδομένων εντός τομέα, και ότι η μεικτή συνάρτηση κόστους μπορεί να χρησιμοποιηθεί αποτελεσματικά ως κριτήριο διακοπής κατά τη διάρκεια της εκπαίδευσης UDA. Επιπλέον, συζητάμε τη σχέση μεταξύ της απόστασης A και του σφάλματος στόχου και διερευνούμε ορισμένους περιορισμούς της προσέγγισης Ανταγωνιστικής Εκπαίδευσης Πεδίου (DAT). Η μέθοδός μας αξιολογείται σε δώδεκα ζεύγη τομέων του συνόλου δε-



Εικόνα 1.6: (a) Το BERT είναι προεκπαιδευμένο στην αγγλική Βικιπαιδεία και το BookCorpus με τις εργασίες MLM και NSP. (b) Συνεχίζουμε την προεκπαίδευση του BERT στο μη επισημειωμένο σύνολο δεδομένων στόχου με MLM. (c) Εκπαιδεύουμε ένα ταξινομητή με τα επισημειωμένα δεδομένα του πηγαίου τομέα, ενώ χρησιμοποιούμε τα μη επισημειωμένα δεδομένα στόχου για το MLM.

δομένων Amazon Reviews Sentiment, αποδίδοντας ακρίβεια 91.74%, η οποία είναι μια απόλυτη βελτίωση 1.11% σε σχέση με την τελευταία λέξη της τεχνολογίας.

1.2.4 Μεθοδος UDALM

Η Εικ.1.6 παρουσιάζει το πλαίσιο UDALM, το οποίο αρχίζει με την προπαίδευση ενός μοντέλου σε γενικά κείμενα (Εικ.1.6α′). Αυτή η φάση περιλαμβάνει τη χρήση του BERT, βασισμένου στην αρχιτεκτονική Transformer, και εκπαιδευμένου μέσω του στόχου MLM. Περιλαμβάνει την πρόβλεψη κρυφών διακριτικών και τη χρήση απώλειας Πρόβλεψης Επόμενης Πρότασης (NSP). Αν υπάρχει διαθέσιμο επισημειωμένο σύνολο δεδομένων, το BERT μπορεί να βελτιστοποιηθεί σε επισημειωμένα σετ δεδομένων.

Η επόμενη φάση (Εικ. 1.6β′) συνεχίζει την προεκπαίδευση του μοντέλου σε συγκεκριμένα δεδομένα του τομέα, χρησιμοποιώντας τον στόχο MLM, προσαρμόζοντας το μοντέλο στον στοχευμένο τομέα χωρίς την ανάγκη εποπτευόμενων δεδομένων.

Στο τελικό βήμα μετεκπαίδευσης (Εικ. 1.6γ′), το μοντέλο υποβάλλεται σε εποπτευόμενη μετεκπαίδευση σε δεδομένα πηγής, χρησιμοποιώντας απώλεια ταξινόμησης, ενώ διατηρεί τον στόχο MLM σε μη επισημειωμένα δεδομένα στόχου ως βοηθητικό καθήκον. Ο ταξινομητής, βασισμένος στην αναπαράσταση του διακριτικού `[CLS]`, περιλαμβάνει ένα εμπρόσθιο επίπεδο. Αυτή η διαδικασία περιλαμβάνει εκπαίδευση σε επισημειωμένα δεδομένα πηγής για ταξινόμηση και σε μη επισημειωμένα δεδομένα στόχου για τη μοντελοποίηση κρυφής γλώσσας. Η μικτή συνάρτηση κόστους χρησιμοποιεί τις απώλειες ταξινόμησης L_{CLF} και γλωσσικής μοντελοποίησης L_{MLM}:

$$L(\mathbf{s}, \mathbf{t}) = \lambda L_{CLF}(\mathbf{s}) + (1 - \lambda) L_{MLM}(\mathbf{t}) \tag{1.7}$$

Πειραματα

Αξιολογούμε το UDALM στο σύνολο δεδομένων πολυ-τομεακής αξιολόγησης συναισθημάτων κριτικών της Amazon, ένα πρότυπο σύνολο δεδομένων για προσαρμογή τομέα. Οι κριτικές με ένα ή δύο αστέρια επισημαίνονται ως αρνητικές, ενώ οι κριτικές με τέσσερα ή πέντε αστέρια επισημαίνονται ως θετικές. Το σύνολο δεδομένων περιέχει κριτικές για τέσσερις τομείς προϊόντων: *Βιβλία* (B), *DVDs* (D), *Ηλεκτρονικά* (E) και *Συσκευές Κουζίνας* (K), παρέχοντας 12 σενάρια προσαρμογής για ζεύγη πηγής-στόχου. Διατίθενται ισορροπημένα σετ από 2000 επισημειωμένες κριτικές για κάθε τομέα. Χρησιμοποιούμε 20000 (τυχαία επιλεγμένες) μη επισημειωμένες κριτικές για τα (B), (D) και (E). Για το (K) είναι διαθέσιμες 17805 μη επισημειωμένες κριτικές. Για κάθε ένα από τα 12 σενάρια προσαρμογής χρησιμοποιούμε το 20 των επισημειωμένων δεδομένων πηγής και των μη επισημειωμένων δεδομένων στόχου για επικύρωση, ενώ τα επισημειωμένα δεδομένα στόχου χρησιμοποιούνται αποκλειστικά για δοκιμές και δεν είναι ορατά κατά τη διάρκεια της εκπαίδευσης ή της επικύρωσης.

Επιλέγουμε τρεις κορυφαίες μεθόδους για σύγκριση από τη βιβλιογραφία. Κάθε μία από τις επιλεγμένες μεθόδους αντιπροσωπεύει μια διαφορετική γραμμή έρευνας, συγκεκριμένα τη βάσει ανταγωνιστικής συνάρτησης κόστους **BERT-DAAT**, την αυτοεκπαίδευση βάσει XLM-R **p+CFd** και τη βάσει κόμβων **R-PERL**.

Επιπρόσθετα, αναφέρουμε αποτελέσματα για τις ακόλουθες ρυθμίσεις με μοντέλα BERT:
Source-Only (SO): Προσαρμόζουμε το BERT σε επισημειωμένα δεδομένα του τομέα πηγής, χωρίς να χρησιμοποιούμε δεδομένα στόχου.
Domain Pre-Training (DPT): Χρησιμοποιούμε τα μη επισημειωμένα δεδομένα του τομέα στόχου για να συνεχίσουμε την προεκπαίδευση του BERT με συνάρτηση κόστους MLM (όπως στο Fig. 1.6β′) και στη συνέχεια προσαρμόζουμε το προκύπτον μοντέλο σε επισημειωμένα δεδομένα του τομέα πηγής.
Domain Adversarial Training (DAT): Ξεκινώντας από το BERT προεκπαιδευμένο στον τομέα (δείτε Fig.1.6β′), στη συνέχεια προσαρμόζουμε το μοντέλο με ανταγωνιστική μάθηση DAT. Για ένα μοντέλο BERT με παραμέτρους θ, με L_{CLF} να είναι μια συνάρτηση κόστους cross-entropy για την πρόβλεψη επισημειωμένα εργασίας, L_{ADV} να είναι μια συνάρτηση κόστους cross-entropy για την πρόβλεψη τομέα και λ_d να είναι ένας παράγοντας βάρους, το DAT αποτελείται από το κριτήριο ελαχιστοποίησης που περιγράφεται στην Εξ. 1.8.

$$\min_{\theta} L_{CLF}(\theta; D_S) - \lambda_d L_{ADV}(\theta; D_S, D_T) \tag{1.8}$$

UDALM: Η προτεινόμενη μέθοδος, όπου προσαρμόζουμε το μοντέλο που δημιουργήθηκε στο βήμα προεκπαίδευσης τομέα χρησιμοποιώντας τη μικτή συνάρτηση κόστους στην Εξ. 1.7.

Παρουσιάζουμε αποτελέσματα για όλες τις 12 ρυθμίσεις προσαρμογής τομέα στον Πίνακα 1.2. Τα αποτελέσματα για το BERT SO, BERT DAT, BERT DPT και UDALM είναι μέσοι όροι από πέντε εκτελέσεις και περιλαμβάνουμε τυπικές αποκλίσεις. Η τελευταία γραμμή του Πίνακα 1.2 περιέχει τη μέση ακρίβεια και τις αποκλίσεις για όλα τα ζεύγη τομέων. Το UDALM υπερτερεί όλων των άλ-

Πίνακας 1.2: Ακρίβεια ταξινόμησης για τεχνικές μη επιβλεπόμενης προσαρμογής μεταξύ πεδίων στα δώδεκα ζευγάρια τομέων του Amazon Reviews.

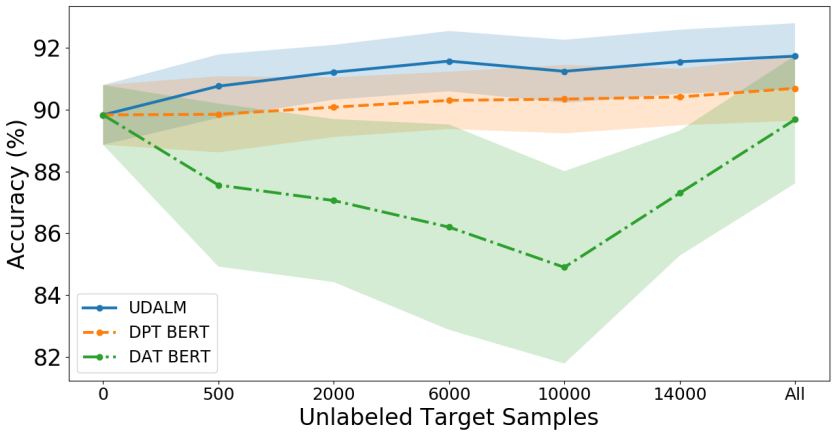
	R-PERL	DAAT	p+CFd	SO BERT	DAT BERT	DPT BERT	UDALM
$B \to D$	87.8	90.9	87.7	89.51 ± 0.76	87.31 ± 2.14	90.49 ± 0.38	$\mathbf{90.97 \pm 0.22}$
$B \to E$	87.2	88.9	91.3	90.51 ± 0.51	86.91 ± 2.71	90.38 ± 1.59	$\mathbf{91.69 \pm 0.31}$
$B \to K$	90.2	88.0	92.5	91.75 ± 0.28	90.59 ± 1.17	92.66 ± 0.43	$\mathbf{93.21 \pm 0.22}$
$D \to B$	85.6	89.7	**91.5**	90.26 ± 0.64	86.30 ± 3.10	91.02 ± 0.75	91.00 ± 0.42
$D \to E$	89.3	90.1	91.6	88.71 ± 1.48	87.85 ± 1.24	91.03 ± 0.82	$\mathbf{92.30 \pm 0.47}$
$D \to K$	90.4	88.8	92.5	91.22 ± 0.69	89.95 ± 1.53	92.30 ± 0.42	$\mathbf{93.66 \pm 0.37}$
$E \to B$	90.2	89.6	88.7	87.96 ± 0.89	85.65 ± 1.91	88.52 ± 0.55	$\mathbf{90.61 \pm 0.30}$
$E \to D$	84.8	**89.3**	88.2	87.37 ± 0.64	83.99 ± 1.31	87.85 ± 0.47	88.83 ± 0.61
$E \to K$	91.2	91.7	93.6	93.30 ± 0.50	92.45 ± 1.35	94.39 ± 0.72	$\mathbf{94.43 \pm 0.24}$
$K \to B$	83.0	**90.8**	89.8	88.15 ± 0.64	85.07 ± 1.03	88.83 ± 0.81	90.29 ± 0.51
$K \to D$	85.6	**90.5**	87.8	87.23 ± 0.49	84.11 ± 0.62	88.52 ± 0.69	89.54 ± 0.59
$K \to E$	91.2	93.2	92.6	93.23 ± 0.34	92.07 ± 0.24	93.42 ± 0.40	$\mathbf{94.34 \pm 0.26}$
Μέσος όρος	87.50	90.12	90.63	89.93 ± 0.65	87.68 ± 1.53	90.78 ± 0.67	$\mathbf{91.74 \pm 0.38}$

λων τεχνικών, επιτυγχάνοντας απόλυτη βελτίωση 1.81% έναντι του βασικού BERT SO. Για δίκαιη σύγκριση, συγκρίνουμε μόνο με μεθόδους βασισμένες σε προεκπαιδευμένα μοντέλα, κυρίως BERT. Παρατηρούμε ότι το BERT που προσαρμόζεται μόνο με επισημειωμένα δεδομένα του τομέα πηγής, χωρίς καμία γνώση του τομέα στόχου, αποτελεί ένα ανταγωνιστικό βασικό μοντέλο. Αυτό το μοντέλο πηγής μόνο καταφέρνει ακόμα να υπερβεί κορυφαίες μεθόδους μη επιβλεπόμενης προσαρμογής τομέα.

Περαιτέρω διερευνούμε την επίδραση της χρήσης διαφορετικού όγκου μη επισημειωμένων δεδομένων του τομέα στόχου στην απόδοση του μοντέλου, για να μελετήσουμε την αποδοτικότητα δειγμάτων του UDALM. Πειραματιζόμαστε με ρυθμίσεις 500, 2000, 6000, 10000 και 14000 δειγμάτων, περιορίζοντας τυχαία τον αριθμό των δεδομένων του τομέα στόχου. Για κάθε ρύθμιση διεξάγουμε τρία πειράματα με μοντέλα BERT: (1) DPT, (2) DAT και (3) UDALM. Όταν δεν υπάρχουν διαθέσιμα δεδομένα στόχου, όλες οι μέθοδοι είναι ισοδύναμες με ένα BERT που έχει προσαρμοστεί μόνο στην πηγή. Επίσης, δεν προσαρμόζουμε τις υπερπαραμέτρους για το DPT ή το UDALM. Το Fig. 1.7 δείχνει τη μέση ακρίβεια στα δώδεκα σενάρια προσαρμογής του μελετημένου συνόλου δεδομένων. Βλέπουμε ότι το UDALM παράγει σταθερή βελτίωση της απόδοσης όταν περιορίζουμε τον όγκο των δεδομένων στόχου, δείχνοντας ότι μπορεί να χρησιμοποιηθεί σε σενάρια με περιορισμένους πόρους. Ωστόσο, η εκπαίδευση του BERT με τρόπο ανταγωνιστικό στον τομέα δείχνει αστάθειες.

1.2.5 Μη επιβλεπομενη προσαρμογη μεταξυ πεδιων για αναγνωριση φωνης στα Νεα Ελληνικα

Τα σύγχρονα συστήματα αναγνώρισης ομιλίας παρουσιάζουν ταχεία υποβάθμιση της απόδοσης όταν αλλάζει ο τομέας εφαρμογής. Αυτό το ζήτημα είναι ιδιαίτερα διαδεδομένο σε περιβάλλοντα με περιορισμένα δεδομένα, όπως γλώσσες με χαμηλούς πόρους, όπου η ποικιλομορφία των δεδομένων εκπαίδευσης είναι περιορισμένη. Σε αυτήν την εργασία, προτείνουμε το M2DS2, μια απλή και αποδοτική ως προς τον αριθμό δειγμάτων στρατηγική μετεκπαίδευσης για μεγάλα προεκπαιδευμένα μοντέλα ομιλίας, που βασίζεται στην αυτοεπίβλεψη με μείξη δεδομένων του πηγαίου τομέα και του

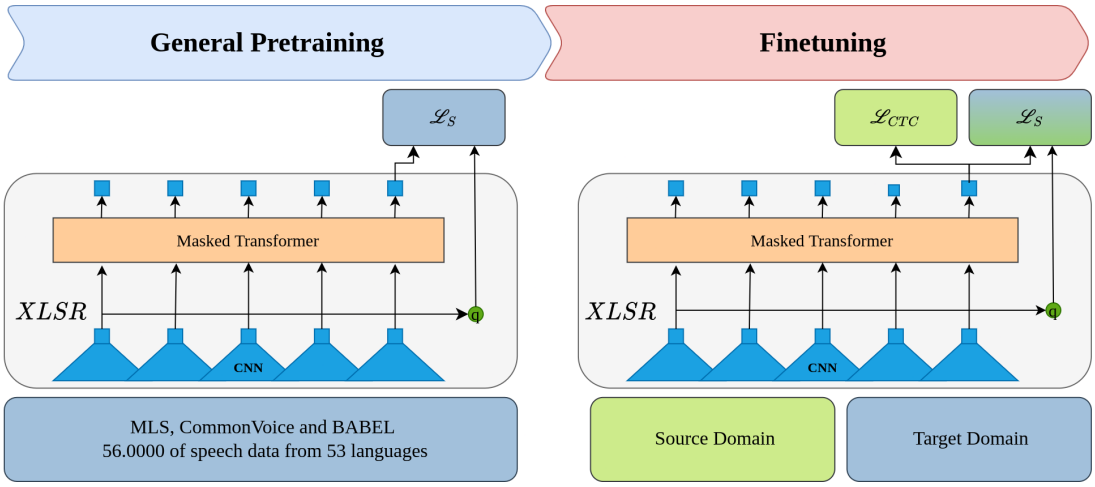


Εικόνα 1.7: Μέση ακρίβεια για διαφορετική ποσότητα διαθέσιμων δεδομένων στόχου για: (1) DPT BERT (2) DAT BERT και (3) UDALM.

τομέα στόχου. Διαπιστώνουμε ότι η συμπερίληψη της αυτοεπίβλεψης με δεδομένα του τομέα πηγής σταθεροποιεί την εκπαίδευση και αποφεύγει την κατάρρευση των εσωτερικών αναπαραστάσεων του δικτύου λειτουργίας. Για αξιολόγηση, συλλέγουμε το HParl, ένα σύνολο μεταγεγγραμμένων δεδομένων ομιλίας 120 ωρών για τα ελληνικά, που αποτελείται από συνόδους ολομέλειας της Βουλής των Ελλήνων. Συγχωνεύουμε την HParl με δύο δημοφιλείς ελληνικές βάσεις δεδομένων για να δημιουργήσουμε το GREC-MD, ένα σύνολο δοκιμής για την αξιολόγηση πολλών τομέων των ελληνικών συστημάτων αναγνώρισης ομιλίας. Στα πειράματά μας, διαπιστώνουμε ότι, ενώ άλλες μέθοδοι προσαρμογής μεταξύ τομέων χωρίς επίβλεψη αποτυγχάνουν σε αυτό το περιβάλλον περιορισμένων πόρων, το M2DS2 αποφέρει σημαντικές βελτιώσεις για την προσαρμογή μεταξύ τομέων, ακόμη και όταν είναι διαθέσιμες μόνο λίγες ώρες ήχου εντός τομέα. Όταν χαλαρώνουμε το πρόβλημα σε πρόβλημα ασθενούς επίβλεψης, διαπιστώνουμε ότι η ανεξάρτητη προσαρμογή ήχου χρησιμοποιώντας M2DS2 και γλώσσας χρησιμοποιώντας απλές τεχνικές επαύξησης γλωσσικών μοντέλων είναι ιδιαίτερα αποτελεσματική, αποδίδοντας ποσοστά λεκτικών λαθών συγκρίσιμα με μοντέλα εκπαιδευμένα με πλήρη επίβλεψη.

Μεθοδος

Η Εικ. 1.8 παρουσιάζει την προτεινόμενη στρατηγική μετεκπαίδευσης. Η βασική διαίσθηση είναι ότι θέλουμε το μοντέλο να μαθαίνει συνεργατικά την εν λόγω έργασία (στην περίπτωσή μας αναγνώριση ομιλίας) ενώ προσαρμόζεται στον στοχευμένο τομέα μέσω αυτοεπίβλεψης σε δεδομένα εντός τομέα. Στα αριστερά, βλέπουμε το γενικό στάδιο προεκπαίδευσης του XLSR-53, το οποίο προεκπαιδεύεται σε 56K ώρες πολυγλωσσικών ηχητικών δεδομένων χρησιμοποιώντας τον αντιθετικό στόχο στην Εξ. (1.9).



Εικόνα 1.8: Προσαρμογή στον στοχευμένο τομέα μέσω αυτοεπίβλεψης. Στα αριστερά, βλέπουμε το γενικό στάδιο προεκπαίδευσης του XLSR-53 χρησιμοποιώντας την αυτοεπιβλεπόμενη συνάρτηση κόστους L_s. Η γενική προεκπαίδευση διεξάγεται σε $56,000$ ώρες ηχητικών δεδομένων σε 53 γλώσσες. Στα δεξιά, βλέπουμε το προτεινόμενο στάδιο προσαρμογής λεπτής επιμόρφωσης, όπου η εργασία αναγνώρισης ομιλίας μαθαίνεται χρησιμοποιώντας δεδομένα από τον πηγαίο τομέα με μεταγραφές, ενώ η προσαρμογή στον στοχευμένο τομέα εκτελείται συμπεριλαμβάνοντας την αυτοεπιβλεπόμενη συνάρτηση κόστους σε ηχητικά δεδομένα από τον πηγαίο και τον στοχευμένο τομέα.

$$L_s = \underbrace{-log\frac{e^{s(z_t,q_t)}}{\sum_{\tilde{q}\sim Q_t} e^{s(z_t,\tilde{q})}}}_{\text{Contrastive Loss}} \overbrace{-\frac{1}{GV}\sum_{g=1}^{G}\sum_{v=1}^{V}\bar{p}_{g,v}log(\bar{p}_{g,v})}^{\text{Diversity Loss}} \tag{1.9}$$

Στα δεξιά, βλέπουμε το προτεινόμενο στάδιο μετεκπαίδευσης, όπου διαμορφώνουμε μια μικτή συνάρτηση κόστους:

$$L = L_{CTC}(x_s, y_s) + \alpha L_s(x_s) + \beta L_s(x_t), \tag{1.10}$$

όπου $(x_s, y_s) \sim \mathcal{S}(x, y)$, $x_t \sim \mathcal{T}(x)$, L_{CTC} είναι η συνάρτηση κόστους Connectionist Temporal Classification (CTC), βελτιστοποιημένη χρησιμοποιώντας δεδομένα μεταγραφής του πηγαίου τομέα, και L_s είναι η αντιθετική συνάρτηση κόστους από την Εξ. (1.9). Κλιμακώνουμε τη συνεισφορά κάθε όρου χρησιμοποιώντας τις υπερ-παραμέτρους α και β.

Σημειώστε ότι σε αντίθεση με προηγούμενες εργασίες, οι οποίες χρησιμοποιούν μόνο αυτοεπίβλεψη στον τομέα στόχο, εμείς αξιοποιούμε και δείγματα από τον πηγαίο και τον στοχευμένο τομέα για τη μικτή αυτοεπίβλεψη. Διαπιστώνουμε ότι αυτό είναι ουσιώδες στην περίπτωσή μας για να αποφύγουμε το πρόβλημα mode collapse, δηλαδή, το μοντέλο να χρησιμοποιεί μόνο μερικά από τα διαθέσιμα διανύσματα διακριτού κώδικα (codevectors). Η ταυτόχρονη αυτοεπίβλεψη στα δεδομένα

Πίνακας 1.3: Το σώμα δεδομένων GREC-MD. Μπορούμε να δούμε τη διάρκεια κάθε υποσυνόλου σε μορφή ώρες : λεπτά : δευτερόλεπτα, καθώς και τον αριθμό των ομιλητών για κάθε ένα από τα υποσώματα.

Σύνολο δεδομένων	Τομέας	Ομιλητές	Train	Dev	Test	Συνολική διάρκεια
HParl	Δημόσιος (πολιτικός) λόγος	387	99:31:41	9:03:33	11:12:28	119:47:42
CV	Ομιλία από πληθοπορισμό	325	12:16:17	1:57:44	1:59:19	16:13:20
Logotypografia	Ειδήσεις	125	51:58:45	9:08:35	8:59:22	70:06:42
Total	-	713	163:46:43	20:09:52	22:11:44	206:08:19

πηγής και στόχου αμβλύνει το mode collapse αγκυρώνοντας τον χώρο των διακριτών κωδικών του στόχου να έχει παρόμοια δομή με αυτούς του τομέα πηγής.

Συνολα Δεδομενων

Για τα πειράματά μας συνθέτουμε ένα σώμα ομιλίας για την ελληνική γλώσσα, το οποίο είναι κατάλληλο για πολυ- και δια-τομεακή αξιολόγηση. Το σώμα GREC-MD περιέχει 206 ώρες ελληνικής ομιλίας. Το ηχητικό υλικό διαιρείται σε ατομικές εκφράσεις και κάθε έκφραση συνδυάζεται με την αντίστοιχη μεταγραφή της. Ο Πίνακας 1.3 περιλαμβάνει μια σύνοψη των περιλαμβανομένων υποσωμάτων, καθώς και των διαίρεσεων εκπαίδευσης, ανάπτυξης και δοκιμών. Το σετ δεδομένων σχεδιάστηκε με τρεις βασικές αρχές στο μυαλό:

1. **Όγκος Δεδομένων**: Συλλέγουμε το μεγαλύτερο δημόσια διαθέσιμο σώμα αναγνώρισης ομιλίας για την ελληνική γλώσσα, που μπορεί να κλιμακωθεί σε εκατοντάδες ώρες μεταγραφής ήχου.

2. **Χρονική Σχετικότητα**: Η γλώσσα αλλάζει με τον καιρό. Στοχεύουμε σε ένα ενημερωμένο σώμα που περιλαμβάνει τους τελευταίους όρους και θέματα που εμφανίζονται στην καθημερινή ομιλία.

3. **Πολυ-Τομεακή Αξιολόγηση:** Η αξιολόγηση σε ένα μόνο τομέα μπορεί να οδηγήσει σε παραπλανητικές εκτιμήσεις της αναμενόμενης απόδοσης για τα μοντέλα αναγνώρισης ομιλίας. Για παράδειγμα, τα κορυφαία στην τεχνολογία μοντέλα αναγνώρισης ομιλίας επιτυγχάνουν κάτω από 5% Word Error Rate (WER) στα σετ δοκιμών του Librispeech , αλλά αυτό αποτελεί υπερεκτίμηση της πραγματικής απόδοσης του συστήματος. Αυτό εντείνεται όταν λαμβάνουμε υπόψη διαφορετικές ακουστικές συνθήκες ή ορολογία. Θεωρούμε την πολυ-τομεακή αξιολόγηση ουσιώδη κατά την ανάπτυξη μοντέλων που θα χρησιμοποιηθούν σε πραγματικές εφαρμογές.

Για να ικανοποιήσουμε τα πρώτα δύο σημεία, συλλέγουμε δεδομένα από μια δημόσια, συνεχώς ενημερωμένη πηγή, δηλαδή από τα Πρακτικά της Ελληνικής Βουλής, όπου οι ηχογραφήσεις των κοινοβουλευτικών συνεδριάσεων ανεβαίνουν τακτικά. Αναφερόμαστε σε αυτό το σώμα δεδομένων ως HParl (HP). Συγκεκιμένα, στο HParl συμπεριλαμβάνουμε συνεδρίες στο διάστημα 2018-2022, όπως φαίνεται στον Πίνακα 1.4. Το πλεονέκτημα της χρήσης αυτής της πηγής είναι η απλή συλλογή ενός συνεχώς αυξανόμενου, μεταγεγγραμμένου σώματος ήχου από πολλούς ομιλητές που είναι πάντα ενημερωμένο, καθώς οι κοινοβουλευτικές συζητήσεις περιστρέφονται γύρω από τρέχοντα θέματα.

Πίνακας 1.4: Συνεδρίες που έχουν συμπεριληφθεί στο HParl. Η στήλη "Ώρες" αναφέρεται στις ώρες ήχου πριν την τμηματοποίηση.

Αρχική ημερομηνία	**Τελική ημερομηνία**	**#Συνεδριών**	**Ώρες**
15-02-2022	01-03-2022	10	55
18-01-2019	01-02-2019	10	52
28-03-2019	10-05-2019	20	108
10-12-2018	21-12-2018	10	88

Για την πολυ-τομεακή αξιολόγηση, συνδυάζουμε το HParl με δύο δημόσια διαθέσιμα σώματα δεδομένων, δηλαδή τη Logotypografia (LG) και το CommonVoice (CV), τα οποία έχουν διαφορετικές ακουστικές και γλωσσικές ιδιότητες. Αναφερόμαστε στο συγχωνευμένο, πολυ-τομεακό σώμα δεδομένων ως GREC-MD.

Πειραματα

Εδώ, αξιολογούμε την αποτελεσματικότητα του M2DS2 για μη επιβλεπόμενη προσαρμογή μεταξύ τομέων. Συγκρίνουμε με τρεις βάσεις:

Πίνακας 1.5: Η επίδοση του M2DS2 για μη επιβλεπόμενη προσαρμογή μεταξύ των HP, CV και LG. Το A $\rightarrow$ B δηλώνει ότι το A είναι ο πηγαίος τομέας και το B είναι ο στοχευμένος τομέας. (G) δηλώνει άπληστη αποκωδικοποίηση. (LM) δηλώνει αναζήτηση δέσμης με επαναβαθμολόγηση από γλωσσικό μοντέλο. Αναφέρουμε το WER στο σετ δοκιμών στόχου, καθώς και το WRR (%) επί του SO, δηλαδή τη σχετική βελτίωση πάνω από το βασικό μοντέλο SO. WER: χαμηλότερο είναι καλύτερο. WRR: υψηλότερο είναι καλύτερο.

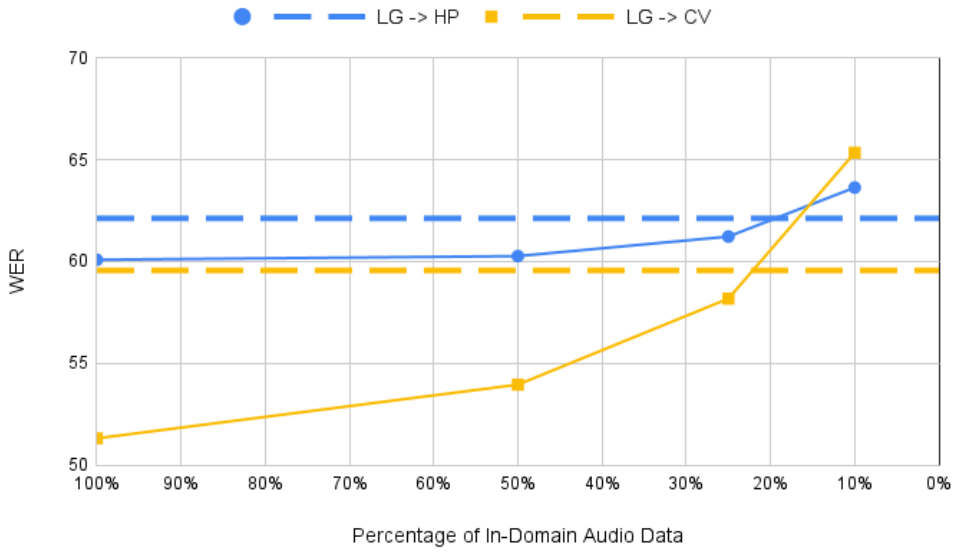
Μέθοδος	SO (G)	CPT (G)		PSL (G)		M2DS2 (G)		SO (LM)	CPT (LM)		PSL (LM)		M2DS2 (LM)	
Σενάριο	WER	WER	WRR	WER	WRR	WER	WRR	WER	WER	WRR	WER	WRR	WER	WRR
HP $\rightarrow$ CV	55.90	54.80	4.1	53.48	9.1	**52.95**	**11.1**	25.26	23.26	12.7	24.34	5.9	**18.35**	**43.9**
HP $\rightarrow$ LG	48.65	47.99	4.0	51.75	−18.6	**46.47**	**12.5**	30.34	33.88	−91.0	31.92	−40.6	**29.56**	**20.1**
LG $\rightarrow$ CV	59.57	60.81	−4.1	63.28	−12.3	**51.31**	**27.3**	25.96	29.10	−19.1	23.46	15.2	**17.30**	**52.7**
LG $\rightarrow$ HP	62.13	60.60	4.3	66.60	−12.4	**60.09**	**5.7**	31.48	31.54	−0.4	39.15	−48.4	**31.36**	**0.8**
CV $\rightarrow$ LG	69.55	68.98	1.5	68.29	3.4	**63.40**	**16.4**	50.80	47.61	13.1	42.53	34.0	**36.93**	**57.0**
CV $\rightarrow$ HP	70.72	71.79	−2.4	69.68	2.3	**68.70**	**4.5**	52.09	48.14	10.8	53.8	−4.7	**41.88**	**28.0**

1. **Source-Only (SO):** Πραγματοποιούμε μετεκπαίδευση του XLSR-53 (CTC) χρησιμοποιώντας μόνο δεδομένα από τον πηγαίο τομέα και εκτελούμε αποκωδικοποίηση στο σετ ελέγχου του στοχευόμενου τομέα. Δεν χρησιμοποιούνται δεδομένα εντός τομέα για προσαρμογή.

2. **Continuous Pre-Training (CPT):** Πραγματοποιούμε μια φάση προεκπαίδευσης χρησιμοποιώντας τη συνάρτηση κόστους στην Εξ. (10.1) στα σετ εκπαίδευσης του πηγαίου και του στοχευόμενου τομέα, για να δημιουργήσουμε προσαρμοσμένες εκδόσεις του XLSR-53. Η προεκπαίδευση διεξάγεται για 20000 βήματα με μέγεθος παρτίδας 4. Χρησιμοποιείται μόνο το ηχητικό υλικό, χωρίς μεταγραφές. Στη συνέχεια, τα προσαρμοσμένα σημεία ελέγχου μετεκπαιδεύονται με τη χρήση της απώλειας CTC στα δεδομένα του πηγαίου τομέα με μεταγραφές. Η αξιολόγηση γίνεται στο σετ ελέγχου του στοχευόμενου τομέα.

3. **Pseudo-Labeling (PSL):** Μετεκπαιδεύουμε το XLSR-53 χρησιμοποιώντας τα δεδομένα του πηγαίου τομέα με το κόστος CTC. Στη συνέχεια, εκτελούμε αποκωδικοποίηση με το μοντέλο του πηγαίου τομέα, για να εξαγάγουμε ασημένιες μεταγραφές για το σετ εκπαίδευσης του στοχευόμενου τομέα. Χρησιμοποιώντας τις ασημένιες μεταγραφές που παράχθηκαν στο πρώτο βήμα, δημιουργούμε το ψευδο-επισημειωμένο σετ εκπαίδευσης του στοχευόμενου τομέα και το συγχωνεύουμε με το σωστά μεταγραμμένο σετ εκπαίδευσης του πηγαίου τομέα. Τέλος, επαναφέρουμε το μοντέλο στα αρχικά βάρη του XLSR-53 και το μετεκπαιδεύουμε στο συνδυασμένο σετ εκπαίδευσης.

Τα αποτελέσματα προσαρμογής φαίνονται στον Πίνακα 1.5.



Εικόνα 1.9: Απόδοση του M2DS2 για τις ρυθμίσεις LG → HP (μπλε κύκλος) και LG → CV (κίτρινο τετράγωνο), όταν μειώνουμε την ποσότητα των διαθέσιμων δειγμάτων στόχου στο 50%, 25%, και 10% του αρχικού σετ δεδομένων (οριζόντιος άξονας). Η απόδοση του SO εμφανίζεται με τις διακεκομμένες γραμμές. Κατακόρυφος άξονας: WER, Οριζόντιος Άξονας: ποσοστό ηχητικού υλικού στόχου (100% → 0%)

Μία βασική παρατήρηση στη βιβλιογραφία και στα πειράματά μας είναι ότι ο CPT απαιτεί μεγάλες ποσότητες ηχητικού υλικού από τον στοχευόμενο τομέα χωρίς μεταγραφές. Αυτό δημιουργεί το ερώτημα, μπορούμε να αξιοποιήσουμε την αυτο-επίβλεψη για προσαρμογή τομέων σε συνθήκες περιορισμένων δεδομένων; Αυτό προσφέρει μια ελπιδοφόρα δυνατότητα για προσαρμογή όταν η συλλογή ηχογραφήσεων εντός τομέα είναι δαπανηρή ή ανέφικτη. Στο Σχήμα 1.9 αξιολογούμε την απόδοση του M2DS2 όταν μειώνουμε την ποσότητα του ηχητικού υλικού του στοχευόμενου τομέα.

Τέλος ο Πίνακας 1.6 συνοψίζει τα κέρδη στην επίδοση για τη μη επιβλεπόμενη προσαρμογή πεδίου όταν χρησιμοποιούμε ακουστική προσαρμογή μέσω του M2DS2 σε συνδυασμό με γλωσσική προσαρμογή μέσω απλών τεχνικών προσαρμογής n-gram LM.

Πίνακας 1.6: Κλείνοντας το χάσμα μεταξύ της εκπαίδευσης SO και της πλήρως επιβλεπόμενης εκπαίδευσης για το σενάριο προσαρμογής LG → CV χρησιμοποιώντας το M2DS2, με διαφορετικές ποσότητες διαθέσιμου μη συζευγμένου ηχητικού υλικού και κειμένου εντός τομέα. (U): μη επιβλεπόμενη ακουστική ή γλωσσική προσαρμογή. (W): ασθενώς επιβλεπόμενη προσαρμογή.

Μέθοδος	#Audio (h)	#Tokens	LM	WER
SO (U)	-	-	N/A	59.57
M2DS2 (U)	3	-	N/A	57.31
M2DS2 (U)	12	-	N/A	51.31
SO (U)	-	-	Γενικό	25.96
SO (U)	-	38, 632	Επαυξημένο	24.67
SO (U)	-	751, 953	Επαυξημένο	20.46
M2DS2 (U)	3	-	Γενικό	20.7
M2DS2 (U)	12	-	Γενικό	17.3
M2DS2 (W)	3	38, 632	Επαυξημένο	19.31
M2DS2 (W)	12	38, 632	Επαυξημένο	16.29
M2DS2 (W)	3	751, 953	Επαυξημένο	12.84
M2DS2 (W)	12	751, 953	Επαυξημένο	10.61
Επιβλεπόμενη	12	751, 953	Γενικό	9.52
Επιβλεπόμενη	12	751, 953	Επαυξημένο	7.94

1.3 Συνεισφορες

1. Μάθηση υποχώρων με πολυδιάστατη κλιμάκωση:

   (α′) Pattern search MDS: Προτείνεται ένας νέος αλγόριθμος μείωσης της διαστατικότητας που συνδυάζει την πολυδιάστατη κλιμάκωση με βελτιστοποίηση χωρίς παραγώγους. Αυτός ο αλγόριθμος προσφέρει γρήγορη και εγγυημένη σύγκλιση, ενώ παρέχουμε και μια βελτιστοποιημένη υλοποίηση.

   (β′) Αξιολόγηση επίδοσης και ανθεκτικότητας: Η απόδοση του αλγορίθμου αξιολογείται μέσω πειραμάτων στη γεωμετρία πολλαπλοτήτων, την ταξινόμηση εικόνων και τη σημασιολογία των λέξεων. Η ανθεκτικότητά του αξιολογείται σε σενάρια με θορυβώδεις εισόδους και απουσία δεδομένων.

   (γ′) Μείωση της διαστατικότητας για την αναγνώριση συναισθημάτων από φωνή: Εξετάζεται η χρησιμότητα του pattern search MDS και άλλων τεχνικών μείωσης διαστατικότητας για την αναγνώριση συναισθημάτων από φωνή, δείχνοντας ότι τα μειωμένα συνόλα χαρακτηριστικών μπορούν να επιτύχουν ανταγωνιστική απόδοση.

2. Μικτή αυτοεπίβλεψη για αποδοτική προσαρμογή τομέα χωρίς επίβλεψη (UDA):

   (α′) Προσαρμογή τομέα χωρίς επίβλεψη μέσω γλωσσικής μοντελοποίησης: Η προσέγγιση της μικτής αυτοεπίβλεψης εφαρμόζεται στην ταξινόμηση κειμένου, ειδικά για την πρόβλεψη της συναισθηματικής αξιολόγησης των κριτικών προϊόντων της Amazon. Επιδεικνύεται η αποτελεσματικότητα με ελάχιστη ρύθμιση υπερπαραμέτρων, ακόμα και με περιορισμένα δεδομένα στον ίδιο τομέα. Αυτή η προσέγγιση αντιμετωπίζει επίσης τους περιορισμούς της ανταγωνιστικής εκπαίδευσης τομέα.

   (β′) Αποδοτική προσαρμογή του τομέα χωρίς επίβλεψη σε συστήματα αναγνώρισης ομιλίας: Αυτή η μελέτη επικεντρώνεται στην μικτή αυτο-επίβλεψη για την ακουστική προσαρμογή τομέα για συστήματα αναγνώρισης ομιλίας, χρησιμοποιώντας ένα πρόσφατα δημιουργημένο σύνολο δεδομένων ομιλίας για τη Νέα Ελληνική. Βρίσκουμε ότι η αποτελεσματικότητα της μικτής αυτο-επίβλεψης ποικίλλει ανάλογα με την εργασία προεκπαίδευσης και υπογραμμίζουμε τη σημασία της χρήσης τόσο δεδομένων εκτός τομέα, όσο και δεδομένων εντός τομέα για αυτοεπίβλεψη, ώστε να αποτραπεί η κατάρρευση των εσωτερικών αναπαραστάσεων. Επιπλέον, δοκιμάζονται τεχνικές επέκτασης για την προσαρμογή του γλωσσικού μοντέλου σε νέους τομείς.

"Deep Blue was intelligent the way your programmable alarm clock is intelligent. Not that losing to a $10 million alarm clock made me feel any better."

Garry Kasparov, Deep Thinking: Where Machine Intelligence Ends and Human Creativity Begins

2 Preface

Chess masters in the late 90s witnessed technology transform their field, and they had to accept that they were no longer the dominant chess entities on the planet and that how they trained and competed would radically change. Almost thirty years later, chess masters still exist and have benefited by incorporating chess engines into their training regime.

Artificial Intelligence (AI) researchers face a similar situation in the past couple of years, with radical shifts in both their practical experience, and, more importantly, with shifts in their intuitions and perspectives. Bias-variance is no longer a trade-off in the over-parameterized region, overfitting can be benign, and scaling-up is now the dominant strategy for good performance. The reader should not infer that I argue against the current trends, nor that I lament the new status quo[1]. That would be an exercise in futility, as the "scale-up" strategy is well-founded, and it is hard to argue with the results.

The key argument of this dissertation is that drawing inspiration from the cognitive sciences, and studying how biological neural networks operate, is essential for building effective artificial neural networks. To me, it would feel disingenuous not to place this argument in the modern context. Therefore, we will begin our discussion with an overview of the scaling laws that characterize artificial and biological neural networks, and the differences in their operation. Then, we will introduce the key ideas of this work, i.e., (i) conceptual spaces that rely on compact representations, and (ii) neural plasticity and how it leads to sample-efficient training schemes. The human brain is the most intelligent machine we know; the word "intelligent" not being considered as a collection of metrics in a set of narrowly defined tasks, rather as a combination of good performance, rapid adaptability, versatility, ability to act, introspect, and communicate. These properties are hard to define rigorously, and even harder to measure, but nevertheless they are desirable for truly intelligent machines. In this dissertation I intend to demonstrate that drawing inspiration from biological neural networks can complement and enhance future architecture design by steering it towards, at least some of, these desiderata. With this context in place, we shall begin our exploration.

"An attempt will be made to find how to make machines use language, form abstractions, and concepts, solve kinds of problems now reserved for humans, and improve themselves. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer."

Dartmouth AI workshop proposal, 1955

3 Introduction

3.1 Bigger and better

In 1955 a group of influential scientists gathered in the Dartmouth AI workshop, coining the term Artificial Intelligence in the process. They viewed the brain as a machine, and learning and intelligence as procedures that can be precisely described and simulated. The key aspects of the artificial intelligence problem they defined were (i) automatic *computation*, (ii) natural *language* usage, (iii) *neural* networks, (iv) *efficient* calculation, (v) *self-improvement*, (vi) ability to form *abstractions* from sensory data, and (vii) *creativity*. 68 years after the Dartmouth AI workshop, the current state-of-the-art models are able to demonstrate close-to-human or super-human ability in almost all of these aspects. Computer vision models have long surpassed human performance in object detection tasks[2]. GPT-4 and ChatGPT[3,4] have impressive reasoning skills, demonstrate theory of mind[5] properties, and can effectively communicate with humans[6]. ChatGPT also achieves genius level scores in standardized verbal IQ assessment tests [*].

3.1.1 Scaling laws of artificial and biological neural networks

Two advancements lie in the center of these technological feats; one model architecture and one learning paradigm. The model architecture is the Transformer[7], a highly scalable attention network based on Multi-Layer Perceptron (MLP), and weak inductive biases[8]. The new learning paradigm is Self-Supervised Learning (SSL)[9], where, through different pretext tasks, a network is learning to predict partial patterns about the data itself. Some example pretext tasks include masked language modeling[10], image inpainting[11], and contrastive learning[12]. The success of Transformers combined with SSL, lies in the fact that it enables scaling of performance with model and training data size. Thus, a dominant strategy has formed in the machine learning community, of building "bigger and better" models [†]. Fig. 3.1 summarizes this trend for recent Natural Language Processing (NLP) models.

[*] https://www.scientificamerican.com/article/i-gave-chatgpt-an-iq-test-heres-what-i-discovered/, accessed 9/11/2023.

[†] This strategy is "dominant" if we only care about model performance given an unbounded budget. There are multiple data-privacy, economic and environmental sustainability concerns that could prove to be a significant limiting factor for model growth in the near future[13–17].

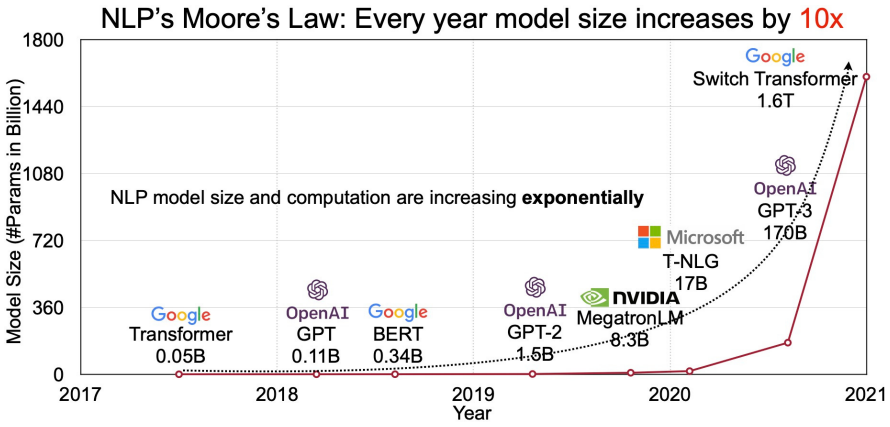


Figure 3.1: The trend in this graph shows an exponential increase of parameters in new NLP models. This results in increased reasoning capabilities, but also the potential forming of a new "Moore's law" for NLP. (image credit: MIT HAN Lab)

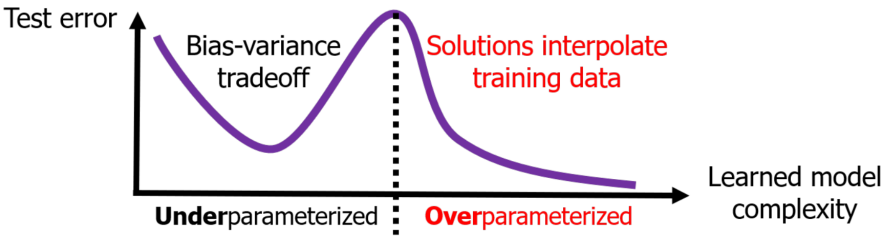
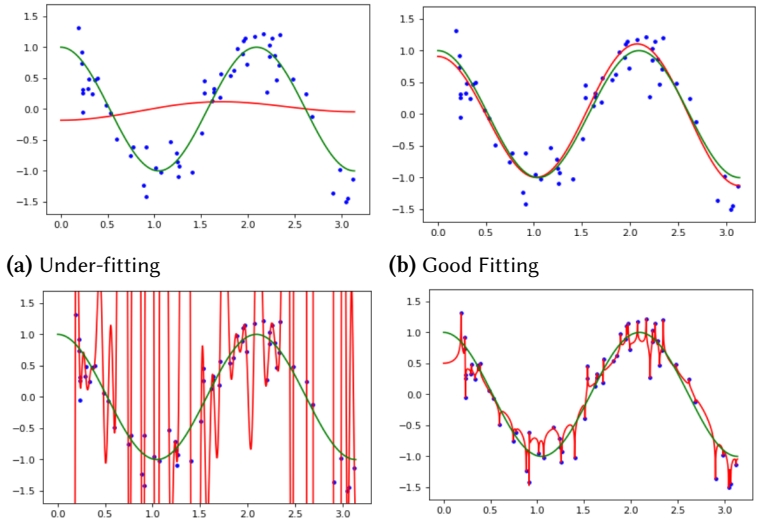


Figure 3.2: The double-descent error curve consists of the classical bias-variance trade-off regime and the modern interpolating regime. Predictors in the interpolating regime have zero training risk (perfectly fit the training data), while the test error can become arbitrarily small. (image credit: Dar et al.[18].)



(a) Under-fitting **(b)** Good Fitting

(c) Classical over-fitting **(d)** Benign Over-fitting

Figure 3.3: Illustration of benign overfitting of a linear regression model on a noisy cosine curve. The over-parameterized model in the last figure is able to accurately interpolate the training data. (image credit: Tsigler and Bartlett [19].)

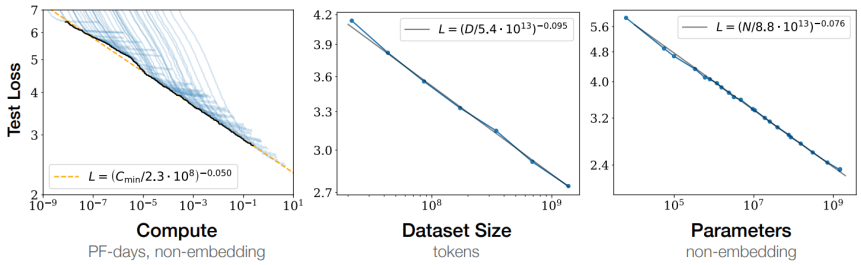


Figure 3.4: The power-law relationship between language model performance and compute (left), training data (center), and model size (right). (image credit: Kaplan et al. [23])

This over-parameterization has theoretical justifications, besides the discussed practical benefits. For example, recent theoretical research [18,20] has identified that over-parameterized models overcome the bias-variance trade-off, by being able to exactly fit the training data, leading to the double descent error curve in Fig. 3.2. This adds the "benign overfitting" [19,21,22] as a new mode of operation for neural networks, where the training error *and* the test error can become arbitrarily small, and the network learns both low frequency patterns and high-frequency noise in the training data (Fig. 3.3).

Consequently, researchers have attempted to empirically assess the fundamental question "how should the expected performance scale with model size? ". Kaplan, McCandlish et al. [23] studied the scaling laws of neural language models, using the test loss as a proxy metric for model performance. They derived empirically a power-law relationship between the test loss and the three scaling factors, i.e. compute, parameter count, and dataset size. This power-

Table 3.1: Number of parameters and training examples for state-of-the-art models across different modalities and tasks. For multitask models (e.g., Whisper also performs speech translation), we list the primary task.

Model	Modality	(Primary) task	Number of parameters	Number of training data
ResNet-152 [25]	Vision	Object recognition	58.5×10^6	1.3×10^6 images
ViT-Huge [26]	Vision	Object recognition	632×10^6	14.2×10^6 images
Stable-Diffusion [27]	Vision	Image generation	890×10^6	2.3×10^{12} images
XLSR-53 [28]	Speech	Automatic speech recognition	300×10^6	50×10^3 hours
Whisper [29]	Speech	Automatic speech recognition	1.6×10^9	680×10^3 hours
Voicebox [30]	Speech	Speech synthesis	330×10^6	60×10^3 hours
AudioLDM [31]	Audio	Audio generation	937×10^6	9×10^3 hours
GPT-3 [32]	Language	Language modeling	175×10^9	300×10^9 tokens
Chinchilla [24]	Language	Language modeling	70×10^9	1.4×10^{12} tokens

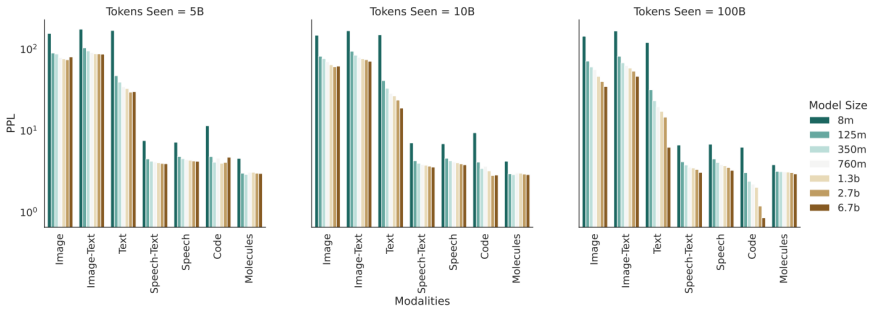


Figure 3.5: Autoregressive modeling performance for different modalities, with varied training data and model sizes. Observe that, for the 100B token setting, the performance of text and code modalities improves drastically with model size, contrary to other modalities. (image credit: Kaplan et al. [33])

law relationship is summarized in Fig. 3.4. The key projections obtained by this power-law relationships are that (i) language models will continue to improve with model size before reaching diminishing returns, (ii) larger models are more sample-efficient, and (iii) the amount of training data should increase sub-linearly in terms of model parameters to avoid overfitting. However, Hoffmann et al. [24] postulate that current language models are largely over-sized and under-trained, and find that the number of training tokens should increase linearly with the model size for optimal performance on a fixed compute budget.

Arguably, this radical need for increasing the model size is more pertinent to the text modality. Vision and speech models performance ranges from close to human to super-human, while their parameter budget stays modest, as can be seen in Table 3.1, where we summarize the parameter and training data budget for state-of-the-art vision, speech, and language models. This can also be observed in a recent study about the scaling laws of multimodal autoregressive models [33], where text-based modalities are shown to benefit more from larger model and dataset sizes. Fig. 3.5 from this paper summarizes the differences in scaling for textual versus speech and visual modalities in an autoregressive setting.

At this stage it is tempting to compare the scale of artificial neural networks to that of the

human brain [‡]. Biological brains are also governed by scaling laws, in terms of size, neuron density and energy consumption[34]. Direct estimations place the number of neurons in the human brain at 86 billion[35], with the number of synapses per neuron being estimated between 1000 and 10000[36,37], or $8.6 - 86$ trillion synapses (parameters). Remarkably, only $\sim 20\%$ of the neurons are in the cerebral cortex, which is associated with higher-order mental functions[35,38] ($16 - 22$ trillion parameters).

The brain has a modular structure, where different regions are more associated with different higher functions, i.e. visual and auditory cortex drive visual and audio perception, while Broca's and Wernicke's areas drive language processing[39]. Using the estimations of Beren Millidge[40], the visual cortex, the auditory cortex and the language-related areas have $3000 - 5000 \times 10^9$, $700 - 1000 \times 10^9$, and $400 - 700 \times 10^9$ synapses respectively. This puts current language models in the same order of magnitude with the brain language processing areas, while visual and audio models are estimated largely more parameter efficient.

In terms of data efficiency though there is a gap. As a thought experiment, a person that reads constantly for 80 years at a rate of 300 words per minute[41] will read $\sim 12.5 \times 10^9$ words, which is one to two orders of magnitude less than the training sets of state-of-the-art language models. Similarly, 80 years correspond to 700800 hours, which is roughly equivalent to the training set of Whisper[§]. In conclusion, while no gap is observed in terms of parameter efficiency, there is an apparent gap in data efficiency between artificial neural networks and the human brain. This may underscore fundamental processing differences, indicating that we have much to learn from the brain's inherent efficiency in achieving complex tasks with relatively few data.

3.2 Scope and Challenges

In this dissertation, two main pillars of machine learning systems are explored, i.e., low-rank representation learning, and data-efficient adaptation strategies. For the development of the techniques discussed throughout this manuscript, we draw from the accumulated knowledge obtained in prior studies of the human brain and its cognitive abilities. Specifically, we explore:

1. How low-dimensional conceptual spaces can be used to encode complex concepts.

2. How a balance can be found between stability and plasticity during neural network adaptation to unseen domains.

[‡]Though the study of the human brain has progressed, there is still much we do not know. Drawing comparisons between artificial and biological neural networks relies on multiple simplifications and Fermi estimations, e.g., $1 - 1$ synapse to parameter equivalence and isotropic distribution of neurons in different brain regions of the cerebral cortex. Nevertheless, the rough comparison can have value in the context of "scaling-up", given that it is limited only in order-of-magnitude estimations, as it can reveal differences and similarities in the learning processes of machines and brains.

[§]For vision such an upper-bound estimate is difficult to obtain, since human visual perception is based on a continuous stream, not on i.i.d. sampled images. Nevertheless, 80 years correspond to 151×10^9 frames at 60 fps.

In this section we provide a summary of the cognitive background that forms the inspiration for the development of the algorithms, architectures, and techniques discussed in the rest of this manuscript.

3.2.1 Cognitive Representations and Conceptual Spaces

The manifold hypothesis[42] is a key concept in machine learning and computational neuroscience, stating that high-dimensional data, encountered in the real-world (e.g. images, speech, neural activity), tend to lie in the vicinity of low-dimensional manifolds. The hypothesis is based on the observation that while the data we encounter in the real world is high dimensional (e.g., every pixel in an image can be considered a separate dimension), it is not entirely random and exhibits structure and dependencies. For example, images of cats differ greatly in pixel space but share underlying visual features such as the shape of the eyes or the structure of the fur. The manifold hypothesis is the base for manifold learning algorithms[43], which aim to extract the underlying low-dimensional structure of high-dimensional input signals. Furthermore, Brahma et al.[44] remark that the success of modern deep learning algorithms lies in the progressive unfolding of the intrinsic low-dimensional manifolds in the input data.

From a cognitive perspective we are motivated by the seminal work of Peter Gärdenfors[45], where a model of conceptual representations is proposed. Specifically, concepts are represented as points that lie along low-dimensional manifolds, e.g. the color hue concept is a point that lies along an 1D circle. The relations between concepts are represented as different similarity metrics between the low-dimensional concept representations. For this, we focus our study on the Multi-Dimensional Scaling (MDS)[46] algorithm, where the relations (distances) between data are directly used to extract low-dimensional representations.

3.2.2 The stability-plasticity dilemma

A key trade-off when training connectionist networks is the stability-plasticity dilemma[47], where plastic connections storing novel patterns, while stability is desirable to keep learned patterns from being erased. McCloskey and Cohen[48] noticed that Artificial Neural Networks (ANN) display a phenomenon known as catastrophic interference, or catastrophic forgetting. This occurs when new patterns alter the weights of previously learned patterns in sequentially trained networks, leading to the forgetting of old patterns. French[49,50] postulated that this is an inherent problem of networks that store information in the form of distributed representations, that is caused due to representational overlap.

More recently, Kirkpatrick et al.[51] proposed Elastic Weight Consolidation (EWC) to overcome catastrophic forgetting, inspired by synaptic consolidation in the brain[52,53], where subsets of synapses are dynamically becoming more stable, enabling long-term memory. EWC takes the form of a regularization term which aims to keep the average change small when learning a new task for all network parameters. Another popular approach for overcoming catastrophic forgetting is information rehearsal[54], where old data are interleaved with novel data in order to reactivate learned patterns and facilitate memory integration. The difference

between EWC and information rehearsal is that in the former, distributional memory consolidation is performed in a direct manner, while in the latter it is performed implicitly. An approach that merges characteristics from both approaches is proposed by Chronopoulou et al.[55], where a pretrained language model is fine-tuned for a supervised task, while performing task replay using a multitask language model loss.

3.3 Contributions and thesis structure

This dissertation is organized in two parts that revolve around the main pillars.

1. **Part I Subspace learning with Multi-Dimensional Scaling:** In this part we focus on low-rank representation learning of conceptual spaces through MDS. We include two studies where we develop a novel MDS algorithm and utilize it in real-world scenarios.

   (a) **Pattern search multi-dimensional scaling:** We propose a novel algorithm for dimensionality reduction, based on multi-dimensional scaling and derivative-free optimization. First we overview the necessary background and algorithmic formulations in Chapter 4, and then we proceed to develop the algorithm, along with a series of optimizations in Chapter 5. Pattern search MDS is shown to have fast and guaranteed convergence.

   (b) **Performance and robustness evaluation:** In Chapter 6, we evaluate the performance of the proposed algorithm in a series of experiments for manifold geometry, image classification, and lexical semantics, and assess its robustness for noisy inputs and missing data.

   (c) **Dimensionality reduction for speech emotion recognition:** In Chapter 7 we explore the effectiveness of pattern search MDS and other dimensionality reduction techniques for Speech Emotion Recognition (SER) with different databases and input features, demonstrating that reduce feature sets achieve competitive performance.

2. **Part II Mixed Self-Supervision for Sample-Efficient Unsupervised Domain Adaptation:** In this part we examine the stability-plasticity dilemma in order to combat catastrophic forgetting during adaptation of pretrained architectures to new domains. We suggest that a mixed self-supervision approach, maintaining the self-supervised pretraining loss during fine-tuning, effectively balances stability and plasticity. Two studies are included, where the proposed approach is employed for the adaptation of transformer-based architectures in diverse settings and different modalities.

   (a) **Unsupervised domain adaptation for text and speech:** In Chapter 8, we formulate the problem of Unsupervised Domain Adaptation (UDA) for classification and sequence to sequence settings. Then we review the relevant prior work in the literature and introduce the proposed fine-tuning strategy.

(b) **Unsupervised domain adaptation through language modeling:** In Chapter 9, we employ a mixed-self supervision approach for performing UDA between domains in a text classification setting, i.e., predicting the sentiment of Amazon reviews for different categories of products. The proposed approach yields successful adaptation with minimal hyperparameter tuning, even when few in-domain data are available. Furthermore, it is demonstrated that the proposed loss function can be a good proxy validation metric in this semi-supervised setting. Finally, we discuss the limitations of domain-adversarial training[56], focused on the difficulty to converge at satisfying solutions and the need for extensive tuning.

(c) **Sample-efficient unsupervised domain adaptation of speech recognition systems:** We propose mixed self-supervision for UDA of the acoustic model in a speech recognition setting for the Greek language in Chapter 10. We create the largest yet transcribed speech corpus for Modern Greek (120 hours) based on parliamentary proceedings. A key finding is that mixed self-supervision is sensitive to the pretraining task; in the case of language modeling, task replay using solely in-domain data is sufficient, while in the case of contrastive pretraining a mix of both out-of-domain and in-domain data is required during task replay to avoid mode-collapse of the internal representations. Finally, we test the effectiveness augmentation techniques for adapting the language model to new domains.

Part I

Subspace Learning with Multi-Dimensional Scaling

"*These creatures you call mice, you see, they are not quite as they appear. They are merely the protrusion into our dimension of vastly hyperintelligent pandimensional beings.*"

Douglas Adams, The Hitchhiker's Guide to the Galaxy

4

Background on dimensionality reduction and derivative-free optimization

4.1 Introduction

In the past decades, we have been witnessing a steady increase in the size of datasets generated and processed by computational systems. Such voluminous data comes from various sources, such as business sales records, the collected results of scientific experiments or real-time sensors used in the internet of things. The most popular way to represent such data is via a set of data points lying in a vector space. The construction of the vector space is often performed using a distance or similarity matrix that can be constructed manually using perceptual ratings or, more commonly, computed automatically using a set of features. In many of these applications high-dimensional data representations are assumed to lie in the vicinity of a low-dimensional, possibly non-linear manifold*, embedded in the high-dimensional space. This is known as the manifold hypothesis [42]. Intuitively human cognition also performs similar mappings when performing everyday tasks, i.e., high-dimensional sensory input get embedded into low dimensional cognitive subspaces [57–59] for rapid and robust decision making, since only a small number of features are salient for each task. Given this assumption *manifold learning* aims to discover such hidden low-dimensional structure and to output a representation with much fewer "intrinsic variables".

In the first part of this dissertation, we study the problem of manifold learning in non-metric topological spaces. The input to this problem is a matrix of (similarities or) dissimilarities† of the dataset objects. "Objects" can be colors, faces, map coordinates, political persuasion scores, or any kind of real-world or synthetic stimuli. For each input dataset object, the output is a low-dimensional vector such that the pairwise Euclidean distances of the output vectors resemble the original dissimilarities. This problem is known as non-metric MDS or Non-Linear Dimensionality Reduction (NLDR) task. An abundance of embedding methods have been developed for dealing with this task as detailed in Section 4.4.

The majority of these algorithms reduce this problem to the optimization of a determin-

*Loosely speaking, a manifold is a topological space that is locally Euclidean.

†It should be mentioned that in many real-world tasks the used dissimilarity measures may correspond in pseudo- or semimetric distance functions that violate the triangular inequality.

istic loss function f. Given this minimization objective, they usually employ gradient-based methods to find a global or a local optimum. In many situations, however, the loss function is non-differentiable or estimating its gradient may be computational expensive. Additionally, gradient-based algorithms usually yield a slow convergence; multiple iterations are needed in order to minimize the loss function.

Inspired by the recent progress in derivative-free optimization tools, we propose an iterative algorithm which treats the non-metric MDS task as a derivative-free optimization problem. Due to the derivative-free formulation, the MDS task can be expanded in spaces where the selected metric results in non-differentiable loss functions. We demonstrate an example of a metric where majorization-based approaches fail to converge to a meaningful solution, while the General Pattern Search (GPS)-based formulation can. The main contributions of this part are as follows: 1) Using the GPS formulation we are able to provide theoretical convergence guarantees for the proposed non-metric MDS algorithm. 2) A set of heuristics are proposed that significantly improve the performance of the proposed algorithm in terms of computational efficiency, convergence rate and solution accuracy. 3) The proposed algorithm is evaluated on a variety of tasks including manifold unfolding, word embeddings and optical digit recognition, and speech emotion recognition, showing consistent performance and good convergence properties. We also compare performance with state-of-the-art MDS algorithms for the aforementioned tasks for clean and noisy datasets. An optimized implementation of pattern search MDS and the experimental code is made available as open source to the research community[‡§].

The first part of this dissertation is based on one preprint publication [60] and one conference publication in Interspeech [62].

4.2 Organization

The remainder of the first part of this dissertation is organized as follows: In this chapter, we begin with an overview of the relevant notation and the related work (Sections 4.3 and 4.4), while we also provide an overview of the MDS problem and the GPS family of algorithms and their convergence characteristics (Sections 4.5, 4.6, and 4.7). In Chapter 5 we present in detail the proposed derivative-free algorithm, coined Pattern Search MDS, the reduction of the algorithm to the GPS formulation and the associated fixed-point convergence guarantees. In Chapter 6 we present the application of Pattern Search MDS to manifold geometry, classification, and lexical similarity problems, and study its robustness with noisy or missing data. Finally, in Chapter 7 we study Dimensionality Reduction (DR) on multiple feature sets in a real-world setting for classifying emotions from speech input.

[‡]Open source code available: https://github.com/georgepar/pattern-search-mds
[§]Chapters 5, 6 consist joint work with Efthymios Tzinis (see [60,61])

4.3 Notation

In this part, we denote real, integer and natural numbers as $\mathbb{R}$, $\mathbb{Z}$, $\mathbb{N}$, respectively. Scalars are represented by no-boldface letters, vectors appear in boldface lowercase letters and matrices are indicated by boldface uppercase letters. All vectors are assumed to be column vectors unless they are explicitly defined as row vectors. For a vector $\mathbf{z} \in \mathbb{R}^n$, $\|\mathbf{z}\|_1 = \sum_{i=1}^{n} |z_i|$ is its ℓ_1 norm and $\|\mathbf{z}\|_2 = \sqrt{\sum_{i=1}^{n} z_i^2}$ is its ℓ_2 norm, where z_i is the ith element of $\mathbf{z}$. By $\mathbf{A} \in \mathbb{R}^{n \times m}$ we denote a real-valued matrix with n rows and m columns. Additionally, the jth column of the matrix $\mathbf{A}$ and its entry at ith row and jth column are referenced as $\mathbf{a}_j$ and a_{ij}, respectively. The trace of the matrix $\mathbf{A}$ appears as $tr(\mathbf{A})$ and its Frobenius norm as $||\mathbf{A}||_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij}^2}$. The square identity matrix with n rows is denoted as $\mathbf{I}_n \in \mathbb{R}^{n \times n}$. For the matrices $\mathbf{A} \in \mathbb{R}^{n \times m}$ and $\mathbf{B} \in \mathbb{R}^{n \times m}$ we indicate their Hadamard product as $\mathbf{A} \odot \mathbf{B}$. The n-ary Cartesian product over n sets $S_1, \ldots, S_n$ is denoted by $\{(s_1, \ldots, s_n) : s_i \in S_i.\ 1 \leq i \leq n\}$ Finally, $\mathbf{X}^{(k)}$ refers to the estimate of a variable $\mathbf{X}$ at the kth iteration of an algorithm.

4.4 Related work

DR algorithms compress data in a low-dimensional space while preserving meaningful statistical and geometrical properties. Such properties are covariance of original data, pairwise distances between samples or local neighborhoods. They can be separated into two general categories, linear and non-linear.

Linear DR aims to find a linear projection $\mathbf{Y} = \mathbf{TX} \in \mathbb{R}^{n \times k}$ of the real data $\mathbf{X} \in \mathbb{R}^{n \times m}$, where $k < m$. Examples of linear DR algorithms are Principal Component Analysis (PCA) and Classical Multi-Dimensional Scaling (cMDS)[63]. PCA projects data into a low-dimensional space, which is formed by an orthogonal basis of linearly uncorrelated vectors called the principal components. Principal components are selected as the axes along which the samples have maximum variance. cMDS takes a geometric approach, finding a set of low-dimensional points that best preserve pairwise euclidean distances between original data points.

In real data applications, such a linearity assumption may be too strong and can lead to suboptimal results. Thus a significant effort has been made by the machine learning community to apply manifold learning in non-linear domains. These algorithms are not limited in linear transformations, like the rotations and stretches that can be induced by a matrix multiplication. Representative manifold learning algorithms include Isometric Feature Mapping (ISOMAP)[64–68], Landmark ISOMAP[69,70], Locally Linear Embedding (LLE)[71–75], Modified LLE[76], Hessian LLE[77], Semi-Definite Embedding (SDE)[78–81], Laplacian Eigenmaps (LE), or Spectral Embedding,[71,75,82], Local Tangent Space Alignment (LTSA)[83], etc. An extension of cMDS is metric MDS[46] where dissimilarity measures are assumed metric, but not necessarily euclidean. When these measures are closely related to the euclidean distance, e.g. cosine distance, metric MDS is still characterized as a linear DR approach. Stress Majorization[46] is an algorithm for metric MDS. The non-metric extension of MDS[84] tries to approximate the rank order of original distances by applying a monotonically increasing function, usu-

ally approximated by isotonic regression. ISOMAP[64–68] finds an isometric mapping of the original data by extending metric MDS to approximate geodesic pairwise distances between original samples space as euclidean pairwise distances in the transformed samples. Geodesic distances are approximated by the shortest path distances between data points. While MDS and ISOMAP consider the global data geometry, LLE[71–75] reconstructs local regions by finding sets of weights which are used to represent samples as a weighted combination of their closest neighbors. Representations are computed by solving a sparse eigenvalue problem. Modified LLE[76] extends LLE, by using multiple neighborhood weights, to produce more robust results. Hessian LLE[77] obtains low-dimensional representations through applying eigenanalysis on a Hessian coefficient matrix. SDE[78–81] attempts to maximize the distance between points that don't belong in a local neighborhood. LE[71,75,82] preserves local manifold geometry by minimizing the Laplacian of the graph formed by neighboring data points. The Laplacian of this graph approximates the Laplacian-Beltrami operator over the manifold, which indicates the divergence of the mapping of a high-dimensional point to the low-dimensionsional manifold. LTSA[83] utilizes local tangent information to represent the manifold geometry and extends this to global coordinates. Also, a common nonlinear method for dimensionality reduction is the kernel extension of PCA[85]. Finally, autoencoders[86] are a class of deep neural networks that can be used for linear and non-linear dimensionality reduction and are composed of an encoder and a decoder. Encoder projects input $\mathbf{x}$ to a low-dimensional space via a hidden layer $\mathbf{h}$, while the attempts to reconstruct $\mathbf{x}$ from $\mathbf{h}$. If no non-linear activations are used, encoder learns a linear projection $\mathbf{W}\mathbf{x} + \mathbf{b}$, whereas if we use a non-linear activation function (e.g., sigmoid or rectified linear unit) in the output of the encoder's layers, a non-linear embedding is learned.

A wide class of derivative-free algorithms for nonlinear optimization has been studied and analyzed in Rios and Sahinidis[87] and Avriel[88]. GPS methods are a subset of the aforementioned algorithms which do not require the explicit computation of the gradient in each iteration-step. Some GPS algorithms are: the original Hooke and Jeeves pattern search algorithm[89], the evolutionary operation by utilizing factorial design[90] and the multi-directional search algorithm[91,92]. In Torczon[93], a unified theoretical formulation of GPS algorithms under a common notation model has been presented as well as an extensive analysis of their global convergence properties. Local convergence properties have been studied later by Dolan et al.[94]. Notably, the theoretical framework as well as the convergence properties of GPS methods have been extended in cases with linear constrains[95], boundary constrains[96] and general Lagrangian formulation[97].

4.5 Multi-dimensional scaling problem formulation

4.5.1 Classical MDS

cMDS was first introduced by Torgerson[63] and can be formalized as follows. Given the matrix $\mathbf{\Delta}$ consisting of pairwise distances or dissimilarities $\{\delta_{ij}\}_{1\leq i,j\leq N}$ between N points in a high dimensional space, the solution to cMDS is given by a set of points $\{\mathbf{x}_i\}_{i=1}^{N}$ which lie on the

manifold $\mathcal{M} \in \mathbb{R}^L$ and their pairwise distances are able to preserve the given dissimilarities $\{\delta_{ij}\}_{1\leq i,j\leq N}$ as faithfully as possible. Each point $\mathbf{x}_i \in \mathbb{R}^L,\ \ 1 \leq i \leq N$ corresponds to a column of the matrix $\mathbf{X}^T \in \mathbb{R}^{L\times N}$. The embedding dimension L is selected as small as possible in order to obtain the maximum dimensionality reduction but also to be able to approximate the given dissimilarities δ_{ij} by the Euclidean distances $d_{ij}(\mathbf{X}) = ||\mathbf{x}_i - \mathbf{x}_j||_2 = \sqrt{\sum_{k=1}^{L}(x_{ik} - x_{jk})^2}$ in the embedded space $\mathbb{R}^L$.

The proposed algorithm uses a centering matrix $\mathbf{H} = \mathbf{I}_N - \frac{1}{N}1_N^T 1_N$ in order to subtract the mean of the columns and the rows for each element. Where $1_N = [1, 1, ..., 1]$ a vector of ones in $\mathbb{R}^N$ space. By applying the double centering to the Hadamard product of the given dissimilarities, the Gram matrix $\mathbf{B}$ is constructed as follows:

$$\mathbf{B} = -\frac{1}{2}\mathbf{H}^T(\mathbf{\Delta} \odot \mathbf{\Delta})\mathbf{H} \tag{4.1}$$

It can be shown (Ch. 12 [98]) that cMDS minimizes the Strain algebraic criterion in Eq. 4.2 below:

$$||\mathbf{X}\mathbf{X}^T - \mathbf{B}||_F^2 \tag{4.2}$$

The eigendecomposition of the symmetric matrix $\mathbf{B}$ gives us $\mathbf{B} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ and thus the new set of points consisting the embedding in $\mathbb{R}^L$ are given by the first L positive eigenvalues of $\mathbf{\Lambda}$, namely $\mathbf{X} = \mathbf{V}_L$. This solution provides the same result as PCA applied on the vector in the high dimensional space [99]. cMDS was originally proposed for dissimilarity matrices $\mathbf{\Delta}$ which can be embedded with good approximation accuracy in a low-dimensional Euclidean space. However, matrices which correspond to embeddings in Euclidean sub-spaces [100], Poincare disks [101] and constant-curvature Riemannian spaces [102] have also been studied.

4.5.2 Metric MDS

Metric MDS describes a superset of optimization problems containing cMDS. Shepard has introduced heuristic methods to enable transformations of the given dissimilarities δ_{ij} [103], [104] but did not provide any loss function in order to model them [105]. Kruskal [46,84] formalized the metric MDS as a least squares optimization problem of minimizing the non-convex Stress-1 function defined in Eq. 4.3 shown next:

$$\sigma_1(\mathbf{X}, \hat{\mathbf{D}}) = \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(\hat{d}_{ij} - d_{ij}(\mathbf{X}))}{\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij}^2(\mathbf{X})}} \tag{4.3}$$

where matrix $\hat{\mathbf{D}}$ with elements $\hat{d}_{ij}$ represents all the pairs of the transformed dissimilarities δ_{ij} that are used to fit the embedded distance pairs $d_{ij}(\mathbf{X})$.

In essence, $\hat{d}_{ij} = \mathcal{F}(\delta_{ij})$ where $\mathcal{F}$ is usually an affine transformation[¶] $\hat{d}_{ij} = \alpha + \beta\delta_{ij}$ for

[¶] Monotone and polynomial regression transformations are employed for nonmetric-MDS, as well as, a wider family of transformations [106].

unknown α and β. Kruskal proposed an iterative gradient-based algorithm for the minimization of σ_1 since the solution cannot be expressed in closed form. Assuming that $d_{ij} \hat{=} \delta_{ij}$ the algorithm iteratively tries to find the coordinates of points $\mathbf{X}$ which are lying in the low embedding space $\mathbb{R}^L$. Trivial solutions ($\mathbf{X} = 0$ and $\hat{\mathbf{D}} = 0$) are avoided by the denominator term in Eq. 4.3.

A weighted MDS raw Stress function is defined as:

$$\sigma^2_{raw}(\mathbf{X}, \hat{\mathbf{D}}) = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij}(\hat{d_{ij}} - d_{ij}(\mathbf{X}))^2 \tag{4.4}$$

where the weights w_{ij} are restricted to be non-negative; for missing data the weights are set equal to zero. By setting $w_{ij} = 1, \forall 1 \leq i, j \leq N$ one can model an equal contribution to the metric MDS solution for all the elements.

4.5.3 SMACOF

Scaling by Majorizing a Complicated Function (SMACOF) is a state-of-the-art algorithm for solving metric MDS and was introduced by Leeuw et al. [107]. By setting $\hat{d_{ij}} = \delta_{ij}$ in raw stress function defined in Eq. 4.4, SMACOF minimizes the resulting stress function $\sigma^2_{raw}(\mathbf{X})$.

$$\sigma^2(\mathbf{X}) = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij}(\delta^2_{ij} - 2\delta_{ij} d_{ij}(\mathbf{X}) + d^2_{ij}(\mathbf{X})) \tag{4.5}$$

The algorithm proceeds iteratively and decreases stress monotonically up to a fixed point by optimizing a convex function which serves as an upper bound for the non-convex stress function in Eq. 4.5. An extensive description of SMACOF can be found in Borg and Groenen [98] while its convergence for a Euclidean embedded space $\mathbb{R}^L$ has been proven by de Leeuw [108].

Let matrices $\mathbf{U}$ and $\mathbf{R}(\mathbf{X})$ be defined element-wise as follows:

$$u_{ij} = \begin{cases} -w_{ij} & i \neq j \\ \sum_{k \neq i} w_{ik} & i = j \end{cases} \tag{4.6}$$

$$r_{ij} = \begin{cases} -w_{ij}\delta_{ij} d^{-1}_{ij}(\mathbf{X}) & i \neq j, d_{ij}(\mathbf{X}) \neq 0 \\ 0 & i \neq j, d_{ij}(\mathbf{X}) = 0 \\ \sum_{k \neq i} r_{ik} & i = j \end{cases} \tag{4.7}$$

The stress function in Eq. 4.5 is converted to the following quadratic form:

$$\sigma^2(\mathbf{X}) = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij}\delta^2_{ij} - 2tr(\mathbf{X}^T \mathbf{R}(\mathbf{X})\mathbf{X}) + tr(\mathbf{X}^T \mathbf{U}\mathbf{X}) \tag{4.8}$$

The quadratic can be minimized iteratively as follows:

$$T(\mathbf{X}, \hat{\mathbf{X}}^{(k)}) = c - 2tr(\mathbf{X}^T\mathbf{R}(\hat{\mathbf{X}}^{(k)})\hat{\mathbf{X}}^{(k)}) + tr(\mathbf{X}^T\mathbf{U}\mathbf{X})$$
$$c = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}\delta_{ij}^2 = const. \tag{4.9}$$

$$\hat{\mathbf{X}}^{(k+1)} = \underset{\mathbf{X}}{\operatorname{argmin}}\ T(\mathbf{X}, \hat{\mathbf{X}}^{(k)}) = \mathbf{U}^{\dagger}\mathbf{R}(\hat{\mathbf{X}}^{(k)})\hat{\mathbf{X}}^{(k)} \tag{4.10}$$

where $\hat{\mathbf{X}}^{(k)}$ is the estimate of matrix $\mathbf{X}$ at the kth iteration and $\mathbf{U}^{\dagger}$ is Moore-Penrose pseudoinverse of $\mathbf{U}$. At iteration k the convex majorizing convex function touches the surface of σ at the point $\hat{\mathbf{X}}^{(k)}$. By minimizing this simple quadratic function in Eq. 4.9 we find the next update which serves as a starting point for the next iteration $k+1$. The solution to the minimization problem is shown in Eq. 4.10. The algorithm stops when the new update yields a decrease $\sigma^2(\hat{\mathbf{X}}^{(k+1)}) - \sigma^2(\hat{\mathbf{X}}^{(k)})$ that is smaller than a threshold value.

4.6 General Pattern Search algorithmic family

The unconstrained problem of minimizing a continuously differentiable function $f\colon \mathbb{R}^n \to \mathbb{R}$ is formally described as

$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathbb{R}^n}{\operatorname{argmin}}\ f(\mathbf{x}) \tag{4.11}$$

Next we present a short description of iterative GPS minimization of Eq. 4.11 based on [93,94]. First we have to define the following components:

- A basis matrix that could be any nonsingular matrix $\mathbf{B} \in \mathbb{R}^{n\times n}$.
- A matrix $\mathbf{C}^{(k)}$ for generating all the possible moves for the kth iteration of the minimization algorithm
$$\mathbf{C}^{(k)} = [\mathbf{M}^{(k)} \ \ -\mathbf{M}^{(k)} \ \ \mathbf{L}^{(k)}] = [\mathbf{\Gamma}^{(k)} \ \ \mathbf{L}^{(k)}] \tag{4.12}$$
where the columns of $\mathbf{M}^{(k)} \in \mathbb{Z}^{n\times n}$ form a positive span of $\mathbb{R}^n$ and $\mathbf{L}^{(k)}$ contains at least the zero column of the search space $\mathbb{R}^n$.
- A pattern matrix $\mathbf{P}^{(k)}$ defined as
$$\mathbf{P}^{(k)} = \mathbf{B}\mathbf{C}^{(k)} = [\mathbf{B}\mathbf{M}^{(k)} \ \ -\mathbf{B}\mathbf{M}^{(k)} \ \ \mathbf{B}\mathbf{L}^{(k)}] \tag{4.13}$$
where the submatrix $\mathbf{B}\mathbf{M}^{(k)}$ forms a basis of $\mathbb{R}^n$.

In each iteration k, we define a set of steps $\{\mathbf{s}_i^{(k)}\}_{i=1}^{m}$ generated by the pattern matrix $\mathbf{P}^{(k)}$ as shown next:

$$\mathbf{s}_i^{(k)} = \Delta^{(k)}\mathbf{p}_i^{(k)}, \ \ \mathbf{P}^{(k)} = [\mathbf{p}_1^{(k)}, \ldots, \mathbf{p}_m^{(k)}] \in \mathbb{R}^{n\times m} \tag{4.14}$$

where $\mathbf{p}_i^{(k)}$ is the ith column of $\mathbf{P}^{(k)}$ and defines the direction of the new step, while $\Delta^{(k)}$ configures the length towards this direction. If the pattern matrix $\mathbf{P}^{(k)}$ contains m columns, then $m \geq n+1$ in order to positively span the search space $\mathbb{R}^n$. Thus, a new trial point of GPS algorithm towards this step would be $\mathbf{x}_i^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{s}_i^{(k)}$ where we evaluate the value of the function f to minimize. The success of a new trial point is decided based on the condition that it takes a step towards further minimizing the function f, i.e., $f(\mathbf{x}^{(k)} + \mathbf{s}_i^{(k)}) > f(\mathbf{x}_i^{(k+1)})$. The steps of a GPS method are presented in Alg. 4.

Algorithm 4 General Pattern Search (GPS)

1: **procedure** GPS_SOLVER($\mathbf{x}^{(0)}$, $\Delta^{(0)}$, $\mathbf{C}^{(0)}$, $\mathbf{B}$)
2: $k = -1$
3: **do**
4: $k = k + 1$
5: $\mathbf{s}^{(k)} = \text{EXPLORE\_MOVES}(\mathbf{BC}^{(k)}, \mathbf{x}^{(k)}, \Delta^{(k)})$
6: $\rho^{(k)} = f(\mathbf{x}^{(k)} + \mathbf{s}^{(k)}) - f(\mathbf{x}^{(k)})$
7: **if** $\rho^{(k)} < 0$ **then**
8: $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{s}^{(k)}$ ▷ Successful iteration
9: **else**
10: $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$ ▷ Unsuccessful iteration
11: $\Delta^{(k+1)}, \mathbf{C}^{(k+1)} = \text{UPDATE}(\mathbf{C}^{(k)}, \Delta^{(k)}, \rho^{(k)})$
12: **while** convergence criterion == False

To initialize the algorithm we select a point $\mathbf{x}^{(0)} \in \mathbb{R}^n$ and a positive step length parameter $\Delta^{(0)} > 0$. In each iteration k, we explore a set of moves defined by the `EXPLORE_MOVES()` subroutine at line 5 of the algorithm. Pattern search methods described using a GPS formalism mainly differ on the heuristics used for the selection of exploratory moves. If a new exploratory point lowers the value of the function f, iteration k is successful and the starting point of the next iteration is updated $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{s}^{(k)}$ as shown in line 8, else there is no update. The step length parameter $\Delta^{(k)}$ is modified by the `UPDATE()` subroutine in line 11. For successful iterations, i.e., $\rho^{(k)} < 0$, the step length is forced to increase in a determistic way as follows:

$$\begin{aligned} &\Delta^{(k+1)} = \lambda^{(k)}\Delta^{(k)}, \quad \lambda^{(k)} \in \Lambda = \{\tau^{w_1}, ..., \tau^{w_{|\Lambda|}}\} \\ &\tau > 1, \quad \{w_1, ..., w_{|\Lambda|}\} \subset \mathbb{N}, \quad |\Lambda| < +\infty \end{aligned} \tag{4.15}$$

where τ and w_i are predefined constants that are used for the ith successive successful iteration. For unsuccessful iterations the step length parameter is decreased, i.e., $\Delta^{(k+1)} \leq \Delta^{(k)}$ as follows:

$$\Delta^{(k+1)} = \theta\Delta^{(k)}, \quad \theta = \tau^{w_0}, \quad \tau > 1, \quad w_0 < 0, \tag{4.16}$$

where τ and the negative integer w_0 determine the fixed ratio of step reduction. Note that the generating matrix $\mathbf{C}^{(k+1)}$ could be also updated for unsuccessful/successful iterations in order

to contain more/less search directions, respectively.

4.7 General Pattern Search convergence

GPS methods under the aforementioned defined framework have some important convergence properties shown in [93–97] and summarized here. For any GPS method which satisfies the specifications of Hyp. 1 on the exploratory moves one may be able to show convergence for Alg. 4.

Hypothesis 1 (Weak Hyp. on Exploratory Moves): *The subroutine* `EXPLORE_MOVES()` *defined in Alg. 4, line 5 guarantees the following:*

- *The exploratory step direction for iteration k is selected from the columns of the pattern matrix $\mathbf{P}^{(k)}$ as defined in Eq. 4.14 and the exploratory step length is $\Delta^{(k)}$ as defined in Eqs. 4.15, 4.16.*
- *If among the exploratory moves $\mathbf{a}^{(k)}$ at iteration k selected from the columns of the matrix*

$$\Delta^{(k)}\mathbf{B}[\mathbf{M}^{(k)} - \mathbf{M}^{(k)}]$$

exist at least one successful move, i.e., $f(\mathbf{x}^{(k)} + \mathbf{a}) < f(\mathbf{x}^{(k)})$, then the `EXPLORE_MOVES()` *subroutine will return a move $\mathbf{s}^{(k)}$ such that $f(\mathbf{x}^{(k)} + \mathbf{s}^{(k)}) < f(\mathbf{x}^{(k)})$.*

Hyp. 1 enforces some mild constraints on the configuration of the exploratory moves produced by Alg. 4, line 5. Essentially, the suggested step $\mathbf{s}^{(k)}$ is derived from the pattern matrix $\mathbf{P}^{(k)}$, while the algorithm needs to provide a simple decrease for the objective function f. Specifically, the only way to accept an unsuccessful iteration would be if none of the steps from the columns of the matrix $\Delta^{(k)}\mathbf{B}[\mathbf{M}^{(k)} - \mathbf{M}^{(k)}]$ lead to a decrease of the objective function f. Based on this hypothesis one can formulate Thm. 1 as follows:

Theorem 1: *Let $L(\mathbf{x}^*) = \{\mathbf{x} : f(\mathbf{x}) \leq f(\mathbf{x}^*)\}$ be closed and bounded and f continuously differentiable on a neighborhood of $L(\mathbf{x}^*)$, namely on the union of the open balls $\bigcup_{\mathbf{a}\in L(\mathbf{x}^*)} B(\mathbf{a}, \eta)$ where $\eta > 0$. If a GPS method is formulated as described in Section 4.6 and Hyp. 1 holds then for the sequence of iterations $\{\mathbf{x}^{(k)}\}$ produced by Alg. 4*

$$\lim_{k\to+\infty} \inf \; ||\nabla f(\mathbf{x}^{(k)})|| = 0$$

Proof: *See [93].*

As shown in [109] one can construct a continuously differentiable objective function and a GPS method with infinite many limit points with non-zero gradients and thus even Thm. 1 holds, the convergence of $||\nabla f(x_k)||$ is not assured. However, the convergence properties of GPS methods can be further strengthened if additional criteria are met. Specifically, a stronger hypothesis on exploratory moves Hyp. 2 regulates the measure of decrease of the objective function for each step produced by the GPS method, as follows:

Hypothesis 2 (Strong Hyp. on Exploratory Moves): *The subroutine* EXPLORE_MOVES() *as defined in Alg. 4, line 5 guarantees the following:*

- *The exploratory step direction for iteration k is selected from the columns of the pattern matrix* $\mathbf{P}^{(k)}$ *as defined in Eq. 4.14 and the exploratory step length is* $\Delta^{(k)}$ *as defined in Eqs. 4.15, 4.16.*
- *If among the exploratory moves* $\mathbf{a}^{(k)}$ *at iteration k selected from the columns of the matrix*

$$\Delta^{(k)}\mathbf{B}[\mathbf{M}^{(k)} - \mathbf{M}^{(k)}]$$

exists at least one successful move, i.e., $f(\mathbf{x}^{(k)} + \mathbf{a}) < f(\mathbf{x}^{(k)})$*, then the* EXPLORE_MOVES() *subroutine will return a move* $\mathbf{s}^{(k)}$ *such that:*
$f(\mathbf{x}^{(k)} + \mathbf{s}^{(k)}) \leq \min_{\mathbf{a}^{(k)}} f(\mathbf{x}^{(k)} + \mathbf{a}^{(k)})$.

Hyp. 2 enforces the additional strong constraint on the configuration of the exploratory moves, namely that the subroutine EXPLORE_MOVES() will do no worse than produce the best exploratory move from the columns of the matrix $\Delta^{(k)}\mathbf{B}[\mathbf{M}^{(k)} - \mathbf{M}^{(k)}]$. Based on this hypothesis and by adding requirements restricting the exploration step direction and length for the GPS method, one can formulate Thm. 2 which is also presented here without proof.

Theorem 2: *Let* $L(\mathbf{x}^*) = \{\mathbf{x} : f(\mathbf{x}) \leq f(\mathbf{x}^*)\}$ *be closed and bounded and f continuously differentiable on a neighborhood of* $L(\mathbf{x}^*)$*, namely on the union of the open balls* $\bigcup_{\mathbf{a}\in L(\mathbf{x}^*)} B(\mathbf{a}, \eta)$ *where* $\eta > 0$*. If a GPS method is formulated as described in Section 4.6,* $\lim_{k\to+\infty} \Delta^{(k)} = 0$*, the columns of the generating matrices* $\mathbf{C}^{(k)}$ *are bounded by norm and Hyp. 2 holds then for the sequence of iterations* $\{\mathbf{x}^{(k)}\}$ *produced by Alg. 4*

$$\lim_{k\to+\infty} ||\nabla f(\mathbf{x}^{(k)})|| = 0$$

Proof: *See*[93].

The additional requirements specify that: 1) the generating matrix $\mathbf{C}^{(k)}$ should have bounded norm in order to produce trial steps from Eq. 4.14 that are bounded by the step length parameter $\Delta^{(k)}$ and 2) $\lim_{k\to+\infty} \Delta^{(k)} = 0$ that can be easily met by selecting $\Lambda = \{1\}$ in Eq. 4.16; this also guarantees a non increasing sequence of $\Delta^{(k)}$ steps[93]. Although these criteria provide much stronger convergence properties, we are faced with a trade off between the theoretical proof of convergence and the efficiency of heuristics in finding a local optimum.

Both theorems 1 and 2 provide a first order optimality condition if their specifications hold. Although the latter theorem premises much stronger convergence results, step-length control parameter $\Delta^{(k)}$, provides a reliable asymptotic measure of first-order stationarity when it is reduced after unsuccessful iterations[94].

5

Pattern Search Multi-dimensional Scaling

5.1 Introduction

In this section we describe Pattern Search MDS, a derivative-free formulation of the MDS algorithm. The input to the algorithm is a distance matrix $\mathbf{T}$, obtained from a set of points in a high-dimensional space with dimension M. The goal is to find a set of points in a low-dimensional space, with target dimension $L < M$ with the same distance matrix. Pattern Search MDS achieves this using a direct search approach, by exploring pertubations on the surface of a hyper-sphere around each point.

After we present the core algorithm, we explore a set of optimizations and perform a complexity analysis of the proposed approach. We further reduce Pattern Search MDS belongs to the General Pattern Search family of derivative-free exploration methods; thus the proposed algorithm inherits the guaranteed convergence properties of GPS. Finally, we show how to automatically tune the hyperparameters for the proposed algorithm and explore the benefits of derivative-free optimization in solving non-differentiable loss functions.

5.2 Core algorithm

The key idea behind the proposed algorithm is to treat MDS as a derivative-free problem, using a variant of general pattern search optimization to minimize a loss function. The input to pattern search MDS is a $N \times N$ target dissimilarity matrix $\mathbf{T}$ and the target dimension L of the embedding space. An overview of the algorithm shown in Alg. 5 is presented next.

The initialization process of the algorithm consists of: 1) random sampling of N points in the embedded space and construction of the matrix $\mathbf{X}^{(0)} = [\mathbf{x}_1^{(0)}, \mathbf{x}_2^{(0)}, ..., \mathbf{x}_N^{(0)}] \in \mathbb{R}^{N \times L}$, 2) computing the embedded space dissimilarity matrix $\mathbf{D}^{(0)}$, where the element $d_{ij}^{(0)}$ is the Euclidean distance between vectors $\mathbf{x}_i^{(0)}$ and $\mathbf{x}_j^{(0)}$ of $\mathbf{X}^{(0)}$, and 3) computing the initial approximation error $e^{(0)} = f(\mathbf{T}, \mathbf{D}^{(0)})$, where e is the element-wise Mean Squared Error (MSE) between the two matrices. The functional f that we attempt to minimize is the normalized square of the Frobenius norm of the matrix $\mathbf{T} - \mathbf{D}$, i.e., $f(\mathbf{T}, \mathbf{D}) = (1/N^2)||\mathbf{T} - \mathbf{D}||_F^2$. Equivalently one may

express f element-wise as follows:

$$f(\mathbf{T}, \mathbf{D}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (t_{ij} - d_{ij})^2, \quad \text{where } \mathbf{T}, \mathbf{D} \in \mathbb{R}^{N \times N} \tag{5.1}$$

Algorithm 5 Pattern Search MDS

procedure MDS($\mathbf{T}$, L, $r^{(0)}$)
  $k \leftarrow 0$ ▷ k is the number of epochs
  $\mathbf{X}^{(k)} \leftarrow$ UNIFORM($N \times L$)
  $\mathbf{D}^{(k)} \leftarrow$ DISTANCE_MATRIX($\mathbf{X}^{(k)}$)
  $e^{(k)} \leftarrow f(\mathbf{T}, \mathbf{D}^{(k)})$
  $e^{(k-1)} \leftarrow +\infty$
  $r^{(k)} \leftarrow r^{(0)}$
  while $r^{(k)} > \delta$ **do**
    if $e^{(k-1)} - e^{(k)} \leq \varepsilon \cdot e^{(k)}$ **then**
      $r^{(k)} \leftarrow \frac{r^{(k)}}{2}$
    $\mathbf{S} \leftarrow$ SEARCH_DIRECTIONS($r^{(k)}$, L)
    for all $x \in \mathbf{X}^{(k)}$ **do**
      $\mathbf{X}^*, e^* \leftarrow$ OPTIMAL_MOVE($\mathbf{X}^{(k)}$,x,$\mathbf{S}$,$e^{(k)}$)
      $e^{(k-1)} \leftarrow e^{(k)}$
      $e^{(k)} \leftarrow e^*$
      $\mathbf{X}^{(k)} \leftarrow \mathbf{X}^*$
    $k = k + 1$

Following the initialization steps, in each epoch (iteration), we consider the surface of a hypersphere of radius r around each point $\mathbf{x}_i^{(k)}$. The possible search directions lie on the surface of a hypersphere along the orthogonal basis of the space, e.g., in the case of 3-dimensional space along the directions $\pm x, \pm y, \pm z$ on the sphere shown in Fig. 5.1. This creates the search directions matrix S and is summarized in Alg. 6

Algorithm 6 Define search directions

function SEARCH_DIRECTIONS(r, L)
  $\mathbf{S}^+ \leftarrow r \cdot \mathbf{I}_L$
  $\mathbf{S}^- \leftarrow -r \cdot \mathbf{I}_L$
  $\mathbf{S} \leftarrow [\frac{\mathbf{S}^+}{\mathbf{S}^-}]$
  return S

Each point is moved greedily along the dimension that produces the minimum error. At this stage we only consider moves that yield a monotonic decrease in the error function. Alg. 7

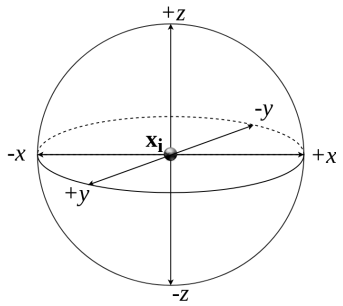


Figure 5.1: Sphere of radius r around point $\mathbf{x_i^{(k)}}$ and possible search directions

finds the optimal move that minimizes $e^{(k)} = f(\mathbf{T}, \mathbf{D}^{(k)})$ for each new point $\tilde{x}$ and moves $\mathbf{X}$ in that direction. Note that when writing $s \in \mathbf{S}$, the matrix $\mathbf{S}$ is considered to be a set of row vectors.

Algorithm 7 Find optimal move for a point

```
1:  function OPTIMAL_MOVE(X^(k), x, S, e)
2:      e* ← e
3:      for all s ∈ S do
4:          x̃ ← x + s
5:          X ← UPDATE_POINT(X^(k), x, x̃)        ▷ Update x point of X^(k) with x̃
6:          D ← DISTANCE_MATRIX(X)
7:          ẽ ← f(T, D)
8:          if ẽ < e* then
9:              e* ← ẽ
10:             X* ← X
11:     return X*, e*
```

The resulting error e^* is computed after performing the optimal move for each point in $\mathbf{X}^{(k)}$. If the error decrease hits a plateau, we halve the search radius and proceed to the next epoch. This is expressed as $e^{(k)} - e^* < \varepsilon \cdot e^{(k)}$, where ε is a small positive constant, namely the error decrease becomes very small in relation to $e^{(k)}$. The process stops when the search radius r becomes very small, namely $r < \delta$, where δ is a small constant, as shown in Alg. 5.

5.3 Optimizations and algorithm complexity

Next, a set of algorithmic optimizations are presented that can improve the execution time and the solution quality of Alg. 5. We also present ways to improve the execution time by searching for an approximate solution, as well as, discuss ways to utilize parallel computation for parts of the algorithm.

5.3.1 Allow for "bad" moves

In Section 5.2 we restrict the accepted moves so that the error decreases monotonically. This is a reasonable restriction that also provides us with theoretical guarantees of convergence. Nonetheless in our experimental setting, we observed that if we relax this restriction and allow each point to always make the optimal move, regardless if the error (temporarily) increases the algorithm converges faster to better solutions. The idea of allowing greedy algorithms to make some "bad" moves in hope to get over local minima can be found in other optimization algorithms, simulated annealing[110] being the most popular. To implement this one can modify line 13 in Alg. 5 to:

Algorithm 8 Allow for bad moves

$\mathbf{X}^*, e^* \leftarrow \text{OPTIMAL\_MOVE}(\mathbf{X}^{(k)}, x, S, +\infty)$

5.3.2 Online computation of dissimilarity matrix

In line 6 of Alg. 7 we observe that we recompute the dissimilarity matrix for each move. This can be avoided because each move modifies only one point $\mathbf{x}_i^{(k)}$, therefore only the row $\mathbf{d}_{i,:}^{(k)}$ and column $\mathbf{d}_{:,i}^{(k)}$ of the dissimilarity matrix $\mathbf{D}^{(k)}$ are affected. Furthermore only one dimension l of the vector $\mathbf{x}_i^{(k)}$ is modified by the move, i.e., only element $x_{i,l}^{(k)}$ of matrix $\mathbf{X}^{(k)}$. In detail, the element $d_{i,j}$ that stores the dissimilarity between points $\mathbf{x}_i$ and $\mathbf{x}_j$ should be updated as follows for the move from $x_{i,l}^{(k)}$ to $x_{i,l}^{(k+1)}$ for $i \neq j$:

$$d_{i,j}^{(k+1)} = \sqrt{\left(d_{i,j}^{(k)}\right)^2 - \left(x_{i,l}^{(k)} - x_{j,l}^{(k)}\right)^2 + \left(x_{i,l}^{(k+1)} - x_{j,l}^{(k+1)}\right)^2} \tag{5.2}$$

5.3.3 Step and move selection

It follows from the need to search for the optimal move across the embedding dimensions L, that the complexity of the algorithm has a linear dependency on L. A large value of L might affect the execution time of the algorithm. An approximate technique to alleviate this is perform a random sampling over all possible directions in the L dimensional space in order to select a "good" direction instead of the optimal, thus restricting the search space*.

An important parameter for our algorithm is the starting radius $r^{(0)}$. This parameter controls how broad the search will be initially and has an effect similar to the learning rate of gradient-based optimization algorithms. If we are too conservative and choose a small initial

*One can potentially do better than random sampling of all possible directions in the L dimensional space. As the geometry of the embedding space starts becoming apparent, after a few epochs of the algorithm, it makes sense to increasingly bias the search towards the principal component vectors of the neighborhood of the point that is being moved.

radius, the algorithm will converge slowly to a local optimum, whereas if we set it too high, the error will overshoot and convergence is not guaranteed. A simple technique to automatically find a good starting radius is to use binary search. In particular, we set the starting radius to an arbitrary value, perform a dry run of the algorithm for one epoch and observe the effect on error. If the error increases we halve the radius. Otherwise we double it and repeat the process. This process is allowed to run for a small number of epochs. The starting radius found using this technique is a not too pessimistic or too optimistic estimate of the best parameter value.

5.3.4 Parallelization

Another way to boost the execution time is to utilize parallel computation to speed up parts of the algorithm. In our case we can parallelize the search for the optimal moves across the embedding dimensions using the map-reduce parallelization pattern. Specifically, we can map the search for candidate moves to run in different threads and store the error for each candidate move in an array $\mathbf{e} = [e_1, e_2.., e_{2L}]$. After the search completes we can perform a reduction operation (min) to find the optimal move and the optimal error $\mathbf{X}^*$, e^*. For our implementation we used the OpenMP parallelization framework[111] and it led to a $2-4$ times speedup in execution time.

5.3.5 Complexity

For each epoch we search across $2L$ dimensions for N points. In each search we also need $\mathcal{O}(N)$ operations to update the distance matrix. Thus, the per epoch computational complexity of the algorithm is $\mathcal{O}(N^2L)$. The optimizations proposed above do not change the complexity of the algorithm per epoch with the notable exception of the move selection optimization: if instead of $2L$ moves per epoch one would consider only $2K$ moves. In this case, the overall complexity per epoch would be $\mathcal{O}(N^2K)$ instead of $\mathcal{O}(N^2L)$. However, as we shall see in the experiments that follow the (rest of the) proposed optimization significantly improve convergence speed, resulting in fewer epochs and less computation complexity overall.

5.4 Reduction to GPS family of algorithms

Pattern Search MDS belongs to the general class of GPS methods and can be expressed using the unified GPS formulation introduced in Section 4.6. Next, we express our proposed algorithm and associated objective function under this formalism.

First, we restate the problem of MDS in a vectorized form. We use matrix $\mathbf{\Delta}$ with elements $\{\delta_{ij}\}_{1\leq i,j\leq N}$ that expresses the dissimilarities between N points in the high dimensional space. The set of points $\{\mathbf{x}_i\}_{i=1}^N$ lie on the low dimensional manifold $\mathcal{M} \in \mathbb{R}^L$ and form the column set of matrix $\mathbf{X}^T$. The matrix $\mathbf{X} \in \mathbb{R}^{N\times L}$ will be now vectorized as an one column vector as shown next:

$$\begin{aligned} \mathbf{x}_i &= [x_{i1}, ..., x_{iL}]^T \in \mathbb{R}^L, 1 \leq i \leq N \\ \mathbf{z} &= vec(\mathbf{X}^T) = [x_{11}, ..., x_{1L}, ..., x_{N1}, ..., x_{NL}]^T \end{aligned} \tag{5.3}$$

Now our new variable $\mathbf{z}$ lies in the search space $\mathbb{R}^{N \cdot L}$. The distance between any two points $\mathbf{x}_i$ and $\mathbf{x}_j$ of the manifold $\mathcal{M}$ remains the same but is now expressed as a function of the vectorized variable $\mathbf{z}$. Namely, $d_{ij}(\mathbf{X}) = ||\mathbf{x}_i - \mathbf{x}_j|| = \sqrt{\sum_{k=1}^{L}(x_{ik} - x_{jk})^2} = d_{ij}(\mathbf{z})$. To this end, our new objective function to minimize g is the MSE between the given dissimilarities δ_{ij} and the euclidean distances d_{ij} in the low dimensional manifold $\mathcal{M}$ as defined in Eq. 5.4 shown next:

$$g(\mathbf{z}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (d_{ij}(\mathbf{z}) - \delta_{ij})^2, \;\; \mathbf{z} \in \mathbb{R}^{N \cdot L} \tag{5.4}$$

Consequently, the initial MDS is now expressed as an unconstrained non-convex optimization problem which is expressed by minimizing the function g over the search space of $\mathbb{R}^{N \cdot L}$ (Eq. 5.5). Specifically, the L coordinates for all N points on the manifold $\mathcal{M}$ now serve as degrees of freedom for our solution.

$$\mathbf{z}^* = \min_{\mathbf{z} \in \mathbb{R}^{N \cdot L}} g(\mathbf{z}) \tag{5.5}$$

Now that we have formulated the problem and the variable $\mathbf{z}$ in the appropriate format we can match each epoch of our initial algorithm with an iteration of a GPS method. Therefore, the moves produced by our algorithm form a sequence of points $\{\mathbf{z}^{(k)}\}$. Moreover, we are going to define the matrices $\mathbf{B}, \mathbf{C}^{(k)}, \mathbf{P}^{(k)}$ for our algorithm as in Eqs. 4.12, 4.13. The choice of our basis matrix $\mathbf{B}$ is the identity matrix as shown in Eq. 5.7.

$$\mathbf{e}_i = [0, .., \underbrace{1}_{index \;\; i}, ..., 0]^T, 1 \leq i \leq N \cdot L \tag{5.6}$$

$$\mathbf{B} = \mathbf{I}_{N \cdot L} = [\mathbf{e}_1, ..., \mathbf{e}_{N \cdot L}] \tag{5.7}$$

While the identity matrix is non singular and its columns span positively the search space $\mathbb{R}^{N \cdot L}$, we also define $\mathbf{M}^{(k)}$ as the identity matrix. In Eq. 5.8 matrix $\mathbf{\Gamma}^{(k)}$ represents the movement alongside the unit coordinate vectors of $\mathbb{R}^{N \cdot L}$. Nevertheless, our generating matrix $\hat{\mathbf{C}}$ also comprises of all the remaining possible directions which are generated by the set $\{-1, 0, 1\}$. In total, we have $3^{N \cdot L} - 2 \cdot N \cdot L$ extra direction vectors inside the corresponding matrix $\mathbf{L}^{(k)}$ as it is shown in Eq. 5.9.

$$\begin{aligned} \mathbf{M}^{(k)} &= \hat{\mathbf{M}} = \mathbf{I}_{N \cdot L} \in \mathbb{Z}^{N \cdot L \times N \cdot L} \\ \mathbf{\Gamma}^{(k)} &= \hat{\mathbf{\Gamma}} = [\hat{\mathbf{M}} \;\; -\hat{\mathbf{M}}] \end{aligned} \tag{5.8}$$

$$
\begin{aligned}
\hat{S} &= \{-1, 0, 1\} \\
\mathbf{L}^{(k)} &= \hat{\mathbf{L}} \\
\hat{\mathbf{L}} &= \{\hat{v} : \hat{v} \in \underbrace{\hat{S} \times \ldots \times \hat{S}}_{N \cdot L} \wedge \hat{v} \notin \{\mathbf{e}_1, \ldots, \mathbf{e}_{N \cdot L}\}\}
\end{aligned}
\tag{5.9}
$$

According to Eqs. 5.8, 5.9, we construct the full pattern matrix $\mathbf{P}^{(k)}$ in Eq. 5.10 in a similar way to Eq. 4.13. For our algorithm the pattern matrix is is equal to our generating matrix $\mathbf{C}^{(k)} = \hat{\mathbf{C}}$ which is also fixed for all iterations. Conceptually, the generating matrix $\hat{\mathbf{C}}$ contains all the possible exploratory moves while a heuristic is utilized for evaluating the objective function g only for a subset of them.

$$
\begin{aligned}
\mathbf{C}^{(k)} = \hat{\mathbf{C}} = [\hat{\mathbf{\Gamma}} \;\; \hat{\mathbf{L}}] = [\hat{\mathbf{M}} \;\; -\hat{\mathbf{M}} \;\; \hat{\mathbf{L}}] \\
\mathbf{P}^{(k)} = \hat{\mathbf{P}} \equiv \mathbf{B}\hat{\mathbf{C}} \equiv \hat{\mathbf{C}}
\end{aligned}
\tag{5.10}
$$

Finally, we configure the updates of the step length parameter for each class of both successful and unsuccessful iterations as they were previously described in Eqs. 4.15, 4.16, respectively. Recalling the notation of Section 4.6, $\hat{\mathbf{s}}^{(k)}$ is the step which is returned from our exploratory moves subroutine at kth iteration. For the successful iterates $g(\mathbf{z}^{(k)} + \hat{\mathbf{s}}^{(k)}) < g(\mathbf{z}^{(k)})$ we do not further increase the length of our moves by limiting $\Lambda = \{1\}$ as follows:

$$
\Delta^{(k+1)} = \Delta^{(k)}, \quad \text{if} \;\; f(\mathbf{z}^{(k)} + \hat{\mathbf{s}}^{(k)}) < f(\mathbf{z}^{(k)}) \tag{5.11}
$$

Similarly, for the unsuccessful iterations $g(\mathbf{z}^{(k)} + \hat{\mathbf{s}}^{(k)}) \geq g(\mathbf{z}^{(k)})$ we halve the distance by a factor of 2 by setting $\theta = \frac{1}{2}$ as it is shown next:

$$
\Delta^{(k+1)} = \frac{1}{2}\Delta^{(k)}, \quad \text{if} \;\; f(\mathbf{z}^{(k)} + \hat{\mathbf{s}}^{(k)}) >= f(\mathbf{z}^{(k)}) \tag{5.12}
$$

A short description of our algorithm as a GPS method for solving the problem stated in Eq. 5.5 follows: In each iteration, we fix the optimal coordinate direction for each one of the points lying on the low dimensional manifold $\mathbf{x}_i \in \mathcal{M}, \;\; 1 \leq i \leq N$. For each internal iteration of Alg. 7, if the optimal direction produces a lower value for our objective function g we accumulate this direction and move alongside this coordinate of the $\mathbb{R}^{N \cdot L}$. Otherwise, we remain at the same position. As a result, the exploration of coordinates for the new point $\mathbf{x}_{i+1}$ begins from this temporary position. This greedy approach provides a potential one-hot vector as described in Eq. 5.6 if the iterate is successful or otherwise, the zero vector $0 \in \mathbb{R}^{N \cdot L}$. The final direction vector $\hat{\mathbf{s}}^{(k)}$ for kth iteration is computed by summing these one-hot or zero vectors. At the kth iteration, the movement would be given by a scalar multiplication of the step length parameter $\Delta^{(k)}$ with the final direction vector in a similar way as defined in Eq. 4.14. This provides a simple decrease for the objective function g or in the worst case represent a zero movement in the search space $\mathbb{R}_{N \cdot L}$. Regarding the movement across $\hat{\mathbf{s}}^{(k)}$, it is trivial to show that this reduction of the objective function g is an associative operation. In other words, accumulating all best coordinate steps for each point $\{\mathbf{x}_i\}_{i=1}^{N}$ and performing the movement

at the end of the kth iteration (as GPS method formulation requires) produces the same result as taking each coordinate step individually. Finally, pattern search MDS terminates when the step length parameter $\Delta^{(k)}$ becomes smaller than a predefined threshold.

5.5 Convergence of Pattern Search MDS

Now that we have homogenized the notation framework as well as have expressed the proposed algorithm as a GPS method one can utilize the theorems stated in Section 4.7 to prove the convergence properties of the proposed algorithm.

First of all, the objective function g is indeed continuously differentiable for all the values of the search space $\mathbb{R}^{N \cdot L}$ by its definition in Eq. 5.4. Moreover, the pattern matrix $\hat{\mathbf{P}}$ in Eq. 5.10 contains all the possible step vectors provided by our exploratory moves routine. Thus, all of our exploratory moves are defined by Eq. 4.14. In each iteration we evaluate the trial steps alongside all coordinates for all the points $\mathbf{x}_i \in \mathcal{M},\ 1 \leq i \leq N$. In our restated problem definition (see Section 5.4), this is translated to searching all over the identity matrices $\mathbf{I}_{N \cdot L}$ and $-\mathbf{I}_{N \cdot L}$ of the search space $\mathbb{R}^{N \cdot L}$. But from our definition of the first columns of our generating matrix in Eq. 5.8 this corresponds to checking all the potential coordinate steps provided by $\hat{\mathbf{\Gamma}} = [\mathbf{I}_{N \cdot L} \ \ -\mathbf{I}_{N \cdot L}]$. Consequently, if there exists a simple decrease when moving towards any of the directions provided by the columns of $\hat{\mathbf{\Gamma}}$ then our algorithm also provides a simple decrease. This result verifies that Hyp. 1 is true for the exploratory moves. By combining the differentiability of our objective function g and Hyp. 1, Thm. 1 holds for pattern search MDS. Hence, $\lim\limits_{k \to +\infty} \inf\ ||\nabla f(\mathbf{z}^{(k)})|| = 0$ is guaranteed.

Trying to further strengthen the convergence properties of the proposed algorithm, we note that most of the requirements of Thm. 2 are met, but we fail to meet the specifications of Hyp. 2 for the minimum decrease provided by the the columns of $\hat{\mathbf{\Gamma}}$. However, our generating matrix $\hat{\mathbf{C}} = [\hat{\mathbf{c}}_1, \ldots, \hat{\mathbf{c}}_{3^{N \cdot L}}]$ is indeed bounded by norm because $||\hat{\mathbf{c}}_j||_1 \leq N \cdot L,\ 1 \leq j \leq 3^{N \cdot L}$. By halving the step length parameter for the unsuccessful iterations we also ensure that $\lim_{k \to \infty} \Delta^{(k)}$. In order to meet the specifications of Thm. 2 we would need a quadratic complexity of $\mathcal{O}((N \cdot L)^2)$ in order to ensure that each iteration provides the same decrease in function g as the decrease provided by the "best" column of $\hat{\mathbf{\Gamma}}$. This is formally stated at the second part of Hyp. 2. If we modify our algorithm in order to meet these requirements we would not be able to implement all the optimizations proposed in Section 5.3 and the overall runtime would be dramatically increased.

5.6 Tuning the hyperparameters

Next we present some guidelines on how to set the hyperparameters for the proposed algorithm and report the values used in the experiments that follow. Specifically:

- The constant ε in line 9 of Alg. 5 determines when the move radius r is decreased. By setting ε to a value very close to 0, e.g., 10^{-10}, the search will take more epochs but the solution will be closer to the local optimum. If we relax ε to a value around 10^{-2}, we can

do a coarse exploration of the search space that will produce a rough solution in a small number of epochs. In our experiments we set $\varepsilon = 10^{-4}$ that provides a good trade-off between solution quality and fast convergence for the datasets used.

- We experimentally found that if L is large, we may only search 50% of the search dimensions and still get a good solution, while significantly reducing the execution time. For this to hold, it is important that we randomly sample a new search space for each epoch.

- The proposed algorithm is relatively robust to the choice of the initial size of the move search radius. However, the choice of $r^{(0)}$ does affect convergence speed. We show the convergence for an example run of the classical swissroll (see Section 6.1) for best-case ($r^{(0)} = 32$), pessimistic ($r^{(0)} = 1$) and optimistic ($r^{(0)} = 65536$) starting radii in Fig. 5.2.

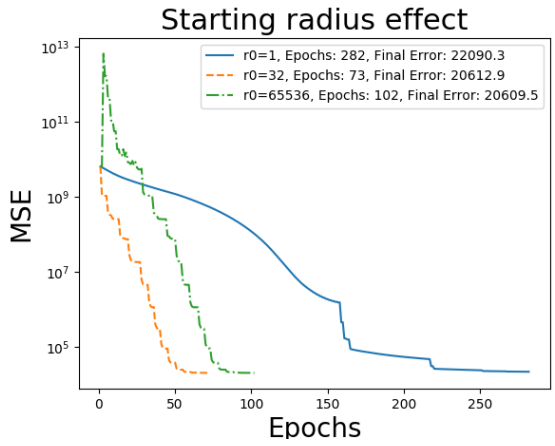


Figure 5.2: Convergence plot of pattern search MDS for different starting radii. Finding the optimal radius (in this case $r = 32$) leads to faster convergence.

5.7 Solving non-differentiable loss functions

Consider the distance function in Eq. 5.13.

$$\delta(x, y) = \begin{cases} \|x - y\|_2 & \text{, if } \|x\|_2 \geq \|y\|_2 \\ \infty & \text{, otherwise} \end{cases} \tag{5.13}$$

This quasimetric results in a non-symmetric distance matrix where many entries are equal to ∞. A physical interpretation can be given to this quasimetric if we imagine x and y as energy states in a physical system, where we can transition from high to low energy states but not vice-versa.

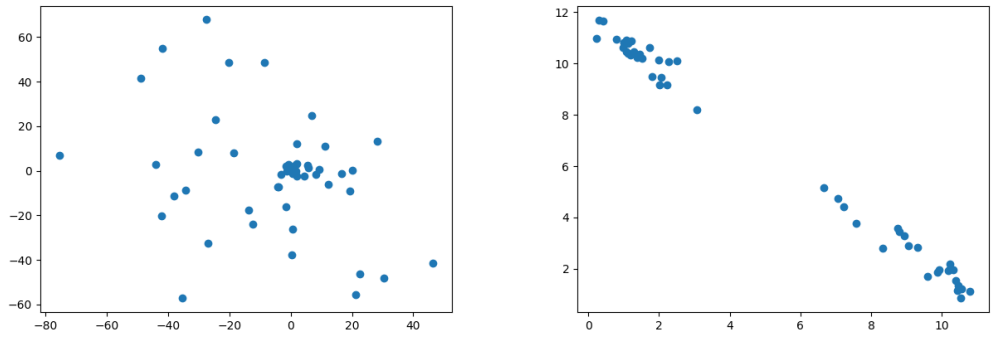
For this experiment we selected to perform MDS for an intuitive geometrical example using the quasimetric of Eq. 5.13 as the target distance function. Specifically consider a set of Npoints in $\mathbb{R}^d$ lying in a straight line. The original high-dimensional points can be summarized in the matrix in Eq. 5.14. For the sake of readability we show the case where the points are sampled as equidistant points on the line that crosses all axes at 45 degrees, but the results of this section can be generalized in the case where the points are randomly sampled on any line.

$$\mathbf{X} = \begin{pmatrix} 0 & 0 & \dots & 0 \\ 1 & 1 & \dots & 1 \\ \dots & \dots & \dots & \dots \\ N-1 & N-1 & \dots & N-1 \end{pmatrix} \quad , \mathbf{X} \in \mathbb{R}^{N \times d} \tag{5.14}$$

The pairwise distance matrix of $\mathbf{X}$ with respect to 5.13 has the pairwise euclidean distances in the upper triangle, zeros in the diagonal and ∞ in the lower triagle, as shown in Eq. 5.15

$$\mathbf{\Delta} = \begin{pmatrix} 0 & \sqrt{2} & 2 \cdot \sqrt{2} & \dots & (N-1) \cdot \sqrt{2} \\ \infty & 0 & \sqrt{2} & \dots & (N-2) \cdot \sqrt{2} \\ \dots & \dots & \dots & \dots & \dots \\ \infty & \infty & \dots & 0 & \sqrt{2} \\ \infty & \infty & \dots & \infty & 0 \end{pmatrix} \quad , \mathbf{\Delta} \in \mathbb{R}^{N \times N} \tag{5.15}$$

When reducing the dimensionality of $\mathbf{X}$ to a lower dimensional space, e.g. in 2 dimensions we would like the straight line geometry to be preserved. Due to the nature of the distance matrix, the loss function contains many saddle points, specifically due to the non-informative nature of the lower triangle. We apply both SMACOF and pattern search MDS and the results are summarized in Fig. 5.3. Specifically in Fig. 5.3a we see that SMACOF fails to converge to a meaningful solution. Compare this to Fig. 5.3b, where Pattern Search MDS puts the points along a straight line, bissecting the 2 axes. The equidistant relationship between the points is lost, because the ∞ terms dominate the loss function, but the general geometry of the data points is preserved.



(a) SMACOF

(b) Pattern Search MDS

Figure 5.3: DR in 2 dimensions using MDS for the distance matrix in Eq. 5.15

6

Manifold Geometry, Classification, and Lexical Similarity Experiments

6.1 Manifold Geometry

The key assumption in manifold learning is that input data lie on a low-dimensional, non-linear manifold, embedded in a high-dimensional space. Thus non-linear DR techniques aim to extract the low-dimensional manifold from the high dimensional space. To showcase this we generated a variety of geometric manifold shapes and compared the proposed MDS to other, well-established DR techniques. We make the code to generate the synthetic data openly available to the community*.

One should note that MDS algorithms with Euclidean distance matrices as inputs cannot infer data geometry, thus we need to provide as input a *geodesic distance matrix*. This matrix is computed by running Djikstra's shortest path algorithm on the nearest neighbors graph trained on the input data. For our experiments we sample 3000 points on 11 $3D$ shapes and reduce them to 2 dimensions using pattern search MDS, SMACOF [107], truncated Singular Value Decomposition (SVD) [112], ISOMAP [64–68], LLE [71–75], Hessian LLE [77], modified LLE [76] and LTSA [83].

Fig. 6.1 shows experiments for NLDR of three manifold shapes. The geodesic distance matrices provided to pattern search MDS and SMACOF is computed using Djikstra's algorithm k Nearest Neighbor (k-NN) graphs. We list the times it took each method to run. Note that pattern search MDS is faster than SMACOF.

We present 3 characteristic shapes selected from the ones we tested. The first shape we examine is the classical swissroll, where a $2D$ plane is "rolled" in $3D$ space and the target is to extract the original $2D$ plane. Results are presented in Fig. 6.1a. We observe that linear DR techniques like truncated SVD have trouble unrolling the swissroll. Also LLE introduces a lot of distortion to the constructed plane.

Next we examine how the algorithms handle sparse distance matrices. To this end, we generate a dataset of $3D$ non-overlapping clusters with a line connecting the centroids, where sparsity of the distance matrix follows because the vast majority of the points are very closely sampled inside the clusters. A good mapping should preserve the cluster structure in lower dimensions. In Fig. 6.1b we see that the truncated SVD and the MDS family of algorithms (pro-

*Open source code available: https://github.com/georgepar/gentlemandata

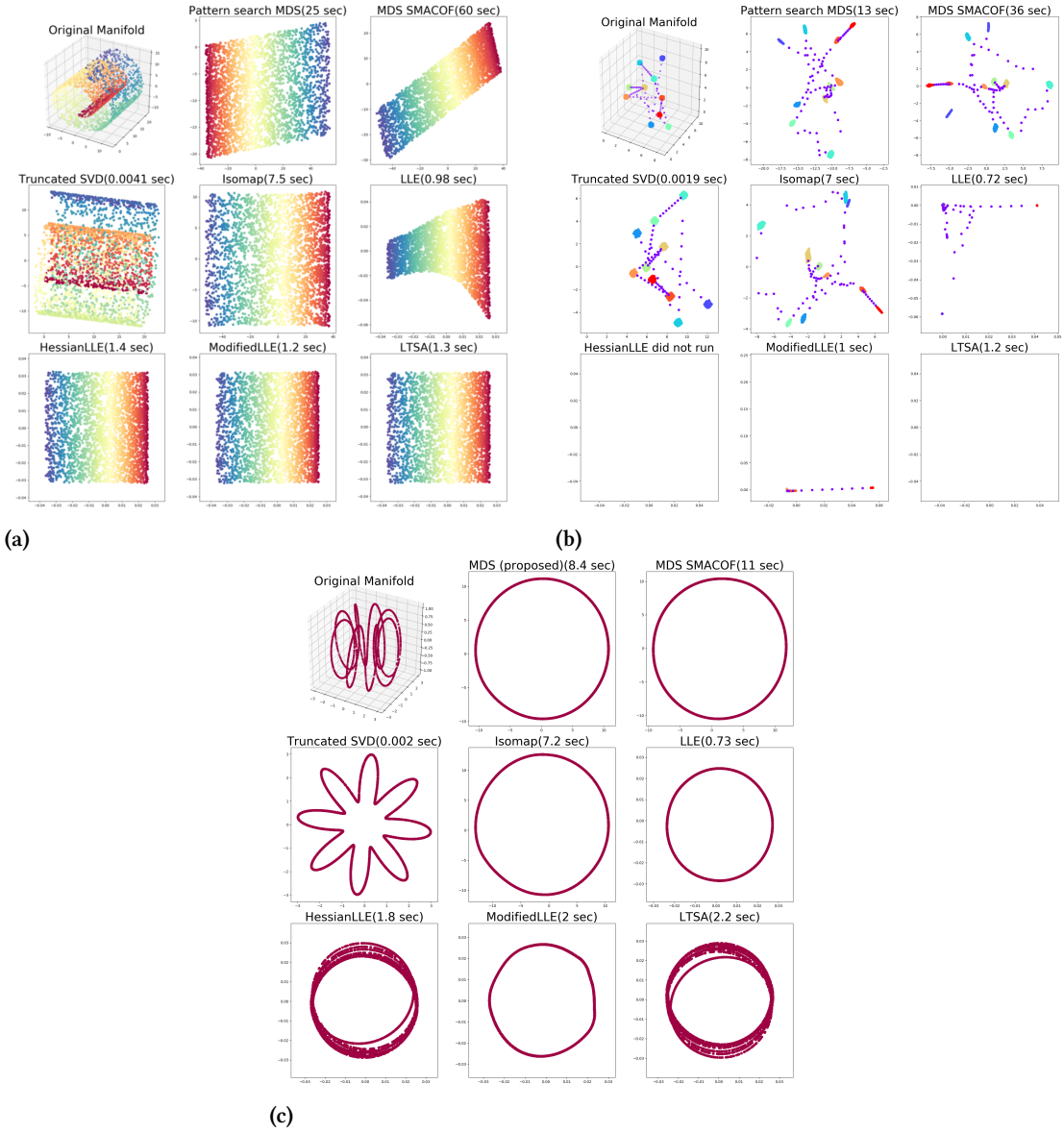


Figure 6.1: Comparison of pattern search MDS with other DR methods when converting: (a) $3D$ swissroll to $2D$ plane, (b) $3D$ clusters to $2D$ clusters, and (c) $3D$ toroid helix to $2D$ circle

posed, SMACOF, ISOMAP) produce good results, while the LLE variants can't handle sparsity in distance matrices very well. In particular Hessian LLE and LTSA do not produce any output because of numerical instability[†] in the eigenvalue decomposition stages of these algorithms. Pattern search MDS does not rely on eigenvalue computation or equation system solvers, and therefore it is numerically stable.

Finally, we showcase how the algorithms perform with transitions from dense to sparse regions with a toroidal helix shape in Fig. 6.1c. We can see that five methods, including pattern search MDS, unroll the shape into the expected $2D$ circle, while truncated SVD provides a daisy-like shape. Hessian LLE and LTSA collapse the helix into multiple overlapping circles.

6.2 DR for semantic similarity

Construction of semantic network models consists of representing concepts as vectors in a, possibly high-dimensional, space $\mathbb{R}^n$. The relations between concepts are quantified as the distances, or inversely the cosine similarities, between semantic vectors. The semantic similarity task aims to evaluate the correlation of the similarities between concepts in a given semantic space against a set of ground truth similarity values provided by human annotators.

We evaluate the performance of the dimensionality techniques investigated also in Section 6.1 for the semantic similarity task. We use the MEN[113] and SimLex-999[114] semantic datasets as ground truth. Both datasets are provided in the form of lists of word pairs, where each pair is associated with a similarity score. This score was computed by averaging the similarities provided by human annotators. As the high-dimensional semantic word vectors, we use the 300-dimensional GloVe vectors constructed by[115] using a large Twitter corpus. We reduce the dimensionality of the vectors to the target dimension L and calculate the Spearman correlation coefficient between the human provided and the automatically computed similarity scores. Results are summarized in Table 6.1 for $L = 10$. We observe that LLE yields the best results for MEN, while pattern search MDS performs best for SimLex-999. In addition, we observe that non-linear DR techniques can significantly improve the performance of the semantic vectors in some cases.

6.3 Dimensionality reduction for k-NN classification

The next set of experiments aims to compare the proposed algorithm to other DR methods for k-NN classification on a real dataset. We choose to use MNIST as a benchmark dataset which contains $70,000$ handwritten digit images. We selected a random subset of 1000 images and reduced the dimensionality from 784 to 20. Performance of the models is evaluated on k-NN with $k = 1$ classification and using 10-fold cross-validation. The evaluation metric is macro-averaged F1 score. Table 6.2 summarizes the results. Observe that DR using pattern

[†]In Hessian LLE the matrices used for the null space computation become singular, while in LTSA the resulting point coordinates are infinite.

Table 6.1: Comparison of DR techniques for the semantic similarity task for MEN and SimLex-999 datasets.

Method	Dimensions	MEN	SimLex-999
-	300	0.635	0.177
Pattern search MDS	10	0.596	**0.242**
SMACOF	10	0.632	0.221
ISOMAP	10	0.625	0.132
Truncated SVD	10	0.562	0.140
LLE	10	**0.657**	0.172
Hessian LLE	10	0.157	0.004
Modified LLE	10	0.643	0.158
LTSA	10	0.154	0.004

search MDS and Truncated SVD can improve classification performance over the original high-dimensional data. Pattern search MDS yields the best results overall. Hessian LLE, Modified LLE and LTSA did not run due to numerical instability.

Table 6.2: Comparison of DR techniques. We use the reduced features for classification on the MNIST dataset.

Method	Dimensions	MNIST (F1-score)
Original features	784	0.861
pattern search MDS	20	**0.878**
SMACOF	20	0.857
ISOMAP	20	0.829
Truncated SVD	20	0.871
LLE	20	0.813
Hessian LLE	20	—
Modified LLE	20	—
LTSA	20	—

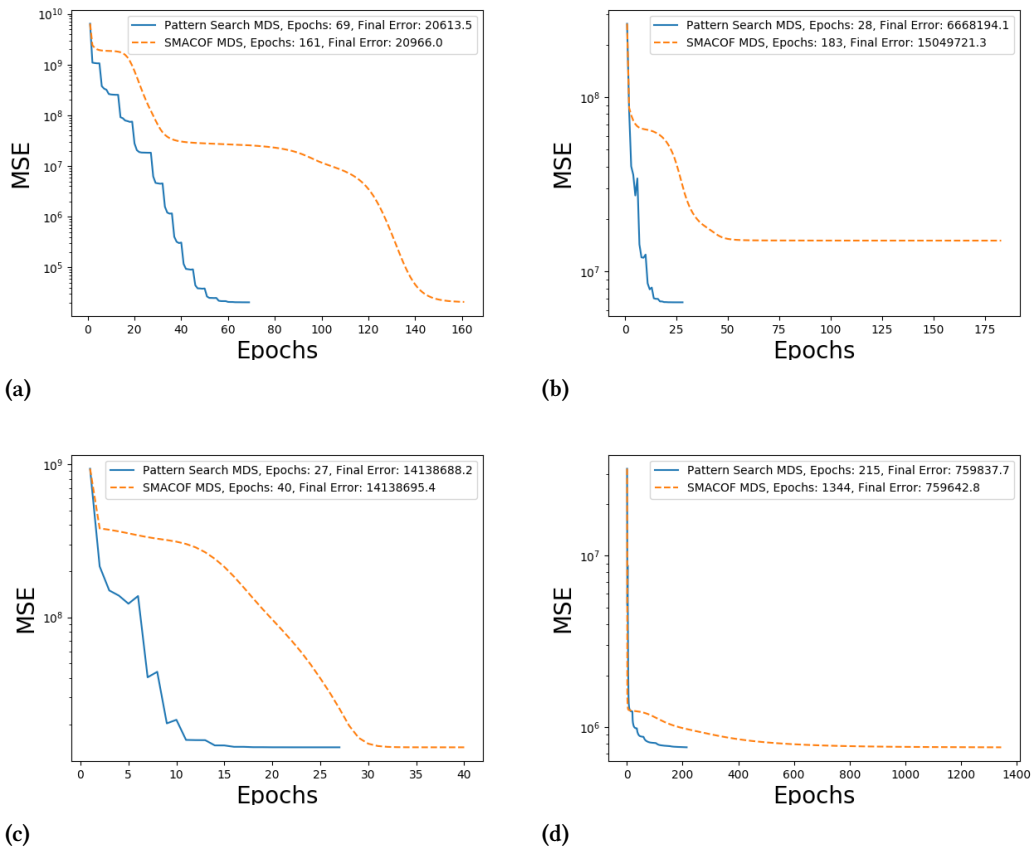


Figure 6.2: Convergence comparison of pattern search MDS and SMACOF for (a) swissroll, (c) toroid helix, (b) 3d clusters and (d) word vectors

6.4 Convergence characteristics

Next we compare speed of convergence of pattern search MDS and SMACOF, in terms of numbers of epochs. To this end we will consider the experiments of Sections 6.1 and 6.2 and present comparative convergence plots. We see the convergence plots for the cases of swissroll, 3*D* clusters, toroid helix in Fig. 6.2a, 6.2b and 6.2c, respectively. The convergence plot for the word semantic similarity task is shown in Fig. 6.2d. The plots are presented in y-axis logarithmic scale because the starting error is many orders of magnitude larger than the local minimum reached by the algorithms.

For all cases, we observe that pattern search MDS converges very quickly to a similar or better local optimum while SMACOF hits regions where the convergence slows down and then recovers. These saw-like structure of the pattern search plots are due to the fact that we allow for "bad moves" as detailed in Section 5.3.1.

From the computational point of view, we see intuitively that pattern search MDS algorithm performs a wider search of the solution space at each epoch, by exploring multiple directions, while gradient or majorization based optimization follows a narrower search path. This highlights a trade-off between the two approaches, where GPS-based optimization converges faster in terms of epochs, but has higher complexity per epoch. Future works could explore hybrid solutions that combine pattern search and gradient descent, in order to find the sweet spot between wide search space exploration and convergence speed.

6.5 Robustness to noisy or missing data

The final set of experiments aims to demonstrate the robustness of pattern search MDS when the input data are corrupted or noisy. To this end two cases of data corruption are considered: additive noise and missing data.

6.5.1 Robustness to additive noise

For this set of experiments, we inject Gaussian noise of variable standard deviation (σ) to the input data and use the dissimilarity matrix calculated on the noisy data as input to each one of the algorithms evaluated.

For the synthetic data of Section 6.1, we will follow a qualitative evaluation by showing the unrolled manifolds for high levels of noise. We perform DR for swissroll, toroid helix and $3D$ clusters for increasing noise levels. We report results for the highest possible noise deviation where one or more techniques still produce meaningful manifolds. Beyond these values of σ the original manifolds become corrupted and the output of all methods is dominated by noise. Figs. 6.3a, 6.3b, 6.3c show the results for noisy swissroll with $\sigma = 0.3$, $3D$ clusters with $\sigma = 0.4$ and toroid helix with $\sigma = 0.07$ respectively. Overall, the pattern search MDS, followed by SMACOF and ISOMAP are more robust to additive noise.

For the semantic similarity task we injected different levels of Gaussian noise in the original word vectors and evaluated the correlation on MEN and Simlex-999. Results are presented in Table 6.3. We observe that the relative performance of the algorithms is maintained under noise injection, except for LLE which cannot handle high amounts of noise. LLE is achieving the best correlation values on MEN at $\sigma = 0.01$ and $\sigma = 0.1$, while pattern search MDS achieving the best performance on Simlex-999.

6.5.2 Robustness to missing data

For the final set of experiments we consider the case of missing data. For this two new synthetic datasets where constructed, namely a dense and a sparse swissroll with a hole as shown in Fig. 6.4. In Fig 6.4a, we show the performance of the various algorithms applied to a dense swissroll with a hole in the middle. As we can see only Hessian LLE, modified LLE and LTSA are able to reconstruct the shape correctly, while MDS algorithms result in distortion around the hole. This is due to the non-convexity we introduced to the space when adding the

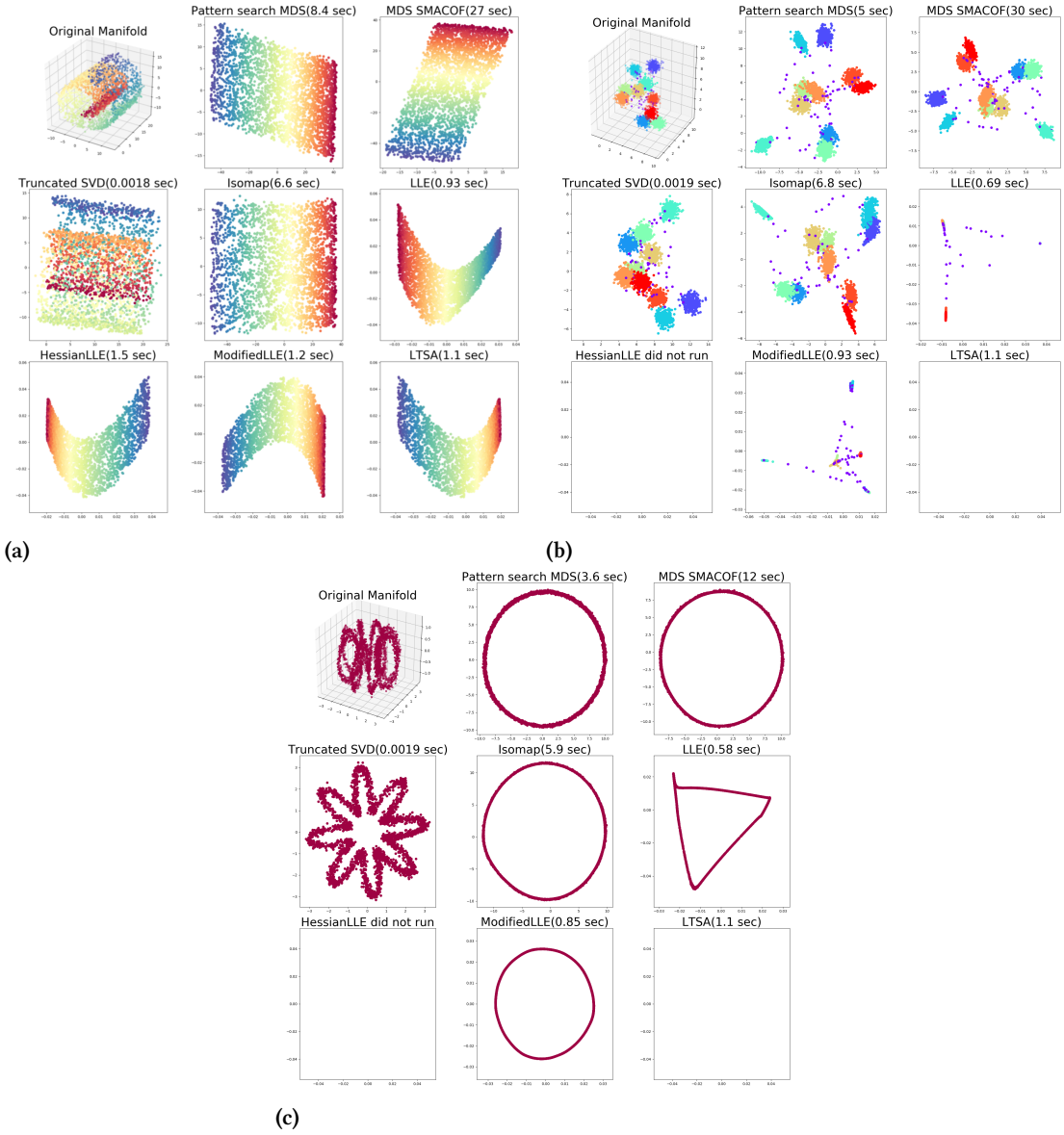


Figure 6.3: Comparison of pattern search MDS with other DR methods when converting noisy (a) $3D$ swissroll to $2D$ plane ($\sigma = 0.4$), (b) $3D$ clusters to $2D$ clusters ($\sigma = 0.3$) and (c) $3D$ toroid helix to $2D$ circle ($\sigma = 0.07$)

Table 6.3: Comparison of DR techniques with noisy word vectors on the semantic similarity task for MEN and SimLex-999 datasets. We introduce additive gaussian noise to the word vectors with increasing standard deviation $\sigma \in \{0.01, 0.1, 0.5\}$

Method	Dimensions	MEN			SimLex-99		
		$\sigma = 0.01$	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 0.01$	$\sigma = 0.1$	$\sigma = 0.5$
Original GloVe	300	0.635	0.619	0.431	0.178	0.169	0.077
pattern search MDS	10	0.593	0.597	0.462	**0.249**	**0.315**	**0.204**
SMACOF	10	0.633	0.620	0.462	0.229	0.222	0.123
ISOMAP	10	0.622	0.613	**0.497**	0.134	0.124	0.079
Truncated SVD	10	0.562	0.551	0.380	0.140	0.136	0.039
LLE	10	**0.659**	**0.649**	0.369	0.175	0.166	0.052
Hessian LLE	10	0.156	0.144	0.023	0.005	0.04	0.018
Modified LLE	10	0.635	0.633	0.489	0.158	0.162	0.096
LTSA	10	0.155	0.141	0.020	0.06	0.04	0.002

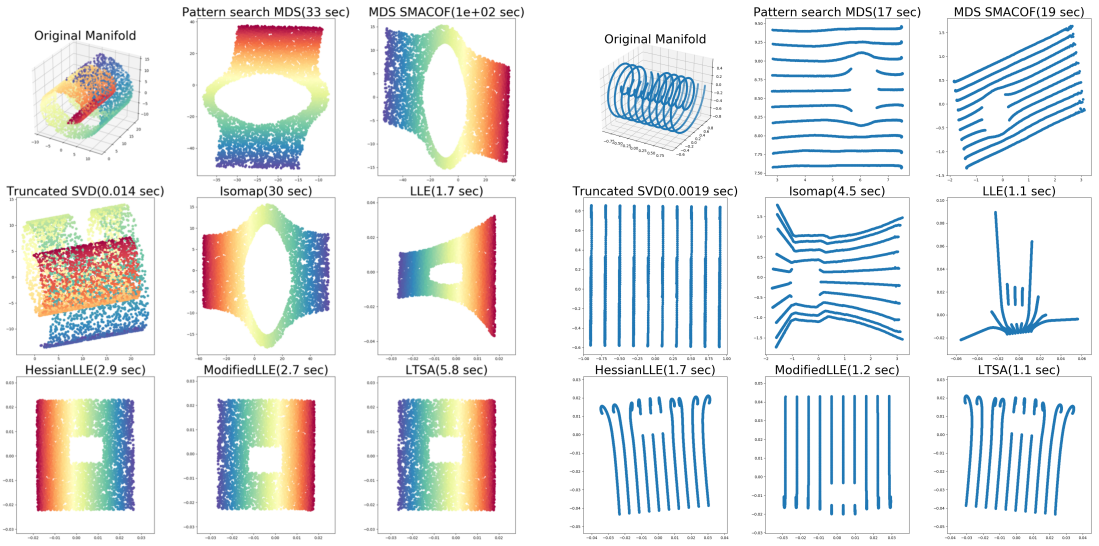
hole. This distortion can still be observed (to a lesser degree) in the sparse variation shown in Fig. 6.4b. For the sparse data case, we observe that LLE methods result in distortion around the edges.

These preliminary experiments indicate that LLE variations can handle non-convexities in input data, while MDS variations can handle sparse data better. This is because LLE methods are based on inferring and combining local data geometry, while MDS methods are inferring global geometry.

6.6 Discussion

We propose pattern search MDS, a novel algorithm for nonlinear DR, inspired by gradient-free optimization methods. Pattern search MDS is formulated as an instance of the wider family of GPS methods, thus providing theoretical guarantees of convergence up to a fixed point. Additional optimizations further improve the performance of our algorithm in terms of computational efficiency, robustness and solution quality. The qualitative evaluation against other popular DR techniques for both clean and noisy manifold geometry shapes indicates that pattern search MDS can accurately infer the intrinsic geometry of manifolds embedded in high-dimensional spaces. Furthermore, the comparison of convergence characteristics against SMACOF show that pattern search MDS converges in fewer epochs to similar or better solutions. Experiments on real data yield comparable to state-of-the-art results both for a lexical semantic similarity task and on MNIST for k-NN classification. Open-source implementations of pattern search MDS and the data generation process are provided to facilitate the reproducibility of our results.

Future work will focus on improving runtime performance and scalability of pattern search MDS. Specifically, an approach for decreasing per epoch computational complexity is to narrow the search space of possible moves as the geometry of the embedding space becomes more apparent by biasing the moves towards the principal component vectors of the neighborhood of the point that is being moved. This can be viewed as a combination of pattern search and



(a) **(b)**

Figure 6.4: Comparison of pattern search MDS with other DR methods for (a) dense and (b) sparse swissroll with hole. Target is a plane with a rectangular hole.

gradient descent, where the search space of moves is wide at the beginning and then gets increasingly biased towards the direction of the gradient. Our algorithm can scale to large numbers of points by utilizing landmark points[69] or fast approximations to MDS[116]. These approaches aim to alleviate the computational and memory cost of computing the full distance matrix, by approximating the data geometry using smaller submatrices. Moreover, stochastic approximations like stochastic SMACOF[117] can be adapted to pattern search MDS.

We also plan to provide more in-depth theoretical insights and ways to enable pattern search MDS to capture complex geometrical properties of input data. We aim to perform a detailed analysis on how heuristics and especially allowing for "bad moves" affect the performance of pattern search MDS. Furthermore, in Sections 6.1 and 6.5.2 we showcased that MDS can better handle sparse data and LLE can better handle non-convexity and missing data. This makes sense, as MDS takes into account the global geometry of the embedding space, while LLE focuses on the geometry of local neighborhoods. We plan to combine the cost functions of these approaches to infer both global and local geometry of the low dimensional data manifold. Another way to increase the expressiveness of the algorithm is to investigate a wider variety of distance metrics, and specifically non-symmetrical distance "metrics", motivated by cognitive sciences[58].

7

Unsupervised low-rank representations for speech emotion recognition

7.1 Dimensionality reduction and speech emotion recognition

Human-machine interaction is constantly evolving towards the use of more natural interfaces, like speech. Still the key difference between human-human and human-machine communication is the ability of humans to recognize the emotion of their conversation peers and modify their communication strategy based on that. Although significant progress has been done in the field of Speech Emotion Recognition (SER), machines have not achieved human-like performance. One of the reasons is the scarcity of available annotated data. SER databases are mostly composed of relatively small number of utterances from few speakers, which limits the generalization abilities of the models. Furthermore modern SER systems rely on feature sets of high dimensions. The small amount of training samples do not cover all combinations of values in the high-dimensional feature spaces and, thus, SER algorithms suffer from the Curse of Dimensionality (CoD)[118]. In this work we postulate that reducing dimensionality of the feature space is an effective way to combat CoD and demonstrate that low-dimensional representations yield simpler models with comparable performance. Dimensionality Reduction (DR) algorithms aim at learning low-dimensional latent representations of real world data. Such representations can be used for exploratory data analysis, to visualize and gain intuition on the statistical properties of data or, as in our case, extract latent features for input to classification or regression models.

Evidence that DR on speech features can create robust representations for SER can be found in the literature. Chiou and Chen[119], use PCA[120] to extract low-dimensional representations for the feature set introduced by Schuller et al.[121] containing 6552 features. The system is evaluated on Berlin emotional database (Emo-DB)[122]. In[123] use Linear Discriminant Analysis (LDA)[124] and PCA for SER, along with a weighted variation of LDA on a feature set of 225 dimensions. Experiments showed no significant performance difference between PCA and LDA. These methods are also compared by You et al.[125] and You et al.[126], along with Sequential Forward Selection[127], on a feature set consisting of 48 prosodic features and 16 formants. PCA representations extracted by You et al.[125] are found to be inferior than LDA, while[126] observed no significant difference. Selective Feature Selection and PCA are also explored by Ververidis et al.[128] for the Danish Emotion Speech database[129]. Lee et al.[130] experimen-

tally found that applying PCA on utterance-level statistics of pitch and energy features gives equivalent SER performance with the original features on a call center dialog corpus. Chuang and Wu [131] have reported that classification accuracy keeps improving when increasing the number of principal components only up to a centain rank for a feature set of 33 dimensions. A supervised variation of PCA along with Greedy Feature Selection [132] and ElasticNet [133] are explored by Fewzee and Karray [134] on two sets of energy-based and MFCC-based feature sets of 400 and 82 dimensions, with inconclusive results as to which approach is superior. The application of Linear and non-linear DR methods on SER is examined on a prosodic feature set of 48 dimensions by Zhang and Zhao [135]. Compared methods include unsupervised methods like PCA, ISOMAP [64] and LLE [136], and supervised methods LDA, Supervised LLE [137], Neighborhood Component Analysis [138], Maximally Component Metric Learning [139], local Fisher Discriminant Analysis [140] and Modified Supervised LLE. Results show better performance of PCA for unsupervised DR while Modified Supervised LLE was superior for supervised DR.

7.2 Features for speech emotion recognition

We consider the following feature sets:

Interspeech 2010 (IS10) set: The IS10 feature set [141] consists of 1582 features. IS10 is obtained by transforming the signal in the Fourier space. Features correspond to 21 statistical functionals (e.g. percentiles, linear regression coefficients) applied to 38 low level descriptors (MFCCs, PCM loudness etc.) and their deltas. Extraction is performed using the openSMILE toolkit.

Recurrence Quantification Analysis (RQA) set: The RQA feature set [142] consists of 432 features. This feature set is obtained by analyzing speech dynamics through phase space representation. The phase space is reconstructed through the use of time-delayed versions of the original signal and then the recurrence plots are calculated as thresholded pairwise distances of points in the phase space. Features are extracted as aggregated RQA measures from the recurrence plots. Source code for feature extraction is publicly available.*

Fused set: We concatenate features from IS10 and RQA into a represenation of 2014 dimensions, modeling both frequency content of speech signals and recurrence dynamics.

7.3 Experimental setup

We use the following databases for evaluation:

Emo-DB: Berlin Database of Emotional Speech (Emo-DB) [122] contains 535 emotional German sentences, voiced by 10 actors (5 male and 5 female). Specifically, 7 emotions are included i.e., 127 anger, 45 disgust, 70 fear, 71 joy, 60 sadness, 81 boredom and 70 neutral.

IEMOCAP: IEMOCAP database [143] contains 12 hours of video data with scripted and improvised dialog recorded by 10 actors. Utterances are organized in 5 sessions of dyadic interactions between pairs of actors. For our experiments we consider 5531 utterances of 4 emotions

*https://github.com/etzinis/nldrp

Table 7.1: Speech emotion recognition results for different combinations of DR techniques and classifiers using the IS10 features for the IEMOCAP dataset. We report unweighted accuracy (UA %).

	SVM (linear)	SVM (RBF)	k-NN	LR
Pattern search MDS	**56.0**	57.5	56.5	55.4
SMACOF	55.8	**58.5**	**56.7**	**55.8**
PCA	55.8	57.7	56.2	**55.8**
ISOMAP	52.3	52.5	51.7	52.2
LLE	53.4	54.2	53.6	53.2
Modified LLE	54.6	47.0	53.9	55.5
LE	54.1	54.3	54.2	55.1
Autoencoder	55.4	57.8	56.3	55.5
Original 1582D	54.7	**59.8**	55.7	**56.9**

(1103 angry, 1636 happy, 1708 neutral and 1084 sad), where we merge excitement class into happiness[144–147].

We consider utterance-level, speaker independent SER for our experiments. In this setup a number of speakers are kept hidden from the training set and used for evaluation. Specifically in the case of Emo-DB we perform leave one speaker out cross-validation, where test folds contain the instances of the unknown speaker. For IEMOCAP we use the leave one session out cross-validation scheme, where two speakers participating in a session are used as the evaluation folds. This results in a 10-fold cross-validation scheme for Emo-DB and 5-fold cross-validation for IEMOCAP. We apply Z-normalization to standardize the features in zero mean and unit variance, where each sample x is transformed according to the formula $z = \frac{x-\mu}{\sigma}$. Note that for speaker independent experiments only samples in the training set are used to calculate μ and σ and test samples are normalized using these statistics.

Representations resulting from all DR approaches are evaluated for k-NN classification. We perform grid search on the optimal number of neighbors k in the $[1, 30]$ range and report results for the optimal value for each dimension and each method. Optimal values of k range from 13 to 20 indicating that consistent neighborhoods are formed in the low-dimensional spaces. We also evaluate low-rank representations on Support Vector Machines (SVM) with linear and Radial Basis Function (RBF) kernels, and Logistic Regression (LR), with optimal value of C in the range $[0.01, 10]$. Autoencoder is trained with 3 encoder layers, 3 decoder layers and 1 hidden layer, using ReLU activations.

7.4 Results

As evaluation metrics we used both weighted accuracy and unweighted accuracy. For brevity we report unweighted accuracy results, noting that same trends form with respect to the weighted accuracy metric. Fig. 7.1a shows the results of DR applied to the RQA features on Emo-DB for all DR methods, for different embedding dimensions L. We observe that Modified LLE achieves the best results when $L = 50$, followed by SMACOF in $L = 25$. Observe that all methods except ISOMAP and LE manage to outperform the original features of 432 dimensions.

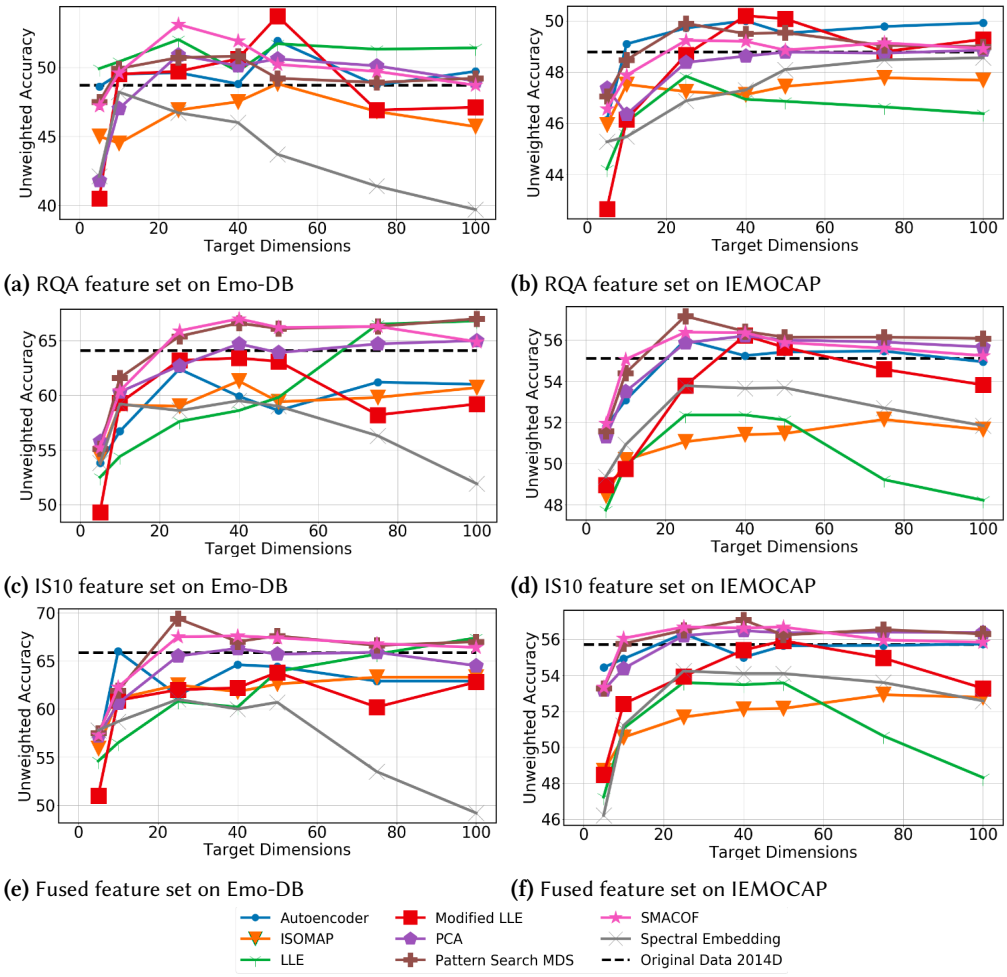
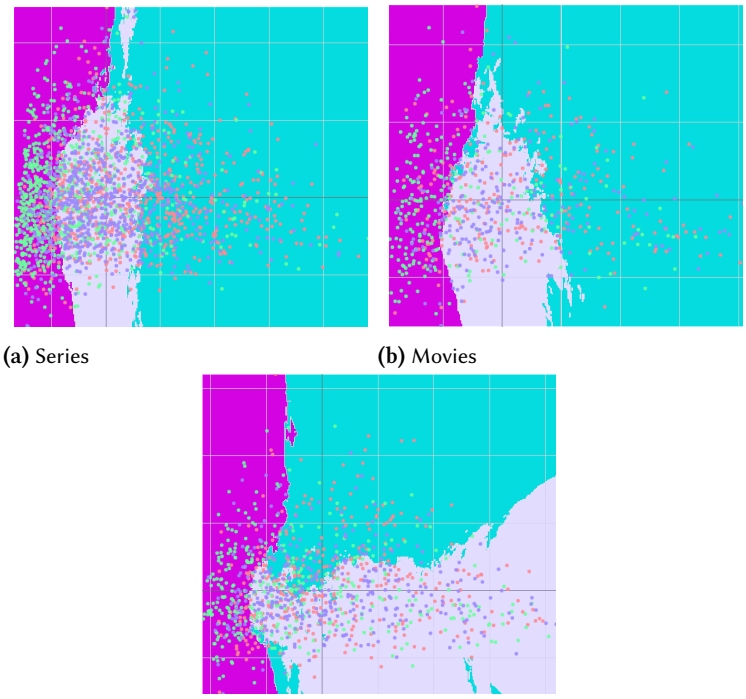


Figure 7.1: DR for the RQA (top), IS10 (middle), and fused (bottom) feature sets on Emo-DB (left) and IEMOCAP (right) for varying target dimensions. We report unweighted accuracy for k-NN classification. Observe that for multiple techniques, when choosing target dimensions in the range 25–100 the reduced features outperform the original features, indicated by the dashed line.

In Fig. 7.1c, which shows results for DR on IS10 features for Emo-DB, we can observe a different pattern. Here the MDS algorithms perform best for every embedding dimension, followed by PCA, all three of these methods outperforming the original feature set of 1582 dimensions. This indicates that this feature set resembles more a hyperplane in the high-dimensional space than a non-linear manifold. Non-linear methods like LLE, ISOMAP and LE underperform. For the fused feature set in Fig. 7.1e we see again that distance-preserving transformations yield the best performance. Same patterns emerge in IEMOCAP in Fig. 7.1b, 7.1d, 7.1f, with Modified LLE achieving better performance for the RQA features and Pattern Search MDS and PCA



(a) Series **(b)** Movies

(c) Interviews

Figure 7.2: Visualization of decision regions with PCA for speech emotion recognition in a large proprietary multi-domain dataset, containing speech segments from movies, tv series, and interviews.

yielding best representations for IS10 features. Notably in IEMOCAP, performance of the Autoencoder is significantly better because there are more training samples. For the experiments with the fused feature set we again observe a consistent pattern in both Emo-DB and IEMOCAP, with MDS yielding again the best representations followed by PCA. Fusion is still beneficial after applying DR though we observe that the structure of IS10 features dominates under fusion.

In Table 7.1 we show results for linear SVM, RBF SVM, k-NN and LR. We reduce dimensionality of IS10 features from 1582 to 25 dimensions and report unweighted accuracy (UA) on IEMOCAP. Low-rank representations produce very competitive results to the original sparse features, while for linear SVM and k-NN they even improve classification accuracy. Overall global, linear DR methods like MDS and PCA produce the best representations.

7.5 Visualizations

We include visualizations of feature maps reduced in 2D. We focus on the best and the worst performing methods and comment on some interesting observations.

Figure 7.2 demonstrates the results of PCA into two dimensions, for a large proprietary

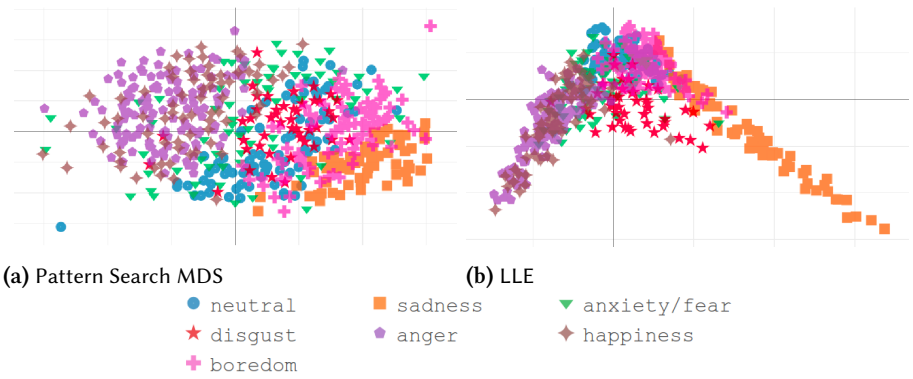


Figure 7.3: Visualization of the fused features for different emotions on Emo-DB using different DR techniques.

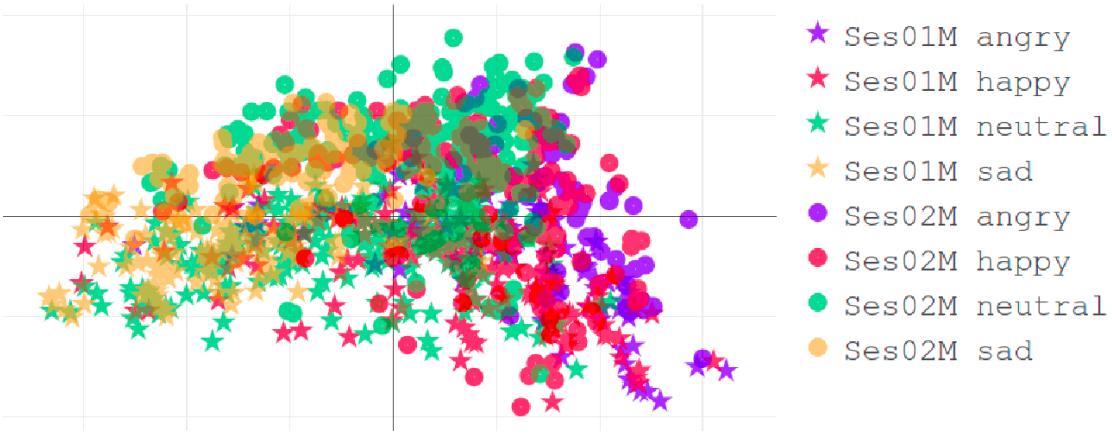


Figure 7.4: Visualization of the fused features using ISOMAP for two IEMOCAP speakers and four emotion classes.

and internally annotated dataset containing speech segments from multiple domains such as movies, TV series and interviews. Subfigures illustrate the distributions of the speech segments into the two PCA dimensions for three emotional classes: anger, happiness and sadness. In addition, we illustrate the decision surfaces for a simple k-NN classifier. The results demonstrate how the blue class (anger) is similarly distributed between the red and green (sadness and happiness respectively) for the two first domains (Series and Movies) in Fig 7.2a and Fig. 7.2b respectively, based on the primary PCA dimension (x axis). On the other hand, for the interviews domain, the primary PCA dimension is not enough to discriminate between the emotional classes as we see in Fig. 7.2c. On the contrary, the anger and happiness classes are mostly discriminated based in the second PCA dimension. Interestingly, this unsupervised distribution is quite similar to the Valence-Arousal affective representation. This example demonstrates how an unsupervised dimensionality reduction can be very sensitive to changes in domain when illustrating emotional content.

Fig. 7.3a shows the 2D space created using Pattern Search MDS, which maps the points inside an elongated disk area. We can see on the left the anger points while the sadness points are on the right. Close to anger is happiness samples, while boredom is close to sadness. Other emotions lie in the middle. So it looks like that even in the 2D space MDS learns meaningful representations, with x axis being a latent feature that can encode arousal. On the contrary LLE, which tries to preserve local neighborhoods and yields poor recognition accuracy on the fused feature set concentrates most samples in the center as we can see in Fig. 7.3b, but still we can observe low arousal emotions (sadness) being separated from high arousal ones (anger). In Fig. 7.4 we show ISOMAP embeddings for two speakers in IEMOCAP. Observe, although ISOMAP cannot separate emotions, it achieves a better discrimination result, in terms of speaker separation, for this experiment. One could consider basing a speaker diarizer on geodesic distances between samples.

7.6 Discussion

In this work we explore the effects of unsupervised linear and non-linear DR on state-of-the-art speech features for SER. We evaluate these algorithms for speaker independent SER on IEMOCAP and Emo-DB. Experiments show that performance of low-rank representations is competitive to original high-dimensional representations. This phenomenon is hypothesized to be caused by the curse of dimensionality, since the number of samples in SER datasets does not span the high-dimensional space. Interpretation of results and vizualization of 2D representations gives interesting insights on the high-dimensional structures. First insight is that IS10 features can be decomposed by use of linear DR, e.g. by use of PCA or MDS algorithms. Second, distance preserving DR can encode meaningful dimensions, e.g. arousal. Third, speaker samples can be separated by isometric mappings. Fourth, unsupervised DR can be rather sensitive when illustrating cross-domain emotional content. Future work will focus on creating end-to-end representations using autoencoders with distance preserving regularization and investigating the interesting insight on using geodesic-distance preserving

representations for speaker separation.

Part II

Mixed Self-Supervision for Sample-Efficient Unsupervised Domain Adaptation

"*They made data a controlled substance.*"

Neal Stephenson, Snow Crash

8

Unsupervised domain adaptation for text and speech

8.1 Introduction

Deep architectures have achieved state-of-the-art results in a variety of machine learning tasks. However, real world deployments of machine learning systems often operate under domain shift, which leads to performance degradation. This introduces the need for adaptation techniques, where a model is trained with data from a specific domain, and then can be optimized for use in new settings. Efficient techniques for model re-usability can lead to faster and cheaper development of machine learning applications and facilitate their wider adoption. Especially techniques for Unsupervised Domain Adaptation (UDA) can have high real world impact, because they do not rely on expensive and time-consuming annotation processes to collect labeled data for domain-specific supervised training.

In the second part of this dissertation we devise training strategies that facilitate the unsupervised adaptation of text and speech systems to unseen domains. Particular focus is placed in the amount of data needed for successful adaptation, and we verify the sample-efficiency of the proposed techniques through extensive experimentation. We validate the proposed method for two diverse application settings, text classification and Automatic Speech Recognition (ASR). The proposed methods work in tandem with popular, state-of-the-art models, trained in a self-supervised manner; for text we verify our approach with BERT[10] and for speech with XLSR-53[148]. This part consolidates our published works in the North American Association of Computational Linguistics (Karouzos et al.[149]) and the IEEE/ACM Transactions on Audio Speech and Language Processing (Paraskevopoulos et al.[150]).

8.2 Organization

The second part of this dissertation is organized as follows: In this chapter we begin in Section 8.3 with a rigorous definition for the problem of UDA in the context of classification, which is extended for ASR in a sequence-to-sequence setting. Then we outline the recent advancements in UDA for text and speech models in Sections 8.4 and 8.5, and we provide a general overview of the proposed training strategy in Section 8.6. In Chapter 9 we employ

the proposed training strategy for UDA of BERT, to perform sentiment classification of textual product reviews. In Chapter 10 we consider the problem ASR, and modify our approach for acoustic adaptation of XLSR-53. We create a 120 hour Greek ASR corpus of parliamentary speech and perform cross-corpus evaluation to validate the performance of the proposed UDA method.

8.3 Problem definition

Formally, the problem of UDA can be defined as follows. Let $X \subseteq \mathbb{R}^n$ be a real-valued space that consists of n-dimentional feature vectors $x \in X$, and Y a finite set of labels $y \in Y$, i.e., $Y = \{1, 2, \ldots, L\}$. Furthermore, assume two different distributions, i.e., the source domain distribution $\mathcal{S}(x, y)$ and the target domain distribution $\mathcal{T}(x, y)$, defined on the cartesian product $X \times Y$.

The goal is to train a model that learns a mapping between feature vectors $x_{\mathcal{T}}$ to their respective labels $y_{\mathcal{T}}$ for samples drawn from the target distribution $(x_{\mathcal{T}}, y_{\mathcal{T}}) \sim \mathcal{T}$.

At training time we have access to samples from the source distribution $\mathcal{S}(x, y)$ and the marginalized target distribution $\mathcal{T}(x)$, i.e., no target labels are provided. We define the training dataset D as the concatenation of the source and target training sets, $D = (D_S, D_T)$. D_S and D_T are defined as sequences of tuples, i.e.,

$$\begin{aligned} D_S &= \{(x_i, y_i) \,|\, (x_i, y_i) \sim \mathcal{S}(x, y),\ 1 \leq i \leq N\} \\ D_T &= \{(x_j, \emptyset) \,|\, x_j \sim \mathcal{T}(x),\ 1 \leq j \leq M\}, \end{aligned} \tag{8.1}$$

where we draw N samples from $\mathcal{S}(x, y)$ and M samples from $\mathcal{T}(x)$. Finally, we augment tuples in D with a domain indicator function:

$$\begin{aligned} D &= \{(x_k, y'_k, 1_k) \,|\, 1 \leq k \leq N + M\} \\ 1_k &= \begin{cases} 0 & \text{if } x_k \sim \mathcal{S}(x), \\ 1 & \text{if } x_k \sim \mathcal{T}(x). \end{cases} \\ y'_k &= \begin{cases} y_k & \text{if } x_k \sim \mathcal{S}(x), \\ \emptyset & \text{if } x_k \sim \mathcal{T}(x). \end{cases} \end{aligned} \tag{8.2}$$

Unsupervised acoustic adaptation for automatic speech recognition

The above definition can be directly extended in the case of speech recognition, with some modifications. In detail, we modify the feature space X, to be the set of (finite) sequences of real-valued feature vectors $(x_m)_{m \in \mathbb{N} \setminus \{\infty\}} \in X \subseteq (\mathbb{R}^n)^*$. Furthermore, the label space Y is modified to be the set of sequences $(y_n)_{n \in \mathbb{N} \setminus \{\infty\}}$, where $Y = (\{1, 2, \ldots, L\})^*$ contains finite-length sequences over a finite lexicon. For Connectionist Temporal Classification (CTC) training

we assume that $m > n$ for any sample (x_m, y_n), i.e., feature sequences are longer than their respective label sequences[151]. The rest of the definitions need no modifications.

Unsupervised language adaptation for automatic speech recognition

Adaptation for ASR systems can also be performed at the language level, i.e., the label space. In this setting, we assume that the target domain samples are drawn from the marginalized target distribution $\mathcal{T}(y)$. The target dataset D_T now consists of tuples in the form $(\emptyset, y_j)$, where y_j is the label word sequence $(y_n)_{n \in \mathbb{N} \setminus \{\infty\}}$ for the j-th sample.

Weakly supervised adaptation for automatic speech recognition

The last setting we explore is the case where both audio and language in-domain samples are available, but the mapping between them is unknown. This situation can be encountered in real-world settings, e.g., in the case in-domain audio and text are collected independently. For example consider the case where audio clips from newscasts are collected, along with contemporary newspaper articles. Another example is the case where long audio clips alongside transcriptions are available, but no fine-grained time alignments*. In this case, the target domain samples are drawn independently from the marginalized distributions $\mathcal{T}(x)$ and $\mathcal{T}(y)$, and the target dataset D_T consists of tuples in the form $(x_j, \emptyset)$ and $(\emptyset, y_j)$.

8.4 Unsupervised domain adaptation in natural language processing

UDA approaches for text can be grouped in three major categories; Pseudo-Labeling (PSL) techniques[152,153], domain adversarial training[56], and pivot-based approaches[154,155]. Pseudo-Labeling (PSL) approaches use a model trained on the source labeled data to produce pseudo-labels for unlabeled target data and then train a model for the target domain in a supervised manner. Domain Adversarial Training (DAT) aims to learn a domain-independent mapping for input samples by adding an adversarial cost during model training, that minimizes the distance between the source and target domain distributions. Pivot-based approaches aim to select domain-invariant features (pivots) and use them as a basis for cross-domain mapping.

Traditionally, UDA has been performed using PSL approaches. PSL techniques are semi-supervised algorithms that either use the same model (self-training)[152,156,157] or multiple ensembles of models (tri-training)[153,158] in order to produce pseudo-labels for the target unlabeled data. Saito et al.[159] proposed an asymmetric tri-training approach. Ruder and Plank[160] introduced a multi-task tri-training method. Rotman and Reichart[161] and Lim et al.[162] study PSL with contextualized word representations. Ye et al.[163] combine self-training with XLM-R[28] to reduce the produced label noise and propose CFd, class aware feature self-distillation.

Another line of UDA research includes pivot-based methods, focusing on extracting cross-domain features. Structural Correspondence Learning (SCL)[154] and Spectral Feature Align-

*While a fully supervised in-domain dataset can be constructed in this case using long / forced alignment methods, this is not a focal point for the experimental part of this work.

ment[155] aim to find domain-invariant features (pivots) to learn a mapping between two domain distributions. Ziser and Reichart[164,165,166] combine SCL with neural network architectures and language modeling.
Miller[167] proposes to jointly learn the task and pivots. Li et al.[168] learn pivots with hierarchical attention networks. Pivot-based methods have also been used in conjunction with BERT[169].

DAT is a dominant approach for UDA[170], inspired by the theory for learning from different domains introduced by Ben-David et al.[171,172]. Ganin et al.[56], Ganin and Lempitsky[173] propose to learn a task while not being able to distinguish if samples come from the source or the target distribution, through use of an adversarial cost. This approach has been adopted for a diverse set of problems, e.g. sentiment analysis, tweet classification and universal dependency parsing[174–176]. Du et al.[177] pose domain adversarial training in the context of BERT models. Zhao et al.[178] propose multi-source domain adversarial networks. Guo et al.[179] propose a mixture-of-experts approach for multi-source UDA. Guo et al.[180] explore distance measures as additional losses and use them to construct dynamic multi-armed bandit controller for the source domains. Shen et al.[181] learn domain invariant features via Wasserstein distance. Bousmalis et al.[182] introduce domain seperation networks with private and shared encoders.

Unsupervised pretraining on domain-specific corpora can be an effective adaptation process. For example BioBERT[183] and SciBERT[184] are specialized BERT variants, where pretraining is extended on large amounts of biomedical and scientific corpora respectively. Sun et al.[185] propose continuing the pretraining of BERT with target domain data and multitask learning using relevant tasks for BERT fine-tuning. Xu et al.[186] introduce a review reading comprehension task and a post-training approach for BERT with an auxiliary loss on a question-answering task. Continuing pretraining on multiple phases, from general to domain specific and task specific data, further improves performance of pretrained language models, as shown by Gururangan et al.[187]. Han and Eisenstein[188] propose AdaptaBERT, which includes a second phase of unsupervised pretraining, in order to use BERT in a unsupervised domain adaptation context.

Recent works have highlighted the merits of using language modeling as an auxiliary task during fine-tuning. Chronopoulou et al.[55] use an auxiliary Language Model (LM) loss to avoid catastrophic forgetting in transfer learning and Jia et al.[189] adopt this approach for cross-domain named-entity recognition. Motivated by these approaches, we use a joint, in-domain LM for UDA.

8.5 Unsupervised domain adaptation in automatic speech recognition

In the following subsections we provide an overview of different adaptation approaches in the literature and link each approach to the UDA problem formulation. Table 8.1 presents a summary of the key adaptation settings and applications that are explored in the literature. We see, that a relatively small amount of methods, and their variants, are used to address multiple real-world ASR problems, for example, cross-lingual, accent, speaker, and noise adaptation. Furthermore, while the majority of the works focus on the English language, there is an effort

Table 8.1: Summary of related works on UDA for ASR.

Work	Method	Model	Adaptation Setting	Language
[190–192]	Teacher-Student Hard and soft labels	Conformer RNN-T[193] Transformer CTC RNN-T[194]	News speech, Voice search, Far-field, Telephony, YouTube	English
[195,196]	Teacher-Student Soft labels	TDNN-LSTM[197]	Noise, Far-field	English
[198]	Teacher-Student Hard and soft labels	NiN-CNN[199]	Dialects Children speech	Japanese
[200]	Teacher-Student Soft labels	Streaming RNN-T[201]	Multilingual	English, Brazilian Portuguese, Russian, Turkish, Nordic/Germanic
[202]	Teacher-Student Hard labels	wav2vec2[203]	Cross-lingual	English, French, German, Italian, Polish, Arabic, Spanish, Portuguese, Dutch
[204–206]	Domain Adversarial Training	TDNN Kaldi[207,208] DNN-HMM DNN-HMM	Noise, Channel	English
[209]	Domain Adversarial Training	RNN-CTC[210]	Far-field	English
[211,212]	Domain Adversarial Training	TDNN Kaldi RNN-T	Accent	Mandarin
[213,214]	Domain Adversarial Training	DNN-HMM CNN-DNN	Speaker, Gender, Accent	English
[215]	Domain Adversarial Training	DSN[182]	Multilingual	Hindi, Sanskri
[216,217]	Continuous Pre-Training	wav2vec2	Audiobooks, Accents, Ted Talks, Telephony, Crowd-sourced, Parliamentary speech	English
[218]	Continuous Pre-Training	wav2vec2	Cross-lingual	Korean
[219,220]	Continuous Pre-Training	XLSR-53[148] wav2vec2	Low resource languages	Ainu Georgian, Somali, Tagalog, Farsi
[221]	Continuous Pre-Training (Adapters)	wav2vec2, HuBERT[222]	Children speech	English

to explore other popular languages, e.g., Mandarin, and under-resourced languages, e.g., Ainu, Somali, etc.

8.5.1 Teacher-student models

Teacher-Student learning is one of the earliest methods in semi-supervised learning[152,223,224]. Teacher-Student learning can be construed as a general framework for Pseudo-Labeling approaches. The key idea is to reduce the problem of unsupervised learning of the task at hand in the target domain to a supervised one. The general methodology is to train a teacher model g_S using the labeled data in the source domain D_S, and then use this for inference on the target domain to produce pseudo-labels $\hat{y}_j = g_S(x_j),\ x_j \sim \mathcal{T}(x)$. The target domain dataset D_T is augmented with these silver labels, to contain tuples $(x_j, \hat{y}_j)$. Finally, a student model g_T is trained in a supervised fashion, using the augmented D_T or a combination of D_S and D_T. This process is usually repeated, with the student model serving as the teacher model for the next iteration, until no further improvement is observed. An end-to-end iterative self-training approach for CTC models is proposed bt Chen et al.[225]. Momentum Pseudo-Labeling[226] and Kaizen[227] have

been proposed to avoid the inefficient retraining required by iterative approaches, by maintaining a slowly evolving teacher model, updated using momentum or exponential moving average. Recently, soft target Teacher-Student learning has been explored for ASR[192,200,228], where the Kullback–Leibler (KL) divergence between the teacher and student output label distributions is used as the loss function.

Being trained only on the source domain data the teacher model is susceptible to error propagation. Filtering is a commonly used technique to achieve the right balance between the size of the target domain used for training the student model and the noise in the pseudo-labels. Confidence scoring based on the likelihood is usually applied, discarding those utterances for which the hypothesized labels are untrustworthy [229]. Khurana et al. [191] use dropout to measure the model uncertainty. The agreement between model predictions with and without dropout is used for confidence scoring. Hwang et al. [190] apply a multi-task training objective with a confidence loss to minimize the binary cross entropy between the estimated confidence and the binary target sequence. To learn more robust and generalizable features from the teacher model, Noisy student training has been proposed by et al [230]. The teacher models generate pseudo-labels for D_T while the student models are trained on a heavily augmented version of D_T[230]. et al[230], Zhang et al. [231] augment the input target data with SpecAugment[232], while Asami et al. [198] perform spectrum frequency augmentation.

Li et al. [195] introduce Teacher-Student learning with soft labels for ASR to tackle noisy, far-field, and children's speech. [196] extend this approach for LF-MMI-based models and used for noisy, far-field, and bandwidth adaptation. In [198] a weighted sum of hard and soft target cross-entropy losses is used for Japanese dialects and children's speech adaptation. Ramabhadran et al. [200] propose a self-adaptive distillation method from multiple teachers which is applied across several multilingual ASR systems for different language groups. A comparison between soft and hard targets for RNN-T models [194] showed that soft targets perform better when both the teacher and student models have the same architecture. Otherwise, hard targets are superior[228].

8.5.2 Domain adversarial training

DAT was initially introduced for image classification[173]. The key idea is to train a model that learns deep features that solve the task at hand in the source domain while being invariant to the domain shift. Concretely, the model is trained end-to-end using the combined loss $L = L_t - \alpha L_a$, where L_t is the supervised task loss L_t, learned on D_S, and L_a is the domain discrimination loss. The loss L_a is binary cross-entropy, trained for domain discrimination using the tuples $(x_k, 1_k)$. Notice the $-$ sign in the loss indicates adversarial learning, i.e., the model should learn features that cannot discriminate between domains while solving the task.

Shinohara [204] employ DAT for noise adaptation on a noise-corrupted version of WSJ[233] as the target dataset. Using the Aurora-4[234] dataset which has labels associated with the noise type, Serdyuk et al. [205] train an adversarial noise classifier. Sun et al. [211] and Hu et al. [212] use DAT for accent adaptation for Mandarin and English respectively. Anoop C.S. et al. [215] propose DAT, to address the scarcity of data in low-resource languages that share a common acoustic

space with a high-resource language, namely Sanskrit and Hindi. They empirically demonstrate the effectiveness of adversarial training, presenting experiments with and without the reversal of the domain classification loss.

8.5.3 Leveraging in-domain self-supervision

These lines of work have roots in NLP tasks[149,187], and explore domain adaptation by leveraging the in-domain data D_T for Self-Supervised Learning (SSL). The core focus is domain adaptation of large pre-trained models, e.g.,[10], and self-supervision is achieved by the use of the pre-training self-supervised loss L_s. This process can either take part in stages, via continual pre-training[187], or by constructing a multitask objective $L = L_t + \alpha L_s$, as suggested by Karouzos et al.[149].

Continuous Pre-Training (CPT), else found as Domain Pre-Training (DPT), has been explored for adaptation of ASR models. Hsu et al.[216] explore the effectiveness of CPT for domain adaptation, indicating the importance of utilizing unlabeled in-domain data. In CASTLE[217], CPT is combined with an online PSL strategy for domain adaptation of wav2vec2. Cross-dataset evaluation for popular English speech corpora indicates that CPT helps to reduce the error rate in the target domain. Kim and Kang[218] and Nowakowski et al.[220] utilize CPT for cross-lingual adaptation of wav2vec2 for Korean and Ainu, respectively. Notably for Ainu, which is an endangered language, CPT has resulted in significant system improvement. DeHaven and Jayadev[219] compare CPT and PSL for adapting XLSR-53 to four under-resourced languages, i.e., Georgian, Somali, Tagalog, and Farsi. They find that both approaches yield similar improvements, with CPT being the more computationally efficient approach. Fan et al.[221] employ CPT based on Adapters[235] to adapt wav2vec2 and HuBERT for child speech.

8.6 Overview of the proposed training strategy

While CPT yields significant improvements in a variety of tasks, one common theme in these works is the assumption of hundreds or thousands of hours of available in-domain data, mostly from online resources, e.g., YouTube. This can be infeasible when we consider more niche adaptation settings, or possible privacy concerns, e.g., how would one collect 1000 hours of psychotherapy sessions in Greek? Furthermore, CPT approaches rely on in-domain pre-training, which is susceptible to catastrophic forgetting[48,236]. Popular methods to combat catastrophic forgetting include Incremental Learning[237] in a few-shot setting and Moment Matching[238] of the learned posterior distributions. More related to our approach, Elastic Weight Consolidation (EWC)[51] combats catastrophic forgetting when learning a new task B by slowing down learning in neurons that are important for a previously learned task A. EWC aims to *explicitly* find a balance in the stability-plasticity trade-off, by including an additive regularization term. However, direct application of EWC for large models may be computationally infeasible, since the proposed regularization term requires the computation of the Fisher information matrix[239] and a norm calculation over all the network parameters θ_i, as can be seen by the second term in

Eq. (8.3). In addition, multiple copies of the network weights are needed to be kept in memory when learning multiple tasks.

$$\mathcal{L}(\theta) = \mathcal{L}_B(\theta) + \sum_i \frac{\lambda}{2} F_i(\theta_i - \theta^*_{A,i}) \tag{8.3}$$

Hence, we opt to *implicitly* find a balance between stability and plasticity, by leveraging in-domain self-supervision to combat catastrophic forgetting for UDA in a data-constrained environment. The key idea is to start from a pretrained model, trained with a self-supervised loss L_{SS}. During fine-tuning, we learn the new task A using the labeled out-of-domain data $x_{\mathcal{S}} \sim \mathcal{S}$, while simultaneously adapting to the target domain using the unlabeled, in-domain data $x_{\mathcal{T}} \sim \mathcal{T}$ using the self-supervised loss. The total fine-tuning loss is formulated in Eq. (8.4). We find that the proposed fine-tuning strategy with mixed self-supervision is robust, leads to effective adaptation, and can be easily adapted to work in diverse application settings.

$$L(x_{\mathcal{S}}, x_{\mathcal{T}}) = L_A(x_{\mathcal{S}}) + \lambda L_{SS}(x_{\mathcal{T}}) \tag{8.4}$$

9

Unsupervised Domain Adaptation through Language Modeling

9.1 Introduction

The rise of Self-Supervised Learning [240–243] has created a paradigm shift in the way we develop machine learning enabled applications. Transfer learning from language models pretrained in massive corpora [10,32,244–246] has yielded significant improvements across a wide variety of NLP tasks, even when small amounts of data are used for fine-tuning. Fine-tuning a pretrained model is a straightforward framework for adaptation to target tasks and new domains, when labeled data are available. However, optimizing the fine-tuning process in Unsupervised Domain Adaptation (UDA) scenarios, where unlabeled in-domain data are available is challenging.

We propose Unsupervised Domain Adaptation through Language Modeling (UDALM), a fine-tuning method for BERT [10] in order to address the UDA problem. Our method is based on simultaneously learning the task from labeled data in the source distribution, while adapting to the language in the target distribution using multitask learning. The key idea of our method is that by simultaneously minimizing a task-specific loss on the source data and a language modeling loss on the target data during fine-tuning, the model will be able to adapt to the language of the target domain, while learning the supervised task from the available labeled data.

Our key contributions are: (a) We introduce UDALM, a novel, simple and robust unsupervised domain adaptation procedure for downstream BERT models based on multitask learning, (b) we achieve state-of-the-art results for the Amazon reviews benchmark dataset, surpassing more complicated approaches and (c) we explore how A-distance and the target error are related and conclude with some remarks on Domain Adversarial Training (DAT), based on theoretical concepts and our empirical observations. Our code and models are publicly available[*] [†].

[*] https://github.com/ckarouzos/slp_daptmlm

[†] This is a joint work with Constantinos Karouzos (see [149,247])

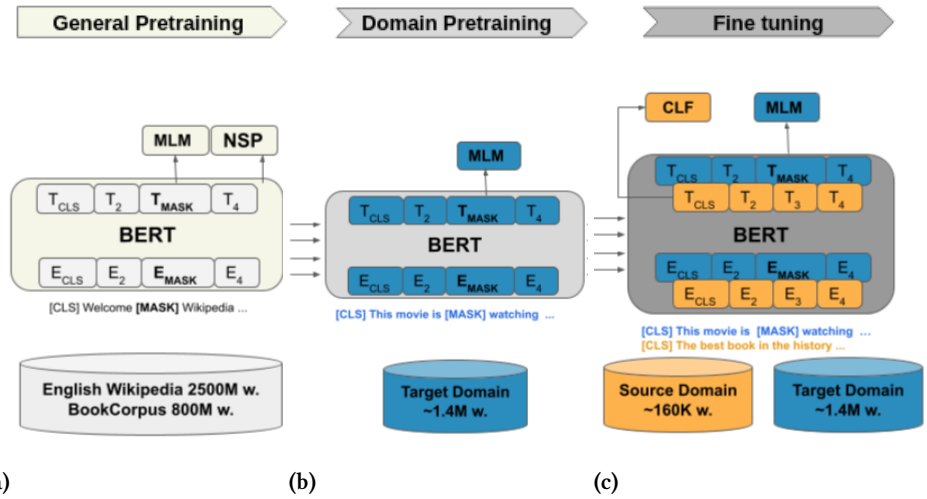


Figure 9.1: (a) BERT[10] is pretrained on English Wikipedia and BookCorpus with the MLM and the NSP tasks. (b) We continue the pretraining of BERT on unlabeled target domain data using the MLM task. (c) We train a task classifier with source domain labeled data, while we keep the MLM objective on unlabeled target domain data.

9.2 Proposed method

Fig. 9.1 gives an overview of the proposed UDALM framework. Starting from a model that is pretrained in general corpora (Fig. 9.1a), we keep pretraining it on target domain data using the masked language modeling task (Fig. 9.1b). On the final fine-tuning step (Fig. 9.1c) we update the model weights using both a classification loss on the labeled source data and Masked Language Modeling (MLM) loss on the unlabeled target data.

In Fig. 9.1a we see the BERT general pretraining phase. BERT[10] is based on the Transformer architecture[7]. During BERT pretraining, input tokens are randomly selected to be masked. BERT is trained using the MLM objective, which consists of predicting the most probable tokens for the masked positions. Additionally it uses a Next Sentence Prediction (NSP) loss, which classifies whether the pair of input sentences are continuous or not. If a labeled dataset is available, a pretrained BERT model can be fine-tuned for the downstream task in a supervised manner with the addition of an output layer.

In Fig. 9.1b we initialize a model using the weights of a generally pretrained BERT and continue pretraining on an unsupervised set of in-domain data, in order to adapt to the target domain. This step does not require use of supervised data, since we use the MLM objective.

For the final fine-tuning step, shown in Fig. 9.1c we perform supervised fine-tuning on the source data, while we keep the MLM objective on the target data as an auxiliary task. Following standard practice, we use the `[CLS]` token representation for classification. The classifier consists of a single feed-forward layer.

During this procedure the model learns the task through the classification objective using the labeled source domain samples, and simultaneously it adapts to the target domain data

through the MLM objective. The model is trained on the source domain labeled data for the classification task and target domain unlabeled data for the masked language modeling task. We mask only the target domain data. During training we interleave source and target data and feed them to the BERT encoder. Features extracted from the source data are then used for classification, while target features are used for MLM.

The mixed loss used for the fine-tuning step, is the sum of the classification loss L_{CLF} and the auxiliary MLM loss L_{MLM}. L_{CLF} is a cross-entropy loss, calculated on labeled examples from source domain, while L_{MLM} is used to predict masked tokens for unlabeled examples from target domain. We train the model over mixed batches, that include both source and target data, used for the respective tasks. The mixed loss is presented in Eq. 9.1:

$$L(\mathbf{s}, \mathbf{t}) = \lambda L_{CLF}(\mathbf{s}) + (1 - \lambda) L_{MLM}(\mathbf{t}) \tag{9.1}$$

We process n labeled source samples $\mathbf{s} \sim D_S$ and m unlabeled target samples $\mathbf{t} \sim D_T$ on a batch. The weighting factor λ is selected as the ratio of labeled source data over the sum of labeled source and unlabeled target data, as stated in Eq. 9.2:

$$\lambda = \frac{n}{n + m} \tag{9.2}$$

9.3 Experimental setup

9.3.1 Dataset

We evaluate UDALM on the Amazon reviews multi-domain sentiment dataset[248], a standard benchmark dataset for domain adaptation. Reviews with one or two stars are labeled as negative, while reviews with four or five stars are labeled as positive. The dataset contains reviews on four product domains: *Books* (B), *DVDs* (D), *Electronics* (E) and *Kitchen appliances* (K), yielding 12 adaptation scenarios of source-target domain pairs. Balanced sets of 2000 labeled reviews are available for each domain. We use 20000 (randomly selected) unlabeled reviews for (B), (D) and (E). For (K) 17805 unlabeled reviews are available. For each of the 12 adaptation scenarios we use 20% of both labeled source and unlabeled target data for validation, while labeled target data are used for testing exclusively and are not seen during training or validation.

9.3.2 Implementation details

We use `bert-base-uncased` as the Language Model on which we apply domain pretraining. The `bert-base-uncased` original English model is a 12-layer, 768-hidden, 12-heads, 110M parameter transformer, trained on the BookCorpus with 800M words and a version of the English Wikipedia with 2500M words. We convert source and target sentences to WordPieces[249]. For target sentences we randomly mask 15% of WordPiece tokens, as in[10]. If a token in a specific

Table 9.1: Accuracy of unsupervised domain adaptation on twelve domain pairs of Amazon Reviews Multi Domain Sentiment Dataset.

	R-PERL	DAAT	p+CFd	SO BERT	DAT BERT	DPT BERT	UDALM
$B \rightarrow D$	87.8	90.9	87.7	89.51 ± 0.76	87.31 ± 2.14	90.49 ± 0.38	$\mathbf{90.97 \pm 0.22}$
$B \rightarrow E$	87.2	88.9	91.3	90.51 ± 0.51	86.91 ± 2.71	90.38 ± 1.59	$\mathbf{91.69 \pm 0.31}$
$B \rightarrow K$	90.2	88.0	92.5	91.75 ± 0.28	90.59 ± 1.17	92.66 ± 0.43	$\mathbf{93.21 \pm 0.22}$
$D \rightarrow B$	85.6	89.7	**91.5**	90.26 ± 0.64	86.30 ± 3.10	91.02 ± 0.75	91.00 ± 0.42
$D \rightarrow E$	89.3	90.1	91.6	88.71 ± 1.48	87.85 ± 1.24	91.03 ± 0.82	$\mathbf{92.30 \pm 0.47}$
$D \rightarrow K$	90.4	88.8	92.5	91.22 ± 0.69	89.95 ± 1.53	92.30 ± 0.42	$\mathbf{93.66 \pm 0.37}$
$E \rightarrow B$	90.2	89.6	88.7	87.96 ± 0.89	85.65 ± 1.91	88.52 ± 0.55	$\mathbf{90.61 \pm 0.30}$
$E \rightarrow D$	84.8	**89.3**	88.2	87.37 ± 0.64	83.99 ± 1.31	87.85 ± 0.47	88.83 ± 0.61
$E \rightarrow K$	91.2	91.7	93.6	93.30 ± 0.50	92.45 ± 1.35	94.39 ± 0.72	$\mathbf{94.43 \pm 0.24}$
$K \rightarrow B$	83.0	**90.8**	89.8	88.15 ± 0.64	85.07 ± 1.03	88.83 ± 0.81	90.29 ± 0.51
$K \rightarrow D$	85.6	**90.5**	87.8	87.23 ± 0.49	84.11 ± 0.62	88.52 ± 0.69	89.54 ± 0.59
$K \rightarrow E$	91.2	93.2	92.6	93.23 ± 0.34	92.07 ± 0.24	93.42 ± 0.40	$\mathbf{94.34 \pm 0.26}$
Average	87.50	90.12	90.63	89.93 ± 0.65	87.68 ± 1.53	90.78 ± 0.67	$\mathbf{91.74 \pm 0.38}$

position is selected to be masked 80% of the time is replaced with a `[MASK]` token, 10% of the time with a random token and 10% of the time remains unchanged.

The maximum sequence length is set to 512 by truncation of inputs. During domain pretraining we train with batch size of 8 for 3 epochs (2 hours on two GTX-1080Ti cards). During the final fine-tuning step of UDALM we train with batch size 36, consisting of $n = 1$ source sub-batch of 4 samples and $m = 8$ target sub-batches of 4 samples each. We update parameters after every 5 accumulated sub-batches. We train for 10 epochs with early stopping on the mixed loss in Eq. 9.1. For all experiments we use AdamW optimizer[250] with learning rate 10^{-5}. Each adaptation scenario requires one hour on one GTX-1080Ti. For the domain adversarial experiments we set $\lambda_d = 0.01$ in Eq. 9.3 [‡] and train for 10 epochs. Models are developed with PyTorch[251] and HuggingFace Transformers[252].

9.3.3 Baselines — compared methods

We select three state-of-the-art methods for comparison. Each of the selected methods represents a different line of UDA research, namely the adversarial loss-based **BERT-DAAT**[177], self-training XLM-R-based **p+CFd**[163] and pivot-based **R-PERL**[169]. We report results for the following settings with BERT models:

Source-Only (SO): We fine-tune BERT on source domain labeled data, without using target data.

Domain Pre-Training (DPT): We use the target domain unlabeled data in order to continue pretraining of BERT with MLM loss (as in Fig. 9.1b) and then fine-tune the resulting model on source domain labeled data.

[‡]We also manually experimented with $\lambda_d = 1$ and $lambda_d = 0.1$, and a sigmoid schedule for λ_d. We report best results.

Domain Adversarial Training (DAT): Starting from the domain-pretrained BERT (see Fig. 9.1b), we then fine-tune the model with DAT as in Ganin et al.[56]. For a BERT model with parameters θ, with L_{CLF} being a cross-entropy loss for supervised task prediction, L_{ADV} being a cross-entropy loss for domain prediction and λ_d being a weighting factor, DAT consists of the minimization criterion described in Eq. 9.3.

$$\min_{\theta} L_{CLF}(\theta; D_S) - \lambda_d L_{ADV}(\theta; D_S, D_T) \tag{9.3}$$

UDALM: The proposed method, where we fine-tune the model created in the domain pretraining step using the mixed loss in Eq. 9.1.

9.4 Comparison to state-of-the-art

We present results for all 12 domain adaptation settings in Table 9.1. Results for SO BERT, DAT BERT, DPT BERT and UDALM are averaged over five runs and we include standard deviations The last line of Table 9.1 contains the macro-averaged accuracy and deviations over all domain pairs. UDALM surpasses all other techniques, yielding an absolute improvement of 1.81% over the SO BERT baseline. For fair comparison, we compare only with methods based on pretrained models, mostly BERT. We observe that BERT fine-tuned only with the source domain labeled data, without any knowledge of the target domain, is a competitive baseline. This source-only model even surpasses state-of-the-art methods developed for UDA, e.g. R-PERL[169].

We reproduce the DAT procedure and present results in the DAT BERT column of Table 9.1. Adversarial training proved to be unstable in our experiments, even after careful tuning of the adversarial loss weighting factor λ_d. This is evidenced by the high standard deviations in the DAT BERT experiments. We observe that adversarial training does not manage to outperform the source-only baseline.[§]

Domain pretraining increases the average accuracy with an absolute improvement of 0.85% over the source-only baseline. Continuing MLM pretraining on the target domain data leads to better model adaptation, and therefore improved performance, on the target domain. This is consistent with previous works on supervised[185–187] and unsupervised settings[177,188].

UDALM yields an additional 0.96% absolute improvement of average accuracy over domain pretraining. Keeping the MLM loss during fine-tuning therefore, leads to better adaptation and acts as a regularizer that prevents the model from overfitting on the source domain. We also observe smaller standard deviations when using UDALM, which indicates that including the MLM loss during fine-tuning can result to more robust training.

UDALM surpasses in terms of macro-average accuracy all other approaches for unsupervised domain adaptation on the Amazon reviews multi-domain sentiment dataset. Specifically, our method improves on the state-of-the-art pseudo-labeling (p+CFd[163]), domain adversarial (DAAT[177]) and pivot-based (R-PERL[169]) approaches by 1.11%, 1.62% and 4.24% respectively.

[§]Note that we did not have to perform extensive tuning for the other methods, including UDALM.

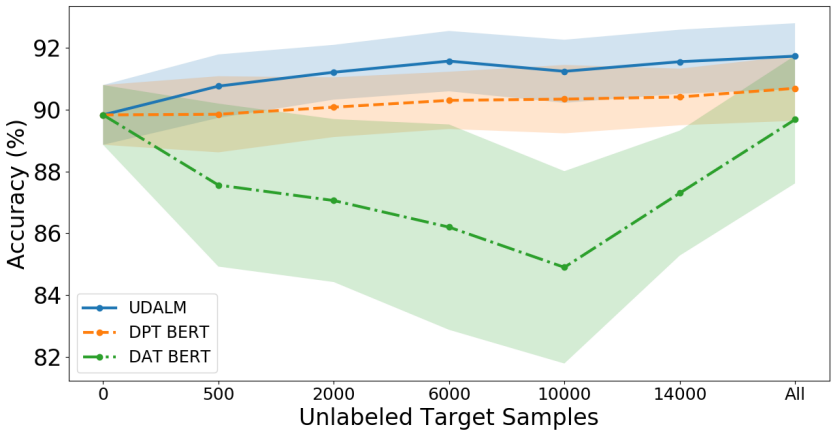


Figure 9.2: Average accuracy for different amount of target domain unlabeled samples of: (1) DPT BERT (2) DAT BERT and (3) UDALM.

9.5 Sample efficiency

We further investigate the impact of using different amount of target domain unlabeled data on model performance, to study the sample efficiency of UDALM. We experiment with settings of 500, 2000, 6000, 10000 and 14000 samples, by randomly limiting the number of unlabeled target domain data. For each setting we conduct three experiments with BERT models: (1) DPT, (2) DAT and (3) UDALM. When no target data are available, all methods are equivalent to a source only fine-tuned BERT. Again, we do not tune the hyper-parameters for DPT or UDALM. Fig. 9.2 shows the average accuracy on the twelve adaptation scenarios of the studied dataset. We see that UDALM produces robust performance improvement when we limit the amount of target data, indicating that it can be used in low-resource settings. However, training BERT in a domain adversarial manner shows instabilities. This is further discussed in Section 9.7.

Table 9.2: Comparison of average accuracy for various validation settings.

Stopping Criterion	Epochs	Av. Acc.
Fixed	1	90.98
Fixed	3	91.65
Fixed	10	**91.75**
Min source loss	10, patience 3	91.30
Min mixed loss	10, patience 3	**91.74**

9.6 On the stopping criteria for UDA training

A common problem when performing UDA is the lack of target labeled data that can be used for hyperparameter validation. For example, Ruder and Plank [160] use a small set of labeled target data for validation, putting the problem in a semi-supervised setting. When training under a domain shift, optimization of model performance on the source data may not result to optimal performance for the target data.

To illustrate this, we examine if the minimization of the mixed loss can be used as a stopping criterion for UDA training. We compare five stopping criteria: (1) fixed training for 1 epoch, (2) fixed training for 3 epochs, (3) fixed training for 10 epochs, (4) stop when the minimum classification loss is reached for the source data and (5) stop when the minimum mixed loss (Eq. 9.1) is reached. For (4) and (5) we train for 10 epochs with patience 3. We report average accuracy of the five stopping criteria over the twelve adaptation scenarios of Amazon Reviews dataset on Table 9.2. Training for a fixed number of 10 epochs and stopping when the minimum mixed loss perform best, yielding comparable accuracies of 91.75% and 91.73% respectively. Note that stopping when the minimum source loss stops the fine-tuning process too soon and does not allow the model to learn the target domain effectively. Overall, we observe that the mixed loss can be effectively used for early stopping, regularizing the model and alleviating the need for extensive search for the optimal number of training steps. This is an indication that the mixed loss could be used for model validation.

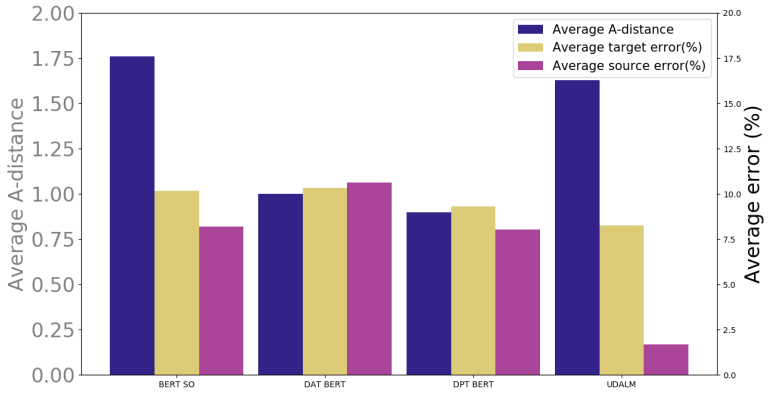


Figure 9.3: Comparison of average A-distance, average source error and average target error rate of different methods over all source - target pairs of the Amazon reviews dataset.

9.7 Discussion

9.7.1 Background Theory

Ben-David et al. [171],[172] provide a theory of learning from different domains. A key outcome of this work is the following theorem:

THEOREM [171,172] Let H be the hypothesis space and let D_S, D_T be the two domains and $\varepsilon_S, \varepsilon_T$ be the corresponding error functions. Then for any $h \in H$:

$$\varepsilon_T(h) \leq \varepsilon_S(h) + \frac{1}{2} d_{H\Delta H}(D_S, D_T) + C \tag{9.4}$$

where $d_{H\Delta H}(D_S, D_T)$ is the $H\Delta H$-divergence [253] between two domains, that is a measure of distance between domains that can be estimated from finite samples.

Eq. 9.4 defines an upper bound for the expected error $\varepsilon_T(h)$ of a hypothesis h on the target domain as the sum of three terms, namely the expected error on the source domain $\varepsilon_S(h)$, the divergence between the source and target domain distributions $\frac{1}{2} d_{H\Delta H}(D_S, D_T)$ and the error of the ideal joint hypothesis C. When such an hypothesis exists, the term is considered relatively small and in practice ignored. The first term, bounds the expected error on the target domain by the expected error in the source domain and is expected to be small, due to supervised learning on the source domain. The second term, gives a notion of distance between the source and target domain extracted features. Intuitively this equation states: "if there exists a hypothesis h that has small error on the source data and the source feature space is close to the target feature space, then this hypothesis will have low error on the target data". DAT aims to learn features that simultaneously result to low source error and low distance between target and source feature spaces based on the combined loss in Eq. 9.3.

9.7.2 A-DISTANCE ONLY PROVIDES AN UPPER BOUND FOR TARGET ERROR

According to Ben-David et al. [171] the $H\Delta H$-divergence can be approximated by proxy A-distance, that is defined by Eq. 9.5 given the domain classification error ε_D.

$$d_A = 2(1 - 2\varepsilon_D) \tag{9.5}$$

We calculate an approximation of the distance between domains. Following prior work [56,159] we create an SVM domain classifier. We feed the SVM with BERT's $[CLS]$ token representations, measure the domain classification error, and compute A-distance as in Eq. 9.5. We train the domain classifier on 2000 samples from each source and target domains. Fig. 9.3 shows the A-distance along with the source and the target error, averaged over the twelve available domain pairs using representations obtained from four methods, namely BERT SO, DAT BERT, DPT BERT and UDALM. DAT BERT minimizes the distance between domains. DPT BERT also reduces the A-distance, to similar levels with DAT, without using an explicit loss to minimize A-distance. To our surprise we found that, although it achieves the lowest error rate, UDALM does not significantly reduce the proxy A-distance compared to the source-only baseline. Additionally, we observe that the source error is correlated to model performance on the target task, i.e. models with lower source error have also lower target error. UDALM specifically, achieves high accuracy on the source task and is able to transfer the task knowledge across domains, while DAT is able to bring domain representations closer, but at the cost of achieving weaker performance on the task at hand.

Overall, we do not observe a correlation between the resulting A-distance and model performance on target domain. Therefore, lower distance between domains, achieved intentionally or not, is not a necessary condition for good performance on the target domain[¶], and our efforts could be better spent towards synergistic learning of the supervised source task and the target domain distribution.

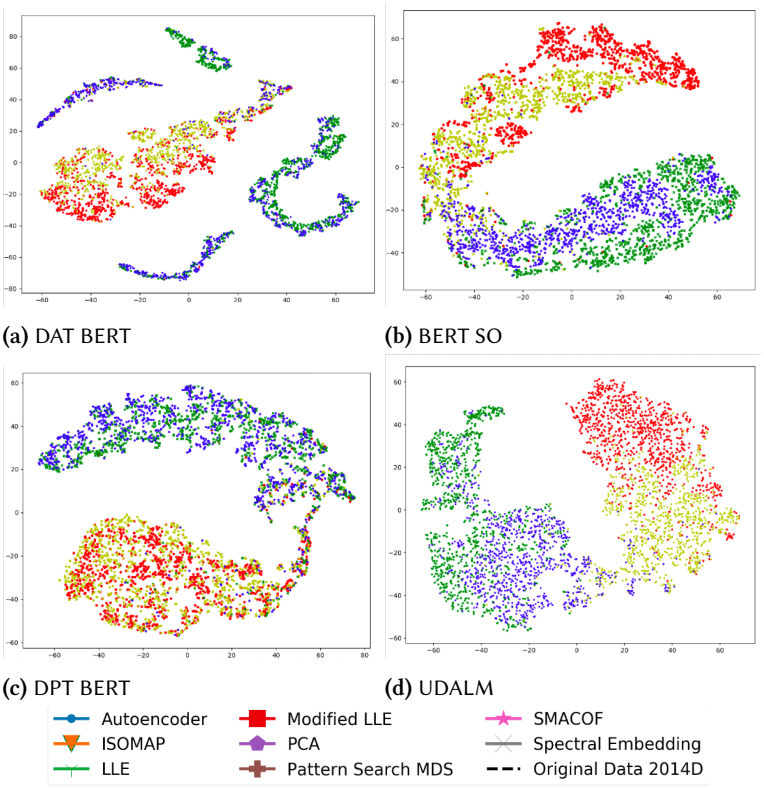


Figure 9.4: $2D$ representations of BERT $[CLS]$ features using t-SNE for the $D \rightarrow K$ task. The goal is to maximize separation between target positive (blue) and target negative (yellow) samples.

9.7.3 Limitations of Domain Adversarial Training

DAT[56] faces some critical limitations that make the method difficult to be reproduced due to high hyper-parameter sensitivity and instability during training.

Such limitations have been highlighted by other authors in the UDA literature. For example, according to Shen et al.[181] when a domain classifier can perfectly distinguish target from source representations, there will be a gradient vanishing problem. Shah et al.[255] state

[¶]Shu et al.[254] state that feature distribution matching is a weak constraint when high-capacity feature extractors are used. Intuitively, a high-capacity feature extractor can perform arbitrary transformations to the input features in order to match the distributions.

that DAT is unstable and needs careful hyper-parameter tuning for their experiments. Wang et al.[256] report results over three multi-domain NLP datasets, where DAT in conjunction with BERT under-performs. Ruder and Plank[160] found that the domain adversarial loss did not help for their experiments on the Amazon reviews dataset.

In our experiments we note that domain-adversarial training results to worse performance than naive source only training. Furthermore, we experienced the need for extensive tuning of the λ_d parameter from Eq. 9.3 every time the experimental setting changed (e.g. when testing for different amounts of available target data as in Section 9.5). This motivated us to further investigate the behavior of BERT fine-tuned with the adversarial cost. For visual inspection, we perform T-SNE[257] on representations extracted from BERT, under four UDA setings in Fig. 9.4. In Fig. 9.4a we observe features extracted using BERT with DAT and we compare it with features from SO BERT (Fig. 9.4b), DPT BERT (Fig. 9.4c) and UDALM (Fig. 9.4d). We observe that DAT manages to group tightly target and source samples, especially in the case of positive samples. Nevertheless, in the process, DAT introduces significant distortion in the semantic space, which is reflected in model performance[‖].

We can attribute this behavior to two factors. First, The formulation of the adversarial loss in Eq. (9.3) can lead to trivial solutions. In order to maximize the L_{ADV} term of Eq. (9.3), the model can just flip all domain labels, namely just predict that source samples belong to the target domain and vice-versa. In this case the model can still discriminate between domains and domain-independent representations are not encouraged. We empirically observed this behavior in our experiments with DAT, and only extensive hyper-parameter tuning could alleviate this issue. Additionally, Eq. (9.3) aims to minimize the upper bound of the target error $\varepsilon_T(h)$ in Eq. (9.4). While this is desirable, reduction of the upper bound does not necessarily result in reduction of the bounded term in all scenarios. Furthermore, optimizing the $L_{ADV}(\theta; D_S, D_T)$ term can lead to increasing $L_{CLF}(\theta; D_S)$, and therefore one must find a balance between the two adversarial terms, again through careful hyper-parameter tuning. These issues could potentially be alleviated by including regularization terms that discourage trivial solutions and improve robustness. Therefore, given the lack of guarantees for good performance and the practical considerations, further investigation should be conducted regarding the robustness and reproducibility of DAT for UDA.

9.7.4 Conclusions and Future Work

Unsupervised Domain Adaptation of pretrained language models is a challenging problem with direct real world applications. In this work we propose UDALM, a robust, plug and play training strategy, which is able to improve performance in the target domain, achieving state-of-the-art results across 12 adaptation settings in the multi-domain Amazon reviews dataset. Our method produces robust results with little hyper-parameter tuning and the proposed mixed-loss can be used for model validation, allowing for fast model development. Furthermore, UDALM scales with the amount of available unsupervised data from the target domain,

[‖]Note, we include this visualization for a single source-domain pair as an example. We performed multiple runs of T-SNE over all 12 source-domain pairs and this behavior appeared consistently.

allowing for adaptation in low-resource settings. In our analysis, we discuss the relationship between the A-distance and the target error. We observe that low A-distance may not suggest low target error for high capacity models. Additionally, we examine limitations of Domain Adversarial Training and highlight that the adversarial cost may lead to distortion of the feature space and negatively impact performance.

In the future we plan to apply UDALM to other tasks under domain-shift, such as sequence classification, question answering and part-of-speech tagging. Furthermore, we plan to extend our method for temporal and style adaptation, by adding more relevant auxiliary tasks that model language shift over time and over different platforms. Finally, we want to investigate the effectiveness of the proposed fine-tuning approach in supervised scenarios.

10

Sample-Efficient Unsupervised Domain Adaptation of Speech Recognition Systems

10.1 Introduction

Automatic Speech Recognition (ASR) models have matured to the point where they can enable commercial, real-world applications, e.g., voice assistants, dictation systems, etc., thus being one of machine learning's success stories. However, the performance of ASR systems rapidly deteriorates when the test data domain differs significantly from the training data. Domain mismatches can be caused by differences in the recording conditions, such as environmental noise, room reverberation, speaker and accent variability, or shifts in the target vocabulary. These issues are made more prominent in the case of low-resource languages, where diversity in the training data is limited due to the poor availability of high-quality transcribed audio. Therefore, specialized domain adaptation approaches need to be employed when operating under domain shift.

Unsupervised Domain Adaptation (UDA) methods are of special interest, as they do not rely on expensive annotation of domain-specific data for supervised in-domain training. In contrast to supervised approaches, where the existence of labeled data would allow to train domain-specific models, UDA methods aim to leverage data in the absence of labels to improve system performance in the domain of interest[56,258]. In the context of speech recognition, the importance of UDA is extenuated, as the transcription and alignment process is especially expensive and time-consuming. Adaptation methods have been explored since the early days of ASR, at different levels of the system and different deployment settings[259]. UDA has been used to improve the robustness of ASR on a variety of recording conditions including far-field speech, environmental noise, and reverberation[195,196,204]. Furthermore, UDA has been used for speaker adaptation, and to improve performance under speaker, gender, and accent variability[211,213]. UDA has also been employed for multilingual and cross-lingual ASR, to improve ASR models for low-resource languages[215], adapt to different dialects[198], and even train speech recognition systems for endangered languages[220].

Classical speech adaptation techniques involve feature-based techniques, e.g., speaker normalization[260], feature-based approaches[261–263], or multi-condition training[264]. Generally, traditional approaches require some knowledge about the target domain, and the domain mismatch, e.g., regarding the noise and reverberation variability[265], and require specific engineer-

ing for each adaptation scenario.

Modern ASR pipelines, increasingly rely on end-to-end neural networks, e.g.,[194,266], or large pre-trained models with self-supervised objectives[148,203]. The key approaches employed for UDA of end-to-end ASR models can be grouped into three categories, namely, teacher-student learning[198], domain adversarial training [267], and target domain self-supervision [190]. The benefit of these techniques is that they do not require any special knowledge about the source or the target domain. This makes end-to-end UDA approaches versatile and able to be utilized in a larger array of adaptation scenarios. In particular, adaptation through self-supervision is a robust, simple, and efficient technique for adaptation of state-of-the-art speech models[216].

We leverage in-domain self-supervision to propose Mixed Multi-Domain Self-Supervision (M2DS2), a fine-tuning strategy enabling sample-efficient domain adaptation of wav2vec2[203] based speech recognition models, even when available in-domain data are scarce. Our key contributions are organized as follows:

1. Inspired by recent advances in UDA for NLP systems[149], we propose a fine-tuning strategy for speech models, where the self-supervised objective is based on a contrastive loss in Section 10.2. While prior works leverage only in-domain self-supervision, we find that mixed source and target domain self-supervision is essential in our setting, to avoid the mode-collapse of latent representations. We demonstrate this empirically in Section 10.6.2.

2. We collect and curate HParl, the largest publicly available* speech corpus for Greek, collected from plenary sessions in the Greek Parliament between 2018 and 2022. We establish a data collection, pre-processing, and alignment pipeline that can be used for continuous data integration, as the parliamentary proceedings get regularly uploaded. We provide a detailed description of our data collection process and the dataset statistics in Section 10.3.1. HParl is merged in Section 10.3 with two popular Greek corpora (Logotypografia and CommonVoice) to create GREC-MD, a testbed for multi-domain evaluation of ASR systems in Greek.

3. We demonstrate that, while other baselines fail at UDA in our resource-constrained setting, M2DS2 can improve model performance in the target domain in multiple adaptation scenarios in Section 10.6. Specifical emphasis is given to the sample efficiency of our approach in Section 10.6.1, where we demonstrate successful adaptation even when we reduce the available in-domain data.

4. When we relax the problem to a weakly supervised adaptation setting, where some in-domain text is available but the pairing between audio and text is unknown, we find that M2DS2 can be effectively combined with simple N-gram adaptation techniques to

*HParl is publicly available under the CC-BY-NC 4.0 license: http://hdl.handle.net/11500/CLARIN-EL-0000-0000-7665-A. The other corpora used in this work are available through their respective distributors.

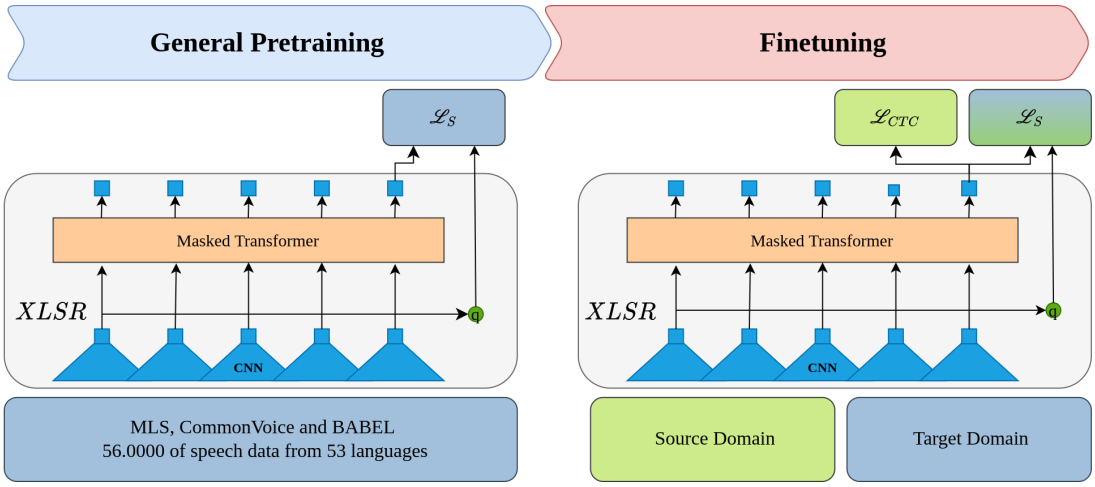


Figure 10.1: Target-domain adaptation through self-supervision. On the left, we see the general pre-training stage of XLSR-53 using the self-supervised loss L_s. General pre-training is performed on 56, 000 hours of audio in 53 languages. On the right, we see the proposed domain-adaptive fine-tuning stage, where the speech recognition task is learned using transcribed source domain data, while adaptation to the target domain is performed by including the self-supervised loss over (audio-only) source and target domain data

get comparable performance with the fully supervised baseline in Section 10.7. Furthermore, we find that a simple text augmentation approach, based on perplexity filtering of a large corpus can produce strong adaptation results, even for small amounts of in-domain text.

Additionally, we provide detailed experimental settings for reproducibility in Section 10.4, and an upper-bound estimation for UDA performance with fully supervised fine-tuning in Section 10.5.

10.2 Domain adaptation through multi-domain self-supervision

The proposed approach is based on end-to-end adaptation of a large pre-trained speech model during the fine-tuning phase, by including in-domain self-supervision. We extend UDALM[149], which has shown promise for NLP tasks, for adaptation of wav2vec2-based acoustic models, and specifically XLSR-53. We focus on the problem of UDA in the context of a low-resource language, i.e., Greek. The key finding of our exploration is that straight-forward extension of UDALM, i.e., by using only target domain self-supervision, underperforms in this setting, and use of both source and target domain data is essential for successful adaptation. In this section, first, we will present a quick overview of the XLSR-53 training procedure, and then we are going to outline the proposed domain adaptation approach, which is shown in Fig. 10.1.

Table 10.1: The GREC-MD corpus. We can see the duration of each split in `hours:minutes:seconds` format, as well as the number of speakers for each of the sub-corpora.

Dataset	Domain	Speakers	Train	Dev	Test	Total Duration
HParl	Public (political) speech	387	99:31:41	9:03:33	11:12:28	119:47:42
CV	Crowd-sourced speech	325	12:16:17	1:57:44	1:59:19	16:13:20
Logotypografia	News casts	125	51:58:45	9:08:35	8:59:22	70:06:42
Total	-	713	163:46:43	20:09:52	22:11:44	206:08:19

10.2.1 XLSR-53

XLSR-53[148] is a massively pre-trained speech model, trained on $56,000$ hours of multilingual speech, covering 53 languages. The model is based on wav2vec2[203], which is composed of a multi-layer convolutional feature encoder, that extracts audio features z_t from the raw audio, and a transformer context encoder that maps the latent audio features to the output hidden states c_t. Each latent feature z_t corresponds to 25 ms of audio with stride 20 ms. A contrastive objective L_c is used for pre-training. For this, product quantization[268] is applied to the features z_t, and then a discrete approximation of z_t is obtained by sampling from a Gumbel-softmax distribution[269], to obtain discrete code vectors q_t, organized into $G = 2$ codebooks with $V = 320$ vocabulary entries each. The contrastive loss aims to identify the correct code vector for a given time step, among a set of distractors Q_t, obtained through negative sampling from other timesteps. To avoid mode collapse, a diversity loss L_d is included by maximizing the entropy over the averaged softmax distribution over the code vector entries $\bar{p}_g$. The total loss is:

$$L_s = \underbrace{-log\frac{e^{s(z_t,q_t)}}{\sum_{\tilde{q}\sim Q_t} e^{s(z_t,\tilde{q})}}}_{\text{Contrastive Loss}} \overbrace{-\frac{1}{GV}\sum_{g=1}^{G}\sum_{v=1}^{V}\bar{p}_{g,v}log(\bar{p}_{g,v})}^{\text{Diversity Loss}} \tag{10.1}$$

10.2.2 Domain adaptive fine-tuning for contrastive learning of speech representations

Fig. 10.1 shows the proposed fine-tuning process. The key intuition is that we want the model to synergistically learn the task at hand (in our case ASR) while being adapted to the target domain by in-domain self-supervision. On the left, we see the general pre-training stage of XLSR-53, which is pre-trained on 56K hours of multilingual audio corpora using the contrastive pre-training objective. On the right, we see the proposed fine-tuning stage.

During fine-tuning, we form a mixed objective function:

$$L = L_{CTC}(x_s, y_s) + \alpha L_s(x_s) + \beta L_s(x_t), \tag{10.2}$$

where $(x_s, y_s) \sim \mathcal{S}(x, y)$, $x_t \sim \mathcal{T}(x)$, L_{CTC} is the Connectionist Temporal Classification (CTC) objective function, optimized using transcribed source domain data, and L_s is the con-

trastive loss from Eq. (10.1). We scale the contribution of each term using hyper-parameters α and β.

Note that contrary to Karouzos et al.[149], who use only in-domain self-supervision, we leverage both source and target domain samples for the mixed self-supervision. We find that this is essential in our case to avoid mode collapse, i.e., the model using only a few of the available discrete code vectors. Simultaneous self-supervision on both the source and target data alleviates mode collapse by anchoring the target code vector space to have a similar structure as the source code vectors.

10.3 The GREC-MD corpus

For our experiments we compose a speech corpus for the Greek language, that is suitable for multi- and cross-domain evaluation. The GREC-MD corpus contains 206 hours of Greek speech. Audio is segmented into individual utterances and each utterance is paired with its corresponding transcription. Table 10.1 summarizes the included sub-corpora, as well as the train, development, and test splits. The dataset is constructed with three core principles in mind:

1. **Data Volume**: We collect the largest publicly available speech recognition corpus for the Greek language, able to scale to hundreds of hours of transcribed audio.

2. **Temporal Relevance**: Language changes over time. We aim at an up-to-date corpus that encompasses the latest terms and topics that appear in daily speech.

3. **Multi-Domain Evaluation**: Single domain evaluation can lead to misleading estimations of the expected performance for ASR models. For example, state-of-the-art ASR models[193] achieve under 5% Word Error Rate (WER) on Librispeech [270] test sets, but this is an over-estimation of system performance in the field. This is extenuated when considering different acoustic conditions or terminology. We consider multi-domain evaluation essential when developing and deploying real-world ASR models. This is further supported by the ablations performed by Likhomanenko et al.[271], who observe that the average WER over multiple test sets is a good proxy metric for real-world ASR performance, and Chan et al.[272] who achieve significant WER reduction through multi-domain training.

To satisfy the first two points, we collect data from a public, continuously updated resource, i.e., the Hellenic Parliament Proceedings, where recordings of the parliamentary sessions are regularly uploaded. We refer to this corpus as HParl. The benefit of using this resource is the straightforward collection of a continuously growing, multi-speaker corpus of transcribed audio that is always up-to-date, as the parliamentary discussions revolve around current affairs. This approach is established in the literature for the creation of open speech corpora for multiple languages, by collecting plenary data from the Japanese[273], Finnish[274], Czech[275], Danish[276], and European[277] parliaments.

For the multi-domain evaluation, we merge HParl with two publicly available corpora, that have different acoustic and language characteristics. We refer to the merged, multi-domain corpus as GREC-MD. In this Section, we will describe the collection and curation process of HParl, and present the relevant statistics for the experiments.

Table 10.2: Plenary sessions included in HParl. The Hours column refers to the raw (unsegmented) hours of collected audio.

Start date	End date	#Sessions	Hours
15-02-2022	01-03-2022	10	55
18-01-2019	01-02-2019	10	52
28-03-2019	10-05-2019	20	108
10-12-2018	21-12-2018	10	88

10.3.1 Collection and curation of HParl

Modern technological advances allow for more direct government transparency, through the commodification of storage and internet speeds. In this spirit, the records of plenary sessions of the Hellenic Parliament are made publicly available, for direct access through a webpage[†]. The available video recordings date back to 2015. For each plenary session, a video recording is uploaded, along with a full transcription that is recorded verbatim, and in real-time by the parliament secretaries. For the creation of HParl, we have built a web crawler that can traverse and download the video recordings, along with the transcriptions from the official website. The collection process is parallelized over multiple threads and parameterized by a range of dates and, optionally, a target corpus size in GB or hours. For this version of HParl, we collect the plenary sessions in four date ranges, as described in Table 10.2. The majority of the collected sessions are from 2019, but we also include sessions from 2018 and 2022 to include coverage of different topics. The individual components of the HParl curation pipeline are Audio Pre-processing, Text Pre-processing, Alignment, Post-processing, and dataset Splitting.

Audio pre-processing

Fig. 10.2 shows the layout of the Hellenic Parliament Chamber. Plenary sessions mainly take place in this room, or in the secondary House Chamber which has a similar setup but is smaller in size. Because of the room and microphone characteristics, the captured audio in the video streams contains reverberation, due to sound reflections. We pass the input video streams through FFmpeg and convert them to monophonic, lossless audio format at 16000 Hz sampling rate. The resulting audio is not passed through any de-reverberation or speech enhancement software. The resulting audio files have a minimum, average, and maximum duration of 6 minutes, 6 hours, and 16 hours respectively.

[†]https://www.hellenicparliament.gr/en/

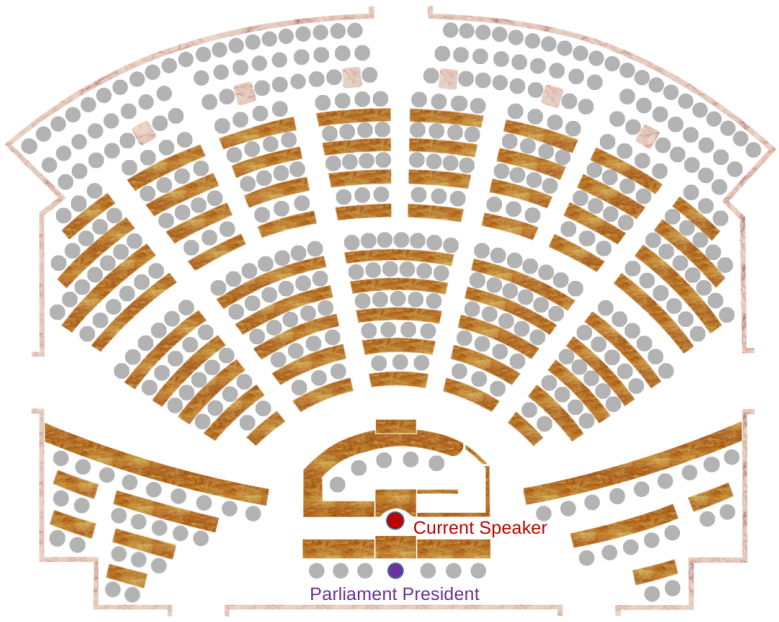


Figure 10.2: Overview of the Hellenic Parliament Chamber. The chamber has an amphitheatrical shape and can accommodate approximately 400 — 450 people. The positions of the key speakers, i.e., the current speaker and the parliament president are annotated in the image.

Text pre-processing

The text files contain full, word-by-word transcription of the speeches and questions asked by members of the audience, as well as extra annotations made by the parliament secretaries. Some annotations are relevant, i.e., the speaker name, while others are plain text descriptions of events happening during the session and need to be filtered out (e.g., "The session is interrupted for a 15-minute break"). We use a rule-based system, based on regular expressions, that filters the unnecessary information, keeping only the transcriptions and the speaker names. The speaker labels are created by transliterating their names and roles from Greek to Greeklish using the "All Greek to Me!" tool[278]. Text is lower-cased and normalized to remove multiple whitespaces. The result is a text file containing the raw transcriptions, and a mapping from speaker labels to their respective text parts.

Aligment and segmentation

The primary challenge of exploiting the plenary sessions for ASR purposes is the length of the plenary recordings, as their durations vary from 6 minutes to 16 hours in length. However, data samples used to train ASR are generally less than 30 seconds long. Computational challenges have limited the length of training utterances for HMM-GMM models[279], and continue to do so in contemporary neural network models. Therefore, we need to segment the sessions into smaller pieces more suitable for ASR training. A second challenge is posed by mismatches

between audio and transcripts. Parliamentary proceedings do not fully capture everything that is said during the parliamentary sessions, and do not account for speech disfluencies.

To obtain smaller, clean segments, that are suitable for ASR training we follow the segmentation procedure proposed by[280]. Initially, the raw recordings are segmented into 30 second segments and the transcriptions are split into smaller segments of approximately 1000 words called *documents*. Each segment is decoded using a seed acoustic model trained on the Logotypografia corpus[281] and a 4-gram biased LM trained on the corresponding transcription of each recording. The best path transcript of each segment is obtained and paired with the best matching *document* via TF-IDF similarity. Finally, each hypothesis is aligned with the transcription using Smith-Waterman alignment[282] to select the best matching sub-sequence of words. The above method yields a list of text utterances, with their corresponding start and end times in the source audio files. The procedure yields 120 hours of useable segmented utterances out of the original 303 hours of raw audio, or a ratio of 39.6%.

Post-processing

After the segments are extracted, we filter out extremely short segments (less than 2 words). Moreover, the iterative alignment algorithm may replace some intermediate words with a `<spoken-noise>` tag. When this tag is inserted, we match the surrounding text with the raw transcriptions and re-insert the missing words. Furthermore, we match each segment to its corresponding speaker label. Segments without a speaker label are discarded. Lastly, speakers are associated with their gender based on name suffixes, using a simple, Greek language-specific, rule: Speaker names which end in a(α), h(η), w(ω) or is(ις) are classified as female, while the rest as male. We format the segments, speaker, and gender mappings in the standard folder structure used by the Kaldi speech recognition toolkit[208].

Table 10.3: Dominant topic words for each dataset split.

Dataset	Topic words (Top-10)
HP Train	New Democracy, work, growth, government, country, time, policy, reality, agreement, development
HP Dev	investments, New Democracy, Greeks, government, problem, national, reality, reason, development, economic
HP Test	state, New Democracy, funding, citizens, work, contract, department, government, public, function
LG Train	president, position, problem, decision, time, subject, reason, policy, group, opinion
LG Dev	time, position, problem, measure, energy, government, subject, region, policy, percent
LG Test	percent, space, agreement, policy, subject, semitics, country, state, officials, conference
CV Train	prince, unnecessary, truth, six, no more, soldiers, eyes, man, work, door
CV Dev	words, compose, women, took, bad guy, prince, eyes, door, deaf, son
CV Test	ships, bad, brother, prince, Irine, paper, cervix, deaf, course, observations

Data splitting

We provide an official train - development - test split. The development set contains 3 plenary sessions, one from 2018, one from 2019, and one from 2022, resulting in 9 hours of segmented speech. Similarly, the test set contains one session from each year, resulting in 11 hours of segmented speech. The rest 99 hours of segmented speech are assigned to the training set.

10.3.2 Including corpora from different domains

We merge HParl with two publicly available corpora to create GREC-MD for multi-domain evaluation.

Common Voice

Common Voice (CV)[283] is a crowd-sourced, multi-lingual corpus of dictated speech, created by Mozilla. The data collection is performed by use of a web app or an iPhone app. Contributors are presented with a prompt and are asked to read it. The prompts are taken from public domain sources, i.e., books, Wikipedia, user-submitted prompts, and other public corpora. The maximum prompt length is 15 words. A rating system is built into the platform, where contributors can upvote or downvote submitted `<audio,transcript>` pairs. A pair is considered valid if it receives two upvotes. Speaker-independent training, development, and test splits are provided. The dataset is open to the research community, and released under a permissive Creative Commons license (CC0). In this work, we use version 9.0 of CV, accessed on April 27, 2022. We keep only the valid utterances, i.e., 16 hours of speech from 325 contributors ($19-49$ years old, 67% male / 23% female).

Table 10.4: Speaker overlap between the splits for each corpus.

Dataset	# Speakers			# Overlapping Speakers		
	Train	Test	Dev	Train-Test	Train-Dev	Dev-Test
HP	378	82	64	77	60	16
LG	78	18	16	0	1	1
CV	110	204	25	79	24	1

Logotypografia

Logotypografia[281] is one of the first corpora for Large Vocabulary Continuous Speech Recognition in Greek. The dataset contains 33,136 newscast utterances or 72 hours of speech. The utterances were collected from 125 speakers (55 male, 70 female), who were staff of the popular "Eleftherotypia" newspaper in Greece, under varied acoustic conditions. Approximately one third of the utterances were collected in a soundproof room, one third in a quiet room, and the last third in an office room. The average utterance duration is 7.8 seconds. The transcriptions contain several speech and non-speech events (e.g., `<cough>`), lower-cased Greek words, and stress marks. Numbers are expanded to full words. We use the whole dataset, and perform light preprocessing in the transcriptions, by discarding the annotated events and punctuation.

We hence refer to each dataset by the abbreviations: HParl: HP, CommonVoice: CV, Logotypografia: LG.

In Table 10.3 we show the dominant topic words for the train, development, and test splits of HP, LG, and CV. For this, we train a Latent Dirichlet Allocation[284] model for each subcorpus

and isolate four dominant topics per subcorpus. We select the top ten topic words from the dominant topics, excluding stop-words. In Table 10.4 we can see the speaker overlap between the data splits of HP, LG, and CV. LG is based on a strict speaker-independent split, while for HP we opt for a balanced split. CV also contains significant speaker overlap between train and test splits.

10.4 Experimental settings

For our experiments, we use the following hyper-parameter settings, unless explicitly stated otherwise. For model training, we use AdamW optimizer[250] with a learning rate of 0.0003. We apply warmup for the first 10% of the maximum training steps, and a linear learning rate decay after that. Models are fine-tuned for a maximum of 10000 steps. For speech recognition training, we make use of the CTC loss[151], optimized using the available transcribed data in each scenario. Validation runs every 500 steps on the development set, and early stopping is employed on the development CTC loss with patience 5. Batch size is set to 8 during fine-tuning for all scenarios, except for M2DS2. In the case of M2DS2, we create mixed batches of size 12, containing 4 transcribed source domain samples and 8 unlabeled target domain samples and train for $10,000$ CTC updates. For memory reasons, we split the mixed batches into mini-batches of 4 and interleave them during model training. Gradients are accumulated over 3 interleaved batches. For the self-supervised objective, we create masks of maximum timestep length 10, with masking probability 0.4. We weigh the contributions of the source and target domain contrastive objectives, and bring them to the same order of magnitude as the CTC loss, by setting $\alpha = 0.01$ and $\beta = 0.02$. The convolutional feature encoder is kept frozen for all experiments. Our code is based on the huggingface [‡] implementation of XLSR-53. For all experiments, we resample the audio files to 16 kHz and downsample to single-channel audio. We exclude utterances in the training set that are longer than 12 seconds. All experiments are run on a single NVIDIA RTX 3090 GPU, with mixed precision training.

For the Language model training, we create a large corpus for the Greek language using a subset of the Greek part of CCNet[285] (approximately 11 billion tokens) and combine it with 1.5 billion tokens from the Greek version of Wikipedia and the Hellenic National Corpus[286]. During preprocessing, we remove all punctuation and accents, deduplicate lines, and convert all letters to lowercase. We will refer to this corpus as the Generic Greek Corpus (GGC). We train a 4-gram language model on GGC using KenLM[287] and prune bigrams, trigrams, and four-grams with counts less than 3, 5 and 7 respectively. We incorporate the n-gram LM at inference time using the pyctcdecode framework[§]. We use language model rescoring over a beam search decoder with 13 beams.

The evaluation metric is the Word Error Rate (WER) over the target test set. For assessing the adaptation effectiveness, we also report the WER recovery of the UDA against fully supervised training[288], which is defined in Eq. (10.3):

[‡] https://huggingface.co/docs/transformers/
[§] https://github.com/kensho-technologies/pyctcdecode

$$WRR = \frac{WER_{unadapted} - WER_{UDA}}{WER_{unadapted} - WER_{supervised}} \times 100\% \tag{10.3}$$

The metric in Eq. (10.3) measures the relative improvement obtained by a UDA approach, compared to the fully supervised baseline, taking into account the difficulty of each scenario. We refer to this metric as Word error Rate Recovery (WRR) for the rest of this paper.

Table 10.5: ASR performance of XLSR-53 over the three corpora for fully supervised in-domain fine-tuning (WER). Left: Decoding without LM, Right: Decoding with 4-gram LM trained on GGC

Dataset \ LM	No LM	4-gram LM
HP	26.21	15.64
CV	29.33	9.52
LG	31.94	26.45

10.5 Supervised in-Domain training

In the first set of experiments, we explore the performance of supervised fine-tuning of XLSR-53 for each domain. This will give an upper-bound estimation for UDA performance. We fine-tune XLSR-53 on CV, HP, and LG (separately) and perform in-domain evaluation on the respective test sets. Results are summarized in Table 10.5. The first column indicates the performance of greedy decoding, while in the second column, we report the performance of the beam search decoder, rescored using the scores of the 4-gram GGC language model. We observe that the greedy decoding performance is under 30 WER for both HP and CV, while for LG we achieve $\sim$ 32 WER. This makes sense, as LG is the most diverse dataset, with respect to the included acoustic conditions. Furthermore, we observe that the incorporation of a language model results in an impressive WER reduction on CV, followed by HP and then LG. While CV includes relatively simple phrases with a common vocabulary, HP and LG contain more specialized terminology.

10.6 Unsupervised domain adaptation using in-domain audio

Here, we evaluate the effectiveness of M2DS2 for UDA. We compare with three baselines:

1. **Source-Only (SO):** We perform supervised fine-tuning of XLSR-53 (CTC) using only the source-domain data and run decoding on the target domain test set. No in-domain data are used for adaptation.

2. **Continuous Pre-Training (CPT):** We perform a pre-training phase using the loss in Eq. (10.1) on the source and target domain train sets, to create adapted versions of XLSR-53. Pre-training is run for 20000 steps with batch size 4. Only the audio is used, without

Table 10.6: M2DS2 performance using greedy decoding for UDA between HP, CV, and LG. A → B indicates that A is the source domain and B is the target domain. (G) indicates greedy decoding. (LM) indicates beam search with LM rescoring. We report the WER on the target test set, as well as the WRR (%) over the SO, i.e., unadapted, baseline. WER: lower is better. WRR: higher is better.

Method	SO (G)	CPT (G)		PSL (G)		M2DS2 (G)		SO (LM)	CPT (LM)		PSL (LM)		M2DS2 (LM)	
Setting	WER	WER	WRR	WER	WRR	WER	WRR	WER	WER	WRR	WER	WRR	WER	WRR
HP → CV	55.90	54.80	4.1	53.48	9.1	**52.95**	**11.1**	25.26	23.26	12.7	24.34	5.9	**18.35**	**43.9**
HP → LG	48.65	47.99	4.0	51.75	−18.6	**46.47**	**12.5**	30.34	33.88	−91.0	31.92	−40.6	**29.56**	**20.1**
LG → CV	59.57	60.81	−4.1	63.28	−12.3	**51.31**	**27.3**	25.96	29.10	−19.1	23.46	15.2	**17.30**	**52.7**
LG → HP	62.13	60.60	4.3	66.60	−12.4	**60.09**	**5.7**	31.48	31.54	−0.4	39.15	−48.4	**31.36**	**0.8**
CV → LG	69.55	68.98	1.5	68.29	3.4	**63.40**	**16.4**	50.80	47.61	13.1	42.53	34.0	**36.93**	**57.0**
CV → HP	70.72	71.79	−2.4	69.68	2.3	**68.70**	**4.5**	52.09	48.14	10.8	53.8	−4.7	**41.88**	**28.0**

transcriptions. The adapted checkpoints are then fine-tuned by the use of CTC loss on the source domain transcribed data. Evaluation is performed on the target test set.

3. **Pseudo-Labeling (PSL):** We fine-tune XLSR-53 using the source domain data with CTC loss. Then we run inference on the source model, to extract silver transcriptions for the target domain training set. Using the silver transcriptions produced in the first step, we create the pseudo-labeled target train set, and merge it with the correctly transcribed source domain train set. Finally, we re-initialize the model to the original XLSR-53 weights and fine-tune it on the combined training corpus.

In Table 10.6 we compare M2DS2 with the SO, CPT, and PSL baselines for six adaptation scenarios, i.e., cross dataset evaluation between the three datasets in GREC-MD. The left half corresponds to greedy decoding, while for the right half, we use the 4-gram LM trained on GGC. First, we observe the SO model performance. The SO models are the fine-tuned models from Table 10.5, evaluated in out-of-domain settings. We see that out-of-domain evaluation results in a large performance hit, e.g., while in the CV → CV in-domain setting we achieve 29.33 WER, in the CV → LG out-of-domain setting we get 69.55 WER. This confirms that for real-world ASR tasks, multi-domain evaluation is of the essence. Second, we observe that in most adaptation scenarios both CPT and PSL fail to surpass the SO (unadapted) baseline. In the case of CPT, we hypothesize that this is due to the relatively data-constrained version of our setting. In the best-case scenario, we have 99 hours of available target domain audio, which is not enough to perform a discrete CPT stage. Note that most of the works in the literature use $\sim$ 1000 hours of target audio for CPT. In the case of PSL, the poor performance is due to the quality of the silver labels created by the seed model. While the performance would improve with more elaborate approaches (e.g., confidence filtering), in challenging adaptation scenarios PSL approaches are limited by the SO model's performance. Lastly, we observe that M2DS2 is the only approach among our baselines that manages to achieve a positive WRR in most adaptation scenarios, by consistently outperforming the SO baseline by significant margins. This is exaggerated when we include an LM during inference.

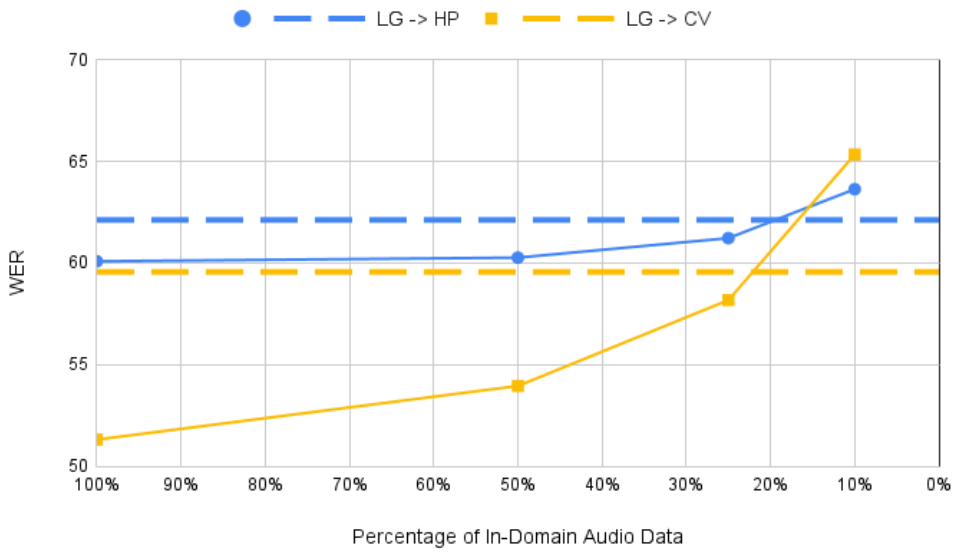


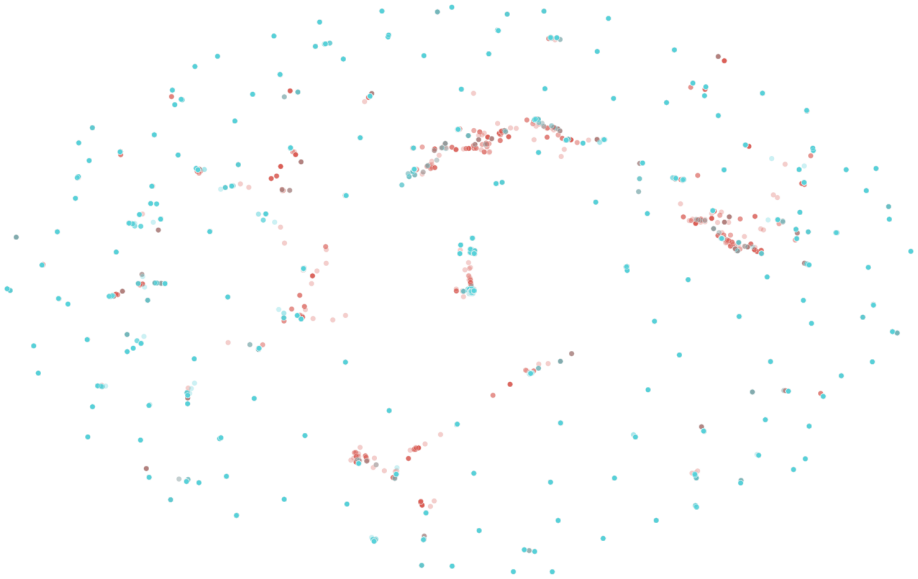
Figure 10.3: Performance of M2DS2 for the LG → HP (blue circle), and the LG → CV settings (yellow square), when reducing the amount of available target samples to 50%, 25%, and 10% of the original dataset (horizontal axis). SO performance is indicated with the dashed lines. Vertical axis: WER, Horizontal Axis: target audio percentage (100% → 0%)

10.6.1 The sample efficiency of M2DS2

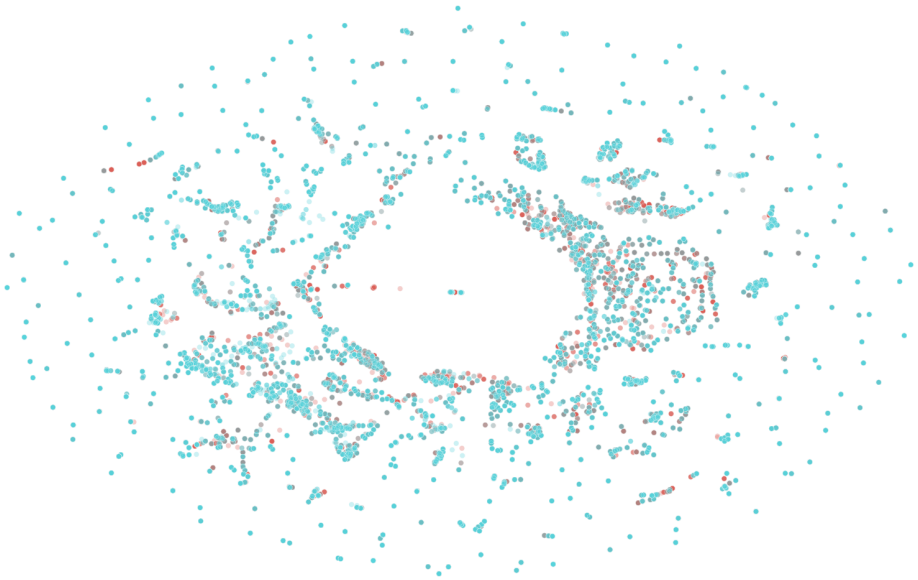
We have observed from our literature review and experimental validation that CPT requires large amounts of un-transcribed target domain audio. This raises the question, can we leverage self-supervision for domain adaptation in data-constrained settings?

This provides a promising avenue for adaptation when the collection of in-domain recordings is expensive, or infeasible.

In Fig. 10.3 we evaluate the performance of M2DS2 when we reduce the amount of target domain audio. Specifically, we focus on two scenarios, LG → CV and LG → HP. We evaluate four settings, where we train M2DS2 with 100%, 50%, 25%, and 10% of the available samples, and plot the resulting WER on the target test set. For the CV target domain, this corresponds to 12, 6, 3, and 1.2 hours of audio respectively. For the HP target, this corresponds to 100, 50, 25, and 10 hours of audio. In all cases, the full source (LG) training corpus is used. We observe that the LG → HP is a more challenging setting than LG → CV, as it demonstrates smaller absolute WER improvement as we include in-domain audio. This can also be observed by the amount of data needed for successful adaptation for each setting, i.e. 25 hours for HP versus 3 hours for CV. In both cases, we observe that M2DS2 achieves lower WER than the SO baseline, even with only 25% of target domain audio. While CPT can suffer from catastrophic forgetting, as most multi-stage training approaches, M2DS2 avoids this issue, being a single-stage approach with a mixed task-specific and self-supervised objective. This provides a promising avenue



(a) Only target domain self-supervision



(b) Target and source domain self-supervision

Figure 10.4: T-SNE scatter plots of code vectors extracted from M2DS2 without source domain self-supervision (top) and with source domain self-supervision (bottom) for LG (red) and CV (teal)

Table 10.7: WER of M2DS2 without source-domain self-supervision when varying the diversity loss weight for the LG→CV setting.

ω	0.0	0.01	0.1	0.5	1.0	1.5	2.0
M2DS2 ($\alpha = 0$)	65.23	70.34	62.77	66.41	60.98	77.51	78.66
M2DS2				**51.31**			

for adaptation when the collection of in-domain recordings is expensive, or infeasible.

10.6.2 The importance of multi-domain self-supervision

In Section 10.2.2 we argue that it is essential to include both source and target domain data for the self-supervised objective of M2DS2. To illustrate the effect of this approach, we train two versions of M2DS2 for the LG → CV scenario. For the first version, we set $\alpha = 0.01$, while for the second we set $\alpha = 0$, removing the second term of Eq. (10.2). We extract the code vectors for the first 100 samples of both LG and CV and flatten them across the time steps, resulting in 60000×768 code vectors corresponding to individual timesteps. We plot these code vectors using T-SNE[257] in Fig. 10.4 for both models. We see that when we do not include the source domain self-supervision, the code vector space collapses in a few tight clusters, and most audio segments correspond to just a few code vectors. This is a visual clue that indicates the mode collapse problem. When we include the source domain term, we see that the code vector space has more structure, and the space coverage is more complete, both for CV (target domain) and LG (source domain). Experimentally we train M2DS2 with $\alpha = 0$ for all source/target domain pairs and we find that the mode collapse is destructive for target domain performance. The simple inclusion of both source and target domain self-supervision stabilizes training, avoids mode collapse, and leads to successful unsupervised adaptation between domains.

In addition, we explore if increasing the contribution of the diversity loss term in Eq. (10.1) is sufficient to combat mode collapse. We weigh the diversity loss with a multiplicative hyper-parameter ω, and explore the following values $\omega \in \{0.0, 0.01, 0.1, 0.5, 1.0, 1.5, 2.0\}$ for M2DS2 without source-domain self-supervision ($\alpha = 0$). For the proposed M2DS2, ω is set to the default value 0.1. The results are summarized in Table 10.7. We observe that varying the contribution of the diversity loss can be important, but mixing source and target domain data is essential for good performance. This observation is in accordance with other works in the literature, which highlight the importance of mixing source and target domain data for CPT[216,217] and pseudo-labeling[226].

10.7 Unsupervised and weakly supervised language adaptation

When small amounts of in-domain textual data are available, simple N-gram LM adaptation techniques can be very effective. In this brief set of experiments, we first explore the unsupervised language adaptation setting, where no in-domain audio is used, and then we relax the problem to the weakly supervised setting, where M2DS2 is combined with the adapted

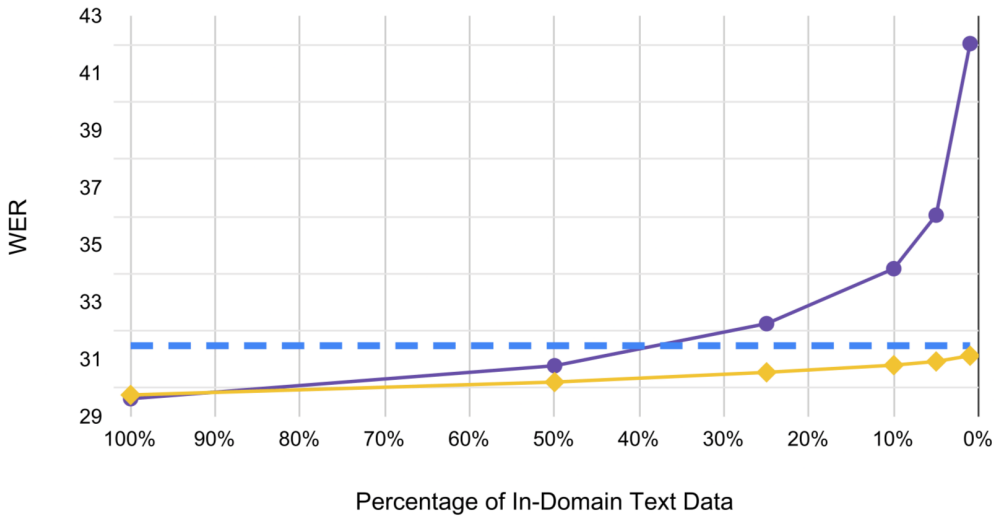


Figure 10.5: Language-only adaptation for LG → HP using the SO model fine-tuned on LG. In-domain text data range from 11M tokens (left) to 110K tokens (right). Blue / dashed: Baseline with generic LM. Purple / circles: Biased LM. Yellow / diamonds: Augmented LM.

N-Gram LM. These settings are described in Sections 8.3 and 8.3 respectively. We explore two approaches for LM adaptation: biased LM, and in-domain data augmentation. To create the biased LM, we train a 4-gram LM on the available in-domain data. Then we replace the generic LM trained on GGC. For LM data augmentation we follow a perplexity filtering approach similar to[285]. We first train a biased LM using available target domain text, and then use it to calculate the perplexity of each line in the GGC corpus. We keep the 10% of the lines with the lowest perplexity. Then we train a 4-gram LM on the augmented "in-domain" corpus and use it for inference.

Fig. 10.5 shows the performance of the SO LG → HP model with biased and augmented

Table 10.8: Language adaptation for M2DS2 in the LG → CV and LG → HP scenarios, using biased and augmented LM. We vary the amount of available in-domain text. LG → CV: 752K to 38K tokens. LG → HP: 11M to 550K tokens.

Setting / Tokens (%)	LG → CV (3 hours)		LG → HP (100 hours)	
	Biased LM	Augmented LM	Biased LM	Augmented LM
100%	11.22	12.84	27.95	28.91
50%	15.13	15.05	29.39	29.44
25%	20.84	16.64	31.06	29.46
10%	27.75	18.47	33.58	29.81
5%	33.04	19.31	35.69	30.16
Baseline (M2DS2 + Generic LM)	20.70		31.36	

Table 10.9: Closing the gap between SO training and fully supervised training for the LG → CV adaptation scenario using M2DS2, with varying amounts of available unpaired in-domain audio and text. (U): unsupervised acoustic or language adaptation. (W): weakly supervised adaptation.

Method	#Audio (h)	#Tokens	LM	WER
SO (U)	-	-	N/A	59.57
M2DS2 (U)	3	-	N/A	57.31
M2DS2 (U)	12	-	N/A	51.31
SO (U)	-	-	Generic	25.96
SO (U)	-	38, 632	Augmented	24.67
SO (U)	-	751, 953	Augmented	20.46
M2DS2 (U)	3	-	Generic	20.7
M2DS2 (U)	12	-	Generic	17.3
M2DS2 (W)	3	38, 632	Augmented	19.31
M2DS2 (W)	12	38, 632	Augmented	16.29
M2DS2 (W)	3	751, 953	Augmented	12.84
M2DS2 (W)	12	751, 953	Augmented	10.61
Supervised	12	751, 953	Generic	9.52
Supervised	12	751, 953	Augmented	7.94

LM, as we reduce the amount of available in-domain text data from 100% to 1% of the in-domain transcriptions (11M tokens to 110K tokens respectively). As a baseline, we include the LG → HP SO model in combination with the generic LM trained on GGC. We observe that the use of biased LM can lead to successful adaptation when an adequate amount of in-domain text data is available. On the other hand, the LM augmentation approach results in successful augmentation, even with very small amounts of in-domain text.

In Table 10.8 we see the results of LM adaptation, combined with M2DS2 for the LG → CV and LG → HP scenarios. To account for different target corpus sizes, we use the variant trained on 3 hours of target domain audio for the LG → CV case, and the variant trained with 100 hour of in-domain audio for the LG → HP case. We compare with M2DS2 combined with the 4-gram GGC LM for inference. We draw similar conclusions, i.e., the use of biased LM performs well for sufficient text data. When we use augmented LM we can leverage very small amounts of in-domain text. Furthermore, when adapting for HP we see smaller improvements from both LM biasing and LM augmentation techniques. This can be attributed to the linguistic content of the HP dataset. HP contains names, dates, law acts, etc., which make the linguistic adaptation harder.

10.8 Discussion

In this work, we have explored Unsupervised and Weakly Supervised Domain Adaptation of ASR systems in the context of an under-resourced language, i.e., Greek. We focus on domain adaptation through in-domain self-supervision for XLSR-53, a state-of-the-art multilingual ASR model. Specifically, we adopt a mixed task and self-supervised objective, inspired by NLP, and show that using only in-domain self-supervision can lead to mode collapse of the representations created by the contrastive loss of XLSR-53. Therefore, we propose the use of mixed task and multi-domain self-supervision, M2DS2, where the contrastive loss leverages both the source and target domain audio data. For evaluation, we create and release HParl, the largest to-date public corpus of transcribed Greek speech (120 hours), collected from the Greek Parliamentary Proceedings. HParl is combined with two other popular Greek speech corpora, i.e., Logotypografia and CommonVoice, for multi-domain evaluation.

In our experiments, we find that while most UDA baselines fail in our low-resource setting, the proposed mixed task and multi-domain self-supervised fine-tuning strategy yields significant improvements for the majority of adaptation scenarios. Furthermore, we focus our ablations on showcasing the sample efficiency of the proposed fine-tuning strategy and demonstrating the necessity of including both source and target domain data for self-supervision. Finally, we show that M2DS2 can be combined with simple language model adaptation techniques in a relaxed weakly supervised setting, where we achieve significant performance improvements with a few hours of in-domain audio and a small, unpaired in-domain text corpus.

More concretely, in Table 10.9 we present a summary of the discussed unsupervised and weakly supervised adaptation combinations, for different amounts of available in-domain audio and text. Note that for the weakly supervised scenarios, the in-domain audio and text are unpaired. We see, that when no in-domain data are available, including an n-gram LM trained on large corpora is recommended. Furthermore, when in-domain audio is available, following a mixed multi-domain fine-tuning strategy using M2DS2 can yield significant WER reductions, even for a few hours of audio. When small amounts of in-domain text are available, using a corpus augmentation strategy, e.g., perplexity filtering, can produce an adapted LM and yield small improvements to the final WER. In the case of sufficient amounts of unpaired in-domain text and audio, the independent adaptation of XLSR-53 using the audio data and the n-gram LM using the text data can yield comparable performance with a fully supervised fine-tuning pipeline.

10.8.1 Future Work

In the future, we plan to explore the effectiveness of the proposed adaptation strategy for other languages, and different adaptation settings, e.g., accent or cross-lingual adaptation. The combination of M2DS2 with different self-supervised models, e.g. WavLM[289], will also be explored, to assess the effectiveness of mixed self-supervision in a non-contrastive setting for English ASR. Of special interest is the investigation of the effectiveness of our approach for endangered languages, e.g., Pomak. Furthermore, we plan to explore the combination of in-

domain self-supervision, when combined with other popular UDA techniques, e.g., teacher-student models, adversarial learning, and data augmentation approaches. On the language adaptation side, we plan to explore multi-resolution learning, which has shown promise for ASR[290], and investigate more elaborate end-to-end weakly supervised adaptation methods. Finally, we plan to expand our study in a multimodal setting, where both audio and video are available, e.g., lip reading.

Discussion

"*His philosophy was a mixture of three famous schools – the Cynics, the Stoics and the Epicureans – and summed up all three of them in his famous phrase, 'You can't trust any bugger further than you can throw him, and there's nothing you can do about it, so let's have a drink.*"

Terry Pratchett, Small Gods

11
Conclusions and Future Directions

11.1 Summary of key contributions

In this work, we explore modern, robust, and flexible machine learning techniques tailored for resource-constrained settings. Our two key contributions, inspired by cognitive sciences, are: firstly, Pattern Search MDS, a novel algorithm for Dimensionality Reduction (DR); and secondly, robust fine-tuning strategies based on mixed self-supervision for Unsupervised Domain Adaptation (UDA). We extensively evaluate both methods in terms of performance and robustness across a diverse set of tasks, with a particular emphasis on speech and language processing. Moreover, we assess the relevance of these techniques in the contemporary "scale-up" context, demonstrating their significant impact in data-scarce scenarios. Finally, beyond these methodological contributions, this dissertation has led to several technical contributions, including the creation of the largest speech recognition dataset for a low-resource language (Modern Greek), named HParl, and the development of multiple open-source projects*.

11.2 Subspace learning with multi-dimensional scaling

We have proposed pattern search MDS, a high-performant, derivative-free algorithm for Dimensionality Reduction through Multi-Dimensional Scaling. Reduction of pattern search MDS to the GPS family of direct search algorithms provides theoretical convergence guarantees. The proposed algorithm has been extensively evaluated in diverse tasks and datasets, i.e., Speech Emotion Recognition, visual classification, word semantic similarity, and manifold geometry. Our experiments show that the reduced feature sets obtained by pattern search MDS geometric, semantic, visual, and affective properties, visual, and affective properties.

The derivative-free formulation of pattern search MDS opens avenues to project features and latent vectors into subspaces, employed with a wide variety of distance metrics, e.g., non-metric distances, quasi-metric distances, etc. This flexibility can be used to produce hierarchical distributed representations, by breaking down semantic neighborhoods into mutliple low-dimensional subspaces, each encoding a different semantic property. A similar idea has been proposed by Karlgren et al. [291], where high-dimensional lexical vector space models are argued to be composed of embedded, local, low-dimensional manifolds. This idea is further

*See https://github.com/georgepar

explored by Athanasopoulou et al. [292], where a Distributional Semantic Model (DSM) for word semantics is decomposed into a sparse set of low-dimensional DSMs, using a set of dissimilarity metrics based on contextual distances in local semantic neighborhoods. In this framework, concepts can be encoded as distributed vectors, while relations are encoded as the similarities between the concept vectors. Multiple subspaces can be created based on different similarity metrics, encoding different relations. Future works can utilize pattern search MDS for the efficient creation of hierarchical DSM, where concepts are encoded in interpretable subspaces, while concept relations are represented using sets of contextual similarity metrics [293].

Another line of exploration can focus on the unsupervised sense disambiguation of distributed word vectors, by viewing a polysemous word vector as a set S of n senses $x_1, x_2, \ldots, x_n$. The word distances in the high-dimensional space can be then assumed to be obtained from a set distance between the sense sets, i.e.:

$$\delta(S_1, S_2) = \min_{x_i \in S_1, x_j \in S_2} d(x_i, x_j) \tag{11.1}$$

The word senses can be disentagled in a low-dimensional space with more points than the original via dimensionality reduction using pattern search MDS, by assuming the original distance matrix is obtained using δ as the underlying distance. This is a challenging scenario, as δ is a pseudo-metric that violates the triagle inequality. Future works can develop of robust algorithms for this problem, i.e. Set-MDS, with the diploma thesis of Lena Fotaki[†] as a starting point. Furthermore, the Set-MDS problem can be viewed as a manifold untangling problem, which has been assumed to underlie the human visual system [294] with strong cognitive motivation. Manifold untangling has also been shown to be an emergent phenomenon in Transformer-based language models [295] and speech recognition systems [296].

11.3 Mixed self-supervision for sample-efficient unsupervised domain adaptation

In the second part of this work, we have investigated the problem of Unsupervised Domain Adaptation in speech and language models. Along these lines, we have proposed a mixed self-supervision fine-tuning strategy. During fine-tuning, the task is learned using annotated out-of-domain data, while adaptation is performed through self-supervision on unlabeled in-domain samples. Mixed self-supervision is motivated as an implicit strategy for solving the stability-plasticity dilemma. It implicitly finds the optimal way to change the network parameters for unseen domains while maintaining the useful knowledge acquired during pretraining. The proposed technique can be applied to various problems and different input modalities. Specifically, in this work, mixed self-supervision leads to successful adaptation in text sentiment analysis and speech recognition settings, across multiple domain pairs. Furthermore, the sample efficiency of mixed self-supervision has been a central point of our investigation. Our ablations have shown that successful adaptation can be achieved with a limited number

[†] Citation not yet available

of in-domain samples; as few as 3 hours of in-domain audio for speech recognition and 500 text samples for sentiment analysis.

In the future, mixed self-supervision can be explored in more challenging adaptation scenarios, particularly in cross-lingual adaptation of text and speech models, a significant issue in creating resources for digitally under-represented languages. Furthermore, source-free domain adaptation[297], where annotated out-of-domain data are not available at the time of adaptation, presents a challenging scenario, especially in commercial settings where data privacy concerns and limited edge computing resources are prevalent. Finally, the combination of Meta-Learning[298] and Self-Supervised Learning is an exciting area for future exploration that has attracted the interest of the machine learning community[299,300], and have been combined for few-shot UDA[301]. On the surface, SSL and meta-learning assume different strategies for model generalization. SSL requires an extensive pretraining stage on vast amounts of unsupervised data, enabling the model to learn the underlying data distribution. In contrast, meta-learning strategies emphasize ease of adaptation, where an easily adaptable model is trained (learning to learn) and then fine-tuned in a few-shot manner at inference time. However, both paradigms ultimately aim for generalization, and their integration could be instrumental in developing advanced System 1/System 2 models, which would combine fast decision-making with the ability to adapt on the fly from new, complex scenarios.

11.4 Beyond unimodal representations: Multimodal fusion and co-learning

Our experience of the world relies on modal systems of perception (e.g., vision, auditory, haptic), action (e.g., motor control, movement), and introspection (e.g., affect). Grounded cognition theory[302] supports the notion that these systems not only allow us to perceive and interact with the environment, but that our understanding of the physical world is deeply rooted in our modal experiences, challenging views of cognition based on amodal, abstract symbols. This theory posits that cognitive processes, rather than being abstract and separate from bodily experiences, are fundamentally shaped and informed by our sensory and motor interactions. Complementing this perspective, the concept of a metamodal brain[303,304] highlights the brain's capacity for cross-modal integration and neuroplasticity[305]. It suggests that the brain can reorganize and adapt, processing sensory information in areas typically dedicated to other modalities. Together, grounded cognition and the metamodal brain concept underscore the dynamic and adaptable nature of the brain, shaping our perception, cognition, and interaction with the world in an integrated and fluid manner.

Given how our understanding of the world is deeply rooted in our sensory experiences, a fruitful direction is to explore the applicability of the proposed techniques in multimodal settings. In the context of machine learning, multimodal processing seeks to emulate aspects of human multimodal perception and cognition. Just as grounded cognition and the metamodal brain concepts illustrate the brain's capacity for cross-modal integration and adaptability, multimodal machine learning aims to integrate and process diverse types of data (e.g., visual, auditory, textual) to enhance learning and decision-making. This integration reflects the

principle that different modalities provide complementary information, enriching the overall understanding of a given context or task[306]. For instance, in image captioning[307], combining visual data with linguistic context leads to more accurate and nuanced descriptions. Furthermore, the field explores how models can adaptively switch or combine modalities, akin to the brain's neuroplasticity, to optimize performance under varying conditions. This involves not only the development of algorithms capable of handling multimodal data efficiently but also understanding how different modalities influence and reinforce each other within these systems. A key goal is to develop AI systems that can process and interpret complex, multimodal information in a way that mirrors human cognitive flexibility and efficiency, leading to more intuitive and effective interactions between humans and machines.

11.4.1 Multimodal and multitask learning

We begin this section with a Proposition demonstrating the close relationship between multitask and multimodal learning[‡].

Proposition 3: *Consider the multi-class classification problem with N data points and C classes:*

$$\min L(\hat{y}, y) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij})$$

Furthermore, consider that predictions $\hat{y}$ are obtained using a late fusion strategy for features from two modalities x^{α}, x^{β}:

$$\hat{y} = \hat{y}^{\alpha} \cdot \hat{y}^{\beta} = \sigma(x^{\alpha}) \cdot \sigma(x^{\beta}),$$

where σ is the softmax function. Then, the loss L can be rewritten as the sum:

$$L(\hat{y}, y) = L(\hat{y}^{\alpha}, y) + L(\hat{y}^{\beta}, y)$$

Proof: *See Appendix A.*

Proposition 3 shows that in its basic form, multimodal learning via late fusion can be equated to multitask learning, by learning the relationship between each modality's features and the target label independently[§]. More complex fusion strategies, can result in more complex (non-linear) loss term combinations, or the addition of cross-modal regularization terms.

This intuition can drive future researchers to devise effective multimodal learning strategies, by borrowing ideas from the multitask learning literature. For example, in our preliminary study[308], we employ mixed self-supervision for fusion of textual and visual information in an industrial application setting with good results. Further research could explore the

[‡]Surprisingly given the simplicity of the derivation, I have not been able to locate a similar remark in the context of multimodal learning in prior works.

[§]It is trivial to show that, if we want to weigh the contribution of each loss term, we only need to modify the logits as $\hat{y} = \sigma(x^{\alpha})^{\lambda} \cdot \sigma(x^{\beta})^{1-\lambda}$

Pareto fronts between different modalities, mitigating cross-modal competition[309], modality imbalance[310,311], and the fact that different modalities learn at different rates[312]. Works such as Dimitriadis et al.[313], Lin et al.[314], Ma et al.[315] provide a starting point for this exploration.

11.4.2 Bottom-up and top-down cross-modal modeling

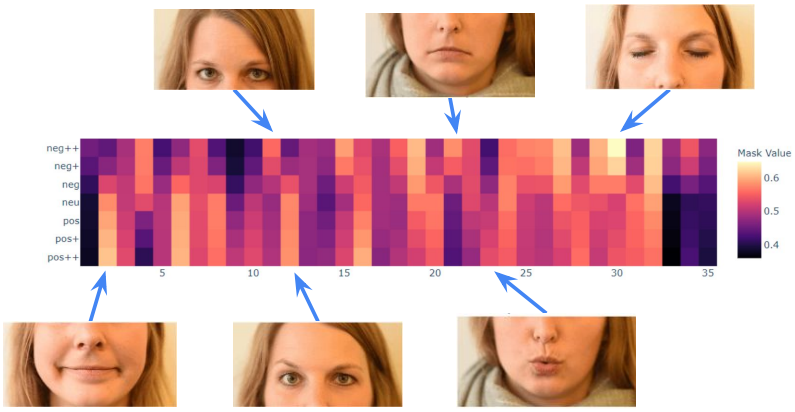


Figure 11.1: Average top-down mask values, learned by MMLatch, over the test set for samples with sentiment values from neg++ (very negative) to pos++ (very positive), and for different facial features. Observe that for negative samples, more higher mask values are put in more negative expressions (top). Vice versa for positive samples higher mask values are associated with more positive expressions (bottom).

Taking it one step further, it is especially intriguing to examine the methods through which cross-modal interactions are facilitated. In common pipelines, only bottom-up interactions are considered in a feedforward manner, i.e., low-level multisensory inputs are processed in modality-systems, and then fused in a high-level multimodal module. Nevertheless, multiple empirical studies[316–323] highlight the importance of top-down regulation. For example, Papale et al.[316] presented occluded images to monkeys, and measured the response of neurons in the primary visual cortex whose receptive field corresponded to the occluded part of the image. Remarkably, they found that the measured activations of the occluded neurons allowed the decoding of the original image. Furthermore, they analyzed the delay in the occluded neurons' response, and found that it indicates the existence of feedback connections from higher-level neurons. The importance of top-down regulation in a cross-modal setting is underscored by Winkowski and Knudsen[324,325]. In their studies, they demonstrated the influence of top-down gain control by electrically stimulating gaze control regions in the barn owl's brain, and examining the impact on auditory spatial perception. Their findings revealed that such stimulation not only modulates but specifically enhances the gain of midbrain auditory responses corresponding to targeted spatial locations, while concurrently attenuating

responses to non-targeted areas, illustrating a sophisticated mechanism for sensory gain modulation across different sensory modalities.

Along these lines, we have reported preliminary results in Paraskevopoulos et al. [326], where we propose MMLatch, a neural architecture that models bottom-up and top-down cross-modal interactions[¶]. MMLatch is a feedback module that performs top-down regulation using high-level representations to produce masks that enhance or attenuate low-level input features. We have demonstrated that incorporating MMLatch to Recurrent Neural Network (RNN) and Transformer-based architectures improves model performance. Furthermore, we have observed that the masks learned by MMLatch can enhance or attenuate input features in an interpretable way. Average top-down mask values, learned using MMLatch are shown in Fig. 11.1, where we can observe the top-down mask values that are associated with different facial expressions for samples with positive or negative sentiment.

One possible limitation of introducing top-down feedback in neural architectures is the difficulty introduced in the training process, by allowing the architecture to modify itself. MMLatch side-steps this difficulty by using only cross-modal top-down feedback. In further experiments we observed convergence issues when trying to introduce within-modality feedback. Future works can focus on separating the training of bottom-up and top-down parts of the network, or formulating the training of bottom-up and top-down parameters in a hierarchical optimization setting [327,328], where the top-down parameters can be viewed as meta-parameters that constrain the bottom-up learning process. Additionally, different top-down functions can be explored. For example, exploring top-down sharpening, instead of masking, can be a fruitful endeavor, where the top-down interactions are more closely resemble the gain control mechanism described in the studies of Winkowski and Knudsen [324,325].

[¶]This study is included in Appendix B. You can refer to the appendix for more details.

Appendices

"*Real stupidity beats artificial intelligence every time.*"

Terry Pratchett, Hogfather

A

Proof of Proposition 3

$$
\begin{aligned}
L(\hat{y}, y) &= -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}) \\
&= -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}^{\alpha}\cdot\hat{y}_{ij}^{\beta}) \\
&= -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}^{\alpha}) + y_{ij}\log(\hat{y}_{ij}^{\beta}) \\
&= -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}^{\alpha}) - \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}^{\beta}) \\
&= L(\hat{y}^{\alpha}, y) + L(\hat{y}^{\beta}, y)
\end{aligned}
$$

■

B

Bottom-up Top-down Fusion for Multimodal Sentiment Analysis

B.1 Introduction

Multimodal processing aims to model interactions between inputs that come from different sources in real world tasks. Multimodality can open ways to develop novel applications (e.g. Image Captioning, Visual Question Answering[307,329] etc.) or boost performance in traditionally unimodal applications (e.g. Machine Translation[330], Speech Recognition[290,331] etc.). Moreover, modern advances in neuroscience and psychology hint that multi-sensory inputs are crucial for cognitive functions[332], even since infancy[333]. Thus, modeling and understanding multimodal interactions can open avenues to develop smarter agents, inspired by the human brain.

Feedback loops have been shown to exist in the human brain, e.g. in the case of vocal production[317] or visual-motor coordination[318]. Human perception has been traditionally modelled as a linear (bottom-up) process (e.g. reflected light is captured by the eye, processed in the prefrontal visual cortex, then the posterior visual cortex etc.). Recent studies have highlighted that this model may be too simplistic and that high level cognition may affect low-level visual[319,320] or audio[321] perception. For example, studies state that perception may be affected by an individual's long-term memory[322], emotions[334] and physical state[323]. While scientists still debate on this subject[335], such works offer strong motivation to explore if artificial neural networks can benefit from multimodal top-down modeling.

Early works on multimodal machine learning use binary decision trees[336] and ensembles of Support Vector Machines[337]. Modeling contextual information is addressed in[338–340] using Recurrent Neural Networks (RNNs), while Poria et al.[341] use Convolutional Neural Networks (CNNs). For a detailed review we refer to Baltruvsaitis et al.[342]. Later works use Kronecker product between late representations[343,344], while others investigate architectures with neural memory-like modules[345,346]. Hierarchical attention mechanisms[347], as well as hierarchical fusion[348] have been also proposed. Pham et al.[349] learn cyclic cross-modal mappings, Sun et al.[350] propose Deep Canonical Correlation Analysis (DCCA) for jointly learning representations. Multitask learning has been also investigated[351] in the multimodal context. Transformers[7] have been applied to and extended for multimodal tasks[352–355]. Wang et al.[356] shift word representations based on non-verbal imformation.[357] propose a fusion gating mechanism.[358]

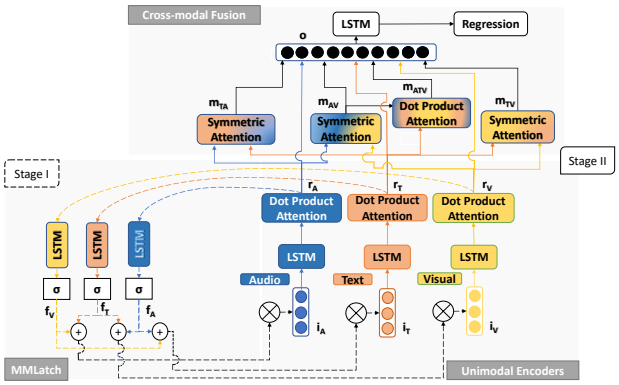


Figure B.1: Architecture overview of three high-level modules, composing the overall system: Unimodal encoders, Cross-modal fusion and MMLatch. Solid lines indicate the feedforward connections (bottom-up processing), while dashed lines indicate feedback connections (top-down processing). Colors indicate different modalities (Blue: Audio, Orange: Text, Yellow: Visual)

use capsule networks[359] to weight input modalities and create distinct representations for input samples.

In this work we propose MMLatch, a neural network module that uses representations from higher levels of the architecture to create top-down masks for the low level input features. The masks are created by a set of feedback connections. The module is integrated in a strong late fusion baseline based on LSTM[360] encoders and cross-modal attention. A similar top-down masking idea is proposed in[361], where feedback masks are produced for interpretability of a unimodal (vision) CNN architecture, trained using an auxiliary loss. Our key contribution is the modeling of *cross-modal* interactions between high-level representations extracted by the network and low-level input features, using an end to end framework, without need for an auxiliary loss. We integrate MMLatch with RNNs and Transformers, but it can be adapted for more architectures. Incorporating top-down modeling shows consistent improvements over our strong baseline, yielding state-of-the-art results for sentiment analysis on CMU-MOSEI. Our code is open source*.

B.2 Proposed Method

Fig. B.1 illustrates an overview of the system architecture. The baseline system consists of a set of unimodal encoders and a cross-modal attention fusion network, that extracts fused feature vectors for regression on the sentiment values. We integrate top-down information by augmenting the baseline system with a set of feedback connections that create cross-modal, top-down feature masks.

Unimodal Encoders: Input features i_A, i_T, i_V for each modality are encoded using three LSTMs. The hidden states of each LSTM are then passed through a Dot Product self-attention mecha-

*https://github.com/georgepar/mmlatch/

Table B.1: Results on CMU-MOSEI for MMLatch. Models indicated with * are reproduced for CMU-MOSEI by Tsai et al.[352]. In row "MMLatch average" we include results averaged over five runs. Since other works do not report standard deviation, we also include row "MMLatch best", where we report the best of the five runs (lowest error).

Model / Metric	Acc@7	Acc@2	F1@2	MAE	Corr
RAVEN[356] *	50.0	79.1	79.5	0.614	0.662
MCTN[349] *	49.6	79.8	80.6	0.609	0.670
Multimodal Routing[358]	51.6	81.7	81.8	-	-
MulT[352]	51.8	82.5	82.3	**0.580**	**0.703**
Baseline (ours)	51.3 ± 0.7	81.9 ± 0.7	82.2 ± 0.6	0.593 ± 0.005	0.695 ± 0.004
Baseline + MMLatch average (ours)	$\mathbf{52.0} \pm 0.2$	82.4 ± 0.3	$\mathbf{82.5} \pm 0.3$	0.585 ± 0.002	0.700 ± 0.004
Baseline + MMLatch best (ours)	**52.1**	**82.8**	**82.9**	0.582	**0.704**

nism to produce the unimodal representations r_A, r_T, r_V, where A, T, V are the audio, text and visual modalities respectively.

Cross-modal Fusion: The encoded unimodal representations are fed into a cross-modal fusion network, that uses a set of attention mechanisms to capture cross-modal interactions. The core component of this subsystem is the symmetric attention mechanism, inspired by Lu et al.[353]. If we consider modality indicators $k, l \in \{A, V, T\}$, $k \neq l$, $r_k, r_l \in \mathbb{R}^{B \times N \times d}$ the input modality representations, we can construct keys $K_l = W_l^K r_l$, queries $Q_k = W_k^Q r_k$ and values $V_l = W_l^V r_l$ using learnable projection matrices $W_{\{k,l\}}^{\{K,Q,V\}}$, and we can define a cross-modal attention layer as:

$$a_{kl} = s\left(\frac{K_l^T Q_k}{\sqrt{d}}\right) V_l + r_k, \tag{B.1}$$

where s is the softmax operation and B, N, d are the batch size, sequence length and hidden size respectively. For the symmetric attention we sum the two cross-modal attentions:

$$m_{kl} = a_{kl} + a_{lk}, \tag{B.2}$$

In the fusion subsystem we use three symmetric attention mechanisms to produce m_{TA}, m_{TV} and m_{AV}. Additionally we create a_{AVT} using a cross-modal attention mechanism (Eq. (B.1)) with inputs m_{AV} and r_T. These crossmodal representations are concatenated ($\|$), along with the unimodal representations m_A, m_T, m_V to produce the fused feature vector $o \in \mathbb{R}^{B \times N \times 7d}$ in Eq. (B.3).

$$o = r_A \,\|\, r_T \,\|\, r_V \,\|\, a_{AVT} \,\|\, m_{AV} \,\|\, m_{TV} \,\|\, m_{TA} \tag{B.3}$$

We then feed o into a LSTM and the last hidden state is used for regression. The baseline system consists of the unimodal encoders followed by the cross-modal fusion network.

Top-down fusion: We integrate top-down information by augmenting the baseline system with MMLatch, i.e. a set of feedback connections composing of three LSTMs followed by sigmoid activations σ. The inputs to these LSTMs are r_A, r_T, r_V as they come out of the unimodal encoders. Feedback LSTMs produce hidden states h_A, h_T, h_V. The feedback masks f_A, f_T, f_V are

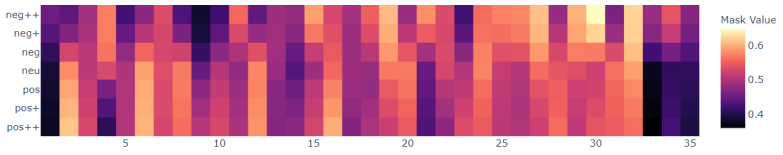


Figure B.2: Averaged top-down mask values for FACET features over all test samples across seven sentiment classes. neg++ indicates a sentiment score ≈ -3, neg+ ≈ -2, neg ≈ -1, neu ≈ 0, pos ≈ 1, pos+ ≈ 2 and pos++ ≈ 3.

produced by applying a sigmoid activation on the hidden states $f_k = \sigma(h_k),\ k \in \{A, T, V\}$ and then applied to the input features i_A, i_T, i_V using element-wise multiplication $\odot$, as:

$$\widetilde{i}_k = \frac{1}{2}(f_j + f_l) \odot i_k \tag{B.4}$$

where $j, k, l \in \{A, V, T\},\ k \neq l \neq m$.

This pipeline is implemented as a two-stage computation. During the first stage we use the unimodal encoders and MMLatch to produce the feedback masks f_A, f_T, f_V and apply them to the input features using Eq. (B.4). During the second stage we pass the masked features $\widetilde{i}_A, \widetilde{i}_T, \widetilde{i}_V$ through the unimodal encoders and the cross-modal fusion module and use the fused representations for regression. Intuitively, this two-stage computation allows the network to use its own representations in order to select which input features should be enhanced or attenuated to solve the task at hand, resulting to a dynamic feature selection process.

B.3 Experimental setup

We use CMU-MOSEI sentiment analysis dataset[346] for our experiments. The dataset contains 23,454 YouTube video clips of movie reviews accompanied by human annotations for sentiment scores from -3 (strongly negative) to 3 (strongly positive) and emotion annotations. Audio sequences are sampled at $20Hz$ and then 74 COVAREP features are extracted. Visual sequences are sampled at $15Hz$ and represented using FACET features. Video transcriptions are segmented in words and represented using GloVe. All sequences are word-aligned using P2FA. Standard train, validation and test splits are used.

For all our experiments we use bidirectional LSTMs with hidden size 100. LSTMs are bidirectional and forward and backward passes are summed. All projection sizes for the attention modules are set to 100. We use dropout 0.2. We use Adam[362] with learning rate 0.0005 and halve the learning rate if the validation loss does not decrease for 2 epochs. We use early stopping on the validation loss (patience 10 epochs). During Stage I of each training step we disable gradients for the unimodal encoders. Models are trained for regression on sentiment values using Mean Absolute Error (MAE) loss. We use standard evaluation metrics: 7-class, 5-class accuracy (i.e. classification in $\mathbb{Z} \cap [-3, 3]$, $\mathbb{Z} \cap [-2, 2]$), binary accuracy and F1-score (negative in $[-3, 0)$, positive in $(0, 3]$), MAE and correlation between model and human predictions. For fair comparison we compare with methods in the literature that use GloVe, COVAREP and FACET features.

Table B.2: Results on CMU-MOSEI when combining top-down feedback with different multimodal encoder networks. MulT with † is reproduced by us. We report results, averaged over five runs, along with standard deviations.

Multimodal Encoder	Feedback Type	Acc@7	Acc@2	F1@2	MAE	Corr
Baseline	-	51.3 ± 0.7	81.9 ± 0.7	82.2 ± 0.6	0.593 ± 0.005	0.695 ± 0.004
Baseline	MMlatch (no LSTM)	51.48 ± 0.41	82.07 ± 0.47	82.29 ± 0.39	0.592 ± 0.002	0.692 ± 0.003
Baseline	MMLatch	**52.0** ± 0.2	**82.4** ± 0.3	**82.5** ± 0.3	**0.585** ± 0.002	**0.700** ± 0.004
MulT†	-	47.91 ± 1.13	80.35 ± 0.36	80.54 ± 0.52	0.643 ± 0.01	0.648 ± 0.02
MulT†	MMLatch	**49.04** ± 0.45	**80.65** ± 0.43	**81.07** ± 0.38	**0.627** ± 0.004	**0.665** ± 0.003

B.4 Experiments

Table B.1 shows the results for sentiment analysis on CMU-MOSEI. The Baseline row refers to our late-fusion baseline described in Section B.2, which achieves competitive to the state-of-the-art performance. Incorporating MMLatch into the baseline constistently improves performance and specifically, almost 1.0% over the binary accuracy and 0.8% over the seven class accuracy. Moreover, we observe lower deviation, w.r.t. the baseline, across experiments, indicating that top-down feedback can stabilize training. Compared to state-of-the-art we achieve better performance for 7-class accuracy and binary F1 metrics in our five run experiments. Since, prior works do not report average results over multiple runs so we also report results for the best (mean absolute error) out of five runs in the last row of Table B.1, showing improvements across metrics over the best runs of the other methods.

In Table B.2 we evaluate MMLatch with different multimodal encoders and different feedback types. The first three rows show the effect of using different feedback types. Specifically, first row shows our baseline performance (no feedback). For the second row we add feedback connections, but instead of using LSTMs in the feedback loop (Stage I in Fig. B.1), we use a simple feed-forward layer. The last row shows performance when we include LSTMs in the feedback loop. We observe that, while the inclusion of top-down feedback, using a simple projection layer results to a small performance boost, when we include an LSTM in the feedback loop we get significant improvements. This shows that choosing an appropriate mapping from high-level representations to low-level features in the feedback loop is important.

For the last two rows of Table B.2 we integrate MMLatch with MulT architecture† [352]. Specifically, we use MMLatch, as shown in Fig. B.1 and swap the baseline architecture (unimodal encoders and cross-modal fusion) with MulT. We use a 4-layer Transformer model with the same hyperparameter set and feature set described in the original paper[352]. The output of the fourth (final) layer is used by MMLatch to mask the input features. First, we notice a performance gap between our reproduced results and the ones reported in the original paper (fourth row of Table B.2). Other works[363,364] have reported similar observations. We observe that the integration of MMLatch with MulT yields significant performance improvements across metrics. Furthermore, similarly to Table B.1, we observe that the inclusion of MMLatch reduces standard deviation across metrics. Overall, we observe that the inclusion of MMLatch results to performance improvements for both our baseline model and MulT with

†We use the original code in this GitHub Link

no additional tuning, indicating the effectiveness of the proposed method.

Fig. B.2 shows a heatmap of the average mask values $\frac{1}{2}(f_T + f_A)$. This mask is applied to the input visual features i_V, i.e. 35 FACET features. The average mask values range from 0.36 to 0.65 and depicted across 7 sentiment classes. Some features are attenuated or enhanced across all classes (e.g. features 1 or 32). Interestingly, some features are attenuated for some classes and enhanced for others (e.g. feature 2). More importantly, mask values change almost monotonically as the sentiment value increases from -3 to $+3$, indicating MMLatch training procedure is well-behaved. We observe the same for COVAREP masks.

B.5 Conclusions

We introduce MMLatch, a feedback module that allows modeling top-down cross-modal interactions between higher and lower levels of the architecture. MMLatch is motivated by recent advances in cognitive science, analyzing how cognition affects perception and is implemented as a plug and play framework that can be adapted for modern neural architectures. MMLatch improves model performance over our proposed baseline and over MulT. The combination of MMLatch with our baseline achieves state-of-the-art results. We believe top-down cross-modal modeling can augment traditional bottom-up pipelines, improve performance in multimodal tasks and inspire novel multimodal architectures.

In this work, we implement top-down cross-modal modeling as an adaptive feature masking mechanism. In the future, we plan to explore more elaborate implementations that directly affect the state of the network modules from different levels in the network. Furthermore, we aim to extend MMLatch to more tasks, diverse architectures (e.g. Transformers) and for unimodal architectures. Finally, we will explore applications of top-down masks for model interpretability.

C
Author Bio

About the Author

Georgios Paraskevopoulos received his M.Eng. in Electrical and Computer Engineering from the National Technical University of Athens (NTUA) in 2016; since 2017 he is a Ph.D. student there. Since 2020 he is an associate researcher at the Institute for Speech and Language Processing, Athena RC. He has worked in the industry (Intracom Telecom, Behavioral Signals, Amazon) and on multiple EU research projects. G.P. has co-authored over 20 publications, with over 600 citations, and has served as a reviewer in professional conferences and journals. His research interests revolve around cognitively-motivated neural architectures, the adaptation of neural networks to unseen domains, and the extraction and fusion of multimodal representations. G.P. is a Student Member of the IEEE since 2020.

Publications included in this dissertation

[1] G. Paraskevopoulos, E. Tzinis, E.-V. Vlatakis-Gkaragkounis, and A. Potamianos, "Pattern search multidimensional scaling," *arXiv preprint arXiv:1806.00416*, 2018.

[2] G. Paraskevopoulos, E. Tzinis, N. Ellinas, T. Giannakopoulos, and A. Potamianos, "Unsupervised Low-Rank Representations for Speech Emotion Recognition," in *Interspeech*, 2019, pp. 939–943.

[3] C. Karouzos, G. Paraskevopoulos, and A. Potamianos, "UDALM: Unsupervised domain adaptation through language modeling," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 2579–2590.

[4] G. Paraskevopoulos, E. Georgiou, and A. Potamianos, "Mmlatch: Bottom-up top-down fusion for multimodal sentiment analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 4573–4577.

[5] G. Paraskevopoulos, T. Kouzelis, G. Rouvalis, A. Katsamanis, V. Katsouros, and A. Potamianos, "Sample-efficient unsupervised domain adaptation of speech recognition systems: A case study for modern greek," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 286–299, 2024.

Other publications during the writing of this dissertation

[1] G. Paraskevopoulos, G. Karamanolakis, E. Iosif, A. Pikrakis, and A. Potamianos, "Sensory-aware multimodal fusion for word semantic similarity estimation," in *MultiLearn Workshop*, 2017.

[2] C. Baziotis, A. Nikolaos, A. Kolovou, G. Paraskevopoulos, N. Ellinas, and A. Potamianos, "NTUA-SLP at SemEval-2018 task 2: Predicting emojis using RNNs with context-aware attention," in *Proceedings of the 12th International Workshop on Semantic Evaluation*, 2018, pp. 438–444.

[3] C. Baziotis, A. Nikolaos, A. Chronopoulou, A. Kolovou, G. Paraskevopoulos, N. Ellinas, S. Narayanan, and A. Potamianos, "NTUA-SLP at SemEval-2018 task 1: Predicting affective content in tweets with deep attentive RNNs and transfer learning," in *Proceedings of the 12th International Workshop on Semantic Evaluation*, 2018, pp. 245–255.

[4] C. Baziotis, A. Nikolaos, P. Papalampidi, A. Kolovou, G. Paraskevopoulos, N. Ellinas, and A. Potamianos, "NTUA-SLP at SemEval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive RNNs," in *Proceedings of the 12th International Workshop on Semantic Evaluation*, 2018, pp. 613–621.

[5] E. Tzinis, G. Paraskevopoulos, C. Baziotis, and A. Potamianos, "Integrating Recurrence Dynamics for Speech Emotion Recognition," in *Interspeech*, 2018, pp. 927–931.

[6] A. Chatziagapi, G. Paraskevopoulos, D. Sgouropoulos, G. Pantazopoulos, M. Nikandrou, T. Giannakopoulos, A. Katsamanis, A. Potamianos, and S. Narayanan, "Data augmentation using gans for speech emotion recognition." in *Interspeech*, 2019, pp. 171–175.

[7] D. Xezonaki, G. Paraskevopoulos, A. Potamianos, and S. Narayanan, "Affective Conditioning on Hierarchical Attention Networks Applied to Depression Detection from Transcribed Clinical Interviews," in *Interspeech*, 2020, pp. 4556–4560.

[8] G. Paraskevopoulos, S. Parthasarathy, A. Khare, and S. Sundaram, "Multimodal and multiresolution speech recognition with transformers," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 2381–2387.

[9] N. Ventoura, K. Palios, Y. Vasilakis, G. Paraskevopoulos, N. Katsamanis, and V. Katsouros, "Theano: A greek-speaking conversational agent for covid-19," in *Proceedings of the 1st Workshop on NLP for Positive Impact*, 2021, pp. 36–46.

[10] E. Georgiou, G. Paraskevopoulos, and A. Potamianos, "M3: Multimodal masking applied to sentiment analysis." in *Interspeech*, 2021, pp. 2876–2880.

[11] V. Kouni, G. Paraskevopoulos, H. Rauhut, and G. C. Alexandropoulos, "Admm-dad net: A deep unfolding network for analysis compressed sensing," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 1506–1510.

[12] E. Zaranis, G. Paraskevopoulos, A. Katsamanis, and A. Potamianos, "Empbot: a t5-based empathetic chatbot focusing on sentiments," *arXiv preprint arXiv:2111.00310*, 2021.

[13] E. Georgiou, K. Kritsis, G. Paraskevopoulos, A. Katsamanis, V. Katsouros, and A. Potamianos, "Regotron: Regularizing the tacotron2 architecture via monotonic alignment loss," in *IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 977–983.

[14] G. Paraskevopoulos, P. Pistofidis, G. Banoutsos, E. Georgiou, and V. Katsouros, "Multimodal classification of safety-report observations," *Applied Sciences*, vol. 12, no. 12, p. 5781, 2022.

[15] G. Bastas, M. Kaliakatsos-Papakostas, G. Paraskevopoulos, P. Kaplanoglou, K. Christantonis, C. Tsioustas, D. Mastrogiannopoulos, D. Panga, E. Fotinea, A. Katsamanis *et al.*, "Towards a dhh accessible theater: Real-time synchronization of subtitles and sign language videos with asr and nlp solutions," in *Proceedings of the 15th International Conference on PErvasive Technologies Related to Assistive Environments*, 2022, pp. 653–661.

[16] C. Sartzetaki, G. Paraskevopoulos, and A. Potamianos, "Extending Compositional Attention Networks for Social Reasoning in Videos," in *Interspeech*, 2022, pp. 1116–1120.

[17] O. S. Chlapanis, G. Paraskevopoulos, and A. Potamianos, "Adapted multimodal bert with layer-wise fusion for sentiment analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[18] G. Paraskevopoulos, C. Lavania, L. Chum, and S. Sundaram, "Multi-scale compositional constraints for representation learning on videos," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[19] I. Triantafyllopoulos, G. Paraskevopoulos, and A. Potamianos, "Depression detection in social media posts using affective and social norm features," *arXiv preprint arXiv:2303.14279*, 2023.

[20] T. Kouzelis, G. Paraskevopoulos, A. Katsamanis, and V. Katsouros, "Weakly-supervised forced alignment of disfluent speech using phoneme-level modeling," in *Interspeech*, 2023, pp. 1563–1567.

[21] M. Plitsis, T. Kouzelis, G. Paraskevopoulos, V. Katsouros, and Y. Panagakis, "Investigating personalization methods in text to music generation," *ICASSP (accepted)*, 2024.

D
Glossary

Acronyms

Abbr.	Description	Greek Translation	Page List
k-NN	k Nearest Neighbor.	k Κοντινότεροι Γείτονες	13, 17, 18, 31, 81, 83, 88, 93–95, 97
AI	Artificial Intelligence.	Τεχνητή Νοημοσύνη	47, 49
ANN	Artificial Neural Networks.	Τεχνητά Νευρωνικά Δίκτυα	54
ASR	Automatic Speech Recognition.	Αυτόματη Αναγνώριση Φωνής	21, 22, 101–107, 121, 122, 124, 125, 127, 128, 131, 132, 138, 139
cMDS	Classical Multi-Dimensional Scaling.	Κλασική Πολυδιάστατη Κλιμάκωση	61–63
CoD	Curse of Dimensionality.	Κατάρα της Διαστατικότητας	91
CPT	Continuous Pre-Training.	Συνεχής Προεκπαίδευση	35, 36, 43, 44, 105, 107, 131–133, 135
CTC	Connectionist Temporal Classification.	Συνδετιστική Χρονική Ταξινόμιση	35, 41, 43, 44, 102, 105, 124, 130–132

Abbr.	Description	Greek Translation	Page List
DAT	Domain Adversarial Training.	Ανταγωνιστική Εκπαίδευση Πεδίου	17, 19, 36, 38–40, 103–106, 109, 112–114, 116–119
DPT	Domain Pre-Training.	Προεκπαίδευση Πεδίου	17, 19, 38–40, 107, 112–114, 116–118
DR	Dimensionality Reduction.	Μείωση Διαστατικότητας	13, 18, 21, 60, 61, 81–84, 86–89, 91–97, 143
DSM	Distributional Semantic Model.	Κατανεμημένα Σημασιολογικά Δίκτυα	144
EWC	Elastic Weight Consolidation.	Ελαστική Συγχώνευση Βαρών	36, 54, 55, 107
GloVe	Global Vectors for word representations.	Συνολικά Διανύσματα για αναπαράσταση λέξεων	27, 83, 88
GPS	General Pattern Search.	Γενική Αναζήτηση Προτύπων	23, 27, 60, 62, 65–69, 73–76, 143
GPT-4	General Pre-Training model version 4.	Μοντέλο Γενικής Προεκπαίδευσης (4η έκδοση)	49
IS10	Interspeech 2010.	Interspeech 2010	18, 21, 29–31, 33, 92–95, 97
ISOMAP	Isometric Feature Mapping.	Ισομετρική Απεικόνιση	17, 18, 26, 29, 33, 61, 62, 81, 83, 84, 86, 88, 92–94, 96, 97
KL	Kullback–Leibler.	Kullback-Leibler	106
LDA	Linear Discriminant Analysis.	Γραμμική Διακριτική Ανάλυση	91, 92
LE	Laplacian Eigenmaps.	Λαπλασιανή Ιδιοαπεικόνιση	61, 62, 93, 94

Abbr.	Description	Greek Translation	Page List
LLE	Locally Linear Embedding.	Τοπικά Γραμμική Ενσωμάτωση	26, 27, 29, 30, 32, 33, 61, 62, 81, 83, 84, 86, 88, 89, 92–94, 96, 97
LM	Language Model.	Γλωσσική Μοντελοποίηση	19, 21, 22, 43–45, 104, 128, 130–132, 135–138
LR	Logistic Regression.	Λογιστική Παλινδρόμηση	93, 95
LTSA	Local Tangent Space Alignment.	Τοπική Εφαπτομενική Χωρική Ευθυγράμμιση	26, 29, 81, 83, 84, 86, 88
M2DS2	Mixed Multi-Domain Self-Supervision.	Μικτή Πολυ-πεδιακή Αυτοεπίβλεψη	17, 19, 21, 22, 39, 40, 43–45, 122, 130–138
MDS	Multi-Dimensional Scaling.	Πολυδιάστατη Κλιμάκωση	17, 18, 23, 26–30, 32, 33, 46, 54, 55, 59–64, 69, 73, 74, 76–78, 81–89, 93–97, 143, 144
MLM	Masked Language Modeling.	Μασκοφόρα Γλωσσική Μοντελοποίηση	17, 19, 36–38, 110–113
MLP	Multi-Layer Perceptron.	Πολυεπίπεδο Αντίληπτρο	49
MSE	Mean Squared Error.	Μέσο Τετραγωνικό Σφάλμα	23, 69, 74
NLDR	Non-Linear Dimensionality Reduction.	Μη Γραμμική Μείωση Διαστατικότητας	59, 81
NLP	Natural Language Processing.	Επεξεργασία Φυσικής Γλώσσας	17, 49, 50, 107, 109, 118, 122, 123, 138
PCA	Principal Component Analysis.	Ανάλυση Κύριων Συνιστωσών	18, 30, 31, 33, 61–63, 91–95, 97
PSL	Pseudo-Labeling.	Ψευδοεπισημείωση	43, 44, 103, 105, 107, 132

Abbr.	Description	Greek Translation	Page List
RBF	Radial Basis Function.	Συνάρτηση Βάσης Ακτινικού Τύπου	93, 95
RNN	Recurrent Neural Network.	Αναδρομικά Νευρωνικά Δίκτυα	148
RQA	Recurrence Quantification Analysis.	Αναδρομική Ποσοτική Ανάλυση	18, 30, 31, 92–94
SDE	Semi-Definite Embedding.	Ημιπροσδιορισμένη Ενσωμάτωση	61, 62
SER	Speech Emotion Recognition.	Αναγνώριση Συναισθήματος από Φωνή	29, 55, 91–93, 97, 143
SMACOF	Scaling by Majorizing a Complicated Function.	Κλιμάκωση μέσω της Κυριαρχίας μιας Σύνθετης Συνάρτησης	18, 26, 27, 29, 64, 78, 81, 83–86, 88, 89, 93
SO	Source-Only.	Πηγαίο-Μόνο	17, 19, 21, 22, 38, 39, 43–45, 112, 113, 116–118, 131–133, 136, 137
SSL	Self-Supervised Learning.	Αυτοεπιβλεπόμενη Μάθηση	49, 107, 109, 145
SVD	Singular Value Decomposition.	Διάσπαση Ιδιάζουσων Τιμών	26, 29, 81, 83, 84, 88
SVM	Support Vector Machines.	Μηχανές Διανυσματικής Υποστήριξης	93, 95
UDA	Unsupervised Domain Adaptation.	Μη Επιβλεπόμενη Προσαρμογή Πεδίου	14, 21, 22, 36, 46, 55, 56, 101–105, 108, 109, 112, 113, 115, 117, 118, 121–123, 130–132, 138, 139, 143–145
UDALM	Unsupervised Domain Adaptation through Language Modeling.	Μη Επιβλεπόμενη Προσαρμογή Πεδίου μέσω Γλωσσικής Μοντελοποίησης	17, 19, 36–40, 109–114, 116–119, 123

Abbr.	Description	Greek Translation	Page List
WER	Word Error Rate.	Λεκτικός Ρυθμός Σφάλματος	17, 19, 21, 22, 42–45, 125, 130–133, 135, 137, 138
WRR	Word error Rate Recovery.	Ανάκτηση Λεκτικού Ρυθμού Σφάλματος	21, 22, 43, 131, 132

Bibliography

[1] J. Togelius and G. N. Yannakakis, "Choose your weapon: Survival strategies for depressed ai academics," *arXiv preprint arXiv:2304.06035*, 2023.

[2] O. Russakovsky, J. Deng *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, pp. 211–252, 2015.

[3] L. Ouyang, J. Wu *et al.*, "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 27 730–27 744.

[4] OpenAI, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[5] H. M. Wellman, *The child's theory of mind*. The MIT Press, 1992.

[6] S. Bubeck, V. Chandrasekaran *et al.*, "Sparks of artificial general intelligence: Early experiments with gpt-4," *arXiv preprint arXiv:2303.12712*, 2023.

[7] A. Vaswani, N. Shazeer *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008.

[8] B. L. Edelman, S. Goel, S. Kakade, and C. Zhang, "Inductive biases and variable creation in self-attention mechanisms," in *Proceedings of the International Conference on Machine Learning*. PMLR, 2022, pp. 5793–5831.

[9] L. Ericsson, H. Gouk, C. C. Loy, and T. M. Hospedales, "Self-supervised representation learning: Introduction, advances, and challenges," *IEEE Signal Processing Magazine*, vol. 39, no. 3, pp. 42–62, 2022.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019, pp. 4171–4186.

[11] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2536–2544.

[12] P. H. Le-Khac, G. Healy, and A. F. Smeaton, "Contrastive representation learning: A framework and review," *Ieee Access*, vol. 8, pp. 193 907–193 934, 2020.

[13] N. Bannour, S. Ghannay, A. Névéol, and A.-L. Ligozat, "Evaluating the carbon footprint of nlp methods: a survey and analysis of existing tools," in *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, 2021, pp. 11–21.

[14] B. Everman, T. Villwock *et al.*, "Evaluating the carbon impact of large language models at the inference stage," in *2023 IEEE International Performance, Computing, and Communications Conference (IPCCC)*. IEEE, 2023, pp. 150–157.

[15] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots: Can language models be too big?" in *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, 2021, pp. 610–623.

[16] S. McGuire, E. Schultz, B. Ayoola, and P. Ralph, "Sustainability is stratified: Toward a better theory of sustainable software engineering," in *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, 2023, pp. 1996–2008.

[17] S. A. Khowaja, P. Khuwaja, and K. Dev, "Chatgpt needs spade (sustainability, privacy, digital divide, and ethics) evaluation: A review," *arXiv preprint arXiv:2305.03123*, 2023.

[18] Y. Dar, V. Muthukumar, and R. G. Baraniuk, "A farewell to the bias-variance tradeoff? an overview of the theory of overparameterized machine learning," *arXiv preprint arXiv:2109.02355*, 2021.

[19] A. Tsigler and P. L. Bartlett, "Benign overfitting in ridge regression," *Journal of Machine Learning Research*, vol. 24, no. 123, pp. 1–76, 2023.

[20] M. Belkin, D. Hsu, S. Ma, and S. Mandal, "Reconciling modern machine-learning practice and the classical bias–variance trade-off," *Proceedings of the National Academy of Sciences*, vol. 116, no. 32, pp. 15 849–15 854, 2019.

[21] P. L. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler, "Benign overfitting in linear regression," *Proceedings of the National Academy of Sciences*, vol. 117, no. 48, pp. 30 063–30 070, 2020.

[22] Y. Cao, Z. Chen, M. Belkin, and Q. Gu, "Benign overfitting in two-layer convolutional neural networks," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 25 237–25 250.

[23] J. Kaplan, S. McCandlish *et al.*, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[24] J. Hoffmann, S. Borgeaud *et al.*, "An empirical analysis of compute-optimal large language model training," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 30 016–30 030.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[26] A. Dosovitskiy, L. Beyer *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[27] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 10 674–10 685.

[28] A. Conneau, K. Khandelwal *et al.*, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 8440–8451.

[29] A. Radford, J. W. Kim *et al.*, "Robust speech recognition via large-scale weak supervision," in *Proceedings of the International Conference on Machine Learning*. PMLR, 2023, pp. 28 492–28 518.

[30] M. Le, A. Vyas *et al.*, "Voicebox: Text-guided multilingual universal speech generation at scale," in *Advances in Neural Information Processing Systems*, 2023.

[31] H. Liu, Z. Chen *et al.*, "AudioLDM: Text-to-audio generation with latent diffusion models," in *Proceedings of the International Conference on Machine Learning*, vol. 202. PMLR, 2023, pp. 21 450–21 474.

[32] T. Brown, B. Mann *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[33] A. Aghajanyan, L. Yu *et al.*, "Scaling laws for generative mixed-modal language models," *arXiv preprint arXiv:2301.03728*, 2023.

[34] S. Herculano-Houzel, "The remarkable, yet not extraordinary, human brain as a scaled-up primate brain and its associated cost," *Proceedings of the National Academy of Sciences*, vol. 109, no. supplement_1, pp. 10 661–10 668, 2012.

[35] S. Herculano-Houzel, "The human brain in numbers: a linearly scaled-up primate brain," *Frontiers in human neuroscience*, p. 31, 2009.

[36] D. A. Drachman, "Do we have brain to spare?" *Neurology*, vol. 64, no. 12, pp. 2004–2005, 2005.

[37] E. H. Chudler, "Brain facts and figures," available at https://faculty.washington.edu/chudler/facts.html. Accessed on 22/11/2023.

[38] G. M. Shepherd, *The synaptic organization of the brain*. Oxford university press, 2003.

[39] D. Purves, G. J. Augustine *et al.*, *Neuroscience, 6th Edition*. Oxford University Press, 2021.

[40] B. Millidge, "The scale of the brain vs machine learning," available at https://www.beren.io/2022-08-06-The-scale-of-the-brain-vs-machine-learning/. Accessed on 22/11/2023.

[41] M. Brysbaert, "How many words do we read per minute? a review and meta-analysis of reading rate," *Journal of Memory and Language*, vol. 109, p. 104047, 2019.

[42] C. Fefferman, S. Mitter, and H. Narayanan, "Testing the manifold hypothesis," *Journal of the American Mathematical Society*, vol. 29, no. 4, pp. 983–1049, 2016.

[43] L. Cayton, "Algorithms for manifold learning."

[44] P. P. Brahma, D. Wu, and Y. She, "Why deep learning works: A manifold disentanglement perspective," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 10, pp. 1997–2008, 2015.

[45] P. Gardenfors, *Conceptual spaces: The geometry of thought*. MIT press, 2004.

[46] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, 1964.

[47] S. Grossberg, "Competitive learning: From interactive activation to adaptive resonance," *Cognitive science*, vol. 11, no. 1, pp. 23–63, 1987.

[48] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of Learning and Motivation*. Academic Press, 1989, vol. 24, pp. 109–165.

[49] R. M. French, "Semi-distributed representations and catastrophic forgetting in connectionist networks," *Connection Science*, vol. 4, no. 3-4, pp. 365–377, 1992.

[50] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[51] J. Kirkpatrick, R. Pascanu *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[52] J. Cichon and W.-B. Gan, "Branch-specific dendritic ca2+ spikes cause persistent synaptic plasticity," *Nature*, vol. 520, no. 7546, pp. 180–185, 2015.

[53] G. Yang, F. Pan, and W.-B. Gan, "Stably maintained dendritic spines are associated with lifelong memories," *Nature*, vol. 462, no. 7275, pp. 920–924, 2009.

[54] T. L. Hayes, G. P. Krishnan *et al.*, "Replay in deep learning: Current approaches and missing biological elements," *Neural Computation*, vol. 33, pp. 2908–2950, 2021.

[55] A. Chronopoulou, C. Baziotis, and A. Potamianos, "An embarrassingly simple approach for transfer learning from pretrained language models," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019, pp. 2089–2095.

[56] Y. Ganin, E. Ustinova *et al.*, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[57] P. Kanerva, *Sparse distributed memory*. MIT press, 1988.

[58] E. Pothos and J. Busemeyer, "A quantum probability explanation for violations of symmetry in similarity judgments," in *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 33, 2011.

[59] E. M. Pothos and J. R. Busemeyer, "Can quantum probability provide a new direction for cognitive modeling?" *Behavioral and Brain Sciences*, vol. 36, no. 3, pp. 255–274, 2013.

[60] G. Paraskevopoulos, E. Tzinis, V. E., and P. A., "Pattern Search Multidimensional Scaling," 2018, arXiv:1806.00416v2.

[61] E. Tzinis, "Manifold learning and nonlinear recurrence dynamics for speech emotion recognition on various timescales," 2018, available at https://dspace.lib.ntua.gr/xmlui/bitstream/handle/123456789/47369/

etzinis_thesis_english.pdf. Accessed on 22/11/2023.

[62] G. Paraskevopoulos, E. Tzinis, N. Ellinas, T. Giannakopoulos, and A. Potamianos, "Unsupervised Low-Rank Representations for Speech Emotion Recognition," in *Proceedings Interspeech*, 2019, pp. 939–943.

[63] W. S. Torgerson, "Multidimensional scaling: I. theory and method," *Psychometrika*, vol. 17, no. 4, pp. 401–419, 1952.

[64] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[65] M. Bernstein, V. D. Silva, J. C. Langford, and J. B. Tenenbaum, "Graph approximations to geodesics on embedded manifolds," 2000.

[66] H. Zha and Z. Zhang, "Isometric embedding and continuum isomap," in *Proceedings of the International Conference on Machine Learning*, 2003, pp. 864–871.

[67] D. L. Donoho and C. Grimes, "Image manifolds which are isometric to euclidean space," *Journal of Mathematical Imaging and Vision*, vol. 23, no. 1, pp. 5–24, 2005.

[68] R. Pless, "Image spaces and video trajectories: Using isomap to explore video sequences." in *ICCV*, vol. 3, 2003, pp. 1433–1440.

[69] V. Silva and J. B. Tenenbaum, "Sparse multidimensional scaling using landmark points," *Technology*, 2004.

[70] V. D. Silva and J. B. Tenenbaum, "Global versus local methods in nonlinear dimensionality reduction," in *Advances in Neural Information Processing Systems*, 2003, pp. 705–712.

[71] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[72] L. Cayton and S. Dasgupta, "Robust euclidean embedding," in *Proceedings of the International Conference on Machine Learning*, 2006, pp. 169–176.

[73] L. K. Saul and S. T. Roweis, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds," *Journal of Machine Learning Research*, vol. 4, pp. 119–155, 2003.

[74] F. Sha and L. K. Saul, "Analysis and extension of spectral methods for nonlinear dimensionality reduction," in *Proceedings of the International Conference on Machine Learning*, 2005, pp. 784–791.

[75] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems*, 2002, pp. 585–591.

[76] Z. Zhang and J. Wang, "Mlle: Modified locally linear embedding using multiple weights," in *Advances in Neural Information Processing Systems*, 2007, pp. 1593–1600.

[77] D. L. Donoho and C. Grimes, "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data," *Proceedings of the National Academy of Sciences*, vol. 100, no. 10, pp. 5591–5596, 2003.

[78] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *International journal of computer vision*, vol. 70, no. 1, pp. 77–90, 2006.

[79] K. Q. Weinberger, B. Packer, and L. K. Saul, "Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization." in *AISTATS*. Citeseer, 2005.

[80] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, no. 1, pp. 49–95, 1996.

[81] D. P. Bertsekas, *Nonlinear programming*. Athena scientific Belmont, 1999.

[82] M. Belkin and P. Niyogi, "Convergence of laplacian eigenmaps," in *Advances in Neural Information Processing Systems*, 2007, pp. 129–136.

[83] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM Journal of Scientific Computing*, vol. 26, no. 1, pp. 313–338, 2005.

[84] J. B. Kruskal, "Nonmetric multidimensional scaling: A numerical method," *Psychometrika*, vol. 29, no. 2, pp.

115–129, 1964.

[85] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[86] D. Ballard, "Modular learning in neural networks," in *Proceedings of the Sixth National Conference on Artificial Intelligence*, 1987, pp. 279–284.

[87] L. M. Rios and N. V. Sahinidis, "Derivative-free optimization: a review of algorithms and comparison of software implementations," *Journal of Global Optimization*, vol. 56, no. 3, pp. 1247–1293, 2013.

[88] M. Avriel, *Nonlinear programming: analysis and methods*. Courier Corporation, 2003.

[89] R. Hooke and T. A. Jeeves, "" direct search" solution of numerical and statistical problems," *J. ACM*, vol. 8, no. 2, pp. 212–229, 1961.

[90] G. E. Box, "Evolutionary operation: A method for increasing industrial productivity," *Applied statistics*, pp. 81–101, 1957.

[91] V. J. Torczon, "Multidirectional search: a direct search algorithm for parallel machines," Ph.D. dissertation, Rice University, 1989.

[92] J. J. E. Dennis and V. Torczon, "Direct search methods on parallel machines," *SIAM Journal on Optimization*, vol. 1, no. 4, pp. 448–474, 1991.

[93] V. Torczon, "On the convergence of pattern search algorithms," *SIAM Journal on optimization*, vol. 7, no. 1, pp. 1–25, 1997.

[94] E. D. Dolan, R. M. Lewis, and V. Torczon, "On the local convergence of pattern search," *SIAM Journal on Optimization*, vol. 14, no. 2, pp. 567–583, 2003.

[95] R. M. Lewis and V. Torczon, "Pattern search methods for linearly constrained minimization," *SIAM Journal on Optimization*, vol. 10, no. 3, pp. 917–941, 2000.

[96] R. M. Lewis and V. Torczon, "Pattern search algorithms for bound constrained minimization," *SIAM Journal on Optimization*, vol. 9, no. 4, pp. 1082–1099, 1999.

[97] A. R. Conn, N. I. M. Gould, and P. Toint, "A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple bounds," *SIAM Journal on Numerical Analysis*, vol. 28, no. 2, pp. 545–572, 1991.

[98] I. Borg and P. J. F. Groenen, *Modern multidimensional scaling: theory and applications*. Springer, 2005.

[99] J. C. Gower, "Some distance properties of latent root and vector methods used in multivariate analysis," *Biometrika*, vol. 53, no. 3-4, pp. 325–338, 1966.

[100] M. A. A. Cox and T. F. Cox, "Multidimensional scaling on the sphere," in *Compstat*. Physica-Verlag HD, 1988, pp. 323–328.

[101] A. Cvetkovski and M. Crovella, "Low-stress data embedding in the hyperbolic plane using multidimensional scaling," *Appl. Math*, vol. 11, no. 1, pp. 5–12, 2017.

[102] H. Lindman and T. Caelli, "Constant curvature riemannian scaling," *Journal of Mathematical Psychology*, vol. 17, no. 2, pp. 89–109, 1978.

[103] R. N. Shepard, "The analysis of proximities: Multidimensional scaling with an unknown distance function. i." *Psychometrika*, vol. 27, no. 2, pp. 125–140, 1962.

[104] R. N. Shepard, "The analysis of proximities: Multidimensional scaling with an unknown distance function. ii," *Psychometrika*, vol. 27, no. 3, pp. 219–246, 1962.

[105] P. Groenen and I. Borg, "Past, present, and future of multidimensional scaling," *Visualization and Verbalization of Data*, pp. 95–117, 2014.

[106] S. L. France and J. D. Carroll, "Two-way multidimensional scaling: A review," *IEEE Transactions on Systems,*

Man, and Cybernetics, Part C (Applications and Reviews), vol. 41, no. 5, pp. 644–661, 2011.

[107] J. D. Leeuw, I. J. R. Barra, F. Brodeau, G. Romier, and B. V. C. (eds, "Applications of convex analysis to multidimensional scaling," in *Recent Developments in Statistics*. North Holland Publishing Company, 1977, pp. 133–146.

[108] J. de Leeuw, "Convergence of the majorization method for multidimensional scaling," *Journal of Classification*, vol. 5, no. 2, pp. 163–180, 1988.

[109] C. Audet, "Convergence results for generalized pattern search algorithms are tight," *Optimization and Engineering*, vol. 5, no. 2, pp. 101–122, 2004.

[110] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.

[111] L. Dagum and R. Menon, "Openmp: an industry standard api for shared-memory programming," *IEEE computational science and engineering*, vol. 5, no. 1, pp. 46–55, 1998.

[112] G. Golub and W. Kahan, "Calculating the singular values and pseudo-inverse of a matrix," *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis*, vol. 2, no. 2, pp. 205–224, 1965.

[113] E. Bruni, N. K. Tran, and M. Baroni, "Multimodal distributional semantics," *Journal Artifial Intelligence Research*, vol. 49, no. 1, pp. 1–47, 2014.

[114] F. Hill, R. Reichart, and A. Korhonen, "Simlex-999: Evaluating semantic models with (genuine) similarity estimation," *Computational Linguistics*, 2015.

[115] C. Baziotis, N. Pelekis, and C. Doulkeridis, "Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017, pp. 747–754.

[116] T. Yang, J. Liu, L. Mcmillan, and W. Wang, "A fast approximation to multidimensional scaling," in *In Proceedings of the of the IEEE Workshop on Computation Intensive Methods for Computer Vision*, 2006.

[117] K. Rajawat and S. Kumar, "Stochastic multidimensional scaling," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 2, pp. 360–375, 2017.

[118] R. Bellman, *Adaptive control processes: a guided tour*. Princeton University Press, 2015, vol. 2045.

[119] B.-C. Chiou and C.-P. Chen, "Feature space dimension reduction in speech emotion recognition using support vector machine," in *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2013, pp. 1–6.

[120] K. F. Pearson, "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[121] B. Schuller, B. Vlasenko, F. Eyben, G. Rigoll, and A. Wendemuth, "Acoustic emotion recognition: A benchmark comparison of performances," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, 2009, pp. 552–557.

[122] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, and B. Weiss, "A database of german emotional speech," in *Proceedings Interspeech*, 2005, pp. 1517–1520.

[123] J. Yuan, L. Chen, T. Fan, and J. Jia, "Dimension reduction of speech emotion feature based on weighted linear discriminate analysis," *Image Processing and Pattern Recognition*, vol. 8, pp. 299–308, 2015.

[124] R. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[125] M. You, C. Chen, J. Bu, J. Liu, and J. Tao, "Emotion recognition from noisy speech," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2006, pp. 1653–1656.

[126] M. You, C. Chen, J. Bu, J. Liu, and J. Tao, "A hierarchical framework for speech emotion recognition," in *Proceedings of the IEEE International Symposium on Industrial Electronics*, 2006, pp. 515–519.

[127] K. Fu, *Sequential methods in pattern recognition and machine learning*. Academic Press, 1968, vol. 52.

[128] D. Ververidis, C. Kotropoulos, and I. Pitas, "Automatic emotional speech classification," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2004, pp. 593–596.

[129] I. Engberg, A. Hansen, O. Andersen, and P. Dalsgaard, "Design, recording and verification of a danish emotional speech database," in *Proceedings of the Fifth European Conference on Speech Communication and Technology (EUROSPEECH)*, 1997, pp. 1695–1698.

[130] C. Lee, S. Narayanan, and R. Pieraccini, "Classifying emotions in human-machine spoken dialogs," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2002, pp. 737–740.

[131] Z.-J. Chuang and C.-H. Wu, "Emotion recognition using acoustic features and textual content," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2004, pp. 53–56.

[132] A. Farahat, A. Ghodsi, and M. Kamel, "An efficient greedy method for unsupervised feature selection," in *Proceedings of the IEEE 11th International Conference on Data Mining (ICDM)*, 2011, pp. 161–170.

[133] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, no. 1, p. 1, 2010.

[134] P. Fewzee and F. Karray, "Dimensionality reduction for emotional speech recognition," in *Proceedings of the ASE/IEEE International Conference on Privacy, Security, Risk and Trust (PASSAT) and International Confernece on Social Computing (SocialCom)*, 2012, pp. 532–537.

[135] S. Zhang and X. Zhao, "Dimensionality reduction-based spoken emotion recognition," *Multimedia Tools and Applications*, vol. 63, no. 3, pp. 615–646, 2013.

[136] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[137] D. De Ridder, O. Kouropteva, O. Okun, M. Pietikäinen, and R. Duin, "Supervised locally linear embedding," in *Artificial Neural Networks and Neural Information Processing ICANN/ICONIP*. Springer Berlin Heidelberg, 2003, pp. 333–341.

[138] J. Goldberger, G. Hinton, S. Roweis, and R. Salakhutdinov, "Neighbourhood components analysis," in *Advances in Neural Information Processing Systems*, 2005, pp. 513–520.

[139] A. Globerson and S. Roweis, "Metric learning by collapsing classes," in *Advances in Neural Information Processing Systems*, 2006, pp. 451–458.

[140] M. Sugiyama, "Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis," *Journal of Machine Learning Research*, vol. 8, no. May, pp. 1027–1061, 2007.

[141] B. Schuller, S. Steidl *et al.*, "The Interspeech 2010 paralinguistic challenge," in *Proceedings Interspeech*, 2010, pp. 2794–2797.

[142] E. Tzinis, G. Paraskevopoulos, C. Baziotis, and A. Potamianos, "Integrating recurrence dynamics for speech emotion recognition," in *Proceedings Interspeech*, 2018, pp. 927–931.

[143] C. Busso, M. Bulut *et al.*, "IEMOCAP: Interactive emotional dyadic motion capture database," *Language Resources and Evaluation*, vol. 42, no. 4, p. 335, 2008.

[144] Z. Aldeneh and E. Provost, "Using regional saliency for speech emotion recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2741–2745.

[145] R. Xia and Y. Liu, "A multi-task learning framework for emotion recognition using 2d continuous space," *IEEE Transactions on Affective Computing*, no. 1, pp. 3–14, 2017.

[146] H. Fayek, M. Lech, and L. Cavedon, "Evaluating deep learning architectures for speech emotion recognition," *Neural Networks*, vol. 92, pp. 60–68, 2017.

[147] S. Ghosh, E. Laksana, L.-P. Morency, and S. Scherer, "Representation learning for speech emotion recognition," in *Proceedings Interspeech*, 2016, pp. 3603–3607.

[148] A. C. et al., "Unsupervised Cross-Lingual Representation Learning for Speech Recognition," in *Proceedings Interspeech*, 2021, pp. 2426–2430.

[149] C. Karouzos, G. Paraskevopoulos, and A. Potamianos, "UDALM: Unsupervised domain adaptation through language modeling," in *Proceedings of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2021, pp. 2579–2590.

[150] G. Paraskevopoulos, T. Kouzelis *et al.*, "Sample-efficient unsupervised domain adaptation of speech recognition systems: A case study for modern greek," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 286–299, 2024.

[151] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the of the 23rd Int. Conf. on Machine Learning*. Association for Computing Machinery, 2006, p. 369–376.

[152] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 1995, pp. 189–196.

[153] Z.-H. Zhou and M. Li, "Tri-training: Exploiting unlabeled data using three classifiers," *IEEE Transactions on knowledge and Data Engineering*, vol. 17, no. 11, pp. 1529–1541, 2005.

[154] J. Blitzer, R. McDonald, and F. Pereira, "Domain adaptation with structural correspondence learning," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2006, pp. 120–128.

[155] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen, "Cross-domain sentiment classification via spectral feature alignment," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 751–760.

[156] D. McClosky, E. Charniak, and M. Johnson, "Reranking and self-training for parser adaptation," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2006, pp. 337–344.

[157] S. Abney, *Semisupervised learning for computational linguistics*. CRC Press, 2007.

[158] A. Søgaard, "Simple semi-supervised training of part-of-speech taggers," in *Proceedings of the ACL 2010 Conference Short Papers*. Association for Computational Linguistics, 2010, pp. 205–208.

[159] K. Saito, Y. Ushiku, and T. Harada, "Asymmetric tri-training for unsupervised domain adaptation," in *Proceedings of the International Conference on Machine Learning*, vol. 70. PMLR, 2017, pp. 2988–2997.

[160] S. Ruder and B. Plank, "Strong baselines for neural semi-supervised learning under domain shift," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2018, pp. 1044–1054.

[161] G. Rotman and R. Reichart, "Deep contextualized self-training for low resource dependency parsing," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 695–713, 2019.

[162] K. Lim, J. Y. Lee, J. Carbonell, and T. Poibeau, "Semi-supervised learning on meta structure: Multi-task tagging and parsing in low-resource scenarios," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, pp. 8344–8351, 2020.

[163] H. Ye, Q. Tan *et al.*, "Feature Adaptation of Pre-Trained Language Models across Languages and Domains for Text Classification," *arXiv:2009.11538 [cs]*, 2020, arXiv: 2009.11538.

[164] Y. Ziser and R. Reichart, "Neural structural correspondence learning for domain adaptation," in *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*. Association for Computational Linguistics, 2017, pp. 400–410.

[165] Y. Ziser and R. Reichart, "Pivot based language modeling for improved neural domain adaptation," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, 2018,

pp. 1241–1251.

[166] Y. Ziser and R. Reichart, "Task refinement learning for improved accuracy and stability of unsupervised domain adaptation," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019, pp. 5895–5906.

[167] T. Miller, "Simplified neural unsupervised domain adaptation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019, pp. 414–419.

[168] Z. Li, Y. Wei, Y. Zhang, and Q. Yang, "Hierarchical attention transfer network for cross-domain sentiment classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[169] E. Ben-David, C. Rabinovitz, and R. Reichart, "Perl: Pivot-based domain adaptation for pre-trained deep contextualized embedding models," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 504–521, 2020.

[170] A. Ramponi and B. Plank, "Neural unsupervised domain adaptation in nlp—a survey," *arXiv preprint arXiv:2005.14672*, 2020.

[171] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Advances in Neural Information Processing Systems*, 2007, pp. 137–144.

[172] S. Ben-David, J. Blitzer *et al.*, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[173] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of the International Conference on Machine Learning*, vol. 37. PMLR, 2015, pp. 1180–1189.

[174] Y. Li, T. Baldwin, and T. Cohn, "What's in a domain? learning domain-robust text representations using adversarial training," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. Association for Computational Linguistics, 2018, pp. 474–479.

[175] F. Alam, S. Joty, and M. Imran, "Domain adaptation with adversarial training and graph embeddings," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2018, pp. 1077–1087.

[176] M. Sato, H. Manabe, H. Noji, and Y. Matsumoto, "Adversarial training for cross-domain Universal Dependency parsing," in *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics, 2017, pp. 71–79.

[177] C. Du, H. Sun, J. Wang, Q. Qi, and J. Liao, "Adversarial and domain-aware BERT for cross-domain sentiment analysis," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 4019–4028.

[178] H. Zhao, S. Zhang *et al.*, "Adversarial multiple source domain adaptation," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 8559–8570.

[179] J. Guo, D. Shah, and R. Barzilay, "Multi-source domain adaptation with mixture of experts," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018, pp. 4694–4703.

[180] H. Guo, R. Pasunuru, and M. Bansal, "Multi-source domain adaptation for text classification via distancenet-bandits." in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 7830–7838.

[181] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, pp. 4058–4065.

[182] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Advances in neural information processing systems*, 2016, pp. 343–351.

[183] J. Lee, W. Yoon *et al.*, "Biobert: a pre-trained biomedical language representation model for biomedical text

mining." *Bioinformatics (Oxford, England)*, vol. 36, no. 4, p. 1234, 2020.

[184] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 2019, pp. 3615–3620.

[185] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune bert for text classification?" in *China National Conference on Chinese Computational Linguistics*. Springer, 2019, pp. 194–206.

[186] H. Xu, B. Liu, L. Shu, and P. Yu, "BERT post-training for review reading comprehension and aspect-based sentiment analysis," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019, pp. 2324–2335.

[187] S. Gururangan, A. Marasović *et al.*, "Don't stop pretraining: Adapt language models to domains and tasks," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 8342–8360.

[188] X. Han and J. Eisenstein, "Unsupervised domain adaptation of contextualized embeddings for sequence labeling," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 2019, pp. 4238–4248.

[189] C. Jia, X. Liang, and Y. Zhang, "Cross-domain NER using cross-domain language modeling," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019, pp. 2464–2474.

[190] D. Hwang, A. Misra *et al.*, "Large-scale asr domain adaptation using self- and semi-supervised learning," in *Advances in Neural Information Processing Systems*, 2022, pp. 6627–6631.

[191] S. Khurana, N. Moritz, T. Hori, and J. L. Roux, "Unsupervised domain adaptation for speech recognition via uncertainty driven self-training," pp. 6553–6557, 2020.

[192] S. Panchapagesan, D. S. Park *et al.*, "Efficient knowledge distillation for rnn-transducer models," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5639–5643.

[193] A. G. et al, "Conformer: Convolution-augmented transformer for speech recognition," in *Proceedings Interspeech*, 2020, pp. 5036–5040.

[194] A. Graves, "Sequence transduction with recurrent neural networks," *CoRR*, vol. abs/1211.3711, 2012.

[195] J. Li, M. L. Seltzer, X. Wang, R. Zhao, and Y. Gong, "Large-scale domain adaptation via teacher-student learning," in *Proceedings Interspeech*, 2017.

[196] V. Manohar, P. Ghahremani, D. Povey, and S. Khudanpur, "A teacher-student learning approach for unsupervised domain adaptation of sequence-trained asr models," in *Proceedings of the Spoken Language Technology Workshop (SLT)*, 2018.

[197] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proceedings Interspeech*, 2014, pp. 338–342.

[198] T. Asami, R. Masumura, Y. Yamaguchi, H. Masataki, and Y. Aono, "Domain adaptation of dnn acoustic models using knowledge distillation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5185–5189.

[199] T. Yoshioka, N. Ito *et al.*, "The ntt chime-3 system: Advances in speech enhancement and recognition for mobile multi-microphone devices," in *Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 436–443.

[200] I. Leal, N. Gaur *et al.*, "Self-adaptive distillation for multilingual speech recognition: Leveraging student

independence." in *Proceedings Interspeech*, 2021.

[201] Y. He, T. N. Sainath *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6381–6385.

[202] S. Khurana, A. Laurent, and J. Glass, "Magic dust for cross-lingual adaptation of monolingual wav2vec-2.0," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6647–6651.

[203] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 12 449–12 460.

[204] Y. Shinohara, "Adversarial Multi-Task Learning of Deep Neural Networks for Robust Speech Recognition," in *Proceedings Interspeech*, 2016, pp. 2369–2372.

[205] D. Serdyuk, K. Audhkhasi *et al.*, "Invariant representations for noisy speech recognition," *CoRR*, 2016.

[206] S. Sun, B. Zhang, L. Xie, and Y. Zhang, "An unsupervised deep domain adaptation approach for robust speech recognition," *Neurocomputing*, vol. 257, pp. 79–87, 2017, machine Learning and Signal Processing for Big Multimedia Analysis.

[207] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proceedings Interspeech*, 2015, pp. 3214–3218.

[208] D. P. et al., "The kaldi speech recognition toolkit," in *Proceedings of the ASRU Workshop*. IEEE, 2011.

[209] S. Mirsamadi and J. H. Hansen, "Multi-domain adversarial training of neural network acoustic models for distant speech recognition," *Speech Communication*, vol. 106, pp. 21–30, 2019.

[210] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[211] S. Sun, C.-F. Yeh, M.-Y. Hwang, M. Ostendorf, and L. Xie, "Domain adversarial training for accented speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4854–4858.

[212] H. Hu, X. Yang *et al.*, "redat: Accent-invariant representation for end-to-end asr by domain adversarial training with relabeling," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[213] Z. Meng, J. Li *et al.*, "Speaker-invariant training via adversarial learning," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018.

[214] A. Tripathi, A. Mohan, S. Anand, and M. Singh, "Adversarial learning of raw speech features for domain invariant speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5959–5963.

[215] A. C S, P. A P, and A. G. Ramakrishnan, "Unsupervised domain adaptation schemes for building asr in low-resource languages," in *Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 342–349.

[216] W.-N. Hsu, A. Sriram *et al.*, "Robust wav2vec 2.0: Analyzing Domain Shift in Self-Supervised Pre-Training," in *Proceedings Interspeech*, 2021, pp. 721–725.

[217] H. Zhu, G. Cheng *et al.*, "Boosting cross-domain speech recognition with self-supervision," *arXiv preprint arXiv:2206.09783*, 2022.

[218] J. Kim and P. Kang, "K-Wav2vec 2.0: Automatic Speech Recognition based on Joint Decoding of Graphemes and Syllables," in *Proceedings Interspeech*, 2022, pp. 4945–4949.

[219] M. DeHaven and J. Billa, "Improving low-resource speech recognition with pretrained speech models: Continued pretraining vs. semi-supervised training," *arXiv preprint arXiv:2207.00659*, 2022.

[220] K. Nowakowski, M. Ptaszynski, K. Murasaki, and J. Nieuważny, "Adapting multilingual speech representation model for a new, underresourced language through multilingual fine-tuning and continued pretraining,"

Information Processing & Management, vol. 60, no. 2, p. 103148, 2023.

[221] R. Fan, Y. Zhu, J. Wang, and A. Alwan, "Towards better domain adaptation for self-supervised models: A case study of child asr," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1242–1252, 2022.

[222] W.-N. Hsu, B. Bolte *et al.*, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[223] H. Scudder, "Probability of error of some adaptive pattern-recognition machines," *IEEE Transactions on Information Theory*, vol. 11, no. 3, pp. 363–371, 1965.

[224] E. Riloff and J. Wiebe, "Learning extraction patterns for subjective expressions," in *Conf. Empirical Methods in Natural Language Processing*, 2003, pp. 105–112.

[225] Y. Chen, W. Wang, and C. Wang, "Semi-supervised asr by end-to-end self-training," in *Proceedings Interspeech*, 2020, pp. 2787–2791.

[226] Y. Higuchi, N. Moritz, J. L. Roux, and T. Hori, "Momentum Pseudo-Labeling for Semi-Supervised Speech Recognition," in *Proceedings Interspeech*, 2021, pp. 726–730.

[227] V. Manohar, T. Likhomanenko *et al.*, "Kaizen: Continuously improving teacher using exponential moving average for semi-supervised speech recognition," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2021, pp. 518–525.

[228] D. Hwang, K. C. Sim, Y. Zhang, and T. Strohman, "Comparison of soft and hard target rnn-t distillation for large-scale asr," 2022.

[229] J. Kahn, A. Lee, and A. Hannun, "Self-training for end-to-end speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7084–7088.

[230] D. S. P. et al, "Improved noisy student training for automatic speech recognition," in *Proceedings Interspeech*, 2020.

[231] Y. Zhang, J. Qin *et al.*, "Pushing the limits of semi-supervised learning for automatic speech recognition," in *Advances in Neural Information Processing Systems*, 2020.

[232] D. S. Park, W. Chan *et al.*, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," in *Proceedings Interspeech*, 2019, pp. 2613–2617.

[233] D. B. Paul and J. Baker, "The design for the wall street journal-based csr corpus," in *Speech and Natural Language: Proceedings of the of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992.

[234] S.-K. A. Yeung and M.-H. Siu, "Improved performance of aurora 4 using htk and unsupervised mllr adaptation," in *Conf. Spoken Language Processing*, 2004.

[235] N. Houlsby, A. Giurgiu *et al.*, "Parameter-efficient transfer learning for NLP," in *Proceedings of the International Conference on Machine Learning*, vol. 97. PMLR, 2019, pp. 2790–2799.

[236] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in Cognitive Sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[237] G. SHI, J. Chen, W. Zhang, L.-M. Zhan, and X.-M. Wu, "Overcoming catastrophic forgetting in incremental few-shot learning by finding flat minima," in *Advances in Neural Information Processing Systems*, 2021.

[238] S.-W. Lee, J.-H. Kim, J. Jun, J.-W. Ha, and B.-T. Zhang, "Overcoming catastrophic forgetting by incremental moment matching," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[239] R. Pascanu and Y. Bengio, "Revisiting natural gradient for deep networks," *arXiv preprint arXiv:1301.3584*, 2013.

[240] X. Han, Z. Zhang *et al.*, "Pre-trained models: Past, present and future," *AI Open*, vol. 2, pp. 225–250, 2021.

[241] Y. Du, Z. Liu, J. Li, and W. X. Zhao, "A survey of vision-language pre-trained models," *arXiv preprint*

arXiv:2202.10936, 2022.

[242] X. Qiu, T. Sun *et al.*, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, vol. 63, no. 10, pp. 1872–1897, 2020.

[243] J. Yang, G. Xiao *et al.*, "A survey of knowledge enhanced pre-trained models," *arXiv preprint arXiv:2110.00269*, 2021.

[244] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2018, pp. 328–339.

[245] Z. Yang, Z. Dai *et al.*, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in Neural Information Processing Systems*, 2019, pp. 5753–5763.

[246] Y. Liu, M. Ott *et al.*, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019.

[247] C. Karouzos, "Unsupervised domain adaptation for natural language processing," 2020, available at https://dspace.lib.ntua.gr/xmlui/bitstream/handle/123456789/52581/ckarouzos_thesis_uda_nlp.pdf. Accessed on 22/11/2023.

[248] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Association for Computational Linguistics, 2007, pp. 440–447.

[249] Y. Wu, M. Schuster *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *CoRR*, vol. abs/1609.08144, 2016.

[250] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2018.

[251] A. Paszke, S. Gross *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 8026–8037.

[252] T. Wolf, L. Debut *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," *ArXiv*, pp. arXiv–1910, 2019.

[253] D. Kifer, S. Ben-David, and J. Gehrke, "Detecting change in data streams," in *VLDB*, vol. 4. Toronto, Canada, 2004, pp. 180–191.

[254] R. Shu, H. Bui, H. Narui, and S. Ermon, "A dirt-t approach to unsupervised domain adaptation," in *International Conference on Learning Representations*, 2018.

[255] D. Shah, T. Lei, A. Moschitti, S. Romeo, and P. Nakov, "Adversarial domain adaptation for duplicate question detection," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018, pp. 1056–1063.

[256] C. Wang, M. Qiu, J. Huang, and X. He, "Meta fine-tuning neural language models for multi-domain text mining," *arXiv preprint arXiv:2003.13003*, 2020.

[257] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, pp. 2579–2605, 2008.

[258] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the International Conference on Machine Learning*. PMLR, 2015, pp. 97–105.

[259] P. Bell, J. Fainberg *et al.*, "Adaptation algorithms for neural network-based speech recognition: An overview," *IEEE Open Journal of Signal Processing*, vol. 2, pp. 33–66, 2021.

[260] S. Furui, "A training procedure for isolated word recognition systems," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 2, pp. 129–136, 1980.

[261] Y. Miao, H. Zhang, and F. Metze, "Towards speaker adaptive training of deep neural network acoustic mod-

els," in *Proceedings Interspeech*, 2014, pp. 2189–2193.

[262] S. H. P. e al., "fmllr based feature-space speaker adaptation of dnn acoustic models," in *Proceedings Interspeech*, 2015, pp. 3630–3634.

[263] V. Gupta, P. Kenny, P. Ouellet, and T. Stafylakis, "I-vector-based speaker adaptation of deep neural networks for french broadcast audio transcription," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

[264] H.-G. Hirsch and D. Pearce, "The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ASR2000-Automatic speech recognition: challenges for the new Millenium ISCA tutorial and research workshop (ITRW)*, 2000.

[265] Y. Qian, T. Tan, and D. Yu, "An investigation into using parallel data for far-field speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[266] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4960–4964.

[267] P. Denisov, N. T. Vu, and M. F. Font, "Unsupervised domain adaptation by adversarial learning for robust speech recognition," in *Speech Communication*, 2018, pp. 1–5.

[268] H. Jegou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 1, pp. 117–128, 2010.

[269] E. Jang, S. Gu, and B. Poole, "Categorical reparametrization with gumbel-softmax," in *Proceedings of the ICLR*, 2017.

[270] V. Panayotov, "Librispeech: an asr corpus based on public domain audio books," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[271] T. Likhomanenko, Q. Xu *et al.*, "Rethinking Evaluation in ASR: Are Our Models Robust Enough?" in *Proceedings Interspeech*, 2021, pp. 311–315.

[272] W. Chan, D. Park *et al.*, "Speechstew: Simply mix all available speech recognition data to train one large neural network," *arXiv preprint arXiv:2104.02133*, 2021.

[273] M. Mimura, S. Sakai, and T. Kawahara, "An end-to-end model from speech to clean transcript for parliamentary meetings," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2021, pp. 465–470.

[274] A. Virkkunen, A. Rouhe, N. Phan, and M. Kurimo, "Finnish parliament asr corpus: Analysis, benchmarks and statistics," *Language Resources and Evaluation*, pp. 1–26, 2023.

[275] J. O. Krůza, "Czech parliament meeting recordings as asr training data," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020, pp. 185–188.

[276] A. S. Kirkedal, M. Stepanovic, and B. Plank, "Ft speech: Danish parliament speech corpus," in *Proceedings Interspeech*. International Speech Communication Association (ISCA), 2020.

[277] G. V. G. Díaz-Munío, J.-A. Silvestre-Cerdà *et al.*, "Europarl-ASR: A Large Corpus of Parliamentary Debates for Streaming ASR Benchmarking and Speech Data Filtering/Verbatimization," in *Proceedings Interspeech*, 2021, pp. 3695–3699.

[278] A. C. et al., "All greek to me! an automatic greeklish to greek transliteration system," in *Proceedings of the LREC*, 2006.

[279] C. Meyer and H. Schramm, "Boosting hmm acoustic models in large vocabulary speech recognition," *Speech Communication*, vol. 48, no. 5, pp. 532–548, 2006.

[280] V. Manohar, D. Povey, and S. Khudanpur, "Jhu kaldi system for arabic mgb-3 asr challenge using diarization, audio-transcript alignment and transfer learning," in *Proceedings of the ASRU Workshop*, 2017, pp. 346–352.

[281] V. D. et al., "Large vocabulary continuous speech recognition in greek: corpus and an automatic dictation system," in *Proceedings of the Eurospeech*, 2003, pp. 1565–1568.

[282] T. Smith and M. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195–197, 1981.

[283] R. A. et al., "Common voice: A massively-multilingual speech corpus," in *Proceedings of the LREC*, 2020, pp. 4218–4222.

[284] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[285] G. Wenzek, M.-A. Lachaux *et al.*, "Ccnet: Extracting high quality monolingual datasets from web crawl data," in *Proceedings of the of the 12th Language Resources and Evaluation Conf.*, 2020, pp. 4003–4012.

[286] N. Hatzigeorgiu, M. Gavrilidou *et al.*, "Design and implementation of the online ilsp greek corpus." in *LREC*, 2000.

[287] K. Heafield, "Kenlm: Faster and smaller language model queries," in *Proceedings of the of the 6th workshop on statistical machine translation*, 2011, pp. 187–197.

[288] J. Ma and R. Schwartz, "Unsupervised versus supervised training of acoustic models," in *Proceedings Interspeech*, 2008, pp. 2374–2377.

[289] S. Chen, C. Wang *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[290] G. Paraskevopoulos, S. Parthasarathy, A. Khare, and S. Sundaram, "Multimodal and multiresolution speech recognition with transformers," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 2381–2387.

[291] J. Karlgren, A. Holst, and M. Sahlgren, "Filaments of meaning in word space," in *Advances in Information Retrieval: 30th European Conference on IR Research, ECIR 2008, Glasgow, UK, March 30-April 3, 2008. Proceedings 30*. Springer, 2008, pp. 531–538.

[292] G. Athanasopoulou, E. Iosif, and A. Potamianos, "Low-dimensional manifold distributional semantic models," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 731–740.

[293] E. Iosif and A. Potamianos, "Unsupervised semantic similarity computation between terms using web documents," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 11, pp. 1637–1647, 2009.

[294] J. J. DiCarlo and D. D. Cox, "Untangling invariant object recognition," *Trends in cognitive sciences*, vol. 11, no. 8, pp. 333–341, 2007.

[295] J. Mamou, H. Le *et al.*, "Emergence of separable manifolds in deep language representations," in *Proceedings of the International Conference on Machine Learning*. PMLR, 2020, pp. 6713–6723.

[296] C. Stephenson, J. Feather *et al.*, "Untangling in invariant speech recognition," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[297] Z. Yu, J. Li, Z. Du, L. Zhu, and H. T. Shen, "A comprehensive survey on source-free domain adaptation," *arXiv preprint arXiv:2302.11803*, 2023.

[298] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 09, pp. 5149–5169, 2022.

[299] J. Wang, Z. Song, W. Qiang, and C. Zheng, "Unleash model potential: Bootstrapped meta self-supervised learning," *arXiv preprint arXiv:2308.14267*, 2023.

[300] H. Peng, "A brief summary of interactions between meta-learning and self-supervised learning," *arXiv preprint arXiv:2103.00845*, 2021.

[301] F. Cappio Borlino, S. Polizzotto, B. Caputo, and T. Tommasi, "Self-supervision and meta-learning for one-

shot unsupervised cross-domain detection," *Computer Vision and Image Understanding*, vol. 223, p. 103549, 2022.

[302] L. W. Barsalou, "Grounded cognition," *Annual Reviews Psychology*, vol. 59, pp. 617–645, 2008.

[303] A. Pascual-Leone and R. Hamilton, "The metamodal organization of the brain," *Progress in brain research*, vol. 134, pp. 427–445, 2001.

[304] S. Lacey and K. Sathian, "Representation of object form in vision and touch," *The neural bases of multisensory processes*, 2012.

[305] M. L. Anderson, "Neural reuse: A fundamental organizational principle of the brain," *Behavioral and Brain Sciences*, vol. 33, no. 4, p. 245–266, 2010.

[306] J. Ngiam, A. Khosla *et al.*, "Multimodal deep learning," in *Proceedings of the International Conference on Machine Learning*, 2011.

[307] Q. Y. et al., "Image captioning with semantic attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[308] G. Paraskevopoulos, P. Pistofidis, G. Banoutsos, E. Georgiou, and V. Katsouros, "Multimodal classification of safety-report observations," *Applied Sciences*, vol. 12, no. 12, p. 5781, 2022.

[309] Y. Huang, J. Lin, C. Zhou, H. Yang, and L. Huang, "Modality competition: What makes joint training of multimodal network fail in deep learning? (Provably)," in *Proceedings of the International Conference on Machine Learning*, vol. 162. PMLR, 2022, pp. 9226–9259.

[310] E. Georgiou, G. Paraskevopoulos, and A. Potamianos, "M3: Multimodal masking applied to sentiment analysis," in *Proceedings Interspeech*, 2021, pp. 2876–2880.

[311] D. Hazarika, Y. Li *et al.*, "Analyzing modality robustness in multimodal sentiment analysis," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2022, pp. 685–696.

[312] W. Wang, D. Tran, and M. Feiszli, "What makes training multi-modal classification networks hard?" in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 12 695–12 705.

[313] N. Dimitriadis, P. Frossard, and F. Fleuret, "Pareto manifold learning: Tackling multiple tasks via ensembles of single-task models," in *Proceedings of the International Conference on Machine Learning*. PMLR, 2023, pp. 8015–8052.

[314] X. Lin, H.-L. Zhen, Z. Li, Q.-F. Zhang, and S. Kwong, "Pareto multi-task learning," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[315] P. Ma, T. Du, and W. Matusik, "Efficient continuous pareto exploration in multi-task learning," in *Proceedings of the International Conference on Machine Learning*. PMLR, 2020, pp. 6522–6531.

[316] P. Papale, F. Wang *et al.*, "The representation of occluded image regions in area v1 of monkeys and humans," *Current Biology*, vol. 33, no. 18, pp. 3865–3871.e3, 2023.

[317] J. F. Houde and E. F. Chang, "The cortical computations underlying feedback control in vocal production," *Current opinion in neurobiology*, 2015.

[318] R. L. S. et al., "Visual feedback during motor performance is associated with increased complexity and adaptability of motor and neural output," *Behavioural Brain Research*, 2019.

[319] M. Bar and A. Bubic, "Top-down effects in visual perception," *The Oxford Handbook of Cognitive Neuroscience, Volume 2*, 2013.

[320] C. Teufel and B. Nanay, "How to (and how not to) think about top-down influences on visual perception," *Consciousness and Cognition*, 2017.

[321] E. Sohoglu, J. E. Peelle, R. P. Carlyon, and M. H. Davis, "Predictive top-down integration of prior knowledge

during speech perception," *Journal of Neuroscience*, 2012.

[322] G. Lupyan, "Objective effects of knowledge on visual perception." *Journal of experimental psychology: human perception and performance*, 2017.

[323] D. R. Proffitt, M. Bhalla, R. Gossweiler, and J. Midgett, "Perceiving geographical slant," *Psychonomic bulletin & review*, 1995.

[324] D. E. Winkowski and E. I. Knudsen, "Top-down gain control of the auditory space map by gaze control circuitry in the barn owl," *Nature*, vol. 439, no. 7074, pp. 336–339, 2006.

[325] D. E. Winkowski and E. I. Knudsen, "Top-down control of multimodal sensitivity in the barn owl optic tectum," *Journal of Neuroscience*, vol. 27, no. 48, pp. 13 279–13 291, 2007.

[326] G. Paraskevopoulos, E. Georgiou, and A. Potamianos, "Mmlatch: Bottom-up top-down fusion for multimodal sentiment analysis," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 4573–4577.

[327] R. Liu, P. Mu, X. Yuan, S. Zeng, and J. Zhang, "A general descent aggregation framework for gradient-based bi-level optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 38–57, 2023.

[328] R. Liu, Y. Liu, S. Zeng, and J. Zhang, "Towards gradient-based bilevel optimization with non-convex followers and beyond," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 8662–8675.

[329] S. A. et al., "Vqa: Visual question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[330] O. Caglayan, P. Madhyastha, L. Specia, and L. Barrault, "Probing the need for visual context in multimodal machine translation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019, pp. 4159–4170.

[331] T. Srinivasan, R. Sanabria, F. Metze, and D. Elliott, "Multimodal speech recognition with unstructured audio masking," in *Proceedings of the First International Workshop on Natural Language Processing Beyond Text*. Association for Computational Linguistics, 2020, pp. 11–18.

[332] J. Klemen and C. D. Chambers, "Current perspectives and methods in studying neural mechanisms of multisensory interactions," *Neuroscience & Biobehavioral Reviews*, 2012.

[333] P. A. N. et al., "Development of multisensory spatial integration and perception in humans," *Developmental science*, 2006.

[334] E. Balcetis and D. Dunning, "Wishful seeing: More desired objects are seen as closer," *Psychological science*, 2010.

[335] C. Firestone and B. J. Scholl, ""top-down" effects where none should be found: The el greco fallacy in perception research," *Psychological science*, 2014.

[336] C. C. L. et al., "Emotion recognition using a hierarchical binary decision tree approach," *Speech Communication*, 2011.

[337] V. R. et al., "Ensemble of svm trees for multimodal emotion recognition," in *Proceedings of the APSIPA*. IEEE, 2012.

[338] A. M. et al., "Context-sensitive learning for enhanced audiovisual emotion classification," *Transactions on Affective Computing*, 2012.

[339] M. W. et al., "Lstm-modeling of continuous emotions in an audiovisual affect recognition framework," *Image and Vision Computing*, 2013.

[340] A. Shenoy and A. Sardana, "Multilogue-net: A context-aware RNN for multi-modal emotion detection and sentiment analysis in conversation," in *Second Grand-Challenge and Workshop on Multimodal Language*

(Challenge-HML). Association for Computational Linguistics, 2020, pp. 19–28.

[341] S. Poria, I. Chaturvedi, E. Cambria, and A. Hussain, "Convolutional mkl based multimodal emotion recognition and sentiment analysis," in *Proceedings of the ICDM*. IEEE, 2016.

[342] T. Baltrušaitis, C. Ahuja, and L. P. Morency, "Multimodal machine learning: A survey and taxonomy," *Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[343] A. Z. et al., "Tensor fusion network for multimodal sentiment analysis," in *Proceedings of the EMNLP*, 2017.

[344] Z. L. et al., "Efficient low-rank multimodal fusion with modality-specific factors," in *Proceedings of the 56th ACL*, 2018.

[345] A. Z. et al., "Multi-attention recurrent network for human communication comprehension," 2018.

[346] A. Bagher Zadeh, P. P. Liang, S. Poria, E. Cambria, and L.-P. Morency, "Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2018, pp. 2236–2246.

[347] Y. G. et al., "Multimodal affective analysis using hierarchical attention strategy with word-level alignment," in *Proceedings of the ACL*, 2018.

[348] E. Georgiou, C. Papaioannou, and A. Potamianos, "Deep hierarchical fusion with application in sentiment analysis," in *Proceedings Interspeech*, 2019.

[349] H. P. et al., "Found in translation: Learning robust joint representations by cyclic translations between modalities," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[350] Z. Sun, P. K. Sarma, W. Sethares, and E. P. Bucy, "Multi-modal sentiment analysis using deep canonical correlation analysis," in *Proceedings Interspeech*, 2019.

[351] A. Khare, S. Parthasarathy, and S. Sundaram, "Multi-modal embeddings using multi-task learning for emotion recognition," in *Proceedings Interspeech*, 2020.

[352] Y.-H. H. Tsai, S. Bai *et al.*, "Multimodal transformer for unaligned multimodal language sequences," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019, pp. 6558–6569.

[353] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *Advances in Neural Information Processing Systems*, 2019.

[354] J.-B. Delbrouck, N. Tits, M. Brousmiche, and S. Dupont, "A transformer-based joint-encoding for emotion recognition and sentiment analysis," in *Second Grand-Challenge and Workshop on Multimodal Language (Challenge-HML)*. Association for Computational Linguistics, 2020, pp. 1–7.

[355] W. Rahman, M. K. Hasan *et al.*, "Integrating multimodal information in large pretrained transformers," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 2359–2369.

[356] Y. W. et al., "Words can shift: Dynamically adjusting word representations using nonverbal behaviors," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[357] A. Kumar and J. Vepa, "Gated mechanism for attention based multi modal sentiment analysis," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020.

[358] Y.-H. H. Tsai, M. Ma, M. Yang, R. Salakhutdinov, and L.-P. Morency, "Multimodal routing: Improving local and global interpretability of multimodal language analysis," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2020, pp. 1823–1833.

[359] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Advances in Neural Information Processing Systems*, 2017.

[360] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, p. 1735–1780, 1997.

[361] H. e. a. Fukui, "Attention branch network: Learning of attention mechanism for visual explanation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[362] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[363] H. Wen, S. You, and Y. Fu, "Cross-modal context-gated convolution for multi-modal sentiment analysis," *Pattern Recognition Letters*, 2021.

[364] S. Sourav and J. Ouyang, "Lightweight models for multimodal sequential data," in *Proceedings of the 11th WASSA*, 2021.