SPONGE: Competing Sparse Language Representations for Effective Knowledge Transfer

Anonymous authors
Paper under double-blind review

Abstract

In domains with privacy constraints, most knowledge resides in siloed datasets, hindering the development of a model with *all* relevant knowledge for a task. Clinical NLP is a prime example of these constraints in practice. Research in this area typically falls back to the canonical setting of sequential transfer learning, where a model pre-trained on large corpora is finetuned on a smaller annotated dataset. An avenue for knowledge transfer among diverse clinics is *multi-step sequential transfer learning* since models[1] are more likely to be shared than private clinical data. This setting poses challenges of cross-linguality, domain diversity, and varying label distributions which undermine generalisation. We propose SPONGE, an efficient prototypical architecture that leverages competing sparse language representations. These encompass distributed knowledge and create the necessary level of redundancy for effective transfer learning across multiple datasets. We identify that prototypical classifiers are critically sensitive to label-recency bias which we mitigate with a novel strategy at inference time. SPONGE in combination with this strategy significantly boosts generalisation performance to unseen data. With the help of medical professionals, we show that the explainability of our models is clinically relevant. We make all source code[2] available.

1 Introduction

In real-world machine learning applications, access to labeled data is often limited, whether due to intellectual property reasons or privacy constraints. Clinical Natural Language Processing (NLP) tasks exemplify this challenge, where these limitations hinder the unification of all medical knowledge. A common approach to handling decentralised and private data is Federated Learning (FL). However, FL performance suffers when data are non-i.i.d. (Nguyen et al., 2023), which is prevalent for most of the clinically relevant outcome classes (diagnoses, medications, procedures). Sequential transfer learning (Howard & Ruder, 2018) is a well-established and simple alternative to FL when model checkpoints, rather than data, are used for knowledge transfer. This approach is also the de facto standard for tasks with limited labeled data, usually involving a two-step approach: pretraining on large data corpora and fine-tuning on a smaller domain-specific dataset. Nevertheless, more than two steps (datasets) have also proven beneficial for knowledge transfer (Poth et al., 2021; Jeong et al., 2020).

At the core of clinical data are Electronic Health Records (EHR), which document the medical history of patients. Upon patient admission, predictions of diagnoses, procedures, medications, and mortality risk help optimize resource allocation in clinical facilities. Clinical Decision Support Systems (CDSS) increasingly rely on NLP methods to assist medical professionals. Models from various clinics and specialties have been made available by researchers in large hosting platforms like *Huggingface* (Wolf et al., 2020). For example, clinical BERT models trained by Zhongshan and Qingpu Hospitals[3] (Wang et al., 2023) or Charite Hospital[4] (Bressem et al., 2024) among others are publicly available and can be finetuned by other clinical facilities for downstream tasks.

[1] Huggingface biomedical models
[2] `https://anonymous.4open.science/r/HSPONGE-6DDD`
[3] BERT from Zhongshan and Qingpu Hospitals.
[4] BERT from Charite Hospital.

In diagnoses prediction state-of-the-art (SOTA) methods augment Transformer representations (BioMedBERT (Tinn et al., 2023)) with prototypical networks (Figueroa et al., 2024; van Aken et al., 2022). Diagnoses prediction generally poses significant challenges: 1) In practice diagnoses occur with widely varying prevalence, exhibiting a pronounced long-tail distribution (Papaioannou et al., 2022). 2) Medical knowledge and data are unevenly distributed across regions and languages. For example, the higher incidence of gastric cancer in Japan (Naylor et al., 2006) or Chagas disease in South America (Martins-Melo et al., 2014). 3) Privacy concerns isolate patient data across medical facilities, hindering unified model training. These challenges are amplified by the fact that Transformer models are usually pretrained on large corpora in high-resource languages (mainly English) limiting performance in low-resource languages.

Sequential transfer learning addresses the constraints of clinical data, namely privacy and data isolation, uneven diagnosis distribution, low-resource settings, and multilinguality. Multiple fine-tuning steps have also been shown to improve downstream performance on low-resource diagnoses prediction (Papaioannou et al., 2022). Research has centered on finding optimal sequences of tasks (Lim et al., 2024; Poth et al., 2021), highlighting the performance sensitivity of Transformers to the task order. However, finding a training order for optimal knowledge transfer requires access to all datasets, which does not comply with the clinical data restrictions mentioned above. We argue that creating robust solutions subject to these constraints involves 1) A realistic simulation of knowledge transfer that reflects the challenges of working with clinical data, 2) An architecture that is *agnostic to the fine-tuning sequence*.

We gather a collection of six clinical datasets encompassing different clinics, writing styles, languages, number of patients, and diagnoses distributions. In Figure 1 we illustrate the label spaces of our five training datasets, which we complement with an additional dataset used only for testing. We train and evaluate SOTA architectures strictly following clinical data constraints, i.e. using model checkpoints rather than data as the medium of knowledge transfer (see Figure 1 right). Additionally, we focus on assessing *sequence-agnostic* performance, reflecting the realistic scenario in which clinics are unaware of the data seen by publicly available models.

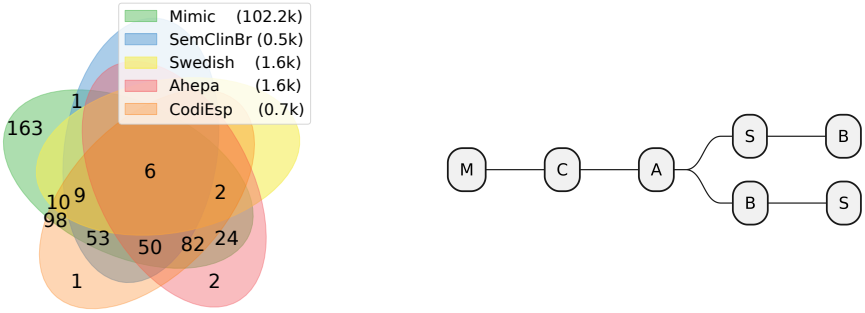

Figure 1: **(Left)** Label space relationships across five clinical datasets used for training. *Legend*: number of training samples for each dataset. **(Right)** Six (two-steps or more) training sequences are generated from two 5-step permutations of the datasets that start with MIMIC (M). Each node is the end of a sequence and results in a model checkpoint; e.g, C indicates the model checkpoint for $M \rightarrow C$. In our experiments, we ensure that the model checkpoints (and not data) are the only medium of knowledge transfer.

We find that SOTA models for diagnoses prediction such as S-Proto (Figueroa et al., 2024) struggle under realistic clinical conditions. We argue that building non-interfering knowledge redundancy during *sequential fine-tuning* is key for knowledge transfer in this scenario. However, deep neural networks are usually dense, i.e., they utilize all parameters during prediction. In sequential settings, deep neural networks often suffer from knowledge overwriting, which impairs effective transfer (Ling et al., 2024).

We introduce SPONGE: an efficient architecture that boosts knowledge transfer in diagnoses prediction by creating end-to-end sparse language representations. We construct these by integrating concepts from parameter-efficient fine-tuning (PEFT) (Pfeiffer et al., 2020; Poth et al., 2023), prototypical networks (Snell et al., 2017), and conditional computation (Bengio et al., 2013) using a non-parametric *winner-takes-all* (Yuille & Grzywacz, 1989) mechanism. Inspired by biological neural circuits, the sparse subnetworks in

SPONGE are full-fledged predictors that compete to predict a diagnosis. While requiring only ≈2.6% of the trainable parameters, our method outperforms current SOTA in diagnoses prediction. We validate this extensively by analyzing both single and multi-step transfer learning scenarios.

Catastrophic forgetting (McCloskey & Cohen, 1989) during sequential fine-tuning impairs generalization. We find that prototypical classifiers tend to focus disproportionately on *recent* labels when compared to traditional Transformers, reducing generalization to unseen datasets. However, explicitly preventing catastrophic forgetting can degrade performance on target tasks (Pfeiffer et al., 2020). We propose an alternative approach: $\mathcal{H}$ydra, a strategy that improves generalization to unseen datasets without modifying SPONGE's architecture. $\mathcal{H}$ydra efficiently improves macro AUROC and PRAUC by 7.5 and 3.5 percentage points, respectively, with a negligible increase in inference parameters. SPONGE and $\mathcal{H}$ydra function as adapters that can be loaded and fine-tuned by clinics as needed to boost knowledge transfer.

Additionally, our method inherits the explainability properties of previous SOTA (Figueroa et al., 2024) by predicting in a latent prototypical space (Chen et al., 2019). With input from medical professionals, we validate that the observed improvement in knowledge transfer is underpinned by clinically meaningful explanations (van Aken et al., 2022).

Contributions

- To our knowledge, our evaluation of multi-step sequential transfer learning is the largest to date in simulating knowledge transfer for diagnoses prediction with real-world clinical data constraints.
- We propose a novel architecture of sparse competing subnetworks that achieves superior knowledge transfer in diagnoses prediction.
- Our analysis identifies weaknesses in current SOTA approaches, which we address with an efficient strategy that mitigates catastrophic forgetting and improves generalization to unseen datasets.

2 Related Work

Diagnoses prediction from textual data. Transformer models have widely been applied to diagnoses prediction (van Aken et al., 2021; Roehr et al., 2024). S-Proto (Figueroa et al., 2024) boosts classification performance by enhancing a Transformer with a sparse prototypical network. In contrast to S-Proto, we create competing end-to-end subnetworks, i.e., networks that include both the prototypical network and the Transformer.

Cross-lingual transfer is a popular topic in NLP (Hämmerl et al., 2024; Philippy et al., 2023). Most research has focused on identifying optimal dataset sequences and configurations, extracting dataset properties that enhance performance on downstream tasks (Lim et al., 2024; Protasov et al., 2024; Lin et al., 2019; Malkin et al., 2022). Given clinical data constraints, Papaioannou et al. (2022) find that sequential training for diagnoses prediction boosts downstream performance for low-resource datasets, however, only for optimal dataset sequences. In contrast, we propose a robust architecture that enables cross-lingual transfer *independently* of sequence order, and the dataset properties; i.e., size, labels, distribution, and language.

Parameter efficient fine-tuning (PEFT) and generalization. Liu et al. (2024); Lialin et al. (2023); Chalkidis et al. (2021) highlight the importance of PEFT using adapters to improve zero-shot classification performance. In our work, adapters play a crucial role in inducing sparsity. We use a pool of adapters to generate multiple Transformer representations, which compete via a *winner-takes-all* (Yuille & Grzywacz, 1989) mechanism. Additionally, PEFT enables our model to scale the number of subnetworks efficiently.

Parameter selective training approaches have centered on using masks to update specific parameters (Somayajula et al., 2024; Sung et al., 2021; Winter et al., 2022). Similarly, we selectively update only a subset of model parameters. However, rather than applying weight level updates, we constrain updates to subnetworks responsible for end-to-end representations. Moreover, Ansell et al. (2022) use sparse fine-tuning to boost cross-lingual transfer using prior knowledge of the target task and language. In contrast, our subnetwork selection strategy operates without prior knowledge of downstream tasks.

Subnetwork routing and lateral inhibition. With ever-growing parameters in Transformers, a Mixture of Experts (MoE) offers an efficient path for scaling computation (Shazeer et al., 2017; Lepikhin et al., 2021; Fedus et al., 2022; Du et al., 2022; Chi et al., 2022). Our approach differs from MoE in that 1) it does not fragment the input (tokens) 2) we use no additional routing loss functions or gating mechanisms 3) our subnetworks are independent predictors that *compete* to produce the best representation. In fact, we leverage conditional computation (Bengio et al., 2013), and are closer to single-layer biologically inspired neural networks (Bricken et al., 2023; Liang et al., 2021; Figueroa et al., 2025) to learn sparse representations through lateral inhibition. However, we extend this to deeper architectures integrating Transformer models with prototypical classification in a latent metric space.

Sequential learning and catastrophic forgetting. Sequential learning presents significant challenges for deep neural networks. A well-established trade-off in this setting is the *stability-plasticity* dilemma, which is central to continual learning (Biesialska et al., 2020; Huang et al., 2021). In cross-lingual transfer, explicitly mitigating catastrophic forgetting (i.e., optimizing for stability) may result in reduced downstream performance (Pfeiffer et al., 2020). We use a non-parametric inhibition approach that creates non-interfering parallel representations, improving both plasticity and stability. However, our objective is to maximize knowledge transfer under clinical data constraints; hence, we evaluate downstream performance on low-resource datasets of clinics.

3 Multilingual Clinical Datasets

We train and test our models on sequences constructed from five different clinical datasets across multiple languages (see Figure 1, right). Additionally, we evaluate their generalization performance on a multilingual dataset with the same EHRs in seven different languages in a zero-shot setting. For all datasets, we map the diagnoses to the CCSR[5] code space. CCSR offers advantages over ICD-10, including improved clinical relevance and simplified categorization.

MIMIC-IV (M) (Johnson et al., 2021; 2023) is an *English* clinical dataset containing cases from the Intensive Care Unit (ICU) and other hospital departments. Admission and discharge notes are simulated from the EHRs as in van Aken et al. (2021).We use only the ICD-10 subset, which we map to the corresponding CCSR categories.

CodiEsp (C) contains clinical case studies in *Spanish*, with patients from various medical specialties, including oncology, urology, cardiology, pneumology, and infectious diseases (Miranda-Escalada et al., 2020).

AHEPA-Cardio (A) are cardiology discharge summaries in *Greek* (Papaioannou et al., 2022).

SemClinBr (B) comprises *Portuguese* clinical notes from multiple Brazilian medical institutions across various specialties (Oliveira et al., 2022).

Stockholm University - Gastrointestinal (S)(Lamproudis et al., 2023) consists of EHRs from a gastroenterology department. These are in *Swedish* and sourced from the Swedish Health Record Research Bank (Dalianis et al., 2015)[6].

Zero-Shot Datasets: DisTEMIST. We use the dataset of Miranda-Escalada et al. (2022) for zero-shot evaluation. It comprises the same EHRs expressed in *Spanish*, *English*, *Catalan*, *Portuguese*, *French*, *Italian*, and *Romanian*.

Data Splits and Label Space. For all datasets, we use stratified sampling (Sechidis et al., 2011) to create train, validation, and test splits (see Table 5 in Appendix A). We remove EHRs with less than four labels. Figure 1 summarizes the label space and training sample counts.

[5]CCSR categories

[6]This research has been approved by the Swedish Ethical Review Authority under permission no.2019-05679. The use of Stockholm EPR Gastro ICD-10 Pseudo Corpus II with the amendment no 2022-02386-02.

4 Sequential Transfer Learning for Diagnoses prediction

Diagnoses prediction is a multilabel classification task with a large label space, in our case, comprising 509 CCSR codes (98% of the full CCSR specification). We focus on knowledge transfer for this task using only model checkpoints for transfer to simulate real-world clinical constraints. Accordingly, we adopt the canonical sequential transfer learning setup involving both single and multi-step fine-tuning (Ruder, 2019). This involves sequential training with all permutations of our datasets, evaluating performance **independently** of any specific fine-tuning order.

We initialize all architectures on this task with MIMIC, our high-resource dataset. Next, we construct all permutations of the four low-resource datasets to generate sequence variations. This process results in $64+1$ distinct sequences (including the one-step sequence on MIMIC, see Appendix E). Figure 1 (*right*) illustrates two example sequences.

Evaluation. The goal of our experiments is to assess how effectively our architecture transfers relevant knowledge and generalizes to unseen data at each step of the sequence. For example, sequences involving one training dataset are evaluated on four unseen test datasets, those trained on two evaluated on three, and so on. DisTEMIST is used exclusively as a test set. We evaluate all resulting model checkpoints $(64 + 1)$, measuring averages of macro AUROC, micro AUROC, and macro PRAUC to capture *sequence agnostic* scores.

MIMIC - (High-resource). We report results on this dataset as it serves as the standard benchmark for diagnosis prediction (Roehr et al., 2024; van Aken et al., 2021). This evaluation corresponds to the one-step sequential transfer learning scenario, i.e., the sole single-dataset sequence in our experiments.

Final datasets (Low-resource). We rank architectures based on the performance on the *final dataset* in each sequence. This evaluation measures downstream performance independently of the datasets seen by the model during sequential training. This is consistent with the performance expected by clinics when fine-tuning models without explicit knowledge of their prior training data. In this evaluation, we average all performances in all sequence configurations. We evaluate 64 sequences of two datasets or more, with a total of 16 sequences for every target dataset (see Appendix E).

Zero-shot generalization. In contrast to the standard *zero-shot cross-lingual transfer setting* (Hu et al., 2020), we focus on models trained sequentially on multiple datasets. This implies that variations in data distributions, label spaces, and language typologies may contribute or interfere with every additional training dataset. We probe the generalization capabilities of all models on DisTEMIST in addition to all unseen datasets. We evaluate $64 + 1$ checkpoints per model and report the average for each metric for 1) all unseen datasets, 2) unseen languages within DisTEMIST, and 3) seen languages within DisTEMIST.

5 Methods

Several works have tackled diagnoses prediction with Transformer networks and architectural augmentations. van Aken et al. (2022) incorporate prototypical classifiers to use latent metric spaces, enhancing both the explainability and performance of model predictions. Figueroa et al. (2024) extend this approach in S-Proto by adding sparsity to the prototypical classification layer using a *winner-takes-all* mechanism. Specifically, S-Proto leverages competing subnetworks to model distinct phenotypes of diagnoses, further boosting performance. These architectural enhancements to Transformer representations achieve state-of-the-art performance in diagnoses prediction. However, they struggle when exposed to multi-step sequential transfer learning, since sequentially updating a single Transformer representation creates inter-task interference. We speculate that this is due to the competing subnetworks, which share a large number of parameters, leading to entangled representations. To address this, we propose extending sparsity throughout the entire network, rather than limiting it to the classification layer.

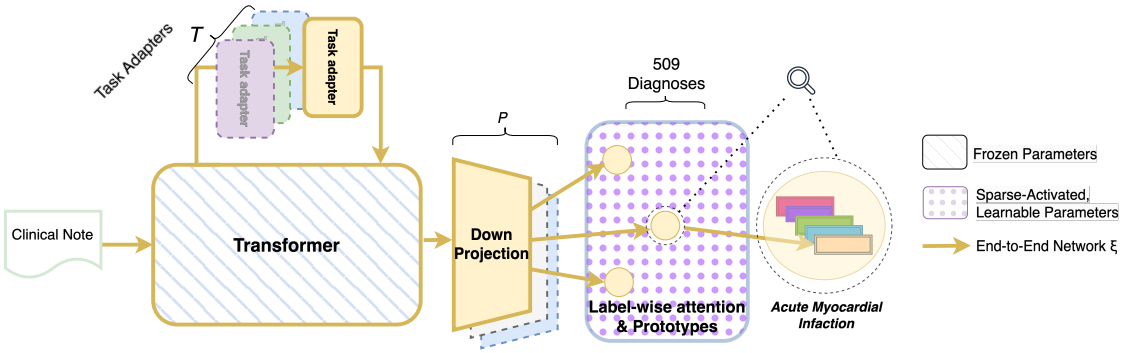

Figure 2: Classifying a clinical note with SPONGE: A pool of T adapters (left) create multiple transformer representations. P down-projection layers map these representations, which are then classified in a latent metric space with sparse label-wise attention. ξ is the winner network among all competing subnetworks that map the input to the label space, resulting in sparse gradient updates.

5.1 End-To-End Sparse Representation

The dense (non-sparse) modules of S-Proto are the Transformer and the projection matrix to the prototypical space. We add competing subnetworks to these components to create end-to-end sparse representations. This entails 1) creating in parallel multiple Transformer representations for the same input, 2) expanding the dimensionality of the projection layer to the prototypical space, and 3) selecting a *winner* subnetwork, hence, the *winner-takes-all* mechanism must be generalized for the entire architecture. We show an overview of the resulting architecture in Figure 2.

Creating multiple Transformer representations would require parallel embedding Transformers, which is computationally prohibitive. We opt for a simpler solution using multiple adapters, which significantly improves efficiency. Trainable adapters combined with a language model have shown competitive performance on classification tasks (Houlsby et al., 2019). We add T task adapters (Pfeiffer et al., 2020) as modular components, freezing the rest of the Transformer while training.

$$TA^t(emb, r) = U^t(ReLU(D^t(emb))) + r \tag{1}$$

D^t and U^t are the task adapter t down and up projections, emb is the encoder representation and r is the residual for a layer within the Transformer. We compute multiple representations, one with every task adapter: $\eta^t \in \mathbb{R}^{s \times h}$, where h is the hidden dimension of the encoder, and $s \in E$ the index of a token in a given EHR.

Expanding the dimensionality of the projection layer. Additionally, we increase the number of down-projection layers $L \in \mathbb{R}^{h \times d}$ to $L \in \mathbb{R}^{P \times h \times d}$, where d is the hidden dimension and P is the number of projection layers. We project η^t which results in a representation $\psi^{t,p} = \langle \eta^t, L^p \rangle \in \mathbb{R}^{E \times d}$, where $t \in T$, $p \in P$ define a subnetwork. We refer to this as the *input subnetwork*, specified by t and p, which enables an end-to-end sparse encoding of the EHR and is core to our contributions.

Let $\gamma \in \Gamma$ be an *output subnetwork* index in the prototypical layer, $c \in C$ be a specific class, therefore $\xi \in T \times P \times \Gamma$ defines the subnetwork that spans through the entire model and involves the combination of t, p, γ. We first map the token vectors $\psi^{t,p}$.

$$\phi^\xi = \langle \psi^{t,p}, W^\gamma \rangle^\top \in \mathbb{R}^{C \times E}$$

where $W^\gamma \in \mathbb{R}^{d \times C}$ are the labelwise-attention vectors. A score $S^\xi = \underset{E}{\text{softmax}}(\phi^\xi)$ is computed for every token s in the EHR for every class c. The mapped EHR, for a subnetwork ξ, in prototypical space is v^ξ.

$$v^\xi = \langle S^\xi, \psi^{t,p} \rangle \in \mathbb{R}^{C \times d}$$

SPONGE: **Generalizing *winner-takes-all*.** The Euclidean distance from a mapped EHR v^ξ to a prototype u^ξ is denoted by $\epsilon^\xi = \|u^\xi - v^\xi\|_2$ and the prediction of a subnetwork $\hat{y}^\xi = \sigma(-\epsilon^\xi)$, where σ is the sigmoid function. We select the *winner* subnetwork $\overset{\star}{\xi}$ with *the winner-takes-all* mechanism inhibiting all other subnetworks:

$$\hat{y} = \hat{y}^\xi \delta_{\xi,\overset{\star}{\xi}} \tag{2}$$

where δ is the Kronecker δ function. To compute the winning subnetwork $\overset{\star}{\xi}$ we start by selecting the *winner output subnetwork*:

$$\overset{\star}{\gamma}^{t,p} = \underset{\gamma}{\text{argmax}}(\hat{y}^\xi) \tag{3}$$

We use $\overset{\star}{\gamma}^{t,p}$ to select the *winner input subnetwork* $\overset{\star}{t}, \overset{\star}{p}$ with the mode over the classes for an EHR.

$$\overset{\star}{t}, \overset{\star}{p} = \underset{c}{\text{mode}}[\underset{t,p}{\text{argmax}}(\hat{y}^\xi \delta_{\gamma,\overset{\star}{\gamma}^{t,p}})] \tag{4}$$

Finally, the *end-to-end winner* subnetwork is:

$$\overset{\star}{\xi} = \underset{\gamma}{\text{argmax}}(\hat{y}^{\overset{\star}{t},\overset{\star}{p},\gamma}) \tag{5}$$

We define the loss as the binary cross-entropy between $\hat{y}$ (see: Equation (2)) and the true labels y.

Subnetwork exploration. Training of sparse networks with *winner-takes-all* may result in *dead* subnetworks that are never selected and, thus, never updated. We modify the approach in (Bricken et al., 2023), inspired by $\epsilon - greedy$ (Sutton, 1995) exploration, to ensure broader subnetwork activation during training. At the start of training, we *explore* the top k subnetworks ξ, choosing one of these randomly. During this phase, we anneal k linearly until $k = 1$. Training proceeds with the top-1 subnetwork as in Equation (2). This strategy helps initialize and train all subnetworks, avoiding dead units, while still limiting computation by updating only one subnetwork per sample.

5.2 Boosting generalization

Prototypical networks use a latent distance to make predictions. Guided by the loss, these classifiers map tokens and shift prototypes in latent space. During sequential training, the learned prototypes become overly biased toward the *most recent* label distribution, severely restricting generalization in different label spaces. We argue that distributed knowledge still exists in the *input subnetworks* of SPONGE due to the overall sparsity (we examine this further in Section 8).

$\mathcal{H}$ydra: **exploiting distributed knowledge.** To counter the *label recency* bias, we store the prototypical parameters (u and W) learned for each dataset in a training sequence, using them only for inference. Importantly, we only use parameters from past training steps for knowledge transfer, respecting the constraints of clinical data. Despite the growing number of *output subnetworks*, the *winner* subnetwork selection strategy of SPONGE remains applicable without requiring architectural changes. Although the number of parameters for inference increases after each dataset in a sequence, the increase is small w.r.t. the Transformer's size. In our experiments, in a five-dataset sequence, the added vectors amount to 6.6M parameters or 2.4% of the Transformer's 270M. A key advantage of the *winner-takes-all* selection is its compatibility with parameter accumulation, requiring no architectural changes nor added complexity like weight consolidation (Kirkpatrick et al., 2017) or knowledge distillation (Ermis et al., 2022).

$\mathcal{H}$SPONGE: $\mathcal{H}$ydra facilitates effective knowledge transfer to previously unseen data distributions. Importantly, SPONGE and $\mathcal{H}$ydra scale very well to larger Transformers since the dimensionality of the parameters is independent of their hidden size h. This approach supports modular scaling to arbitrary dataset sequences during *sequential training*. Furthermore, in the case of growing label spaces, this modularity would enable our models to train only the new parameters.

6 Experiments

Although generative large language models (LLMs) have shown strong performance in many language tasks, they still fall short of SOTA in many classification scenarios (Yang et al., 2024), particularly those with large label spaces. For diagnoses prediction, the best-performing approach remains fine-tuning Transformer representations with a classification head, for both encoder (Roehr et al., 2024; Figueroa et al., 2024) and decoder models (Gema et al., 2024).

We investigate how sparse subnetworks impact sequential transfer learning for diagnoses prediction. Therefore, we control for one Transformer architecture. We use *XLM-R* (Conneau et al., 2020) for all experiments, as it has been extensively studied for cross-lingual transfer (Philippy et al., 2023; Pfeiffer et al., 2020; Hu et al., 2020; Choi et al., 2020; Conneau et al., 2020). Although our methods are applicable to any Transformer architecture, we favor *XLM-R* over significantly larger Transformers, given the large number of experiments.

Baselines. We focus on the following architectures: 1) *XLM-R*, which is a cross-lingual model and the least sparse method (single dense network); 2) *XLM-R + A* to ablate the effect of PEFT and a single adapter in our architecture; and 3) S-PROTO$_{\text{XLM-R}}$ (Figueroa et al., 2024), the current SOTA architecture for diagnoses prediction, which combines a dense Transformer with a sparse prototypical classifier. For the latter, we replace the BiomedBERT (Tinn et al., 2023) with XLM-R, since the former is not a cross-lingual Transformer.

SPONGE Variations. We evaluate multiple configurations of sparse subnetworks by varying the number of adapters and projection matrices. We name each variant SPONGE^{T,P} where T and P stand for the number of task adapters and projections respectively. SPONGE^{1,1} establishes the effect of having *no* sparsity in the encoder. SPONGE^{1,6} displays the effect of having sparsity only in the projection stage before the prototypical layer. SPONGE^{6,1} highlights the impact of no sparsity in the projection, but only in the Transformer using a pool of adapters. Given the effectiveness of adapters for cross-lingual transfer (He et al., 2021), our sparsest model, SPONGE^{6,3}, uses more adapters than projections.

All prototypical models (SPONGE and S-PROTO$_{\text{XLM-R}}$) have five *output subnetworks* as in Figueroa et al. (2024). We detail hyperparameters in Appendix C.

7 Results

Performance on MIMIC. This dataset is a standard benchmark for NLP models in diagnoses prediction. We present the performance of all models in Table 1 (left). S-PROTO$_{\text{XLM-R}}$ outperforms all dense baselines (lack subnetworks): XLM-R and XLM-R + A. Our variant with the largest number of subnetworks: SPONGE^{6,3}, outperforms all evaluated methods.

Subnetwork exploration. We show the impact of this strategy in the bottom section of Table 1 (left). The models SPONGE^{1,1} and SPONGE^{6,3} (marked with *-x*) indicate that they do *not* use *exploration*. In both cases, *exploration* improves PRAUC, benefiting the sparser SPONGE^{6,3} more.

Table 1: (Left) model performance when trained solely on MIMIC. (Right) average performance on the last dataset of each sequence. SPONGE^{6,3} consistently outperforms across both settings.

	MIMIC Only			Sequential Training		
Model	macro AUROC	micro AUROC	macro PRAUC	macro AUROC	micro AUROC	macro PRAUC
S-PROTO$_{\text{XLM-R}}$	87.51	93.89	28.08	80.08	86.62	37.01
SPONGE^{1,1}	89.68	94.67	33.18	87.47	91.78	52.07
SPONGE^{1,6}	86.63	93.75	22.94	83.57	89.71	43.71
SPONGE^{6,1}	86.55	90.42	31.42	86.19	90.71	48.93
SPONGE^{6,3}	**89.70**	**94.70**	**34.17**	**88.58**	**92.09**	**54.24**
XLMR	86.21	93.67	27.27	83.10	88.89	45.92
XLMR + A	85.74	93.31	26.38	83.81	89.10	45.11
SPONGE^{1,1}-x	89.28	94.66	32.51	–	–	–
SPONGE^{6,3}-x	89.48	94.56	32.56	–	–	–

Sequential training performance. We show in Table 1 (right) the average performance of all models across the 64 training sequences (16 per target dataset; see Appendix E). XLM-R+A performs similarly to XLM-R, reinforcing the effectiveness of PEFT. S-PROTO$_{\text{XLM-R}}$ performs the worst. This emphasizes the inter-task interference created by updating a single Transformer representation for sequential training. PEFT with *output subnetworks* shows significant gains. Specifically, 7.39 points in macro AUROC and 15.06 points in macro PRAUC when comparing SPONGE^{1,1} to S-PROTO$_{\text{XLM-R}}$.

Creating *input subnetworks* exclusively with either the adapter component or the projection is not as beneficial as creating them with both simultaneously. This is evident when comparing SPONGE^{6,1}, SPONGE^{1,6} against SPONGE^{6,3}. In general, increasing sparsity via more subnetworks boosts PRAUC, aligning with findings from Figueroa et al. (2024).

SPONGE^{6,3} consistently demonstrates the best performance regardless of dataset order, language mix, label space diversity, or fine-tuning sequence length, while requiring $\approx 2.6\%$ of the training parameters of XLM-R or S-PROTO$_{\text{XLM-R}}$(see Table 8 in Appendix F). We therefore focus on SPONGE^{6,3} for all subsequent analyses.

Downstream dataset performance. We further analyze sequential training results by examining all permutations ending with each specific target dataset. We present this in Table 2 (top) and highlight how SPONGE^{6,3} outperforms all methods for all metrics, showcasing the robustness to large distribution shifts (languages, dataset sizes, and labels). Generally, downstream dataset performance significantly benefits from multi-step sequential transfer learning, we detail this in Appendix B.

Number of training datasets. We analyze the performance w.r.t. the number of datasets used in a training sequence. We present this in Table 2 (bottom). SPONGE^{6,3} also significantly outperforms all other methods regardless of the number of datasets. We observe that updating all Transformer parameters (XLM-R, and S-PROTO$_{\text{XLM-R}}$) leads to performance degradation as the sequences increase in length.

Table 2: Classification performance per target dataset (top) and number of datasets (bottom) averaged over all permutations of sequences. SPONGE^{6,3}outperforms all methods in both settings.

	macro AUROC				micro AUROC				macro PRAUC			
Model/Dataset	A	B	C	S	A	B	C	S	A	B	C	S
S-PROTO$_{\text{XLM-R}}$	81.83	70.75	83.33	84.42	88.18	80.81	88.51	88.96	42.22	23.83	34.66	47.33
SPONGE^{6,3}	**90.05**	**82.65**	**91.03**	**90.62**	**94.18**	**86.68**	**93.97**	**93.53**	**59.93**	**42.16**	**55.22**	**59.67**
XLM-R	85.78	73.30	85.67	87.63	91.39	81.78	89.70	92.69	58.85	27.73	39.20	57.90
XLM-R+A	84.83	75.93	86.38	88.08	90.27	83.05	90.33	92.76	55.65	28.54	38.77	57.52
Model/ # datasets	2	3	4	5	2	3	4	5	2	3	4	5
S-PROTO$_{\text{XLM-R}}$	81.83	70.75	83.33	84.42	88.18	80.81	88.51	88.96	42.22	23.83	34.66	47.33
SPONGE^{6,3}	**90.05**	**82.65**	**91.03**	**90.62**	**94.18**	**86.68**	**93.97**	**93.53**	**59.93**	**42.16**	**55.22**	**59.67**
XLM-R	85.78	73.30	85.67	87.63	91.39	81.78	89.70	92.69	58.85	27.73	39.20	57.90
XLM-R+A	84.83	75.93	86.38	88.08	90.27	83.05	90.33	92.76	55.65	28.54	38.77	57.52

Performance on unseen data. In Table 3 (left) we present the results for all unseen datasets: DisTEMIST and all datasets absent in a training sequence. Additionally, we adapt S-PROTO$_{\text{XLM-R}}$ and SPONGE with the $\mathcal{H}$ydra strategy. We denote this with $\mathcal{H}$ preceding the name of the model. S-PROTO$_{\text{XLM-R}}$ performs worse than XLM-R and XLM-R+A and fails to improve when employing $\mathcal{H}$ydra. SPONGE^{6,3} performs better than XLM-R and XLM-R + A in PRAUC; however, it falls behind on AUROC. $\mathcal{H}$ydra proves effective, since $\mathcal{H}$SPONGE^{6,3} outperforms all other methods.

Performance on DisTEMIST. The clinical notes of this dataset are multiple versions of the same EHR in different languages, therefore it is a good test bed for generalization. We distinguish between *unseen* and *seen* languages for all trained sequences. We present results in Table 3 center and right. Intuitively, for all models the performance on *seen* languages is higher than on *unseen* languages. $\mathcal{H}$SPONGE^{6,3} consistently outperforms in both cases.

Table 3: Average zero-shot performance of all sequences. *Left*: all datasets (DisTEMIST + those unseen during training). *Center, right*: DisTEMIST only, for both seen and unseen languages respectively.

	Unseen datasets + DisTEMIST			DisTEMIST unseen languages			DisTEMIST seen languages		
Model	macro AUROC	micro AUROC	macro PRAUC	macro AUROC	micro AUROC	macro PRAUC	macro AUROC	micro AUROC	macro PRAUC
S-PROTO$_{\text{XLM-R}}$	68.77	65.40	12.58	69.36	65.11	10.91	72.69	67.23	14.97
$\mathcal{H}$S-PROTO$_{\text{XLM-R}}$	54.39	54.76	8.32	54.69	54.32	6.86	51.41	51.93	8.08
SPONGE^{6,3}	74.46	71.15	20.91	74.44	70.69	19.27	77.74	73.23	23.47
$\mathcal{H}$SPONGE^{6,3}	**81.99**	**77.86**	**24.40**	**82.22**	**77.26**	**22.21**	**85.80**	**81.11**	**27.28**
XLM-R	75.15	73.38	19.03	75.40	72.60	16.42	81.52	78.23	25.17
XLM-R+A	76.18	72.65	18.67	77.14	72.82	17.51	80.50	75.69	21.97

8 Analysis and Discussion

Our experiments show that multi-step sequential transfer learning is generally beneficial for diagnoses prediction. SPONGE outperforms all other methods on all target datasets, regardless of the sequence. Nevertheless, SPONGE struggles to generalize to unseen datasets, often underperforming XLM-R and XLM-R + A (Table 3). However, we demonstrate that $\mathcal{H}$ydra is an effective strategy to address this. We attribute the increased performance of $\mathcal{H}$SPONGE^{6,3} to two main factors: 1) the sparsity and the modular architecture in SPONGE induced by *winner-takes-all*, which creates distributed Transformer representations; and 2) $\mathcal{H}$ydra's ability to leverage these representations, alleviating label recency bias.

Distributed Representations in SPONGE. We investigate the existence of distributed knowledge after sequential training. For this, we test on the first dataset (MIMIC) after training on each of the 64 dataset sequences. As before, we report the average performance of all checkpoints which we denote as *Original* in Table 4. Additionally, we evaluate a variant of each checkpoint where we replace the classification layer with the one trained only on MIMIC; we denote this as $\mathcal{M}$. This allows the model's feature-building network to better align existing knowledge with the MIMIC label space. In other words, we decouple the Transformer representation from the classification layer and measure its impact. $\mathcal{M}$ favors the label space of MIMIC and thus mitigates *label recency* bias.

We observe that S-PROTO$_{\text{XLM-R}}$ does not improve when employing $\mathcal{M}$. We argue that this degradation in performance is related to the dense projection L which is crucial to map to the prototypical space. In contrast, XLM-R and XLM-R + A benefit from $\mathcal{M}$, hinting that the bare Transformer representations can recover knowledge from MIMIC, further emphasizing the need for sparsity in L for S-PROTO$_{\text{XLM-R}}$. Although SPONGE underperforms XLM-R and XLM-R + A, it outperforms all methods when employing $\mathcal{M}$. This significant boost in performance confirms that SPONGE retains MIMIC knowledge but is unable to access it once the prototypical parameters u and W have shifted after sequential-training. The $\mathcal{M}$ strategy is a simplification of $\mathcal{H}$ydra that explains how the latter enables access to distributed knowledge in the model for better generalization as seen in Table 3.

Table 4: Average performance on MIMIC of all 64 checkpoints. Every checkpoint has been trained on two or more datasets. *Original* denotes scores of the unmodified checkpoints; $\mathcal{M}$ corresponds to checkpoints modified by replacing their classification head with the one trained only on MIMIC; Δ denotes the difference.

Model	Macro AUROC			Micro AUROC			Macro PRAUC		
	Original	$\mathcal{M}$	Δ	Original	$\mathcal{M}$	Δ	Original	$\mathcal{M}$	Δ
S-PROTO$_{\text{XLM-R}}$	63.97	46.91	-17.06	68.08	63.34	-4.74	6.85	6.08	-0.77
SPONGE^{6,3}	68.43	**82.30**	+13.87	79.62	83.50	+3.88	14.58	**20.39**	+5.81
XLM-R	76.87	78.91	+2.04	80.74	**87.48**	+6.74	16.68	18.53	+1.85
XLM-R+A	73.00	78.79	+5.79	76.68	87.04	+10.36	11.27	17.40	+6.13

Winner-takes-all and optimal subnetwork. We expand on the analysis of distributed knowledge by focusing on all subnetworks (full-fledged predictors) of SPONGE^{6,3}. For every k^{th} subnetwork (90 in total)

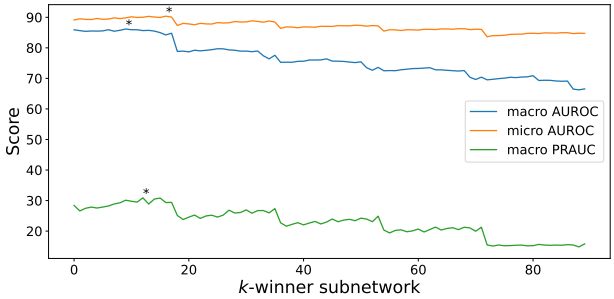

Figure 3: Scores achieved by SPONGE^{6,3} for all subnetworks, ordered by the *winner-takes-all* selection mechanism (see Equation (5)). The decreasing nature of the curves highlights how the top activated subnetworks highly correlate with classification performance, reinforcing the suitability of this selection strategy. Asterisks $*$ point to the k with the highest scores which are close to the *winner-takes-all* selection at $k = 0$.

of this model, we compute all metrics which we present in Figure 3. The *Winner-takes-all* mechanism defines the order of the subnetworks indexed by k (x-axis in the figure). For instance, $k = 89$ is the most inhibited subnetwork. In contrast, $k = 0$ is the single *winner* subnetwork that yields a prediction. We highlight with * the best-performing subnetwork, which proves to be within the top-18 for all metrics. Generally, given the aggregation in Equation (4), predictions are bundled in groups of 18 where performance is very similar, stemming from the 6 adapters T and 3 projections L. This supports the idea that the inhibition mechanism in *winner-takes-all* produces multiple non-interfering predictors that are almost *optimal*.

Hydra and label recency. Prototypical classifiers underperform when not modified with $\mathcal{H}$ydra (see S-PROTO$_{\text{XLM-R}}$ and SPONGE^{6,3} vs. XLM-R and XLM-R + A in Table 3). $\mathcal{H}$SPONGE^{6,3} outperforms all methods on unseen data. We investigate the impact of label recency on prototypical models when exposed to unseen data. Therefore, we compare $\mathcal{H}$SPONGE^{6,3} and S-PROTO$_{\text{XLM-R}}$ since it is the SOTA architecture for this task. For a fair comparison to S-PROTO$_{\text{XLM-R}}$ we focus on its best-performing checkpoint on DisTEMIST i.e., the sequence M→B→C (see performance of S-PROTO$_{\text{XLM-R}}$ in Table 7 in Appendix D and Table 3). We analyze the performance of both models w.r.t. the *recency* of the labels for this sequence. For each label, we define *recency* as the proportion of the most recently trained samples (C) w.r.t all samples in M, B, and C.

Recency and PRAUC. We divide the labels into 8 logarithmically spaced *recency* groups, each containing at least 10 labels to ensure reliable estimation. Figure 4 (*bottom left*) shows the label distribution across these *recency* groups. In Figure 4 (*top left*) we present the difference(Δ) between the two models in macro PRAUC for the labels sorted by *recency* (see x-axis). $\mathcal{H}$SPONGE^{6,3} performs significantly better on all but one label group, thus, Δ PRAUC is positive. This difference decreases as *recency* increases; notably, it is significantly larger for labels with lower *recency*. This illustrates the positive effect of $\mathcal{H}$ydra at mitigating the *recency bias*.

Recency and explainability. Prototypical methods create token saliencies that explain their predictions. Faithfulness quantifies how gradually masking the input according to these saliencies, deteriorates model performance. A lower faithfulness score indicates greater explainability (Atanasova et al., 2020). We generate token saliencies for $\mathcal{H}$SPONGE^{6,3} and S-PROTO$_{\text{XLM-R}}$, and compute faithfulness for macro AUROC and macro PRAUC for the aforementioned *recency* groups. Figure 4 (*right*) presents the difference in faithfulness for $\mathcal{H}$SPONGE^{6,3} and S-PROTO$_{\text{XLM-R}}$. We expand these scores for all DisTEMIST languages (rows in the Figure). Similar to Δ PRAUC, the largest difference is in the first (least recent) label group. This relationship appears to hold independently of the language.

Both classification (PRAUC) and explainability (Faithfulness) metrics confirm that the prototypical layer alone (S-PROTO$_{\text{XLM-R}}$) struggles to generalize in sequential training, likely due to an overemphasis on the most *recent* labels. However, $\mathcal{H}$SPONGE^{6,3} overcomes this in both classification and explainability, reinforcing the importance of combining the distributed representations of SPONGE with $\mathcal{H}$ydra.

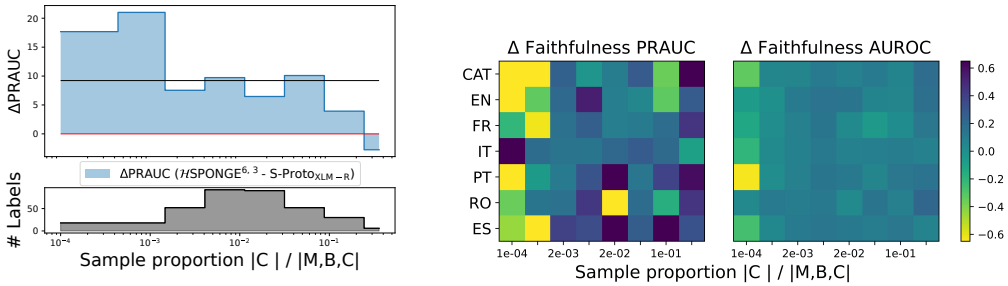

Figure 4: **Left** *top*: $\Delta(\mathcal{H}\text{SPONGE}^{6,3} - \text{S-PROTO}_{\text{XLM-R}})$ in PRAUC over label *recency*. *Bottom*: number of labels by *recency* group. On average (gray dotted line) $\mathcal{H}\text{SPONGE}^{6,3}$generalizes better for all label groups. **Right** $\Delta(\mathcal{H}\text{SPONGE}^{6,3} - \text{S-PROTO}_{\text{XLM-R}})$ Difference in faithfulness for zero-shot datasets (lower is better). Although faithfulness is similar for both models (Δ is roughly 0), S-PROTO$_{\text{XLM-R}}$ is less explainable for the under-represented labels in the last fine-tuning (most negative left region).

Analysis of saliencies with doctors. We further examine 30 patients and five diagnoses with the help of medical doctors. We compare the saliencies of S-PROTO$_{\text{XLM-R}}$ and $\mathcal{H}\text{SPONGE}^{6,3}$ for the previously analyzed sequence M→B→C. We aim to understand how the performance gap between the two models manifests in qualitative differences from a domain expert's perspective. We focus on testing AHEPA (A), which is not included in the sequence and contains the typologically most distant language (Greek). We observe the same performance gap in faithfulness on the least *recent* labels as with the DisTEMIST languages. The physicians were tasked with ranking patients according to their relevance to describe a diagnosis. Remarkably, both models rank the patients similarly, i.e., they successfully identify *prototypical* patients. In terms of the qualitative differences of the saliencies, doctors find that S-PROTO$_{\text{XLM-R}}$is more sensitive to learning annotation artifacts (Zellers et al., 2019) like stopwords or explicit mentions of acronyms of the diagnosis, explaining its weakness in generalization to distinct languages. This issue is compounded by S-PROTO$_{\text{XLM-R}}$ assigning near-uniform relevance scores to most tokens. In contrast, $\mathcal{H}\text{SPONGE}^{6,3}$ sharply highlights key terms related to each diagnosis. These qualitative findings suggest that the zero-shot generalization capabilities of $\mathcal{H}\text{SPONGE}^{6,3}$ is underpinned by clinically meaningful and explainable saliencies.

9 Conclusion

We address the challenges of sequential transfer learning when data is subject to privacy constraints by the example of the clinical domain. We focus on a multi-step sequential transfer learning setting, transferring knowledge via model checkpoints rather than raw data. We propose SPONGE, a novel architecture that excels at knowledge transfer and achieves superior performance in diagnoses prediction. We generalize *winner-takes-all*, a mechanism of neural inhibition inspired by nature, creating competing representations. Extensive analysis reveals that these representations combined with the ability of $\mathcal{H}$ydra to mitigate label-recency bias, are key to strong generalization performance. We confirm with medical doctors that the explanations provided by SPONGE are supported by clinical relevance.

Future Work. We examine a set of six datasets in nine different Indo-European languages, in multi-step sequential transfer learning. While this evaluation is significant, it is not exhaustive; further analysis will be conducted as more clinical data becomes available. Although we focus on a multi-label classification task, Transformers are widely used for solving generative tasks. Given that our architecture may also enhance decoder-based Transformer models, future evaluation on generative tasks is warranted as LLMs become more computationally efficient.

Deployment considerations. In our evaluation, SPONGE performs strongly across high- and low-resource data, sequential training, and generalization tasks. However, the scores achieved still leave room for improvement, which is crucial for the clinical domain. We emphasize that this work does not aim to replace doctors; the applications of technologies to the clinical environment must meet stringent legal and ethical requirements.

References

Alan Ansell, Edoardo Maria Ponti, Anna Korhonen, and Ivan Vulic. Composable sparse fine-tuning for cross-lingual transfer. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pp. 1778–1796. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.ACL-LONG.125. URL https://doi.org/10.18653/v1/2022.acl-long.125.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. A diagnostic study of explainability techniques for text classification. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 3256–3274. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.263. URL https://doi.org/10.18653/v1/2020.emnlp-main.263.

Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *CoRR*, abs/1308.3432, 2013.

Magdalena Biesialska, Katarzyna Biesialska, and Marta R. Costa-jussà. Continual lifelong learning in natural language processing: A survey. In Donia Scott, Núria Bel, and Chengqing Zong (eds.), *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pp. 6523–6541. International Committee on Computational Linguistics, 2020. doi: 10.18653/V1/2020.COLING-MAIN.574. URL https://doi.org/10.18653/v1/2020.coling-main.574.

Keno K. Bressem, Jens-Michalis Papaioannou, Paul Grundmann, Florian Borchert, Lisa C. Adams, Leonhard Liu, Felix Busch, Lina Xu, Jan P. Loyen, Stefan Markus Niehues, Moritz Augustin, Lennart Grosser, Marcus R. Makowski, Hugo J. W. L. Aerts, and Alexander Löser. medbert.de: A comprehensive german BERT model for the medical domain. *Expert Syst. Appl.*, 237(Part C):121598, 2024. doi: 10.1016/J.ESWA.2023.121598. URL https://doi.org/10.1016/j.eswa.2023.121598.

Trenton Bricken, Xander Davies, Deepak Singh, Dmitry Krotov, and Gabriel Kreiman. Sparse distributed memory is a continual learner. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=JknGeelZJpHP.

Ilias Chalkidis, Manos Fergadiotis, and Ion Androutsopoulos. Multieurlex - A multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 6974–6996. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EMNLP-MAIN.559. URL https://doi.org/10.18653/v1/2021.emnlp-main.559.

Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan Su. This looks like that: Deep learning for interpretable image recognition. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 8928–8939, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/adf7ee2dcf142b0e11888e72b43fcb75-Abstract.html.

Zewen Chi, Li Dong, Shaohan Huang, Damai Dai, Shuming Ma, Barun Patra, Saksham Singhal, Payal Bajaj, Xia Song, Xian-Ling Mao, Heyan Huang, and Furu Wei. On the representation collapse of sparse mixture of experts. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/df4f371f1f89ec8ba5014b3310578048-Abstract-Conference.html.

Hyunjin Choi, Judong Kim, Seongho Joe, Seungjai Min, and Youngjune Gwon. Analyzing zero-shot cross-lingual transfer in supervised NLP tasks. In *25th International Conference on Pattern Recognition, ICPR*

2020, Virtual Event / Milan, Italy, January 10-15, 2021, pp. 9608–9613. IEEE, 2020. doi: 10.1109/ICPR48806.2021.9412570. URL https://doi.org/10.1109/ICPR48806.2021.9412570.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *ACL*, pp. 8440–8451. Association for Computational Linguistics, 2020.

Hercules Dalianis, Aron Henriksson, Maria Kvist, Sumithra Velupillai, and Rebecka Weegar. Health bank - a workbench for data science applications in healthcare. In John Krogstie, Gustaf Juell-Skielse, and Vandana Kabilan (eds.), *Proceedings of the CAiSE-2015 Industry Track co-located with 27th Conference on Advanced Information Systems Engineering (CAiSE 2015)*, volume Vol-1381, pp. 1–18, Stockholm, Sweden, June 11 2015. CEUR. URL http://www.dsv.su.se/healthbank.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten P. Bosma, Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, Kathleen S. Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc V. Le, Yonghui Wu, Zhifeng Chen, and Claire Cui. Glam: Efficient scaling of language models with mixture-of-experts. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5547–5569. PMLR, 2022.

Beyza Ermis, Giovanni Zappella, Martin Wistuba, Aditya Rawal, and Cédric Archambeau. Memory efficient continual learning with transformers. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/4522de4178bddb36b49aa26efad537cf-Abstract-Conference.html.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.*, 23:120:1–120:39, 2022.

Alexei Figueroa, Jens-Michalis Papaioannou, Conor Fallon, Alexandra Bekiaridou, Keno Bressem, Stavros Zanos, Felix Gers, Wolfgang Nejdl, and Alexander Löser. Boosting long-tail data classification with sparse prototypical networks. In *ECML/PKDD (7)*, volume 14947 of *Lecture Notes in Computer Science*, pp. 434–449. Springer, 2024.

Alexei Figueroa, Justus Westerhoff, Golzar Atefi, Dennis Fast, Benjamin Winter, Felix Alexader Gers, Alexander Löser, and Wolfang Nejdl. Comply: Learning sentences with complex weights inspired by fruit fly olfaction. *arXiv preprint arXiv:2502.01706*, 2025.

Aryo Gema, Pasquale Minervini, Luke Daines, Tom Hope, and Beatrice Alex. Parameter-efficient fine-tuning of LLaMA for the clinical domain. In Tristan Naumann, Asma Ben Abacha, Steven Bethard, Kirk Roberts, and Danielle Bitterman (eds.), *Proceedings of the 6th Clinical Natural Language Processing Workshop*, pp. 91–104, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.clinicalnlp-1.9. URL https://aclanthology.org/2024.clinicalnlp-1.9/.

Katharina Hämmerl, Jindrich Libovický, and Alexander Fraser. Understanding cross-lingual alignment - A survey. In *ACL (Findings)*, pp. 10922–10943. Association for Computational Linguistics, 2024.

Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jia-Wei Low, Lidong Bing, and Luo Si. On the effectiveness of adapter-based tuning for pretrained language model adaptation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pp. 2208–2222. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.ACL-LONG.172. URL https://doi.org/10.18653/v1/2021.acl-long.172.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika

Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, 2019. URL http://proceedings.mlr.press/v97/houlsby19a.html.

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pp. 328–339. Association for Computational Linguistics, 2018. doi: 10.18653/V1/P18-1031. URL https://aclanthology.org/P18-1031/.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net, 2022.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080, 2020. URL https://arxiv.org/abs/2003.11080.

Yufan Huang, Yanzhe Zhang, Jiaao Chen, Xuezhi Wang, and Diyi Yang. Continual learning for text classification with information disentanglement based regularization. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pp. 2736–2746. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.NAACL-MAIN.218. URL https://doi.org/10.18653/v1/2021.naacl-main.218.

Ayyoob Imani, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André F. T. Martins, François Yvon, and Hinrich Schütze. Glot500: Scaling multilingual corpora and language models to 500 languages. In *ACL (1)*, pp. 1082–1117. Association for Computational Linguistics, 2023.

Minbyul Jeong, Mujeen Sung, Gangwoo Kim, Donghyeon Kim, Wonjin Yoon, Jaehyo Yoo, and Jaewoo Kang. Transferability of natural language inference to biomedical question answering. In Linda Cappellato, Carsten Eickhoff, Nicola Ferro, and Aurélie Névéol (eds.), *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, Thessaloniki, Greece, September 22-25, 2020*, volume 2696 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2020. URL https://ceur-ws.org/Vol-2696/paper_44.pdf.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. Mimic-iv, 2021. URL https://physionet.org/content/mimiciv/1.0/.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. Mimic-iv, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, Jan 2023. ISSN 2052-4463. doi: 10.1038/s41597-022-01899-x. URL https://doi.org/10.1038/s41597-022-01899-x.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114. URL https://www.pnas.org/doi/abs/10.1073/pnas.1611835114.

Anastasios Lamproudis, Thomas Olsen Svenning, Trond Torsvik, Taridzo Chomutare, Andrius Budrionis, Phuong Dinh Ngo, Thomas Vakili, and Hercules Dalianis. Using a large open clinical corpus for improved icd-10 diagnosis coding. In *Proceedings of AMIA 2023, Annual Symposium*, New Orleans, LA, USA, November 11-15 2023. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10785868/.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *ICLR*. OpenReview.net, 2021.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 3045–3059. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EMNLP-MAIN.243. URL https://doi.org/10.18653/v1/2021.emnlp-main.243.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL/IJCNLP (1)*, pp. 4582–4597. Association for Computational Linguistics, 2021.

Vladislav Lialin, Vijeta Deshpande, and Anna Rumshisky. Scaling down to scale up: A guide to parameter-efficient fine-tuning. *CoRR*, abs/2303.15647, 2023. doi: 10.48550/ARXIV.2303.15647. URL https://doi.org/10.48550/arXiv.2303.15647.

Yuchen Liang, Chaitanya K. Ryali, Benjamin Hoover, Leopold Grinberg, Saket Navlakha, Mohammed J. Zaki, and Dmitry Krotov. Can a fruit fly learn word embeddings? In *ICLR*. OpenReview.net, 2021.

Seong Hoon Lim, Taejun Yun, Jinhyeon Kim, Jihun Choi, and Taeuk Kim. Analysis of multi-source language training in cross-lingual transfer. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 712–725. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.42. URL https://doi.org/10.18653/v1/2024.acl-long.42.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. Choosing transfer languages for cross-lingual learning. In *ACL (1)*, pp. 3125–3135. Association for Computational Linguistics, 2019.

Charles X Ling, Ganyu Wang, and Boyu Wang. Sparse and expandable network for google's pathways. *Front. Big Data*, 7:1348030, August 2024.

Chen Liu, Jonas Pfeiffer, Ivan Vulic, and Iryna Gurevych. FUN with fisher: Improving generalization of adapter-based cross-lingual transfer with scheduled unfreezing. In Kevin Duh, Helena Gómez-Adorno, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pp. 1998–2015. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.NAACL-LONG.111. URL https://doi.org/10.18653/v1/2024.naacl-long.111.

Dan Malkin, Tomasz Limisiewicz, and Gabriel Stanovsky. A balanced data approach for evaluating cross-lingual transfer: Mapping the linguistic blood bank. In Marine Carpuat, Marie-Catherine de Marneffe, and Iván Vladimir Meza Ruíz (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pp. 4903–4915. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.NAACL-MAIN.361. URL https://doi.org/10.18653/v1/2022.naacl-main.361.

Francisco Rogerlândio Martins-Melo, Alberto Novaes Ramos, Jr, Carlos Henrique Alencar, and Jorg Heukelbach. Prevalence of chagas disease in brazil: a systematic review and meta-analysis. *Acta Trop.*, 130: 167–174, February 2014.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Antonio Miranda-Escalada, Aitor Gonzalez-Agirre, Jordi Armengol-Estapé, and Martin Krallinger. Overview of automatic clinical coding: annotations, guidelines, and solutions for non-english clinical cases at codiesp track of clef ehealth 2020. In *Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings*, 2020.

Antonio Miranda-Escalada, Luis Gascó, Salvador Lima-López, Eulàlia Farré-Maduell, Darryl Estrada, Anastasios Nentidis, Anastasia Krithara, Georgios Katsimpras, Georgios Paliouras, and Martin Krallinger. Overview of distemist at bioasq: Automatic detection and normalization of diseases from clinical texts: results, methods, evaluation and multilingual resources. In *CLEF (Working Notes)*, volume 3180 of *CEUR Workshop Proceedings*, pp. 179–203. CEUR-WS.org, 2022.

G M Naylor, T Gotoda, M Dixon, T Shimoda, L Gatta, R Owen, D Tompkins, and A Axon. Why does japan have a high incidence of gastric cancer? comparison of gastritis between UK and japanese patients. *Gut*, 55(11):1545–1552, November 2006.

Dinh C. Nguyen, Quoc-Viet Pham, Pubudu N. Pathirana, Ming Ding, Aruna Seneviratne, Zihuai Lin, Octavia A. Dobre, and Won-Joo Hwang. Federated learning for smart healthcare: A survey. *ACM Comput. Surv.*, 55(3):60:1–60:37, 2023. doi: 10.1145/3501296. URL `https://doi.org/10.1145/3501296`.

Luiz E. S. E. Oliveira, Ana Carolina Peters, Anderson M. P. da Silva, Caroline P. Gebeluca, Yohan B. Gumiel, Luciana M. M. Cintho, Diego R. Carvalho, Sami Al Hasan, and Claudia M. C. Moro. Semclinbr - a multi-institutional and multi-specialty semantically annotated corpus for portuguese clinical nlp tasks. *Journal of Biomedical Semantics*, 13(1):13, 2022. doi: 10.1186/s13326-022-00269-1. URL `https://doi.org/10.1186/s13326-022-00269-1`.

Jens-Michalis Papaioannou, Paul Grundmann, Betty van Aken, Athanasios Samaras, Ilias Kyparissidis, George Giannakoulas, Felix A. Gers, and Alexander Löser. Cross-lingual knowledge transfer for clinical phenotyping. In *LREC*, pp. 900–909. European Language Resources Association, 2022.

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, and Sebastian Ruder. MAD-X: an adapter-based framework for multi-task cross-lingual transfer. In *EMNLP (1)*, pp. 7654–7673. Association for Computational Linguistics, 2020.

Fred Philippy, Siwen Guo, and Shohreh Haddadan. Towards a common understanding of contributing factors for cross-lingual transfer in multilingual language models: A review. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pp. 5877–5891. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.ACL-LONG.323. URL `https://doi.org/10.18653/v1/2023.acl-long.323`.

Clifton Poth, Jonas Pfeiffer, Andreas Rücklé, and Iryna Gurevych. What to pre-train on? efficient intermediate task selection. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 10585–10605. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EMNLP-MAIN.827. URL `https://doi.org/10.18653/v1/2021.emnlp-main.827`.

Clifton Poth, Hannah Sterz, Indraneil Paul, Sukannya Purkayastha, Leon Engländer, Timo Imhof, Ivan Vulic, Sebastian Ruder, Iryna Gurevych, and Jonas Pfeiffer. Adapters: A unified library for parameter-efficient and modular transfer learning. In Yansong Feng and Els Lefever (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023 - System Demonstrations, Singapore, December 6-10, 2023*, pp. 149–160. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-DEMO.13. URL `https://doi.org/10.18653/v1/2023.emnlp-demo.13`.

Vitaly Protasov, Elisei Stakovskii, Ekaterina Voloshina, Tatiana Shavrina, and Alexander Panchenko. Super donors and super recipients: Studying cross-lingual transfer between high-resource and low-resource languages. In Atul Kr. Ojha, Chao-hong Liu, Ekaterina Vylomova, Flammie Pirinen, Jade Abbott, Jonathan Washington, Nathaniel Oco, Valentin Malykh, Varvara Logacheva, and Xiaobing Zhao (eds.), *Proceedings of the Seventh Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2024)*, pp. 94–108, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.loresmt-1.10. URL `https://aclanthology.org/2024.loresmt-1.10`.

Tom Roehr, Alexei Figueroa, Jens-Michalis Papaioannou, Conor Fallon, Keno K. Bressem, Wolfgang Nejdl, and Alexander Löser. Revisiting clinical outcome prediction for MIMIC-IV. In *ClinicalNLP@NAACL*, pp. 208–217. Association for Computational Linguistics, 2024.

Sebastian Ruder. *Neural Transfer Learning for Natural Language Processing*. PhD thesis, National University of Ireland, Galway, 2019.

Konstantinos Sechidis, Grigorios Tsoumakas, and Ioannis P. Vlahavas. On the stratification of multi-label data. In *ECML/PKDD (3)*, volume 6913 of *Lecture Notes in Computer Science*, pp. 145–158. Springer, 2011.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *ICLR (Poster)*. OpenReview.net, 2017.

Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *NIPS*, pp. 4077–4087, 2017.

Sai Ashish Somayajula, Youwei Liang, Li Zhang, Abhishek Singh, and Pengtao Xie. Generalizable and stable finetuning of pretrained language models on low-resource texts. In Kevin Duh, Helena Gómez-Adorno, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pp. 4936–4953. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.NAACL-LONG.277. URL https://doi.org/10.18653/v1/2024.naacl-long.277.

Yi-Lin Sung, Varun Nair, and Colin Raffel. Training neural networks with fixed sparse masks. In *NeurIPS*, pp. 24193–24205, 2021.

Richard S. Sutton. Generalization in reinforcement learning: Successful examples using sparse coarse coding. In David S. Touretzky, Michael Mozer, and Michael E. Hasselmo (eds.), *Advances in Neural Information Processing Systems 8, NIPS, Denver, CO, USA, November 27-30, 1995*, pp. 1038–1044. MIT Press, 1995.

Robert Tinn, Hao Cheng, Yu Gu, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Fine-tuning large neural language models for biomedical natural language processing. *Patterns*, 4(4):100729, 2023.

Betty van Aken, Jens-Michalis Papaioannou, Manuel Mayrdorfer, Klemens Budde, Felix A. Gers, and Alexander Löser. Clinical outcome prediction from admission notes using self-supervised knowledge integration. In *EACL*, pp. 881–893. Association for Computational Linguistics, 2021.

Betty van Aken, Jens-Michalis Papaioannou, Marcel Ganesh Naik, Georgios Eleftheriadis, Wolfgang Nejdl, Felix A. Gers, and Alexander Löser. This patient looks like that patient: Prototypical networks for interpretable diagnosis prediction from clinical text. In *AACL/IJCNLP (1)*, pp. 172–184. Association for Computational Linguistics, 2022.

Guangyu Wang, Xiaohong Liu, Zhen Ying, Guoxing Yang, Zhiwei Chen, Zhiwen Liu, Min Zhang, Hongmei Yan, Yuxing Lu, Yuanxu Gao, Kanmin Xue, Xiaoying Li, and Ying Chen. Optimized glycemic control of type 2 diabetes with reinforcement learning: a proof-of-concept trial. *Nature Medicine*, 29(10):2633–2642, Oct 2023. ISSN 1546-170X. doi: 10.1038/s41591-023-02552-9. URL https://doi.org/10.1038/s41591-023-02552-9.

Benjamin Winter, Alexei Figueroa Rosero, Alexander Löser, Felix Alexander Gers, and Amy Siu. KIMERA: injecting domain knowledge into vacant transformer heads. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Jan Odijk, and Stelios Piperidis (eds.), *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022, Marseille, France, 20-25 June 2022*, pp. 363–373. European Language Resources Association, 2022. URL https://aclanthology.org/2022.lrec-1.38.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pp. 38–45. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-DEMOS.6. URL https://doi.org/10.18653/v1/2020.emnlp-demos.6.

Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *ACM Trans. Knowl. Discov. Data*, 18(6), April 2024. ISSN 1556-4681. doi: 10.1145/3649506. URL https://doi.org/10.1145/3649506.

Alan L. Yuille and Norberto M. Grzywacz. A winner-take-all mechanism based on presynaptic inhibition feedback. *Neural Comput.*, 1(3):334–347, 1989. doi: 10.1162/NECO.1989.1.3.334. URL https://doi.org/10.1162/neco.1989.1.3.334.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In Anna Korhonen, David R. Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 4791–4800. Association for Computational Linguistics, 2019. doi: 10.18653/V1/P19-1472. URL https://doi.org/10.18653/v1/p19-1472.

A Datasets

We present in Table 5 all data splits created with stratified sampling for the datasets used for training and evaluation in our experiments. DisTEMIST comprehends the same 439 EHR notes in seven different languages, for a total of 3,073 samples.

Table 5: Dataset splits.

Dataset	Train	Val	Test
(M) MIMIC (Johnson et al., 2023)	102,199	9,358	7,618
(A) Achepa (Papaioannou et al., 2022)	1,590	407	394
(S) Stockholm University (Lamproudis et al., 2023)	1,583	256	232
(C) CodieEsp (Miranda-Escalada et al., 2020)	656	158	182
(B) SemclinBr (Oliveira et al., 2022)	453	107	109
(D) DisTEMIST (Miranda-Escalada et al., 2022)	—	—	3,073 (439 × 7)

B Impact of sequential training on downstream performance

We compare the average performance of models trained sequentially against models trained exclusively on a single dataset. Table 6 shows the absolute performance difference for every dataset. With very few exceptions, the performance boost is significant, highlighting the positive effect of knowledge transfer in sequential training. SPONGE^{6,3} outperforms all models on all datasets on the average sequence when finetuned sequentially. S-PROTO$_{\text{XLM-R}}$ outperforms XLM-R and XLM-R + A when trained and evaluated only on MIMIC(single dataset). However, for the other datasets that are trained in isolation S-PROTO$_{\text{XLM-R}}$ is the worst-performing model, indicating that it is not suitable for low-resource datasets. Moreover, SPONGE^{6,3} outperforms all models also for the case of single datasets which shows the data efficiency of our approach.

Generally, all models perform better when sequentially trained, this is confirmed by the positive performance difference for both metrics (bottom of Table 6).

Table 6: Performance difference when comparing sequential training results versus single-dataset training for every downstream dataset. We underline the best results when training on a single dataset (top), and **boldface** (middle) stands for the best results in sequential-transfer learning. All models benefit significantly(bottom) from multi-step sequential transfer learning.

		macro AUROC					macro PRAUC				
	Model/Dataset	M	A	B	C	S	M	A	B	C	S
Single dataset	S-PROTO$_{\text{XLM-R}}$	87.51	78.70	56.55	65.46	69.29	28.08	34.41	7.17	13.38	26.73
	SPONGE^{6,3}	89.70	89.48	67.05	82.92	84.17	34.17	60.65	20.53	36.32	43.33
	XLM-R	86.21	87.37	65.18	77.74	87.01	27.27	55.80	18.29	28.18	63.80
	XLM-R+A	85.74	81.43	63.68	79.69	87.35	26.38	52.08	16.73	31.54	53.12
Sequence average	S-PROTO$_{\text{XLM-R}}$	-	81.83	70.75	83.33	84.42	-	42.22	23.83	34.66	47.33
	SPONGE^{6,3}	-	**90.05**	**82.65**	**91.03**	**90.62**	-	**59.93**	**42.16**	**55.22**	**59.67**
	XLM-R	-	85.78	73.30	85.67	87.63	-	58.85	27.73	39.20	57.90
	XLM-R+A	-	84.83	75.93	86.38	88.08	-	55.65	28.54	38.77	57.52
Performance Difference	S-PROTO$_{\text{XLM-R}}$	-	+3.13	+14.20	+17.87	+15.13	-	+7.81	+16.66	+21.28	+20.60
	SPONGE^{6,3}	-	+0.57	+15.6	+8.11	+6.45	-	-0.72	+21.63	+18.9	+16.34
	XLM-R	-	-1.59	+8.12	+7.93	+0.62	-	+3.05	+9.44	+11.02	-5.90
	XLM-R+A	-	+3.4	+12.25	+6.69	+0.73	-	+3.57	+11.81	+7.23	+4.4

C Hyperparameters

We use subnetwork exploration for 10 epochs to build distributed knowledge of MIMIC, with a batch size of 8 and learning rate as per Pfeiffer et al. (2020). We disable exploration for the remaining datasets of the training sequences. For S-PROTO$_{\text{XLM-R}}$ we use hyperparameters as in Figueroa et al. (2024). SPONGE and S-PROTO$_{\text{XLM-R}}$ use the same hidden dimensions for the prototypical layer, XLM-R as a Transformer, and are trained on four A100 GPUs.

Adapter choice We evaluate SPONGE^{1,1} and XLM-R+A across the four low-resource datasets using different adapter types: prefix-tuning (Li & Liang, 2021), LoRA (Hu et al., 2022), prompt-tuning (Lester et al., 2021) and bottleneck adapters (Pfeiffer et al., 2020). Consistent with findings from Poth et al. (2023), bottleneck adapters are the most effective for downstream classification tasks for SPONGE. For XLM-R, LoRA delivered the best performance. We use the best performing adapters for each model family. All models with adapters have significantly less trainable parameters than both S-PROTO$_{\text{XLM-R}}$ and XLM-R.

Cross-lingual Transformer choice. Because of the large amount of experiments and the extent of our evaluation, we consider smaller Transformers like XLM-R instead of significantly larger LLMs. We favor XLM-R over Glot500 (Imani et al., 2023) since the latter performs worse on text classification tasks in all but one of the languages of our training datasets (see Table 19 in Imani et al. (2023)).

D Best performing sequence checkpoint of S-Proto for DisTEMIST

In Table 7 we report the best performing sequence results of S-PROTO$_{\text{XLM-R}}$ and compare them to those of $\mathcal{H}$SPONGE^{6,3}. This sequence corresponds to Mimic(English) $\rightarrow$ SemClinBr(Portuguese) $\rightarrow$ CodiEsp(Spanish) or M$\rightarrow$B$\rightarrow$C. The two models resulting from this sequence significantly outperform the average scores over all sequences of all models reported in Table 3.

Table 7: Sequence $M \rightarrow B \rightarrow C$ yields the best S-Proto model for DisTEMIST, we report results for this sequence for $\mathcal{H}$SPONGE^{6,3}.

Model	macro AUROC	micro AUROC	macro PRAUC
S-Proto	86.16	84.73	25.25
$\mathcal{H}$SPONGE^{6,3}	92.26	91.89	34.19

E All training sequences

In Figure 5 we list all generated 64 dataset sequences that start with MIMIC. Every node corresponds to the end of a sequence and a model checkpoint.

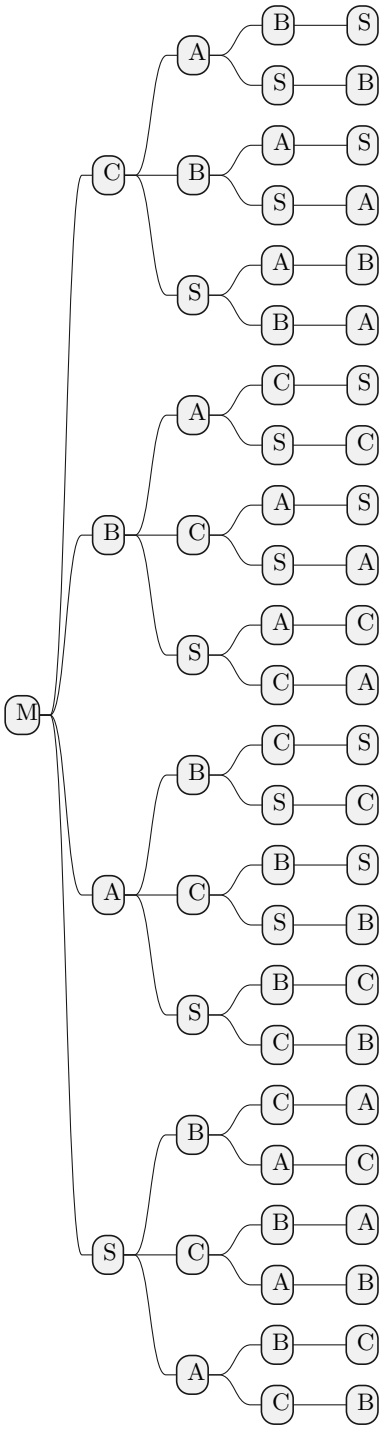

Figure 5: All 64 training sequences that we evaluate in our experiments.

F Trainable parameters

Table 8 demonstrates the number of trainable parameters of the models in our experiments. We decompose these according to the corresponding modules of the architectures. Although SPONGE^{6,3} features the largest number of competing subnetworks (90) it is the second smallest model w.r.t. the number of parameters that are updated. Note, that $\mathcal{H}$ydra is used only for inference and involves no training parameters.

Table 8: Number of subnetworks and trainable parameters for each model

Model	Subnetworks	Transformer	Adapters	Projection L	Prototypical parameters (W and u)	Total
S-PROTO$_{\text{XLM-R}}$	1×5	270M	-	0.19M	5×0.25M	271.5M
XLM-R	1	270M	-		-	270M
SPONGE^{6,3}	6×3×5	-	6×0.88M	3×0.19M	5×0.25M	7.15M
XLM-R+A	1	-	0.88M		-	0.88M