ZERO/FEW-SHOT DARK KINASE-PHOSPHOSITE ASSOCIATION PREDICTION WITH BIOLOGICALLY GROUNDED DATA AUGMENTATION

by
MERT PEKEY

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfilment of
the requirements for the degree of Master of Science

Sabancı University
July 2025

ZERO/FEW-SHOT DARK KINASE-PHOSPHOSITE ASSOCIATION PREDICTION WITH BIOLOGICALLY GROUNDED DATA AUGMENTATION

Approved by:

Assoc. Prof. Dr. Öznur TAŞTAN OKAN ....
(Thesis Supervisor)

Asst. Prof. Dr. Dilara KEKÜLLÜOĞLU ....

Assoc. Prof. Dr. A. Ercüment ÇİÇEK ....

Date of Approval:

ABSTRACT

ZERO/FEW-SHOT DARK KINASE-PHOSPHOSITE ASSOCIATION PREDICTION WITH BIOLOGICALLY GROUNDED DATA AUGMENTATION

MERT PEKEY

COMPUTER SCIENCE & ENGINEERING
MSc. THESIS
JULY 2025

Thesis Supervisor: Assoc. Prof. Öznur Taştan Okan

Keywords: Protein Sequence Classification, Zero/Shot Learning, Phosphorylation, Dark Kinases, Post-translational Modifications, Conditional Generative Models, Data Augmentation

Protein phosphorylation, a fundamental cellular process mediated by kinases, is crucial for signaling, and its dysregulation is implicated in numerous human diseases. A significant challenge persists in identifying substrate phosphosites for the vast number of understudied 'dark' kinases, for which conventional supervised machine learning methods are ineffective due to data scarcity. To address this gap, this thesis develops a zero- and few-shot learning framework and introduces biologically grounded data augmentation strategies, all evaluated on the DARKIN benchmark.

We introduce two novel deep learning architectures: DARKIN-FT, a compatibility-based model that enhances performance through end-to-end fine-tuning of phosphosite encoder, and DARKIN-Interact, a binary classification model that directly captures kinase–substrate interactions via joint attention over sequence pairs. The central contribution is a systematic investigation into biologically grounded data augmentation, evaluating three distinct strategies: (i) kinase-conditional phosphosite generation via a fine-tuned ProGen2 model, (ii) weak supervision using predictions from the Kinase Substrate Specificity Atlas (KSSA), and (iii) augmentation with homologous sequences.

Our results demonstrate that DARKIN-FT and DARKIN-Interact significantly outperform existing baselines on the DARKIN benchmark. The investigation into data augmentation yielded mixed results: while kinase conditional generation with ProGen2 and weak labeling with KSSA degraded the performance, augmentation with homologous sequences improved the Macro Average Precision of the DARKIN-Interact model. While the results are promising, challenges persist in disambiguating kinases with high sequence similarity.

Overall, this thesis establishes a framework for kinase–phosphosite interaction prediction in low-data regimes and provides valuable insights into the strengths and limitations of data augmentation in the dark kinase-phosphosite association task.

ÖZET

BİYOLOJİK TEMELLİ VERİ ARTIRIMIYLA SIFIR/AZ ÖRNEKLİ
KARANLIK KİNAZ–FOSFOSİT İLİŞKİSİ TAHMİNİ

MERT PEKEY

BİLGİSAYAR BİLİMİ & MÜHENDİSLİĞİ
YÜKSEK LİSANS TEZİ
TEMMUZ 2025

Tez Danışmanı: Doç. Dr. Öznur Taştan Okan

Anahtar Kelimeler: Protein Dizisi Sınıflandırması, Sıfır/ Az Örnekli Öğrenme, Fosforilasyon, Karanlık Kinazlar, Translasyon Sonrası Modifikasyonlar, Koşullu Üretici Modeller, Veri Artırma

Protein fosforilasyonu, kinazlar tarafından düzenlenen temel bir hücresel süreç olup hücre içi sinyal iletiminde kritik bir rol oynar. Bu sürecin bozulması, birçok insan hastalığında etkili olmaktadır. Ancak, geleneksel denetimli makine öğrenimi yöntemleri, veri yetersizliği nedeniyle henüz yeterince çalışılmamış 'karanlık' kinazlara ait substrat fosfositleri belirlemede yetersiz kalmaktadır. Bu eksikliği gidermek amacıyla, bu tezde sıfır ve az örnekli öğrenmeye dayalı bir çerçeve geliştirilmiş ve biyolojik temelli veri artırma stratejileri sunulmuştur. Tüm yöntemler DARKIN karşılaştırma seti üzerinde değerlendirilmiştir.

Bu kapsamda iki yeni derin öğrenme mimarisi önerilmiştir: Fosfosit kodlayıcısını uçtan uca ince ayar yoluyla optimize eden ve uyumluluğa dayalı bir model olan DARKIN-FT, ile dizi çiftleri üzerinde ortak dikkat mekanizması kullanarak doğrudan kinaz-fosfosit etkileşimlerini modelleyen ikili sınıflandırma tabanlı DARKIN-Interact. Tezin temel katkısı, biyolojik gerçekliğe dayanan veri artırma yaklaşımlarının sistematik olarak incelenmesidir. Bu amaçla üç farklı strateji değerlendirilmiştir: (i) ProGen2 modelinin ince ayarlanmasıyla gerçekleştirilen kinaz-koşullu fosfosit üretimi; (ii) Kinase Substrate Specificity Atlas (KSSA) kullanılarak yapılan zayıf denetimli etiketleme; ve (iii) homolog sekanslarla veri artırımı.

Elde ettiğimiz sonuçlar, DARKIN-FT ve DARKIN-Interact modellerinin DARKIN karşılaştırma setindeki mevcut temel yöntemlere kıyasla anlamlı performans artışı sağladığını göstermektedir. Veri artırma stratejilerinin etkisi ise karışıktır: ProGen2 ile yapılan üretim ve KSSA ile zayıf etiketleme performansı düşürürken, homolog sekanslarla yapılan veri artırımı özellikle DARKIN-Interact modeli için Makro Ortalama Kesinliği artırmada etkili olmuştur. Her ne kadar elde edilen sonuçlar umut verici olsa da, yüksek dizi benzerliğine sahip kinazları ayırt etme konusunda zorluklar devam etmektedir.

Genel olarak bu tez, düşük veri koşullarında kinaz–fosfosit etkileşimi tahmini için bir çerçeve sunmakta ve karanlık kinaz–fosfosit ilişkisinin modellenmesinde veri artırma stratejilerinin güçlü ve zayıf yönlerine dair önemli içgörüler sağlamaktadır.

ACKNOWLEDGEMENTS

This thesis would not have been possible without the unwavering support and guidance of many individuals.

First and foremost, I extend my deepest gratitude to my advisor, Assoc. Prof. Öznur Taştan, for her invaluable mentorship throughout my undergraduate and graduate education. Her consistent support, profound insights, and endless motivation have been fundamental to my academic journey and have brought me to this significant point in my career.

I am also immensely grateful to Assoc. Prof. Ramazan Gökberk Cinbiş, our dedicated co-advisor from the Tübitak 1001 Project (122E500). His significant contributions and expert guidance were crucial in navigating and resolving the challenges encountered during our project.

My sincere thanks go to the esteemed members of my thesis jury, Asst. Prof. Dilara Keküllüoğlu and Assoc. Prof. Ercüment Çiçek, for dedicating their valuable time to participate in the defense and for offering their insightful comments and constructive feedback.

I would like to express my appreciation to Prof. Albert Levi for his mentorship during my undergraduate graduation project and for his crucial support as I began my graduate studies. Furthermore, I am thankful to Prof. Erchan Aptoula and Prof. Ayşe Berrin Yanıkoğlu for their continuous support throughout my master's studies, especially for the opportunity to serve as a teaching assistant in their courses.

I am very grateful to the Scientific and Technological Research Council of Turkey (TÜBİTAK) for funding through the TUBITAK 1001 project 122E500. The computational results presented in this thesis were carried out using the resources of TÜBİTAK ULAKBİM, High Performance and Grid Computing Center, TRUBA.

I am also deeply indebted to my friends and fellow group members from our Tübitak Project, Emine Ayşe Sunar and Zeynep Işık. Our two years of close collaboration were incredibly productive, and their assistance was invaluable during this period.

My graduate journey was significantly enriched by the friendship and assistance of all my friends, who helped me overcome numerous obstacles. Their friendship was truly invaluable. A special mention goes to Müge Kuşkon, Kutay Altıntaş, Alperen

Doğan, Anes Abdennebi, Uğur Berk Çelik, Simge Demir, Begüm Arslanhan, Ceren Yıldırım, Veysel Oğulcan Kaya, and İrem Ergün.

Moreover, I am grateful for the strong and lasting bonds I made with Deniz, Furkan, Barış, Batuhan, Ali Barış, Kuzey, Ayça, and countless others from our shared experiences in high school and university.

Finally, and most importantly, I express my deepest gratitude to my beloved sister, İrem, and my wonderful parents, Hakan and Beyhan. Their endless love, unwavering belief, and steady support have been the cornerstone of my life, without which I could not have reached this milestone.

Dedicated
to my family

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

$\mathcal{Y}_{\text{tr}}$:	Set of seen kinase classes
$\mathcal{Y}_{\text{te}}$:	Set of unseen kinase classes
k:	Number of few-shot classes
$\theta(x)$:	Phosphosite representation
$\theta(y)$:	Kinase representation
$f_{\text{fam}}(y)$:	Kinase family (one-hot)
$f_{\text{grp}}(y)$:	Kinase group (one-hot)
$f_{\text{ec}}(y)$:	EC number (one-hot)
$\mathbf{W}$:	Bilinear compatibility matrix
$\mathcal{M}$:	Tokens used in Progen2 fine-tuning loss
$\mathcal{V}$:	Amino acid vocabulary for BLOSUM loss
α:	Non-central residue weight (BLOSUM)
β:	Central weight for S/T correctness (BLOSUM)
γ:	Central weight for S/T vs Y confusion (BLOSUM)
δ:	Central weight for other residue confusion (BLOSUM)
$\mathbf{W}_g$:	Gated fusion projection (sequence + attributes)
$\mathbf{W}_a$:	Kinase attribute projection
$\sigma(\cdot)$:	Sigmoid activation
l:	Number of final encoder layers

LIST OF ABBREVIATIONS

DARKIN:	Dark-Kinase benchmark dataset
BZSM:	Bilinear Zero-Shot Model
DARKIN-FT:	BZSM with fine-tuned phosphosite encoder
DARKIN-Interact:	DARKIN-FT variant modeling kinase–substrate interactions
DARKIN-KL-ST:	DARKIN containing only S/T kinases validated in KSSA
DARKIN-KL-Y:	DARKIN containing only Y kinases validated in KSSA
ESM:	Evolutionary-Scale Modeling
KSSA:	Kinase–Substrate Specificity Atlas
Acc:	Accuracy
AP:	Average Precision
Macro AP:	Mean per-class Average Precision
Fgrain AP:	Fine-grained Cluster Average Precision
EC:	Enzyme Commission number
CE:	Cross-Entropy loss
BCE:	Binary Cross-Entropy loss
LoRA:	Low-Rank Adaptation
pLM:	Protein Language Model
CLS:	Classification token
EOS:	End-of-sequence token
PTM:	Post-Translational Modification
MSA:	Multiple Sequence Alignment
PHACT:	Phylogeny-Aware Computing of Tolerance
GPU:	Graphics Processing Unit

Chapter 1

INTRODUCTION

A key mechanism in cellular control is protein phosphorylation, which involves the reversible attachment of a phosphate group to a protein (Hunter, 1995). This common post-translational modification (PTM) functions as a molecular switch, altering a protein's behavior, location, and stability. The reaction is mediated by protein kinases, a large class of enzymes that catalyze the transfer of a phosphate group from ATP onto specific serine, threonine, or tyrosine residues within their target substrates. Because phosphorylation is integral to most cellular activities, from signaling to division, its dysregulation is often associated with human diseases such as cancer and neurodegenerative disorders (Ardito et al., 2017). The residue in the target protein is called a phosphorylation site (phosphosite). Identifying the specific kinase responsible for modifying a given phosphosite is therefore essential for advancing both basic biological research and the development of targeted therapies.

High-throughput phosphoproteomics methods can identify phosphosites on a massive scale (Hornbeck et al., 2012), but they cannot report the kinase that performs the modification. Pinpointing which kinase catalyzes each site requires further experimentation, which is often costly and labour intensive. Although many phosphosites have been identified, most of them do not have a known kinase. Over 95% of human phosphosites lack an experimentally confirmed kinase (Needham et al., 2019). Also, around 25% of kinases have no known targets, and for about 35% of kinases, only a few phosphosites are known. As a result, a large portion of the phosphoproteome and kinome remains in the dark (Needham et al., 2019; Moret et al., 2020; Deznabi et al., 2020; Vella et al., 2022). This leaves a significant portion of the human kinome unexplored, and its role in health and disease is poorly understood.

Because experimental techniques to find the associated kinase of a phosphosite are labor-intensive, computational methods have been developed (Blom et al., 1999;

Yaffe et al., 2001; Koenig and Grabe, 2004; Wong et al., 2007; Li et al., 2008; Saunders et al., 2008; Gao et al., 2010; Xue, Z. Liu, et al., 2010; L. Zou et al., 2013; Horn et al., 2014; Patrick et al., 2014; Qin et al., 2016; Song et al., 2017). While these early approaches relied on sequence motifs or traditional machine learning, later approaches rely on deep learning-based methods. In particular, Protein Language Models (pLMs), pre-trained on vast databases of protein sequences, have shown a remarkable ability to learn the complex sequence to protein structure and function, capturing subtle sequence features far beyond simple motifs (Rives et al., 2021; Elnaggar et al., 2021; Lin et al., 2023; Madani et al., 2023; Nijkamp et al., 2023; Hayes et al., 2025; Lv et al., 2025). These models have therefore been increasingly adopted in phosphorylation-related tasks. However, their effectiveness remains constrained by the scarcity of labeled data for understudied kinases. As a result, their performance significantly drops when applied to the dark kinome, highlighting the need for alternative modeling strategies.

Traditional supervised models fail when no labeled examples exist for a given kinase. To handle this, the problem of finding the cognate kinase of a phosphosite have been casted as a zero-shot learning problem for the first time in the DeepKinZero work. Zero-shot learning (ZSL) enables prediction for dark kinases by learning a mapping between the phosphosite sequence space and a semantic space of kinase attributes. Instead of learning to classify kinases directly, the model learns to associate a phosphosite's sequence patterns with generalizable kinase characteristics (e.g., features from their own sequence, family, or group). This allows the model to predict interactions for a kinase it has never encountered during training (the zero-shot scenario) by projecting it into the shared semantic space. The framework naturally extends to the few-shot scenario, where predictions for a novel kinase can be refined using just a handful of available examples. This paradigm, which builds on prior work of Deznabi et al. (2020), is particularly well-suited to exploring the dark kinome, offering a principled way to generate hypotheses for thousands of unannotated phosphosites and understudied kinases.

While the zero-shot framework provides a powerful foundation, its predictive accuracy is still limited by the diversity and volume of the initial training data. This data scarcity problem is intensified by the very structure of the DARKIN benchmark introduced by Sunar et al. (2024), which is designed for rigorous zero-shot evaluation. By assigning data-poor "dark" kinases to the training set while reserving well-characterized "light" kinases for testing, DARKIN creates an intentionally challenging and sparse training environment. DARKIN is structured this way to ensure a proper evaluation of zero- and few-shot models. This inherent limitation necessitates strategies that go beyond the available data. The central and most novel contribution of this thesis is the design and systematic evaluation of biologically grounded data augmentation techniques tailored to this low-data problem. We move beyond standard methods and introduce three distinct strategies to create a large corpus of realistic, synthetic training instances.

First, we fine-tune a large generative pLM, as introduced by Nijkamp et al. (2023), to perform kinase-conditional phosphosite generation, effectively learning to "write" new substrate sequences for a given kinase. Second, we use the Kinase Substrate Specificity Atlas, a massive experimental resource developed by Johnson et al. (2023) and Yaron-Barir et al. (2024), as a source of weak supervision to assign plausible kinase labels to thousands of otherwise unlabeled phosphosites. Third, we utilize multiple sequence alignments to generate homologous sequence variants with an established method proposed by Kuru et al. (2022), introducing evolutionarily plausible diversity into the training set.

To test these ideas, this thesis introduces two distinct deep learning architectures, DARKIN-FT and DARKIN-Interact, which use state-of-the-art pLMs and are evaluated on the rigorous DARKIN benchmark dataset (Sunar et al., 2024). We present a comprehensive series of experiments that not only establish the performance of these models but also analyze the impact of different model components and critically assess the effectiveness of each data augmentation strategy. Ultimately, this thesis provides a robust framework for predicting kinase-substrate interactions in data-scarce environments and offers a valuable investigation into the potential and challenges of using biologically informed data augmentation in this domain, providing new insights into the dark kinome.

The main contributions of this thesis can be summarized as follows:

- **Advancement of Zero-Shot Architectures:** We introduce two deep learning models, DARKIN-FT and DARKIN-Interact, that significantly improve upon existing zero-shot prediction framework for the dark kinase-phosphosite association task. By incorporating novel strategies for the integration and end-to-end fine-tuning of modern protein language models, our architectures are demonstrated to substantially outperform the BZSM baseline on the DARKIN benchmark. Furthermore, our models show superior predictive performance on the DARKIN dataset compared to scores from the experimental Kinase Substrate Specificity Atlas (KSSA) (Johnson et al., 2023).

- **Systematic Investigation of Data Augmentation:** We present a comprehensive study of biologically informed data augmentation for the dark-kinase phosphosite prediction task. We design three distinct strategies: (1) conditional phosphosite generation using a fine-tuned generative model; (2) weak supervision by labeling unannotated sites with predictions from the experimental Kinase Substrate Specificity Atlas (KSSA) developed by Johnson et al. (2023) and Yaron-Barir et al. (2024); and (3) data augmentation with homologous sequences.

- **Rigorous Benchmarking and Analysis:** All models and strategies are evaluated on the DARKIN benchmark, a dataset specifically designed for reproducible zero-shot evaluation. We provide a thorough analysis of the results, including extensive ablation studies, which yield valuable insights into the strengths and weaknesses of each approach and provide a clear direction for future work.

The remainder of this thesis is structured as follows:

Chapter 2 provides the necessary background on protein phosphorylation, kinase biology, zero- and few-shot learning, the Transformer models, and the Protein Language Models that underpin this work, and it also reviews related work in the fields of phosphorylation site prediction, kinase-specific prediction, protein sequence generation, and data augmentation in protein modeling.

Chapter 3 details the methodology, including the formal problem definition, the architecture of the proposed DARKIN-FT and DARKIN-Interact models, the datasets used, the evaluation metrics, and the implementation of the data augmentation strategies.

Chapter 4 presents the results of our experiments, including baseline model performance, ablation studies, and a comprehensive evaluation of the data augmentation techniques. The findings are discussed in detail.

Chapter 5 concludes the thesis by summarizing the key findings, discussing the limitations of the current work, and suggesting directions for future research.

Chapter 2

RELATED WORK

2.1 Background Information

2.1.1 Phosphorylation and Kinases

Post-translational modifications (PTMs) are chemical changes that proteins acquire after translation, and they can profoundly influence a protein's activity, localization, and binding partners (Walsh and Jefferis, 2006). The most thoroughly investigated PTM is phosphorylation, a reversible reaction in which a kinase transfers a phosphate group from ATP to a target residue—most often serine, threonine, or tyrosine in eukaryotes (Cohen, 2002). To capture the local sequence context required for reliable prediction, phosphorylation sites ("phosphosites") are typically examined in windows of 15 residues centered on the modified amino acid (D. Wang et al., 2017).

Protein kinases, the enzymes that catalyze this transfer, share a conserved catalytic core of roughly 250–300 residues embedded within otherwise diverse sequences (Manning et al., 2002). By modulating protein activity, stability, localization, and interaction networks, phosphorylation acts as a central switch in cellular signaling. Consequently, irregular kinase activity or misregulated phosphorylation patterns underpin many diseases, including cancer and neurological disorders (Ardito et al., 2017). The human kinome comprises more than 500 kinases, each with its own substrate spectrum, yet assigning the correct kinase to an experimentally detected phosphosite remains challenging. High-throughput phosphoproteomics can identify thousands of modified residues, but for most of them, the upstream kinase is still unknown (Deznabi et al., 2020). This knowledge gap motivates research for (i) predicting which residues in a protein can be phosphorylated and (ii) inferring which kinase is most likely to target a given site.

2.1.2 Zero- and Few-Shot Learning

Zero-shot learning is the capability of a model to identify classes it has never encountered during training by leveraging additional information that connects these unseen classes to those it has previously learned. In essence, the model uses its understanding of known categories to make educated predictions about new, unfamiliar ones. Kinase identification is well suited to a zero-shot learning formulation: many kinases still lack experimentally confirmed substrates, yet they share sequence motifs and structural features with better-characterized family members. Zero-shot learning tackles such situations by using auxiliary information, such as semantic attributes, domain descriptors, or learned embeddings, to recognize classes that never appear in the training set (Xian et al., 2017). By projecting phosphosite and kinase information into a common embedding space, the model can infer associations for kinases that have no direct training examples, relying on relationships encoded in the shared representation rather than on class-specific labels.

Few-shot learning sits between conventional supervised and zero-shot settings. Here, the model is given only a handful of labeled examples for a new class, and its decision boundary must be adjusted using this sparse data. While zero-shot methods depend solely on the auxiliary information, few-shot approaches refine the representation with the limited examples provided.

Both paradigms are crucial for kinase annotation because a large fraction of the human kinome lacks substrate data, leaving little or no pairing information for many enzymes. By learning informative embeddings of phosphosite context and kinase sequences, zero-shot models can generalize to completely unseen kinases, whereas few-shot models can further improve predictions when a small number of examples becomes available.

2.1.3 Transformers

Transformers were introduced as encoder-decoder models for sequence-to-sequence tasks and quickly set new performance records in natural language processing by replacing recurrence with multi-head self-attention and position-wise feed-forward layers (Vaswani et al., 2017). In the encoder, each token embedding is combined with a positional signal and passed through several identical layers, allowing the model to learn both short- and long-range relationships. The decoder generates one token at a time using masked self-attention to look only at earlier outputs, while a cross-attention module links these partial outputs to the encoder's representations.

Variants that keep only the encoder, such as BERT and RoBERTa, introduced by Devlin et al. (2019) and Y. Liu et al. (2019), specialize in learning contextual embeddings for classification and retrieval. Models that keep only the decoder, including the GPT family introduced by Radford and Narasimhan (2018), Radford, Wu, et al. (2019), and Brown et al. (2020), are designed for left-to-right text generation. Architectures that retain both halves, for example, T5, balance these strengths and are widely used for tasks like translation and summarization (Raffel et al., 2020).

The core idea behind all of these models is attention: each token can assign different weights to every other token when building its representation. In the encoder, this self-attention is bidirectional, whereas in the decoder, it is masked to preserve the left-to-right generation order. In an encoder-decoder architecture, cross-attention allows the decoder to attend to the input sequence by forming queries from its current hidden states while drawing keys and values from the encoder's outputs. This enables the decoder to selectively focus on relevant parts of the input, effectively guiding the generation process based on the encoded information.

Training objectives fall into two main categories. Masked language modeling hides a fraction of input tokens and asks an encoder to recover them, promoting bidirectional understanding (Devlin et al., 2019). Autoregressive language modeling trains a decoder to predict the next token given all previous ones, which encourages fluent generation (Radford, Wu, et al., 2019). Because attention can be computed in parallel across positions, Transformers make efficient use of modern hardware, though the quadratic cost of self-attention with respect to sequence length has motivated research into efficient sparse attention variants for very long sequences (Zaheer et al., 2020).

2.1.4 Protein Language Models

Protein sequences can be viewed as sentences written in an alphabet of 20 standard amino acids, with special symbols such as 'X' for unknown residues or '-' for alignment gaps (Elnaggar et al., 2021). Depending on the task, tokenization may operate at the single–residue level or on short sequence motifs, allowing models to generalize to patterns not observed during training. With only minor adjustments, Transformer architectures originally developed for text can process these sequences:

amino acids replace word embeddings, and positional encodings preserve residue order.

Like their natural-language counterparts, protein language models (pLMs) are trained with objectives such as masked language modeling, where the network must recover hidden residues from their context and thus learn biologically meaningful features (Rives et al., 2021). Decoder-only pLMs can be optimized autoregressively to generate new protein sequences that are likely to fold or function. However, purely sequence-based approaches do not explicitly encode three-dimensional structures. Sequence-level pLMs have already advanced secondary-structure prediction, contact mapping, and functional annotation, yet incorporating structural or evolutionary signals remains an active challenge. Recent systems, including AlphaFold by Jumper et al., 2021 and ESMFold by Lin et al., 2023, extend the Transformer framework to predict or exploit 3D conformations, narrowing the gap between sequence embeddings and structural biology.

2.2 Phosphosite-Kinase Association

2.2.1 Deep Learning Methods for Phosphorylation Site Prediction

In the past decade, deep learning has revolutionized sequence-based protein prediction tasks (D. Wang et al., 2017; Luo et al., 2019; Guo et al., 2021; Elnaggar et al., 2021; Rives et al., 2021; Zhou, Yeung, Gravel, et al., 2023; Zhou, Yeung, Soleymani, et al., 2024). Deep learning models can automatically extract complex features from raw sequences without explicit manual feature design. One of the first deep learning frameworks in this domain was MusiteDeep (D. Wang et al., 2017). MusiteDeep implemented a multi-layer convolutional neural network (CNN) to identify phosphorylation sites from sequence windows, and it was capable of both general site prediction and kinase-family-specific predictions. By applying an attention mechanism on the CNN outputs, MusiteDeep aim to focus on important positions in the sequence window. It achieved improved accuracy over earlier shallow models.

A notable advancement came with DeepPhos (Luo et al., 2019). DeepPhos introduced densely connected convolutional blocks (inspired by DenseNet architectures) to capture multi-scale sequence patterns effectively. Instead of a simple stacked CNN, DeepPhos's dense connections allowed information from earlier convolutional layers (detecting small local motifs) to be concatenated with later layers (detecting broader patterns), yielding a rich representation of the sequence surrounding a candidate site. DeepPhos was applied to general S/T/Y site prediction and extended to kinase-specific prediction at various levels such as kinase group, family, or individual kinase. Similarly, DeepPSP combined global and local sequence information using parallel CNN-LSTM modules: one module analyzed a wide window (global protein context) while another focused on the immediate local sequence of the site (Guo et al., 2021). By fusing global and local features, DeepPSP achieved higher accuracy than MusiteDeep, particularly for tyrosine sites.

More recently, transformer-based architectures have been explored for phosphorylation prediction. Transformers can capture long-range dependencies in protein sequences using self-attention. Phosformer, developed by Zhou, Yeung, Gravel, et al. (2023), fine-tunes a pre-trained protein-language model on 11-mer peptide windows and full kinase-domain sequences, coupling them with a multi-head attention module that learns cross-sequence interactions. Trained with hard- and easy-negative sampling plus pLM-compatible augmentations, Phosformer outperforms family-specific CNNs and achieves kinome-wide generalization. Building on the same design, Zhou, Yeung, Soleymani, et al. (2024) introduced Phosformer-ST and extended coverage to serine/threonine kinases profiled by Johnson et al. (2023). A multitask objective (masked-language modeling + kinase-substrate classification) and SHAP-based interpretability reveal that the model captures both substrate-motif cues and evolutionary signals. TransPhos is another transformer-based model that maps protein sequence fragments into high-dimensional representations via a transformer encoder, followed by densely connected CNN layers for the final prediction (X. Wang et al., 2022). By using self-attention, TransPhos can model the influence of residues farther away from the phosphosite, potentially identifying distant modulatory motifs. X. Wang et al. (2022) showed that TransPhos outperformed earlier CNN/RNN models in general phosphosite prediction, suggesting that the attention mechanism adds valuable context that simpler sliding-window models might miss. Our previous work, DARKIN, curates balanced train/val/test splits emphasizing understudied (“dark”) kinases and enforces sequence-identity exclusivity, providing a stringent benchmark for zero-/few-shot prediction of pLMs (Sunar et al., 2024).

Along with these computational methods, large-scale experimental atlases have recently mapped kinase substrate specificities. Johnson et al. (2023) generated more than ten million quantitative measurements for 303 serine/threonine kinases using exhaustive synthetic peptides. Yaron-Barir et al. (2024) took this further by studying how 76 human tyrosine kinases naturally choose their targets.

2.2.2 Kinase-Specific Prediction and Zero-Shot/Few-Shot Learning

In this thesis, as opposed to the methods discussed in the previous problem, we aim to predict the responsible kinase for a given phosphorylation site, which is still an experimental challenge. Traditional kinase-specific predictors required training data for each kinase or kinase family. For example, Scansite developed by Obenauer et al. (2003), a computational tool that predicts short protein sequence motifs likely to be phosphorylated by specific serine/threonine or tyrosine kinases, provided predictions for specific kinase motifs, and MusiteDeep only produced kinase-specific models for families with at least 100 known phosphosites (D. Wang et al., 2017). Many kinases, however, have very few known substrates, making it infeasible to train individual models. To address this data scarcity, researchers have developed methods that generalize across kinases. One strategy is to group kinases by sequence similarity or by classification (e.g., AGC, CAMK, etc.) and train a model per group, assuming kinases in the same group have similar substrate preferences (Xue, Ren, et al., 2008). This was the idea behind the GPS series and others, which improved coverage but still left "orphan" kinases with no data unaddressed.

2.2.3 DeepKinZero: A Zero-Shot Approach

DeepKinZero was the first model that went beyond the supervised learning approaches. DeepKinZero is a zero-shot approach that can suggest a kinase for a phosphosite even if that kinase had no known training examples (Deznabi et al., 2020). DeepKinZero achieves this by learning vector embeddings for kinases and phosphosite sequences in a shared space. During training, it uses kinases with many known substrates to learn the relationship between a phosphosite's sequence features and the corresponding kinase's features. Phosphosite sequences (typically represented as a fixed-length window around the modified residue) are processed by a bidirectional recurrent neural network to capture the site's contextual features, yielding a phosphosite embedding. For kinases, instead of one-hot class labels, DeepKinZero derives kinase embeddings from multiple sources of information: it incorporates the kinase domain protein sequence, the kinase's family hierarchy, and pathway membership to construct a continuous feature representation for each kinase. A compatibility function is then learned between phosphosite embeddings and kinase embeddings. In effect, the model learns what characteristics of a site make it likely to be phosphorylated by a certain type of kinase. At test time, for a novel kinase with no prior substrates (an unseen class), the model can still produce an embedding for that kinase (from its sequence and metadata) and then find the phosphosite embedding that best matches it.

DeepKinZero was shown to significantly improve prediction accuracy for understudied kinases (with zero or few known sites) compared to baseline methods that might default to generic predictions (Deznabi et al., 2020). This zero-shot framework expands the coverage of kinase–substrate predictions across the kinome. In this study, we built our approach on the DeepKinZero framework, replacing the original RNN/LSTM components with Transformer-based protein language models. We also used an updated dataset, DARKIN, from our previous work, which provides up-to-date information and carefully designed splits to enable more reliable zero- and few-shot evaluation (Sunar et al., 2024).

2.3 Deep Learning Models for Protein Sequence Generation

Over the past few years, deep learning has increasingly been applied to de novo protein sequence generation, yielding a variety of model architectures. Early approaches took inspiration from natural language processing (NLP). For example, recurrent neural networks were explored to model protein sequences, and variational autoencoders (VAEs) demonstrated the ability to capture protein family variability. Riesselman et al. (2018) trained one of the first deep generative models on protein sequence alignments, showing that a VAE could model the distribution of natural variants in a protein family and predict mutational effects by sampling novel variants that respected evolutionary constraints (Riesselman et al., 2018). Similarly, Greener et al. (2018) applied a VAE to design new protein sequences with specified attributes, such as metal-binding sites or novel fold topologies. Their VAE-based framework generated sequences conditioned on desired properties (e.g., the presence of a metal-binding motif), and some designs were predicted to adopt stable structures, illustrating the feasibility of VAE-driven protein design. Generative adversarial networks (GANs) have also been introduced: Repecka et al. (2021) developed ProteinGAN, an unconditional GAN that learns protein sequence diversity from a specific enzyme family. Trained on malate dehydrogenase sequences, ProteinGAN was able to produce highly diverse variants, 24% of which were experimentally confirmed to be functional enzymes, including variants with over 100 amino acid differences from any natural sequence. These early studies established that deep generative models could sample the protein sequence space in meaningful ways, often yielding sequences with realistic statistics and sometimes retaining biological function.

A major breakthrough in protein sequence generation came with the adoption of Transformer-based language models. Just as Transformers have excelled in natural language generation, they have proven powerful for modeling protein sequences. Rives et al. (2021) trained a high-capacity Transformer on over 250 million sequences (the ESM-1b model), reporting that "biological structure and function emerge" from scaling up unsupervised learning on protein sequences. In other words, the latent representations from such models captured meaningful protein features like secondary structure and family relationships. While ESM-1b was primarily a masked language model used for representations, its success paved the way for autoregressive sequence generators. Shortly thereafter, researchers introduced GPT-like models for proteins. Madani et al. (2023) developed ProGen, a 1.2-billion-parameter Transformer decoder trained on 280 million sequences. Notably, ProGen was a conditional language model: each training sequence was prepended with control tags (such as the protein's taxonomic origin or functional class), allowing the model to learn sequence generation in context. ProGen demonstrated the ability to generate full-length protein sequences that not only resembled natural proteins but also exhibited functional activity. In one striking result, Madani et al. showed that artificial enzymes generated by ProGen (fine-tuned to lysozyme families) had catalytic activity comparable to natural enzymes despite as low as 30% sequence identity to any known protein. Around the same time, Ferruz et al. (2022) introduced ProtGPT2, an unsupervised protein language model based on GPT-2 architecture. They trained ProtGPT2 on the UniProt database to enable de novo protein sequence generation without any conditioning. Despite the lack of explicit conditioning, ProtGPT2 learned to generate sequences with realistic amino acid compositions and secondary structure propensities. Ferruz and colleagues reported that 88% of ProtGPT2's generated sequences were predicted (by computational disorder predictors) to be well-folded, mostly globular proteins, aligning with natural protein statistics.

Subsequent efforts have pushed model sizes and training data further. ProGen2, introduced by Nijkamp et al. (2023), is a suite of protein language models scaling up to 6.4 billion parameters. By training on over a billion protein sequences (including general and targeted datasets), ProGen2 achieved state-of-the-art perplexity in modeling natural protein sequences and improved the fidelity of generated sequences. The authors showed that increasing model size and training diversity enhanced the model's ability to capture the distribution of evolutionary sequences and even generalize to fitness prediction tasks without additional training. Meanwhile, Meta AI's team scaled their ESM models to 15 billion parameters in the ESM2 family, demonstrating that extremely large protein LMs can produce embeddings rich enough to infer protein structure directly (via the ESMFold system) (Lin et al., 2023). Although ESM2 and ESMFold were aimed at structure prediction, the underpinning language model's capacity to model sequence constraints at atomic detail underscores how far these models have progressed. In summary, recent years have seen a progression from relatively small deep generative models to enormous protein language models. These models can generate protein sequences that conform to the complex statistical patterns of natural proteins, and a subset of the generated sequences have been validated to fold properly or perform biochemical functions.

2.3.1 Conditional Protein Sequence Generation Approaches

An important aspect in protein generation is controlling the properties or attributes of generated sequences. In practical protein design, one often seeks sequences that not only are valid proteins but also have a specific function or structure. Thus, researchers have developed conditional generation methods to guide models toward desired outcomes (Zhu et al., 2024). One straightforward strategy is to incorporate feature tags or prompts with the input sequence. As mentioned, ProGen was trained with auxiliary tags (for instance, specifying a protein's molecular function or organism) to enable controllable output. Madani et al. (2023) demonstrated that such tag-based control, combined with fine-tuning on specific protein families, allowed targeted design: ProGen could focus on generating members of a given enzyme class with a high success rate. In a similar vein, Ferruz et al. (2022) discussed how combining multiple control tags (e.g., a desired cellular compartment, enzyme class, or ligand-binding property) can enable more nuanced design objectives. In their perspective on controllable protein design, they envisioned that dedicated protein Transformers could be fine-tuned or guided to produce sequences with specific functional attributes, analogous to prompting an NLP model to write text in a certain style or topic.

Beyond simple tags, more sophisticated conditional models have been developed for functional control. Kucera et al. (2022) proposed ProteoGAN, a conditional GAN that allows users to specify a protein's function via Gene Ontology (GO) labels. ProteoGAN was trained on millions of protein sequences annotated with GO terms, learning a mapping from functional categories to sequence distribution. Likewise, conditional VAEs have been applied to guide output properties: for example, earlier work on peptide design used a conditional VAE (PepCVAE) to generate antimicrobial peptides by supplying the model with a target activity label during generation (Das et al., 2018). These approaches show that by incorporating task-specific information (class labels, property values, etc.) into the generative process, we can bias the random sequence generation toward sequences that meet design criteria.

The latest protein language models have begun to incorporate multi-faceted control in a single framework. A notable example is ProLLaMA, introduced by Lv et al. (2025), which represents a multitask protein language model capable of both generating sequences and interpreting them. ProLLaMA adapts a large general language model (LLaMA) to protein sequences, training it with an instruction-tuning paradigm on protein tasks. It was supplied with a diverse set of "prompts" and tasks, for instance, prompts to "generate a protein with function X" or "predict the family of this protein." As a result, ProLLaMA can handle controllable sequence generation via natural-language-like prompts or instructions. Experiments showed that it achieved state-of-the-art performance in unconditional sequence generation, and importantly, in a controllable generation benchmark, it could design novel proteins with specified functionalities on demand.

2.4 Data Augmentation Techniques in Protein Modeling

2.4.1 Application to Low-Data and Zero-Shot/Few-Shot Scenarios

Augmentation strategies help by either bootstrapping additional examples or by leveraging foundation models that have been pre-trained on enormous unlabeled datasets (Rives et al., 2021). A popular approach is to pre-train a protein language model on millions of sequences and then fine-tune it on the small task-specific data. Even here, data augmentation can play a role: one can fine-tune in a multitask fashion or with additional unlabeled data via semi-supervised learning. For instance, ProLLaMA was trained on a multitask protein instruction set, effectively augmenting each task with information from others (Lv et al., 2025). In doing so, it achieved strong performance across tasks, including protein generation and classification, despite limited data per task, because it learned a rich shared representation. In low-data regimes, it is also common to integrate multiple augmentation techniques. For example, in a few-shot protein engineering experiment, Hie and Yang (2022) combined a language model (to generate candidate sequences likely to have the desired property) with an active learning loop.

2.4.2 Weak Supervision and Other Augmentation Strategies

When direct labels are limited, weakly supervised learning can generate additional pseudo-labeled data to train on. In weak supervision, one uses noisy or indirect signals to label data automatically. An example in the protein context is to use predictions from a pre-trained model or a heuristic as surrogate labels. A recent approach combined molecular simulation with a transformer-based zero-shot predictor to label mutant sequences with predicted fitness effects, creating a large training set that, in turn, improved a supervised model's accuracy in data-scarce conditions (Deguchi et al., 2025). From general machine learning, techniques like Mixup have been adapted to protein sequences to a limited extent. Mixup typically takes two training examples and creates a synthetic example that is an interpolation of the two features and two labels (Zhang et al., 2018). In their work, Chen et al. (2023) introduced a latent-space Mixup strategy during prompt-based meta-learning, interpolating between molecular structure embeddings and condition prompts to improve sample efficiency and generalization in protein simulation models.

Chapter 3

METHODOLOGY

3.1 Problem Formulation

Let $\mathcal{X}$ denote the set of *phosphosite windows*, where each $x \in \mathcal{X}$ is a 15-residue amino-acid sequence whose central residue is experimentally verified to be phosphorylated. Let $\mathcal{Y}$ be the catalog of human protein kinases. Because one site can be catalyzed by several kinases, the problem is cast as *multilabel classification*: for any x, we aim to recover the subset $\mathcal{Y}(x) \subseteq \mathcal{Y}$ of its true associated kinases.

Following the zero-shot learning protocol of Xian et al. (2017), we split the kinases into *seen* classes $\mathcal{Y}_{\text{tr}}$ and *unseen* classes $\mathcal{Y}_{\text{te}}$, with $\mathcal{Y}_{\text{tr}} \cap \mathcal{Y}_{\text{te}} = \varnothing$. During training, labels are available only for $\mathcal{Y}_{\text{tr}}$, yet the trained model must score *all* kinases at test time. In the few-shot regime, each unseen kinase contributes at most $k \in \{1, \ldots, 5\}$ labeled sites; the zero-shot setting corresponds to $k = 0$.

3.2 Modelling Framework

In this thesis, we tried two different approaches to model this problem:

(A) *DARKIN-FT* is a compatibility-based multiclass model which extends the models introduced in our previous work (Sunar et al., 2024). The model updates the pLM that generates the phosphosite embeddings in an end-to-end manner. It computes a score for every kinase in a forward pass (see 3.3.1).

(B) *DARKIN-Interact* is a sequence-pair binary classifier model that predicts

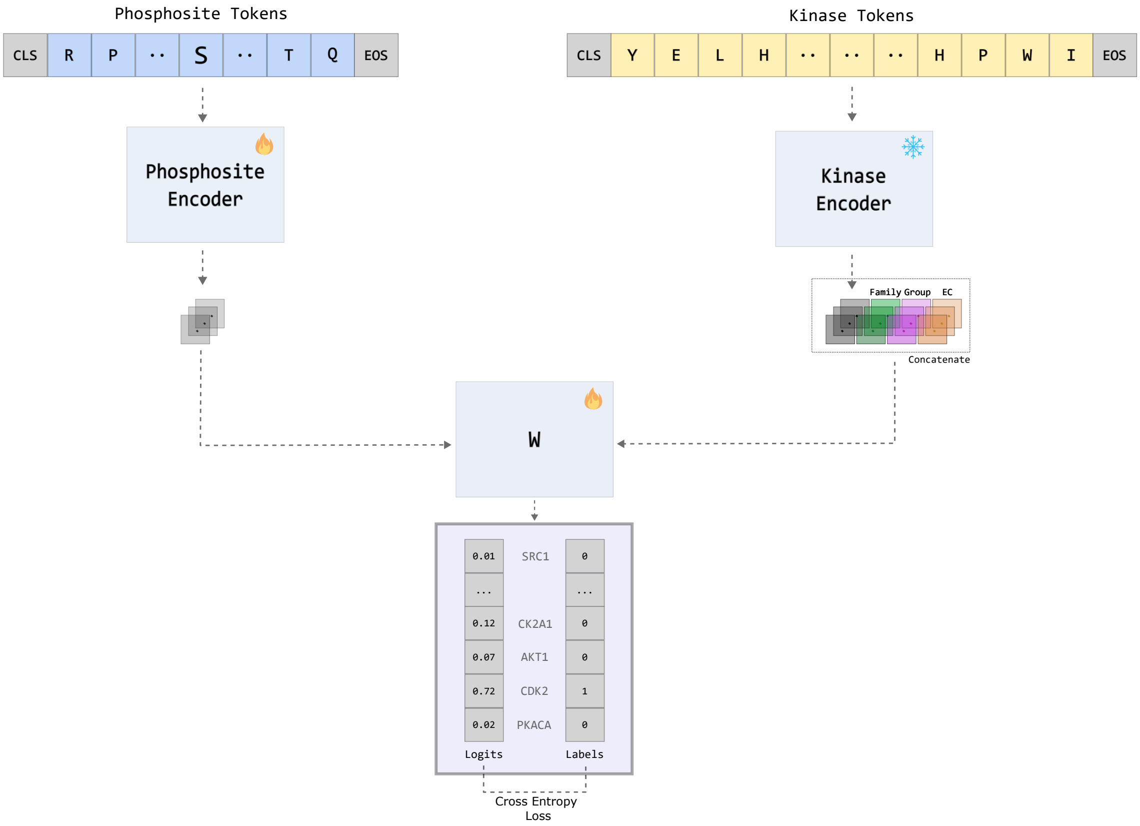

Figure 3.1 Schematic overview of DARKIN-FT, an extension of the BZSM baseline of the DARKIN benchmark in which the phosphosite encoder is fine-tuned end-to-end rather than kept frozen.

whether an individual (site, kinase) pair is compatible (see 3.6.1).

Both models are evaluated in the same data splits and evaluation metrics.

3.2.1 DARKIN-FT: Compatibility-Based Multiclass Model

DARKIN-FT framework, illustrated in Figure 3.1, is a modified version of DeepKinZero, developed by Deznabi et al. (2020) and the BZSM baseline of the DARKIN benchmark. While DeepKinZero employs RNN/LSTM-based architectures for phosphosite encoding and BZSM uses frozen phosphosite embeddings, DARKIN-FT utilizes a transformer-based encoder and updates phosphosite embeddings end-to-end during training.

Representations. A protein language model embeds a phosphosite window as $\theta(x) \in \mathbb{R}^d$. Each kinase y is represented by

$$\phi(y) = \left[\phi_{\text{seq}}(y) \parallel f_{\text{fam}}(y) \parallel f_{\text{grp}}(y) \parallel f_{\text{ec}}(y)\right] \in \mathbb{R}^M,$$

where $\phi_{\text{seq}}(y) \in \mathbb{R}^m$ is the pLM embedding of its catalytic domain and f_{fam}, f_{grp}, and f_{ec} represent the one-hot encodings of family, group, and EC number, respectively.

The dimensionality of $\phi_{\text{seq}}(y)$ depends on the chosen protein language model. For instance, using ESM1B yields a 1280-dimensional embedding for the catalytic domain. The one-hot encodings for kinase family, group, and EC number together contribute 144 additional dimensions, resulting in a total kinase embedding size of 1424 in this setting.

Compatibility functions. Two forms are considered:

i. **Bilinear form with bias**

$$s_{\text{bil}}(x,y) = \left[\theta(x)^\top \; 1\right] W \left[\phi(y)^\top \; 1\right]^\top, \qquad W \in \mathbb{R}^{(d+1)\times(M+1)} \tag{3.1}$$

Here, W is a learnable compatibility matrix that captures the interaction between phosphosite and kinase representations. This formulation is a variant of Deznabi et al. (2020)

ii. **Dot product**

$$s_{\text{dot}}(x,y) = \theta(x)^\top \phi(y). \tag{3.2}$$

Learning objective. For a phosphosite x_i with a multi-hot label vector $\mathbf{y}_i \in \{0,1\}^{|\mathcal{Y}_{\text{tr}}|}$.

The model normalizes the scores over all kinases with a softmax and minimizes the cross-entropy loss.

$$\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{N} \sum_{y \in \mathcal{Y}_{\text{tr}}} y_{i,y} \log \frac{\exp s_\Theta(x_i,y)}{\sum_{y'} \exp s_\Theta(x_i,y')}, \tag{3.3}$$

In the equation above, $\Theta = \mathit{bil}$ for Eq. (3.1) and $\Theta = \mathit{dot}$ for Eq. (3.2). Although each site may have multiple correct kinases, Eq. (3.3) is applicable during training since each (phosphosite, kinase) pair is treated as a separate training instance; that is, multilabel annotations are converted into multiple single-label examples. At test time, we do not split multilabel sites; instead, we retain the full label set and evaluate the raw prediction scores $s_\Theta(x,y)$ across all candidate kinases.

3.2.2 DARKIN-Interact: Sequence-Pair Binary Classifier

DARKIN-Interact is an alternative method to DARKIN-FT, allowing for the examination of the attention mechanism's power by reformulating the problem as a binary classification task. In Section 3.6.1, we explain this approach.

Joint representation. The amino-acid sequence of kinase y is concatenated with the phosphosite window x, separated by a `[EOS]` token, and fed to ESM2 (Lin et al., 2023). The CLS token yields

$$z(x,y) = \mathrm{CLS}\Big(\mathrm{ESM2}(\texttt{[CLS]} \parallel x \parallel \texttt{[EOS]} \parallel y \parallel \texttt{[EOS]})\Big) \in \mathbb{R}^h.$$

Scoring and loss. A linear head produces the logit $s_{\text{pair}}(x,y) = w^\top z(x,y) + b.$ of the `[CLS]` embedding. Given a positive pair (x, y^+), n negative kinases $\{y_j^-\}_{j=1}^n$ are sampled and the binary cross-entropy

$$\mathcal{L}_{\text{BCE}} = -\sum_{(x,y^+)} \log \sigma\Big(s_{\text{pair}}(x,y^+)\Big) - \sum_{j=1}^{n} \log\Big(1 - \sigma(s_{\text{pair}}(x, y_j^-))\Big) \tag{3.4}$$

is minimized. We do not use explicit class balancing, but for each positive pair, we sample n negatives randomly, which introduces variability and helps mitigate bias in the zero-shot setting. During the evaluation, a given site is paired with *every* kinase, the logits are ranked, and multilabel metrics are computed exactly as for DARKIN-FT (see Figure 3.10).

3.2.3 Data Augmentation and Regularization

To address the severe scarcity of labeled examples, we augment the training set using three complementary strategies: (i) kinase-conditioned phosphosite generation, (ii) homology-based label transfer, and (iii) weak supervision through labeling of unlabeled data using an external model. For either model, the objective is minimizing

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{core}}(D_{\text{tr}}) + \alpha\,\mathcal{L}_{\text{core}}(\tilde{D}) + \lambda\,\|\Theta\|_2^2, \tag{3.5}$$

where $\mathcal{L}_{\text{core}}$ is $\mathcal{L}_{\text{CE}}$ for DARKIN-FT and $\mathcal{L}_{\text{BCE}}$ for DARKIN-Interact; $\alpha \in [0,1]$ balances real and synthetic data, and λ is the weight-decay coefficient.

3.3 Dataset Description and Preprocessing

3.3.1 Overview of the DARKIN Benchmark

A reliable evaluation of learning algorithms—especially in zero-shot and few-shot scenarios—depends on a dataset that aligns with the characteristics of the task. In the context of phosphosite–kinase prediction, this entails avoiding information leakage, stratifying splits properly based on their class/group memberships, and maintaining a sufficient number of positive instances per class to ensure reliable evaluation of the test performance.

DARKIN (short for *DARk KINase*) benchmark dataset was introduced in our earlier work to meet these challenges (Sunar et al., 2024). It integrates kinase and phosphosite annotations from publicly available databases and partitions the dataset into train, validation and test splits based on the problem characteristics avoiding leakage. them into These predefined partitions are designed to support both conventional zero-shot learning (ZSL) and the more demanding generalized zero-shot learning (GZSL) framework.

The DARKIN set includes the human kinase list and their associated phosphosites. The starting kinase list is the 557 human protein kinases that contain at least one kinase domain, as reported by Moret et al. (2020). The following filtering steps are applied:

- **Removal of isoforms:** Protein isoforms are alternative splice variants derived from the same gene. Because isoforms may differ in non-catalytic regions or include truncations that complicate sequence-level comparisons, we retained only the canonical isoform recorded in UniProt.
- **Removal of fusion proteins:** The fusion kinases contain multiple kinase domains. We removed them from the dataset.
- **Retention of canonical sequences:** UniProt designates one representative protein per gene, the canonical sequence, which typically corresponds to the most studied or functionally relevant form. Using these canonical sequences provides consistency across analyses.

For each kinase that passed these filters, we extracted the following features:

- **Kinase domain sequence:** Rather than using the full-length protein, we focused on the domain region responsible for catalytic activity, typically spanning 250-300 amino acids. These domain-level sequences serve as the direct input to protein language models (pLMs).

- **Kinase group:** We used the 11 major kinase groups defined by Manning et al. (2002), which cluster kinases based on domain sequence similarity. These groups reflect shared evolutionary relationships and functional similarity. Missing group labels were inferred from sequence similarity, while any remaining unassigned kinases were placed in a generic "Other2" group. During model training, group labels were one-hot encoded.
- **Kinase family:** A more granular classification into 129 families, also based on Manning et al. (2002), capturing finer distinctions in substrate specificity and regulation. As with groups, missing family labels were inferred, and a fallback category "otherFamily" was assigned to kinases that could not be classified.
- **Enzyme Commission (EC) number:** Each kinase was assigned a four-level EC code that reflects its catalytic function (e.g., EC 2.7.11.* for serine/threonine kinases). Unknown EC codes were imputed via nearest-neighbor transfer, and the resulting EC numbers were encoded as binary vectors.

Phosphosite data were obtained from the PhosphoSitePlus database as of May 2023 (Hornbeck et al., 2012). Each phosphosite is represented as a 15-residue peptide sequence centered on the modified residue: Serine (S), Threonine (T), or Tyrosine (Y). If a site occurs within seven residues of the protein's N- or C-terminus, "_" is used as padding to maintain a consistent window length. After filtering out entries involving non-human kinases, the dataset comprises approximately 14,000 unique phosphosites and around 25,000 kinase–phosphosite interaction pairs. Notably, about 20% of phosphosites are annotated with multiple kinases, making them multilabel examples and highlighting the complexity of kinase-substrate relationships.

3.3.2 Construction of Zero-Shot Learning Splits

In the DARKIN dataset, kinases with abundant phosphosite annotations are assigned to the validation and test splits, while understudied "dark" kinases are reserved for the training split. This design choice ensures that each held-out class (i.e., each kinase in the validation or test set) has a sufficient number of positive examples to support the computation of reliable performance metrics, such as average precision and top-k accuracy. Consequently, the evaluation is both reliable and meaningful.

Beyond this dark/light kinase separation, the dataset is further partitioned by applying a set of constraints to a random split procedure. This enables the generation of multiple reproducible dataset variants. Unless otherwise stated, all experiments in this thesis are conducted using Split 1. Each split assigns 80% of the kinase–phosphosite interactions to training, and 10% each to validation and test while adhering to the following constraints:

- **Minimum support:** Every kinase in the validation and test splits must be associated with at least ten annotated phosphosites, ensuring the reliability of per-class evaluation metrics.
- **Stratified groups:** The distribution of the 11 high-level kinase groups is preserved across training, validation, and test sets, maintaining representational balance.
- **Sequence identity exclusivity:** Kinases that share 90% or more sequence identity within their kinase domains are assigned to the same split. This prevents knowledge leakage through highly similar sequences and avoids inflating performance via trivial transfer.

For DARKIN Split 1, the final dataset comprises 8,560 training, 1,485 validation, and 1,415 test examples. Kinases with fewer than ten associated phosphosites are always placed in the training set, whereas those with richer annotation are preferentially used in validation and testing. This approach increases the difficulty of generalization, placing a greater burden on the model to perform well on truly unseen and well-characterized classes while minimizing overfitting to rare patterns.

3.4 Evaluation Metrics

To evaluate the proposed methods and interpret their outcomes rigorously, we rely on a small set of core metrics complemented by task-specific variants tailored to the DARKIN dataset.

Macro Average Precision (Macro-AP): Macro-AP is the principal metric. For each kinase class, we rank all test phosphosites, compute the Average Precision (AP) for that class, and then take the unweighted mean across classes. DARKIN places *dark* kinases (classes with very few examples) in the training set and reserves *light* kinases for testing (see Section 3.3 for details). Macro-AP offers a balanced view: every kinase, regardless of how many sites it phosphorylates, contributes equally to the final score, preventing well-represented kinases from obscuring performance on rarer ones.

Top-*k* Accuracy: To complement kinase-centric Macro-AP with a phosphosite-centric perspective, we report Top-k Accuracy for $k \in \{1, 3, 5\}$. A prediction is counted as correct if any of the true kinase labels of the site appear within the top-k positions. This metric naturally accommodates the multilabel nature of the task and highlights how far down the ranked list a user must look before encountering a correct kinase.

Phosphosite Average Precision (Phosphosite AP): Phosphosite AP is computed analogously to Macro-AP but in the opposite direction: for each phosphosite, we rank all candidate kinases, compute the AP, and average the resulting APs. This metric captures how accurately the model ranks kinases for a given phosphosite, aligning closely with the retrieval experience of an end user. However, we do not report Phosphosite AP as the primary evaluation metric because the kinase label space in the test set is relatively small. In such settings, site-level AP can be disproportionately affected by the number of true labels per phosphosite or the frequency of certain kinases, which can lead to misleading impressions of model performance.

Attribute-level Metrics: Kinases can be grouped by broader functional or evolutionary attributes, and these coarse labels often have practical significance. We, therefore, aggregate the model's per-site predictions at three levels, *family*, *group*, and *fine-grained cluster* (abbreviated as *F.grain*), and report both AP and Accuracy at each level. After aggregation, the same ranking-based formulas used for Macro-AP and Top-k Accuracy are applied to the attribute labels.

Fine-grained clusters extend the family and group hierarchies with an additional layer derived from phylogenetic proximity. Using the kinase phylogenetic tree published by KinBase (2024), we convert branch lengths into pairwise similarity scores, normalize them, and form clusters of kinases that share a high evolutionary similarity. These clusters typically contain few kinases. Thus, only very similar kinases are grouped together. Kinases may not be very specific in certain cases. Very similar kinases can indeed associate with the same phosphosite, and the available experimental data might not always show this specificity accurately due to a lack of experimentation on both kinases' association with the same site.

The aggregated AP and accuracy scores provide users with an alternative perspective: a model may misidentify the exact kinase yet still recover the correct family, group, or cluster, offering biologically meaningful guidance even when fine-grained predictions are uncertain.

3.5 Data Augmentation Strategies

DARKIN addresses phosphosite–kinase prediction in a zero-shot learning scenario. To obtain a realistic evaluation, the dataset holds out well-characterized “light” kinases for testing while training on dark kinases that possess far fewer annotated phosphosites. Light kinases are fewer in number but have more annotated sites, whereas dark kinases are more numerous but less well studied. Although this split highlights the model’s ability to generalize to unseen classes, it amplifies class imbalance and reduces the diversity of training examples. In this section, we investigate several augmentation strategies to enrich the training set, broaden class coverage, and ultimately improve the zero-shot performance.

Data augmentation is a widely adopted practice in machine learning because it increases sample diversity, improves generalization, and mitigates overfitting (Perez and J. Wang, 2017; C. N. Vasconcelos and B. N. Vasconcelos, 2017). In computer vision tasks, for example, an image may be rotated, blurred, or color-shifted; in natural-language processing, augmentation often involves deleting words or substituting them with synonyms (Wei and K. Zou, 2019). Crucially, augmented instances must preserve the true structure of the data so that the model is not misled. This is even more critical for protein sequences. Arbitrarily deleting or swapping amino acids can disrupt functional motifs and produce incorrect labels. In the DARKIN dataset, every phosphosite sequence is exactly fifteen residues long, with the phosphosite residue at the middle position. Preserving this structural prior is critical for downstream phosphosite–kinase association.

3.5.1 Kinase–Conditional Phosphosite Generation

Generative protein language models aim to capture the grammatical and functional regularities of natural sequences by training on extensive corpora. Recent work has adapted many of the training strategies that propelled progress in natural language processing, yet protein models remain an active area of development with notable limitations (Rives et al., 2021; Elnaggar et al., 2021; Hayes et al., 2025; Lv et al., 2025). ProGen and its successor, ProGen2, belong to the family of autoregressive Transformer decoders for protein design. Both models share a similar architectural blueprint, but ProGen2 was trained on substantially larger datasets and is available in several parameter sizes. Figure 3.2 illustrates the inference procedure of ProGen2. Throughout this study, we employ the 650M parameter *ProGen2–Base* variant, whose increased capacity has been linked to stronger sequence modeling performance

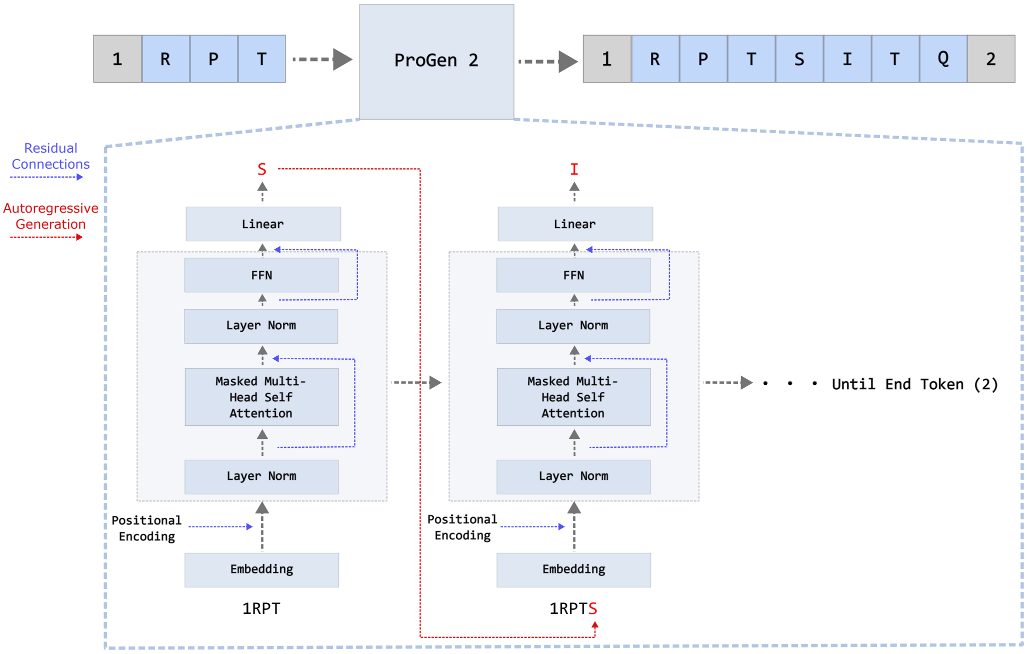

Figure 3.2 Illustration of ProGen2 inference: given a kinase-conditioned prompt, the model autoregressively generates the remaining amino-acid sequence.

(Nijkamp et al., 2023).

In ProGen2, the special token 1 signals the beginning of the sequence, and 2 marks the end of the sequence. Supplying 1 initiates left-to-right generation until the model decides to emit 2 or a user-specified length limit is reached. Conditioning is straightforward: any sequence placed before 1 provides additional context. Decoding quality and diversity can be tuned by switching from greedy selection to stochastic schemes such as temperature scaling, top-k, or nucleus (top-p) sampling.

Fine-tuning protocol. To generate synthetic phosphosites, we fine-tuned ProGen2 on the training split of the DARKIN dataset. Each training sample combined a 15-residue phosphosite with its cognate kinase in two complementary orientations:

$$\big[\text{kinase tokens}\big]\ \texttt{1}\ \big[\text{phosphosite tokens}\big]\ \texttt{2}$$

$$\big[\text{kinase tokens}\big]\ \texttt{2}\ \big[\text{reverse phosphosite tokens}\big]\ \texttt{1}$$

During fine-tuning, the loss was computed only on the phosphosite tokens together with 1 and 2; kinase residues served purely as non-trainable contexts. Therefore, the model learned that (i) the kinase information precedes the first special token, (ii) exactly fifteen residues should appear between 1 and 2, and (iii) the central position must carry the target phosphorylatable amino acid. The reverse orientation exposes the same constraints in the opposite direction, encouraging the model to capture both N-to-C and C-to-N dependencies within the site. We refer to the

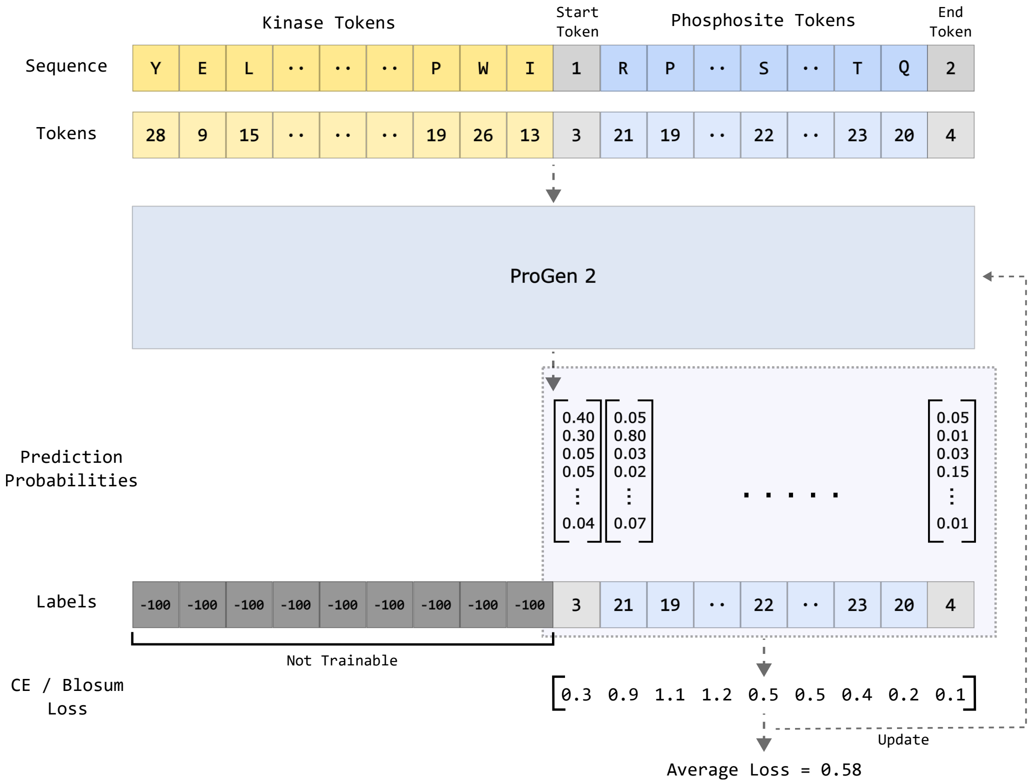

Figure 3.3 Fine-tuning workflow of *ProGen2–Phospho* for kinase-conditioned phosphosite generation, supporting either cross-entropy or BLOSUM loss.

resulting model as *ProGen2–Phospho*. The complete fine-tuning procedure of *ProGen2–Phospho* is illustrated in Figure 3.3.

LoRA is a parameter-efficient fine-tuning technique (PEFT), which keeps all original weights frozen while injecting trainable low-rank adapters into linear layers (Hu et al., 2022). The rank and scaling factor α were tuned to balance expressiveness against overfitting and compute cost. Higher ranks increase the number of trainable parameters but can diminish generalization if chosen unwisely.

Because *ProGen2–Phospho* is evaluated by generation rather than classification, we removed from the training split any phosphosite that also appeared in validation or test sets. This precaution prevents the model from memorizing specific sites and inflating downstream performance. Using DARKIN splits for *ProGen2–Phospho* also eliminates leakage when synthetic sequences are later merged with DARKIN for data augmentation.

We first trained *ProGen2–Phospho* with the token-level autoregressive cross-entropy (CE) loss

$$\mathcal{L}_{\text{CE}} = -\frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \log p_\theta\left(x_i \,\middle|\, x_{<i}\right), \tag{3.6}$$

where x_i is the ground-truth token at position i, p_θ is the model's predictive distribution, and $\mathcal{M}$ is the set of positions on which the loss is evaluated (the 15-mer phosphosite window and the two delimiter tokens "1","2"). Although (3.6) is widely used in language modeling, it penalizes based on amino acid mismatch, even though some substations (e.g. S↔T) are known to be structurally and functionally tolerated (Betts and Russell, 2003).

BLOSUM smoothed loss: To encourage the model to (i) respect biochemical similarity encoded in BLOSUM62, introduced by S. Henikoff and J. G. Henikoff (1992), and (ii) focus on the central residue while remaining strong to mild peripheral variations, we replace the one-hot target in (3.6) with a smoothed target distribution $q_i(\cdot)$ built from BLOSUM scores. The resulting loss is

$$\mathcal{L}_{\text{BLOSUM}} = -\frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \sum_{t \in \mathcal{V}} q_i(t) \log p_\theta\Big(t \,\Big|\, x_{<i}\Big), \tag{3.7}$$

where $\mathcal{V}$ is the amino-acid vocabulary and q_i depends on (a) whether position i is the central residue in the 15-mer window and (b) the true amino-acid $a_i^\star$.

(i) Non-central amino-acids: For every amino-acid position $i \neq i_{\text{mid}}$ we compute

$$q_i(t) \;=\; \operatorname{softmax}\Big(\alpha\, B[a_i^\star, t]\Big),$$

where $B[\cdot,\cdot]$ is the BLOSUM62 score matrix shifted to non-negative values and $\alpha > 0$ controls how close q_i is to a one-hot vector ($\alpha \to \infty$ recovers CE).

(ii) Central phosphosite residue: Let $r \in \{\mathrm{S}, \mathrm{T}, \mathrm{Y}\}$ be the true phospho-acceptor. We assign three scalar weights before normalization:

$$q_{i_{\text{mid}}}(t) \propto \begin{cases} \beta, & t = r, \\ \beta, & r \in \{\mathrm{S},\mathrm{T}\}, t \in \{\mathrm{S},\mathrm{T}\} \setminus \{r\}, \\ \gamma, & r \in \{\mathrm{S},\mathrm{T}\}, t = \mathrm{Y} \quad \text{or} \quad r = \mathrm{Y}, t \in \{\mathrm{S},\mathrm{T}\}, \\ \delta, & \text{otherwise}, \end{cases}$$

with $\beta > \gamma > \delta > 0$. After normalization, the central target distribution rewards S↔T confusions mildly, penalizes any S/T↔Y confusion more strongly and discourages all other substitutions.

(iii) Delimiter tokens: For the delimiter tokens "1" and "2" we keep the original one-hot CE targets, ensuring the model learns their exact identities.

Unless stated otherwise, we keep the implementation defaults α=1.0, β=1.0, γ=0.4, and $\delta = 0.01$. A brief grid search on the validation set showed that raising α pushes the objective back toward vanilla cross-entropy while lowering it over-smooths the targets; the chosen value, therefore, preserves informative gradients without ignoring biochemical similarity. Setting γ to 40% of β and δ two orders of magnitude smaller balances the cost of S↔T confusions (about half that of an S/T↔Y error) and makes all other substitutions comparatively negligible.

Sequence generation and filtering: After training *ProGen2–Phospho*, we used it to 100 candidate phosphosite sequences for each kinase across all DARKIN dataset splits, using the prompt format

$$\Big[\text{kinase\_tokens}\Big]\ \texttt{1}$$

The generated sequences underwent a multi-step filtering process to ensure biological plausibility and consistency with the modeling objectives. Sequences were excluded if they did not have a length of exactly 15 amino acids, lacked a phosphorylatable residue (S, T, or Y) at the central position, or contained non-standard amino acid symbols. Additionally, a binary classifier based on the ESM1B model fine-tuned as part of our TÜBİTAK project (122E500) was employed to further assess sequence validity. This classifier was trained on a dataset comprising known 15-length phosphosites and randomly sampled non-phosphorylated sequences using a binary classification objective. The model achieved a test accuracy of 94% and an F1-score of 94%. We retained only those sequences for which the classifier assigned a positive-class probability greater than 90%. Figure 3.4 illustrates the distribution of predicted positive-class probabilities for *ProGen2–Phospho*-generated sequences, as scored by the fine-tuned classifier trained with a standard cross-entropy loss. Following this filtering process, the remaining synthetic phosphosites were selectively integrated into the DARKIN dataset in controlled amounts to evaluate their impact on downstream performance. An illustration of this process is shown in Figure 3.5.

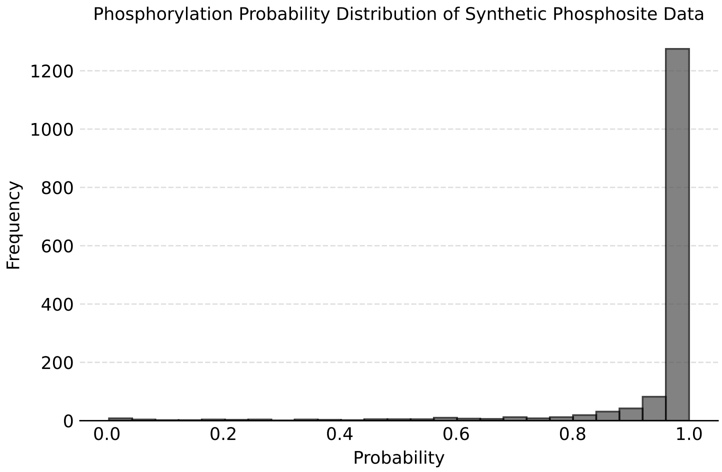

Figure 3.4 Distribution of positive-class probabilities assigned by an ESM1B phosphorylation predictor to synthetic phosphosites produced by *ProGen2–Phospho* (cross-entropy loss).

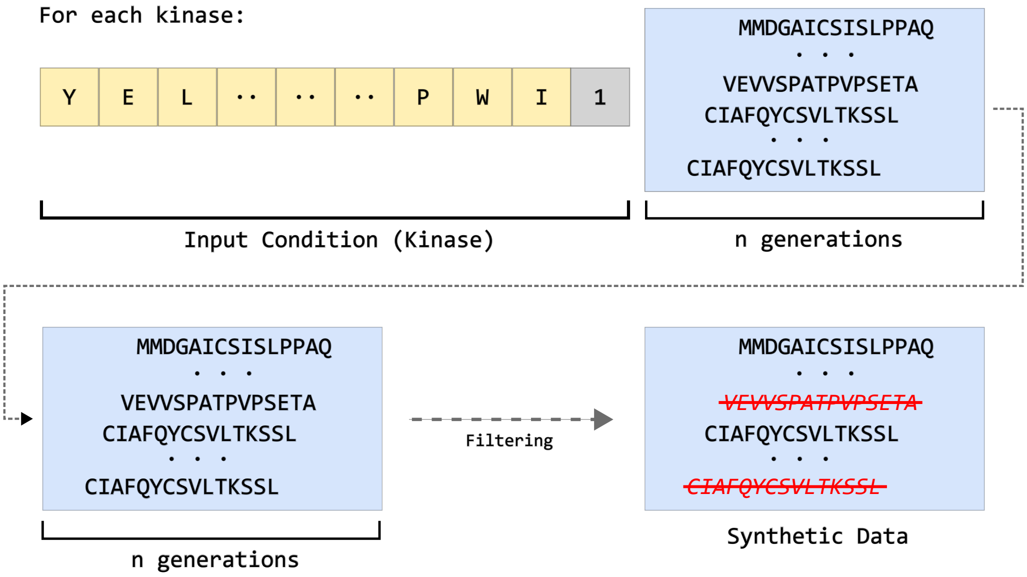

Figure 3.5 Pipeline for generating and filtering synthetic phosphosite–kinase pairs with *ProGen2–Phospho* prior to downstream training.

3.5.2 Leveraging Kinase Atlas Predictions as Synthetic Training Data

Johnson et al. (2023) and Yaron-Barir et al. (2024) systematically profiled substrate specificities for 303 human Ser/Thr kinases and 78 human Tyr kinases, respectively, by screening each enzyme against millions of synthetic 15-mer peptides. The resulting position-weighted motif score matrices permit high-resolution predictions: for any query phosphosite, the interface returns a percentile score (0–1) that ranks the likelihood of phosphorylation across the surveyed kinome. Percentiles correspond to $-\log P$ values, so larger scores indicate stronger model support. For clarity, throughout this thesis, we refer to their method as the Kinase Substrate Specificity Atlas (KSSA).

We exploited this experimental resource to label previously unlabeled phosphosites collected from PhosphoSitePlus (Hornbeck et al., 2012). Each sequence was submitted to the KSSA interface, and the best two kinases whose scores exceeded the 0.99-percentile threshold were retained. These kinase–phosphosite pairs were then added to the DARKIN training split as synthetic examples. This labeling procedure is depicted in Figure 3.7. The model generates its predictions based on either a predefined kinase set provided as input or, alternatively, by considering all valid kinases available to it. Because the KSSA can only consider kinases profiled in the original experiments, its accessible kinase set does not fully overlap with the DARKIN kinases. In some cases, KSSA includes kinases that are not in DARKIN due to filtering steps applied during DARKIN construction (see Section 3.3.1). Figure 3.6 illustrates the difference between the number of kinases in DARKIN and those applicable within the KSSA for the training, validation, and test sets, highlighting the distinct experimental designs of the two studies. Johnson et al. (2023) focus exclusively on Ser/Thr kinases, whereas Yaron-Barir et al. (2024) examine Tyr kinases; therefore, the model can generate predictions separately for these two kinase groups.

Before applying augmentation, we collected all DARKIN kinases that were compatible with the KSSA, enabling the construction of alternative configurations comprising (i) training kinases only, (ii) training and validation kinases, and (iii) the union of all three. This design allows us to investigate the impact of synthetic data originating from distinct class categories. Each phosphosite was restricted to its top two predicted kinases to reduce the risk of false positives while preserving kinase diversity. These synthetic examples were subsequently integrated into the DARKIN dataset and used to train the DARKIN-FT and DARKIN-Interact models. Although including predictions involving validation and test kinases introduces

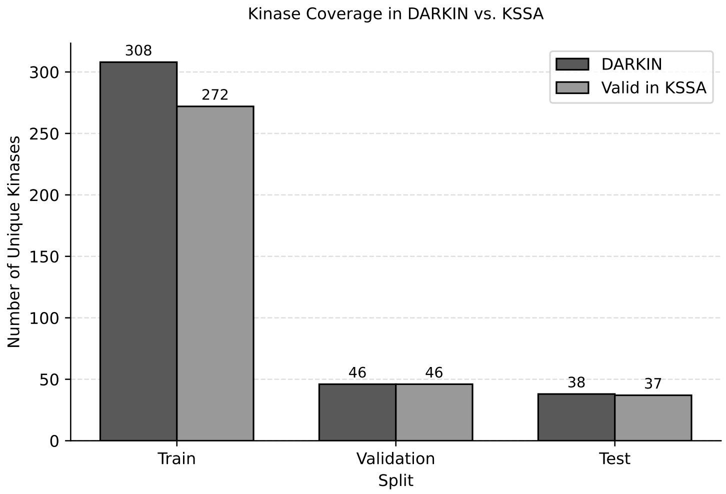

Figure 3.6 Coverage of DARKIN kinases by the KSSA.

supervised information for otherwise unseen classes, no explicit kinase–phosphosite pairs were presented to the prediction interface. This strategy prevents information leakage while offering a rigorous test of the model's generalization capabilities.

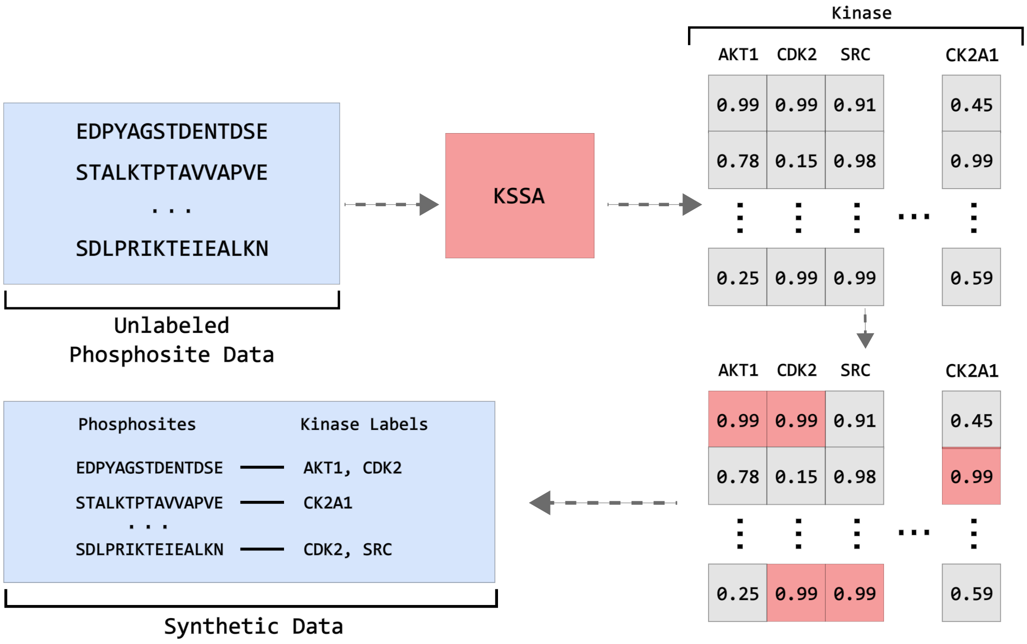

Figure 3.7 Procedure for weakly labeling unannotated phosphosite–kinase pairs using the Kinase Substrate Specificity Atlas (KSSA).

3.5.3 Using Homologous Sequences for Data Augmentation

Homologous sequences refer to evolutionarily related protein sequences originating from a common ancestor and often retaining similar functions. In this section, we aimed to augment the training dataset by incorporating homologous variants of phosphosite sequences to enrich the data with biologically plausible evolutionary variants, thereby increasing the diversity and size of the training set. We obtained the homologous sequences from the study by Kuru et al. (2022). For multiple sequence alignment construction, PHACT identifies homologous sequences using PSI-BLAST against the UniProtKB/Swiss-Prot reference proteomes (The UniProt Consortium, 2021). The retrieved sequences are aligned with MAFFT FFTNS developed by Katoh and Standley (2013), and the alignments are trimmed using trimAl to remove poorly aligned regions.

In this application, we leveraged the homologous sequences of human proteins. Consequently, we excluded isoforms and sequences from non-human organisms from the reference sequences list. For each 15-mer phosphosite sequence in the training set, we located the corresponding aligned position in the full-length protein alignments. We then extracted up to five most similar aligned 15-mer windows from homologous sequences, using global alignment scores computed via the `globalxx` method from Biopython (Cock et al., 2009). These homologous sequences were treated as augmented phosphosite candidates. Each synthetic sequence retained the label of its original kinase association, assuming functional equivalence among close homologs. This approach assumes that evolutionarily close substitutions maintain similar biochemical roles. Through this method, we incorporated approximately 40,000 synthetic samples into the dataset. This homology-driven augmentation strategy not only expanded the data volume but also introduced evolutionary diversity aligned with biological relevance.

3.6 Reassessing and Extending DeepKinZero

3.6.1 Reformulating Kinase–Phosphosite Association as Binary Classification

Transformer encoder protein language models leverage self–attention to weigh nonlocal relationships within an input sequence. Inspired by the sentence-pair architecture of BERT and its Next Sentence Prediction (NSP) task introduced by Devlin et al. (2019), we rethink the kinase–phosphosite association problem as a binary classification: given a phosphosite and a kinase sequence, the model predicts whether the kinase can phosphorylate the site. We call this method *DARKIN-Interact.*

All experiments begin from a pre-trained ESM2 backbone, chosen for its family

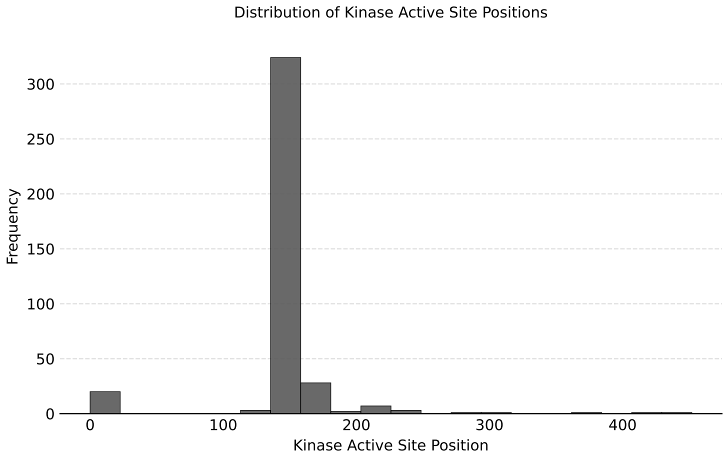

Figure 3.8 Histogram of active-site residue indices within kinase domain sequences in the DARKIN dataset.

of models at various parameter sizes, which allows us to probe the influence of representational capacity. Inputs are constructed as

$$\texttt{<CLS>}\ \underbrace{\text{phosphosite}}_{\text{15 aa}}\ \texttt{<EOS>}\ \underbrace{\text{kinase}_{[:200]}}_{\leq 200\text{ aa}}\ \texttt{<EOS>},$$

where the kinase sequence is truncated to 200 residues to conserve GPU memory while retaining most active-site motifs (see Figure 3.8). The output embedding of `<CLS>` is passed through a two-layer classifier to predict the binary label.

Training Phase. Because DARKIN contains only positive kinase–phosphosite pairs, we augment each positive instance with n randomly chosen negative kinases. Negative sampling is refreshed every epoch, exposing the model to diverse counterexamples and encouraging it to learn both binding and non-binding patterns. Training minimizes binary cross-entropy over the positive and negative pairs. Figure 3.9 illustrates the training framework of DARKIN-Interact.

We investigate two parameter-efficient strategies: (i) *partial fine-tuning*, where only the top l Transformer layers and the classification head are updated while earlier layers remain frozen and (ii) *LoRA* adapters proposed by Hu et al. (2022) inserted into every linear projection, which keeps the original weights intact and trains only low-rank matrices. Varying l or the LoRA rank allows us to balance task-specific adaptation against overfitting and computational cost.

Evaluation Phase. For evaluation, each phosphosite in the validation or test set is paired with every kinase present in that split, and the classifier's scores are ranked from highest to lowest. Macro Average Precision (AP) and top-k accuracy are then

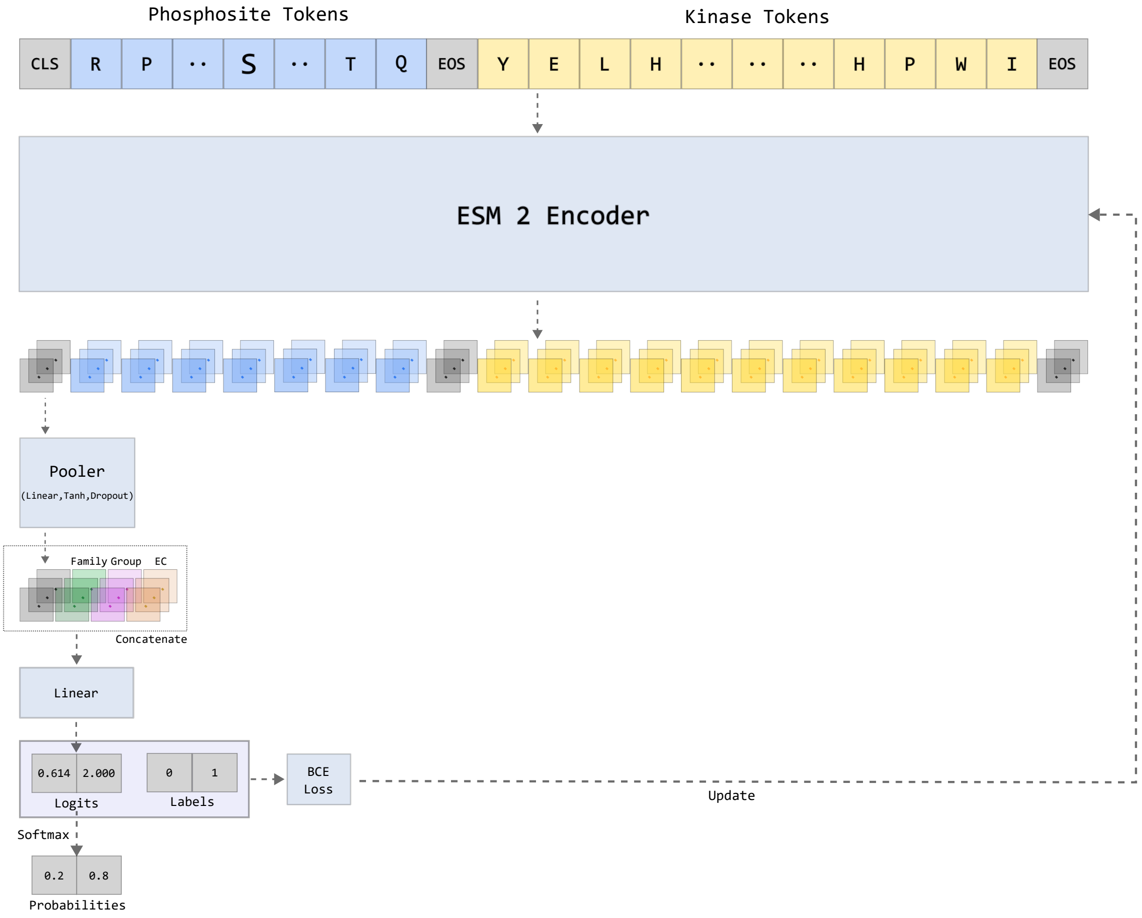

Figure 3.9 Training scheme of DARKIN-Interact, a binary compatibility model for phosphosite–kinase interaction prediction.

computed in the same multiclass setting used by DARKIN-FT, enabling a like-for-like comparison. The desired behavior is a high score for true kinase partners and a low score for unrelated kinases; AP is particularly diagnostic because it rewards correct ordering across all classes. An overview of the evaluation process is shown in Figure 3.10.

Incorporating Kinase Family, Group, and EC Information. DeepKinZero and DARKIN benchmarks show that concatenating one-hot encodings of kinase family, group, and EC annotations to sequence embeddings boosts performance (Deznabi et al., 2020; Sunar et al., 2024). We adopt a similar strategy: the family, group, and EC vectors are appended to the pooled <CLS> representation immediately before the final linear layer in the two-layer classifier. The first linear layer thus acts as a pooler for the sequence embedding, while the second integrates both sequence and kinase attribute information. DeepKinZero and DARKIN studies have demonstrated that these attributes capture functional similarity beyond primary sequence and consistently enhance predictive accuracy; for fairness, we include them

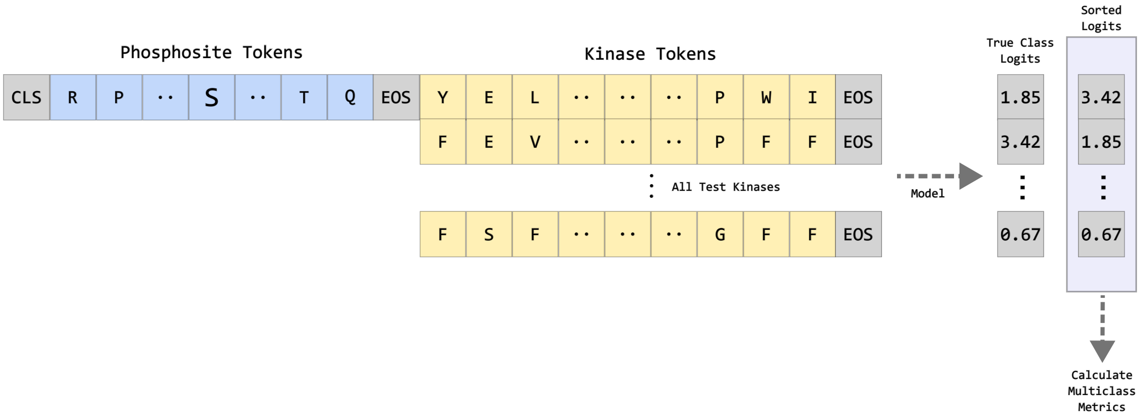

Figure 3.10 Multiclass evaluation with DARKIN-Interact obtained by ranking binary compatibility scores, analogous to the DARKIN-FT evaluation protocol.

in all models unless stated otherwise.

3.6.2 Ablation Studies and Representation Improvements

DeepKinZero originally demonstrated that learning a bilinear compatibility matrix **W** between phosphosite and kinase embeddings is an effective strategy for zero-shot interaction prediction (Deznabi et al., 2020). Building upon this idea, our DARKIN benchmark systematically evaluates the impact of alternative protein language model embeddings on the same architecture (Sunar et al., 2024). Because the model comprises several modular components, it provides a convenient test-bed for dissecting how each module contributes to overall performance. In this section, we report ablation experiments that replace or remove individual blocks and investigate improved projection strategies for the underlying representations.

3.6.2.1 Removing W Bilinear Compatibility Function

The bilinear compatibility function **W** maps phosphosite and kinase vectors into a shared latent space and returns a scalar compatibility score. While expressive, this layer adds many parameters and places no constraint on the embedding dimensions. To investigate its necessity, we removed **W** and replaced it with a simple dot product, which requires that phosphosite and kinase embeddings share the same dimensionality.

Adding a Projection to Phosphosite Representations. To satisfy the equal-dimension requirement introduced by the dot product, we first project phosphosite embeddings to the kinase dimension using a single linear layer, followed by a `tanh` activation. The transformed phosphosite vector is then combined with kinase embeddings through the dot product to yield compatibility scores. This configuration allows us to isolate the contribution of a lightweight projection from that of the bilinear form.

Adding a Projection to Kinase Representations. Unlike phosphosite embeddings, kinase vectors concatenate sequence information with one-hot encodings of family, group, and EC attributes. We tested two projection schemes:

- **Single-stage projection**: Sequence and attribute vectors are concatenated and passed through one linear layer plus `tanh`, mimicking the role previously played by $\mathbf{W}$.
- **Two-stage projection**: Sequence and attribute parts are processed separately by their own linear+`tanh` layers, then merged. At merge time, we evaluated both simple concatenation and a gated fusion that learns adaptive weights for the two sources. Gated fusion is implemented as follows:

  $$\mathbf{z} = \mathbf{s} \odot \sigma(\mathbf{W}_g[\mathbf{s};\mathbf{a}]) + (\mathbf{W}_a\mathbf{a}) \odot (1 - \sigma(\mathbf{W}_g[\mathbf{s};\mathbf{a}])) \tag{3.8}$$

  Here, $\mathbf{s} \in \mathbb{R}^d$ is the sequence embedding, $\mathbf{a} \in \mathbb{R}^{d'}$ is the attribute embedding, and $[\mathbf{s};\mathbf{a}]$ denotes their concatenation. The gating weights are computed by passing this concatenation through a learned linear transformation $\mathbf{W}_g$ followed by a sigmoid activation $\sigma(\cdot)$. The attributes are first projected into the sequence embedding space using $\mathbf{W}_a \in \mathbb{R}^{d \times d'}$. The final merged vector $\mathbf{z}$ blends information from both sources, where the learned gate softly controls each dimension's contribution. This allows the model to emphasize either the sequence or the attribute features based on context.

These experiments clarify how best to integrate categorical kinase metadata with sequence embeddings when the bilinear compatibility function is absent.

3.6.2.2 Regularization on W Matrices

The bilinear compatibility function $\mathbf{W}$ is a crucial component in this zero-shot framework: it projects phosphosite and kinase embeddings into a joint space where their compatibility is scored. Because $\mathbf{W}$ must implicitly encode kinase family, group, and EC information while generalizing beyond the training split of DARKIN, regularizing this layer is essential for strong performance. We already apply weight decay during the training of all modules. In addition, we explore three complementary regularization strategies: i) Ranked Dropout, ii) ℓ_1 regularization, and iii) spectral norm regularization, and analyze their effects on model performance.

Ranked Dropout: Classical dropout randomly masks a subset of activations during training, preventing co-adaptation and encouraging the model to rely on distributed cues rather than memorizing specific dimensions (Srivastava et al., 2014). Targeted dropout improves stability by ranking weights (or activations) by absolute magnitude and stochastically masking a fixed fraction of the smallest ones (A. N. Gomez et al., 2019). We adopt the same magnitude-ranking mechanism but invert the selection: at every update step, we zero the top-γ fraction of the largest-magnitude elements, a variant we call Ranked Dropout. When applied to the bilinear compatibility matrix $\mathbf{W}$, this strategy prevents the model from concentrating signal in a few dominant directions, compels information to flow through lower-magnitude dimensions, and thus improves generalization from the limited set of training kinases to the full human kinome.

ℓ_1 **Regularization:** Adding an ℓ_1 penalty to the loss encourages sparsity in $\mathbf{W}$, effectively selecting a compact subset of informative interactions while driving irrelevant entries toward zero. A sparser $\mathbf{W}$ is less prone to capturing noise from limited training data and is easier to interpret biologically.

Spectral Norm Regularization: Spectral norm regularization constrains the largest singular value of the weight matrices in a neural network (Yoshida and Miyato, 2017). This method reduces the sensitivity of the model to input perturbation, ensuring that a trained model exhibits slight sensitivity to the perturbation of test data. This contributes to better generalizability.

Chapter 4

RESULTS AND DISCUSSION

In this chapter, we first present the results of DARKIN-FT and DARKIN-Interact methods, analyze their strengths and weaknesses, evaluate how modifying different components affects model performance, and assess the impact of incorporating synthetic data along with the performance of models used to generate such data. While doing so, we also discuss the evaluation metrics used (refer to 3.4) and how we interpret them to better understand model behavior.

4.1 Experimental Setup

For the approaches, we report not only the zero-shot results but also their few-shot performance. To evaluate the latter, we conduct 5-shot experiments, in which five test examples are randomly selected from each test class and added to the training set. This sampling process is repeated ten times to generate a pool of 50 examples. After removing any potential duplicate entries from this pool, the final 5-shot dataset is formed by randomly selecting five unique examples. Although the primary goal of these methods is to learn the phosphosite-kinase association in a zero-shot setting so that predictions can be made for kinases not seen during training, which would be particularly valuable in cases of novel kinase discoveries, few-shot performance offers additional insights and another perspective from which to analyze the problem.

The hyperparameter search space explored for all models discussed in this section is detailed in Appendix Table A.15. Hyperparameter optimization was conducted under a zero-shot learning setting, using Macro AP as the evaluation metric. The optimal hyperparameters, highlighted in bold in the table, were subsequently utilized for the few-shot learning experiments. For the DARKIN-FT and DARKIN-

Interact models, training was performed for 100 and 50 epochs, respectively. The *ProGen2–Phospho* models were trained for 5000 steps. In all instances, the model checkpoint corresponding to the best validation performance was selected for subsequent analysis. After identifying the optimal hyperparameters, the training and validation sets were merged, and the models were retrained using the same number of epochs (100 for DARKIN-FT and 50 for DARKIN-Interact). For *ProGen2–Phospho*, the final training was limited to 2000 steps, as performance consistently declined beyond this point in the initial training runs.

4.2 Performance of the Compatibility-Based Zero-Shot Model

Inspired by the DeepKinZero framework proposed by Deznabi et al. (2020) and extended in the DARKIN study introduced by Sunar et al. (2024), the method, DARKIN-FT, learns a phosphosite-kinase compatibility function through a learned compatibility matrix $\mathbf{W}$, applied to phosphosite and kinase embeddings. In the BZSM method of DARKIN, the matrix $\mathbf{W}$ is trained using fixed pLM embeddings, which are projected into a shared space where compatibility is evaluated. For the DARKIN-FT, we also fine-tuned the phosphosite encoder during training to better adapt its representations to the task. The goal is to assess whether end-to-end fine-tuning leads to more informative phosphosite embeddings. To that end, both randomly initialized and pre-trained versions of the pLM were evaluated in order to observe the role of prior knowledge in learning this association. These experiments not only aim to assess baseline model performance but also inform the selection of phosphosite encoders for subsequent methods introduced in the following sections.

To perform this analysis, we selected four pLMs that performed well in the DARKIN benchmark: ESM1B, ESM2-150M, ESM2-650M, and ProtT5-XL (Elnaggar et al., 2021; Rives et al., 2021; Lin et al., 2023). ESM2-150M, the smallest among these, was included to examine how the number of parameters, and hence the model's prior knowledge, affects performance on this specific problem. As demonstrated in our earlier work Sunar et al., 2024; Deznabi et al., 2020, incorporating kinase family, group, and EC information with the kinase domain embeddings significantly improves performance. Thus, these attributes are included in the kinase representations for all models in this study. Each model was run with three different random seeds, and the average results were reported. All results are based on the DARKIN Split 1 unless otherwise noted.

Table 4.1 and 4.2 compare the results of the BZSM approach across the four models, with both fixed **W** compatibility learning and end-to-end fine-tuning of the phosphosite encoder. As expected, when the phosphosite encoder is fine-tuned, thus introducing more trainable parameters, performance improves notably over the fixed BZSM setting. This highlights the limitations of relying solely on static embeddings and the **W** matrix for learning phosphosite-kinase associations. While ESM2-150M performed significantly worse than ESM2-650M in the BZSM setting due to its smaller size, this gap is reduced substantially once fine-tuning is enabled. This suggests that learning task-specific embeddings compensates for the lack of pre-trained knowledge in the smaller model. Interestingly, models with randomly initialized weights outperformed those initialized with pre-trained weights. This counterintuitive result indicates that protein language models trained on unrelated tasks may struggle to converge when repurposed for phosphosite-kinase prediction, whereas randomly initialized models adapt more easily. Among the four models, ProT5-XL uses an encoder-decoder structure in its original task. Since this setup only trains the encoder component, its performance lags behind the other three encoder-only models. Consistent with previous findings in the DARKIN study, ESM1B again yields the best performance when its embeddings are fine-tuned for this specific task.

Table 4.1 and 4.2 also include phosphosite-level Phosphosite AP scores, which are important when considering practical use cases. The macro-averaged AP scores (Kinase AP) represent the average of AP scores calculated separately for each test kinase class. Ultimately, the goal is to provide a model that, given a phosphosite input, can predict the most likely responsible kinase. While Phosphosite AP is included for completeness, we place less emphasis on it during performance interpretation due to the relatively small number of test kinase classes, which can make this metric less reliable. Still, we observe that Phosphosite AP scores follow trends consistent with those of the Macro AP scores reported across models.

Appendix Table A.1 and A.2 present the performance of the four pLMs on kinase attribute-level metrics. Specifically, the AP and Accuracy scores are computed by grouping predictions based on kinase attributes such as family, group, or EC number. To compute these values, predictions for kinases sharing the same attribute were pooled, and evaluation was performed at the attribute level. This analysis reveals some of the challenges faced by the models. While models achieve good discrimination between broad kinase groups, they struggle to differentiate between kinases within the same group. This difficulty likely stems from high sequence similarity among kinases in the same group, making it hard for the model to learn fine-grained distinctions. Figure 4.1 shows the distribution of cosine similarity be-

Table 4.1 Zero-shot performance of four pLM backbones under three strategies: BZSM (DARKIN benchmark), DARKIN-FT, and DARKIN-FT–PT (the phosphosite encoder is initialized with pre-trained weights).

Model	**Method**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
ESM1B	BZSM	0.1746	0.1320	0.3314	0.4698	0.2879
	DARKIN-FT	**0.2250**	**0.1751**	**0.3977**	**0.5228**	**0.3321**
	DARKIN-FT - PT	0.1822	0.1648	0.3819	0.5177	0.3191
ProtT5XL	BZSM	0.1673	0.1339	0.2953	0.4394	0.2790
	DARKIN-FT	0.1710	0.1376	0.2720	0.3936	0.2631
	DARKIN-FT - PT	0.1637	0.1188	0.2744	0.4894	0.2705
ESM2-650M	BZSM	0.1659	0.1280	0.3128	0.4429	0.2744
	DARKIN-FT	0.1966	0.1673	0.3717	0.4868	0.3113
	DARKIN-FT - PT	0.1865	0.1471	0.3847	0.5127	0.3149
ESM2-150M	BZSM	0.1325	0.1058	0.3177	0.4448	0.2629
	DARKIN-FT	0.1862	0.1416	0.3465	0.4835	0.2915
	DARKIN-FT - PT	0.1713	0.1174	0.2892	0.4314	0.2617

Table 4.2 Few-shot (5-shot) performance of four pLM backbones under three strategies: BFSM (DARKIN benchmark BZSM, renamed for few-shot), DARKIN-FT, and DARKIN-FT–PT (the phosphosite encoder is initialized with pre-trained weights).

Model	**Method**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
ESM1B	BFSM	0.1894	0.1443	0.3202	0.4582	0.2965
	DARKIN-FT	**0.2386**	0.1830	**0.4132**	**0.5457**	**0.3465**
	DARKIN-FT - PT	0.1956	0.1616	0.3967	0.5315	0.3298
ProtT5XL	BFSM	0.1766	0.1285	0.3139	0.4377	0.2835
	DARKIN-FT	0.1797	0.1514	0.3052	0.4211	0.2858
	DARKIN-FT - PT	0.1759	0.1167	0.2847	0.4448	0.2740
ESM2-650M	BFSM	0.1866	0.1325	0.2934	0.4251	0.2776
	DARKIN-FT	0.2272	**0.1885**	0.3785	0.5102	0.3330
	DARKIN-FT - PT	0.1859	0.1593	0.3257	0.4740	0.3025
ESM2-150M	BFSM	0.1431	0.0946	0.2981	0.4613	0.2589
	DARKIN-FT	0.2055	0.1601	0.3841	0.5331	0.3188
	DARKIN-FT - PT	0.1740	0.1254	0.3044	0.4361	0.2739

tween test kinases, illustrating that many kinase sequences are highly similar. Given the available training data, conveying these subtle differences to the model remains a challenging task.

4.3 Reformulating Kinase–Phosphosite Association as Binary Classification

PLMs, trained on massive protein sequence databases, inherently contain structural and functional information about proteins. Therefore, leveraging the prior knowledge embedded in these models is particularly valuable. Although the DARKIN-FT framework achieved strong results even when training the phosphosite encoder from

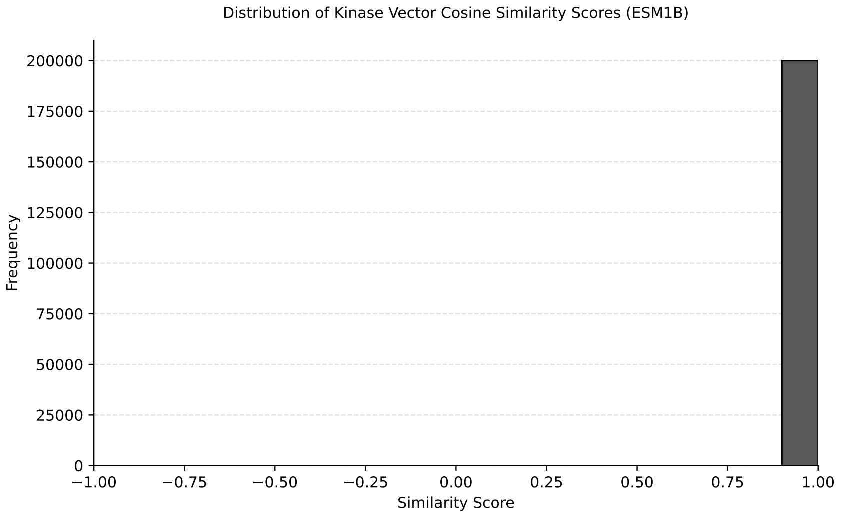

Figure 4.1 Pairwise cosine-similarity distribution of DARKIN kinase embeddings extracted from ESM1B, shown over the full range [–1, 1].

scratch, it relied on pre-trained kinase embeddings combined with a learned **W** compatibility matrix to project them into a shared embedding space.

In contrast, the binary classification approach, DARKIN-Interact, described in this section, aims to model the interaction between kinase and phosphosite sequences directly using a joint input representation. Consequently, we only used pre-trained pLMs in this setup. This method differs significantly from DeepKinZero, BZSM baseline of DARKIN, and DARKIN-FT by explicitly modeling the sequence-level interaction between kinase and phosphosite and taking full advantage of the attention mechanism and transformer architecture. Reformulating the task as binary classification provided a complementary perspective on the kinase–substrate association problem.

This approach also allowed us to examine different aspects of the problem through its modular design. One key distinction was reframing the original multilabel classification task as a binary classification problem. Since all kinase–phosphosite pairs in the DARKIN dataset represent positive associations, we had to generate negative samples during training. The negative samples were randomly drawn from a pool of non-interacting kinase–phosphosite pairs in the DARKIN dataset. This sampling was performed randomly during each training batch. The number of negative samples per positive instance, denoted as n, plays a significant role in both performance and computational cost. For each batch, the effective batch size becomes batch_size $\times (n+1)$. During evaluation, each validation or test phosphosite is paired with all candidate kinases, leading to a batch size of batch_size × #test kinases.

A major difference between this method and the DARKIN-FT is the inclusion of

Table 4.3 Zero-shot performance of DARKIN-Interact with different pLM encoders fine-tuned via LoRA.

Model	n	Layers	Macro AP	Top 1 Acc	Top 3 Acc	Top 5 Acc	Phosphosite AP
ESM1B	4	LoRA	**0.1880**	**0.1464**	**0.2871**	**0.4194**	**0.2740**
ESM2-650M	4	LoRA	0.1701	0.1350	0.2892	0.4045	0.2658
ESM2-150M	4	LoRA	0.1594	0.1145	0.2638	0.3918	0.2415
ESM2-35M	4	LoRA	0.1278	0.1124	0.2397	0.3564	0.2342

Table 4.4 Few-shot (5-shot) performance of DARKIN-Interact with different pLM encoders fine-tuned via LoRA.

Model	n	Layers	Macro AP	Top 1 Acc	Top 3 Acc	Top 5 Acc	Phosphosite AP
ESM1B	4	LoRA	0.1699	**0.1593**	0.3320	0.4574	**0.2978**
ESM2-650M	4	LoRA	**0.1814**	0.1553	**0.3541**	**0.4700**	0.2958
ESM2-150M	4	LoRA	0.1486	0.1301	0.2878	0.4164	0.2639
ESM2-35M	4	LoRA	0.1229	0.1332	0.2792	0.3975	0.2597

the kinase domain sequence as part of the model input. To construct the input, the kinase and phosphosite sequences are concatenated. Since kinase sequences can be long, they were truncated to the first 200 amino acids, with the aim of preserving the active domain regions as much as possible. Figure 3.8 shows the locations of kinase active domains within their sequences. This input formulation, like the choice of n, also poses computational challenges.

4.3.1 Baseline Model Performance Analysis

To evaluate the baseline performance of this approach and select a model for deeper analysis, we trained four encoder-only protein language models: ESM1B, ESM2-35M, ESM2-150M, and ESM2-650M. We excluded ProT-T5-XL from this section because, as discussed in Section 4.2, its encoder-decoder architecture underperforms in this encoder-only setup. Moreover, to better understand how model size affects performance, we included the smaller 35M variant of ESM2.

All models were trained using Low-Rank Adaptation (LoRA), with the rank hyperparameter set to 32. This allowed us to fine-tune models while keeping the number of trainable parameters manageable, which is particularly important given the increased input length and computational demand due to negative sampling. In Section 4.3.3, we explore alternatives to LoRA for fine-tuning.

We fixed the number of negative samples per positive instance to $n = 4$ for all models. This value provided a balance between dataset size and informativeness, enabling a reasonable number of negative samples without making the dataset excessively

Table 4.5 Effect of the number of negative samples per positive (n) on DARKIN-Interact's zero-shot performance. The ESM1B encoder is utilized and fine-tuned via LoRA.

Model	n	Layers	Macro AP	Top 1 Acc	Top 3 Acc	Top 5 Acc	Phosphosite AP
ESM1B	1	LoRA	0.1847	**0.1584**	**0.3685**	**0.5085**	**0.3088**
ESM1B	2	LoRA	0.1899	0.1506	0.2772	0.4045	0.2730
ESM1B	4	LoRA	0.1880	0.1464	0.2871	0.4194	0.2740
ESM1B	8	LoRA	**0.1910**	0.1534	0.3210	0.4561	0.2909
ESM1B	12	LoRA	0.1868	0.1407	0.2793	0.3769	0.2604

imbalanced or computationally intensive. Table 4.5 and 4.6 show the effect of varying n values on performance, which we will discuss in the following section. Table 4.3 and 4.4 reports model-wise metrics. A notable observation is that ESM2 models exhibit consistent improvement with increasing size. As shown in Appendix Table A.3 and A.4, attribute-level metrics also improve with larger model sizes for both zero- and few-shot cases. Among all models, ESM1B again achieved the best performance.

A striking result is that when we train the model in a binary classification setting and then evaluate it in the same multilabel setting used in DARKIN-FT, the performance remains comparable. One key reason for this success is the use of attention mechanisms to model phosphosite–kinase interactions directly. In addition, unlike previous methods where kinase attribute information (e.g., family, group, EC) is embedded prior to training, here we append these attributes after processing the sequence pair through the transformer. Together, these observations point to the utility of alternative modeling strategies in this task. Based on these findings, we selected ESM1B for further experiments.

4.3.2 Impact of Negative Sample Count on Model Performance

The number of negative samples (n) is a critical hyperparameter in this task. Increasing n exposes the model to more non-interacting kinase–phosphosite pairs during training, potentially improving its ability to discriminate true interactions. However, as previously mentioned, larger n values also increase the computational load.

To analyze the impact of n, we fixed all other settings and evaluated model performance with $n \in \{1, 2, 4, 8, 12\}$. An n of 1 yields a balanced dataset, while an n of 12 leads to a highly imbalanced dataset but greatly increases the number of training examples.

As shown in Table 4.5 and 4.6, performance does not vary dramatically across

Table 4.6 Effect of the number of negative samples per positive (n) on DARKIN-Interact's few-shot (5-shot) performance. The ESM1B encoder is utilized and fine-tuned via LoRA.

Model	n	Layers	Macro AP	Top 1 Acc	Top 3 Acc	Top 5 Acc	Phosphosite AP
ESM1B	1	LoRA	0.1816	**0.1790**	**0.3375**	0.4408	**0.3052**
ESM1B	2	LoRA	0.1777	0.1506	0.3147	0.4211	0.2824
ESM1B	4	LoRA	0.1699	0.1593	0.3320	**0.4574**	0.2978
ESM1B	8	LoRA	0.1830	0.1624	0.3233	0.4219	0.2816
ESM1B	12	LoRA	**0.1834**	0.1695	0.3351	0.4227	0.2976

different values of n. The attribute-level metrics, presented in Appendix Table A.5 and A.6, also reflect this stability, indicating only a marginal improvement at $n = 1$. While incrementing n increases the number of training instances, the resulting imbalance and the high similarity among kinase classes (see Figure 4.1) likely limit the benefits of additional negatives. Therefore, for all subsequent experiments, we fix $n = 4$ to maintain computational feasibility while ensuring adequate representation of negative examples.

4.3.3 Effect of Fine-Tuning Different pLM Layers

Fine-tuning a pre-trained model typically involves updating all model parameters. However, with the increasing size of modern language models, parameter-efficient fine-tuning strategies have gained popularity. LoRA, which we used in prior experiments, updates only low-rank matrices injected into frozen layers, enabling efficient training with fewer trainable parameters.

In earlier sections, we fine-tuned ESM1B using LoRA with rank 32. This setup helped manage memory usage due to long inputs and negative sampling. However, performance remained slightly below that of the BZSM baseline of DARKIN. To investigate whether this gap was due to limited parameter updates or differences in modeling strategy, we compared LoRA with another fine-tuning approach: updating only the top l layers of the model. We hypothesized that lower layers of pLMs capture general protein features, whereas higher layers can be better adapted to task-specific learning. Therefore, we selected $l \in \{1, 2, 4, 8, 12\}$ and retrained the model, updating only the top l layers.

Table 4.7 and 4.8 compare the LoRA and layer-wise fine-tuning strategies. Results indicate that updating the top layers improves performance, with results approaching those of Section 4.2. Performance gains plateau after two layers, but show a slight increase in both Accuracy and AP when extended to the last 12 layers. This

Table 4.7 Zero-shot impact of unfreezing the last l ESM1B encoder layers versus LoRA on DARKIN-Interact.

Model	n	**Layers**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
ESM1B	4	Last 1	0.1541	0.1711	0.3302	0.4371	0.2933
ESM1B	4	Last 2	0.2005	**0.1916**	0.3395	0.4455	0.3062
ESM1B	4	Last 4	**0.2078**	0.1732	0.3154	0.4491	0.3006
ESM1B	4	Last 8	0.2041	0.1810	0.2935	0.4186	0.2899
ESM1B	4	Last 12	**0.2078**	0.1669	**0.3501**	**0.5035**	**0.3109**
ESM1B	4	LoRA	0.1880	0.1464	0.2871	0.4194	0.2740

Table 4.8 Few-shot (5-shot) impact of unfreezing the last l ESM1B encoder layers versus LoRA on DARKIN-Interact.

Model	n	**Layers**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
ESM1B	4	Last 1	0.1604	0.1672	0.3572	0.4763	0.3066
ESM1B	4	Last 2	0.1935	0.1806	0.3336	0.4290	0.2968
ESM1B	4	Last 4	0.2010	0.1806	0.3754	0.4976	0.3142
ESM1B	4	Last 8	0.2179	**0.1932**	0.3706	0.4850	0.3179
ESM1B	4	Last 12	**0.2284**	0.1885	**0.4077**	**0.5260**	**0.3383**
ESM1B	4	LoRA	0.1699	0.1593	0.3320	0.4574	0.2978

trend is also observed in the attribute-level metrics, shown in Appendix Table A.7 and A.8.

These findings demonstrate that effective performance can be achieved even with a fundamentally different modeling strategy from DeepKinZero and the BZSM baseline. The goal of these experiments was to observe how various training strategies and architectural decisions affect performance. The results underscore the effectiveness of transformer-based architectures, and particularly the attention mechanism, in modeling phosphosite–kinase associations.

4.4 Ablation Studies and Representation Improvements

Building upon the best-performing model identified in Table 4.1, where we jointly trained the phosphosite encoder and the **W** compatibility matrix in an end-to-end manner as in the BZSM baseline of DARKIN framework, we conducted a series of ablation studies to examine how replacing or modifying individual components affects performance. These experiments are particularly valuable for understanding the role and strength of the **W** compatibility function in capturing the kinase–phosphosite relationship.

4.4.1 Removing the W Bilinear Compatibility Function

In the absence of the **W** function, we compute the compatibility between phosphosite and kinase vectors via a simple dot product. However, one challenge in this approach is that the dimensions of the phosphosite and kinase vectors may not match. Although both vectors are generated from the same pLM, the kinase embeddings include additional metadata such as family, group, and EC number, which results in a dimensional mismatch.

Throughout the DARKIN study and previous experiments, we used the CLS token from the last layer of the encoder to represent each sequence. While the DARKIN paper compared using CLS token embeddings versus averaging all token embeddings and reported minor differences, the CLS token generally led to slightly better results and was, therefore, preferred.

Projecting the phosphosite embeddings: To address the dimensional mismatch between phosphosite and kinase embeddings, we first applied a projection step on the phosphosite side. Specifically, we introduced a "pooler" module consisting of a linear layer followed by a Tanh activation function, which projects the CLS token embedding of the phosphosite into the same dimensional space as the kinase embeddings. This projection module was trained jointly with the model. As shown in Table 4.9, although this approach resulted in a slight drop in performance compared to the **W** matrix, the decrease was minimal, suggesting that dot product compatibility serves as a feasible alternative, although with a small decrease in effectiveness.

Projecting the kinase embeddings: Next, we applied a similar projection operation to the kinase embeddings to assess whether symmetric projection would further improve performance. On the kinase side, additional modeling flexibility was possible due to the inclusion of structured attributes. We explored two designs: a single-stage and a two-stage projection. In the single-stage approach, the kinase domain sequence embedding and the attribute embedding were concatenated and passed through a single projection layer. In the two-stage approach, each component, the domain sequence, and the attribute embedding were projected separately through independent layers before being merged.

For both projection strategies, we tested two fusion mechanisms: simple concatenation and gated fusion. While concatenation preserves the structure of individual vectors, gated fusion combines the two representations via element-wise addition with learned weights, enabling the model to emphasize or suppress components dy-

Table 4.9 Ablation study of zero-shot DARKIN-FT exploring alternative compatibility projections, merging strategies (concatenation vs gated fusion), kinase-pooler designs, and optional attribute projections.

Phosphosite Pooler	Kinase Pooler	Merging	Attribute Projection	Macro AP	Top 1 Acc	Top 3 Acc	Top 5 Acc	Phosphosite AP
Default Model Setting				**0.2250**	0.1751	0.3977	0.5228	0.3321
✓	–	–	–	0.2218	0.1732	**0.4045**	**0.5304**	0.3330
✓	Single Stage	Concat	–	0.2220	0.1831	0.3925	0.5220	**0.3364**
			✓	0.2092	0.1782	0.3932	0.5092	0.3304
		Gated	–	0.1990	0.1676	0.3592	0.5063	0.3125
			✓	0.2154	0.1824	0.3911	0.5240	0.3341
	Two Stage	Concat	–	0.2120	0.1697	0.3826	0.5014	0.3275
			✓	0.2116	0.1782	0.3621	0.4823	0.3211
		Gated	–	0.2124	**0.2125**	0.3868	0.4957	0.3236
			✓	0.2053	0.1782	0.3642	0.4929	0.3241

namically.

According to the results in Table 4.9 and the attribute-level performances provided in Appendix Table A.9, adding a projection layer to the kinase embedding did not lead to a significant performance improvement. Interestingly, gated fusion outperformed concatenation in the two-stage setup but underperformed in the single-stage setup. We believe this discrepancy arises because, in the single-stage design, fusion occurs before projection. If the fusion weights are not well-initialized or optimized, the quality of the resulting representation may suffer, thus degrading projection performance.

To further explore the representation power of the kinase attributes, we experimented with projecting the binary attribute vectors to higher-dimensional representations using an additional linear layer. This was followed by the same sequence of projection and fusion operations as before. The results, also shown in Table 4.9, indicate that projecting attribute vectors to higher dimensions generally led to slightly worse performance compared to using the binary vectors directly. However, when gated fusion was applied in the single-stage design, performance improved, suggesting that projection prior to fusion can be more effective when using high-dimensional representations instead of one-hot encodings. A similar trend is observed in the attribute-level metrics presented in Appendix Table A.9.

Overall, this section shows that while the **W** matrix remains the most effective compatibility mechanism, alternative formulations using projection and dot product can approximate its performance. These findings also illustrate the nuanced tradeoffs between representation learning and architecture design in phosphosite–kinase

Table 4.10 Regularization strategies for the compatibility matrix **W** in zero-shot DARKIN-FT.

Regularization	Macro AP	Top 1 Acc	Top 3 Acc	Top 5 Acc	Phosphosite AP
L2 (Default)	0.2250	0.1751	0.3977	0.5228	0.3321
L2 + L1	0.1711	0.1223	0.3487	0.5042	0.2871
L2 + Ranked Dropout	0.2240	**0.1817**	**0.4087**	**0.5297**	**0.3383**
L2 + Spectral Norm	**0.2287**	0.1768	0.3981	0.5233	0.3336

association modeling.

4.4.2 Regularization on W Matrices

The **W** matrix plays a central role in modeling the interaction between phosphosite and kinase embeddings, directly influencing prediction quality. As observed in Appendix Table A.1, while the model achieves strong performance at the group level, it struggles to distinguish between kinases within the same group. This suggests that the model may overfit to certain group-level patterns, raising concerns about generalization and stability. To address this, we applied three different regularization techniques to the **W** matrix and evaluated their effect on model performance. These methods were designed to constrain or stabilize the learning process in different ways, potentially mitigating overfitting.

The comparison of performance with and without regularization is presented in Table 4.10. As with previous analyses, the attribute-level results presented in Appendix Table A.10 follow a similar behavior to the main performance metrics reported in Table 4.10. Among the methods tested, L1 regularization produced a clear degradation in performance. We hypothesize that this is due to L1 penalizing many features too harshly, effectively shrinking informative weights to near zero and impairing the model's ability to learn from limited data.

Ranked Dropout, in contrast, yielded strong results in terms of Phosphosite AP and Accuracy, even though its Macro AP was similar to the baseline. These results suggest that Ranked Dropout is particularly effective in improving phosphosite retrieval by helping the model avoid over-reliance on specific features.

Spectral Norm regularization slightly outperformed the baseline in both Macro AP and Accuracy. This performance gain suggests that constraining the largest singular value of the weight matrices effectively improves generalizability, likely by limiting the model's sensitivity to input perturbations and encouraging smoother decision boundaries.

These results demonstrate that carefully selected regularization techniques can bring modest but meaningful improvements to compatibility-based models. While the **W** matrix is undoubtedly powerful, it is also sensitive to overfitting. Regularization can play a critical role in balancing this power and ensuring better generalization, especially across kinase subtypes with high sequence similarity or small sample sizes.

4.5 Data Augmentation Strategies

In this section, we first evaluate the performance of the models used or trained for data augmentation. Then, we assess the impact of incorporating the synthetic data generated by these methods into the DARKIN dataset.

4.5.1 Finetuning ProGen2 for Kinase-Conditional Phosphosite Generation

ProGen2 is an autoregressive protein generation model trained on a large-scale protein sequence database (Nijkamp et al., 2023). Its training mechanism resembles that of natural language models. It uses special tokens analogous to [CLS] and [EOS], represented as "1" and "2", respectively. Since ProGen2 was originally trained to generate entire protein sequences, we adapted it to this work by applying task-specific fine-tuning.

The aim was to augment the training data for kinase classes with limited examples in the DARKIN dataset by generating phosphosite sequences that could plausibly be phosphorylated by specific kinases. To achieve this, we constructed the input sequence as follows: the kinase sequence was placed at the beginning, followed by the special "1" token, then the phosphosite sequence, and finally the "2" token. During fine-tuning, the loss was computed only on the phosphosite segment, ensuring that the model focused on generating this region while conditioning on the kinase.

The model was trained using the DARKIN Split 1 to ensure consistency with previous experiments. However, unlike the zero-shot design in DARKIN, which ensures kinase classes are split across training and test sets, the same phosphosite sequences can appear in both training and testing sets due to the dataset's multilabel nature. Although this overlap is not problematic for most of the experiments, it may lead to overfitting in generative modeling. To prevent data leakage, we removed overlapping sequences from the train set during ProGen2 training. This resulted in the removal

Table 4.11 Position-wise negative-log-likelihood for DARKIN test phosphosites under vanilla ProGen2 and *ProGen2–Phospho* trained with cross-entropy or BLOSUM loss.

Finetuning Method	**Positions**														
	-7	**-6**	**-5**	**-4**	**-3**	**-2**	**-1**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
No Finetuning	17.09	3.24	2.96	2.94	2.91	2.91	2.87	**2.72**	2.79	2.77	2.78	2.74	2.74	2.71	2.71
Cross Entropy Loss	2.83	2.81	2.79	2.56	2.32	2.21	2.19	**0.53**	1.75	2.05	1.98	2.00	1.86	1.99	1.98
BLOSUM Loss	2.84	2.82	2.79	2.62	2.36	2.24	2.22	**0.95**	1.80	2.01	2.01	1.93	1.92	1.92	1.92

of 1054 sequences.

Table 4.11 compares the position-wise loss values on the test set for the fine-tuned ProGen2 models and the original pre-trained ProGen2 model. We experimented with two loss functions: standard cross-entropy and a modified version we refer to as "BLOSUM Loss," which adjusts the cross-entropy loss using a BLOSUM-based amino acid similarity matrix. The goal was to encourage the model to generate amino acids with similar biochemical properties when mistakes occur. Synthetic sequences generated from both training regimes were later used for data augmentation, and their downstream effects are discussed in the following subsection.

As shown in Table 4.11, the fine-tuned models outperformed the original ProGen2. This is expected, as the goal of fine-tuning was to adapt a general-purpose model trained on diverse proteins to the specific phosphosite generation task. While ProGen2 may not have captured detailed phosphosite-specific knowledge from general training alone, it was able to learn kinase–phosphosite associations more effectively during fine-tuning. The model successfully learned that phosphosites must be 15 amino acids long and that the phosphorylation site must appear in the center of the sequence. Figure 4.2 presents the confusion matrix for the central residue predictions. The BLOSUM Loss, which imposes softer penalties for confusing serine (S) and threonine (T), reflects these similarities while effectively distinguishing S/T from tyrosine (Y). The results show that the model trained with standard cross-entropy loss performs better at discriminating the phosphosite residue, likely due to the sharper penalties it applies for incorrect predictions. Although the impact of BLOSUM Loss on overall performance appears more limited due to its milder penalization, it plays an important role in preserving biological plausibility. Since this approach aims to generate entirely novel protein sequences, we consider preserving realistic phosphosite structures a critical factor, and BLOSUM Loss contributes meaningfully to that goal.

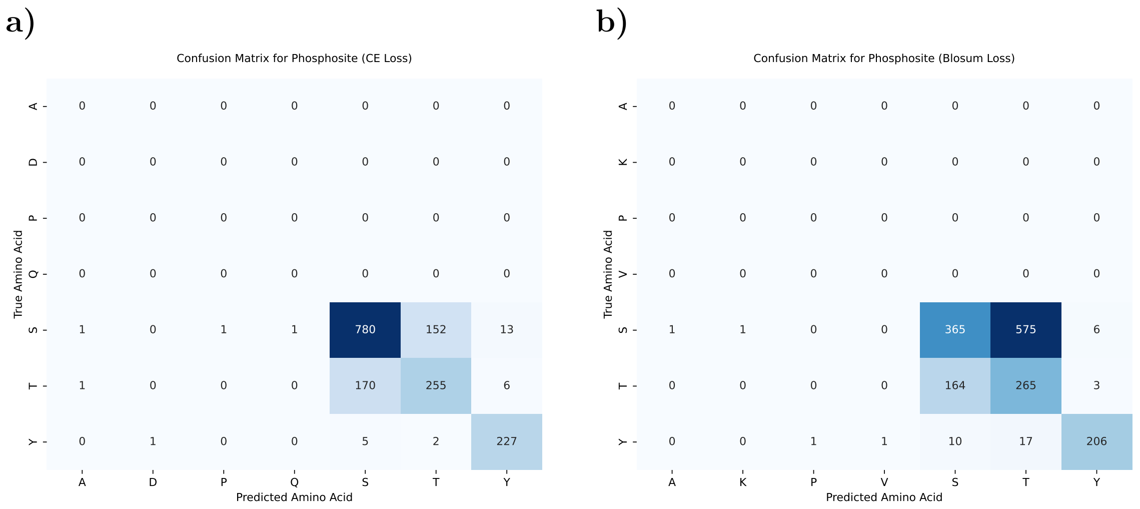

Figure 4.2 Residue-level confusion matrices at the phosphosite position comparing *ProGen2–Phospho* trained with (a) Cross-Entropy loss and (b) BLOSUM loss.

Table 4.12 Zero-shot effect of augmenting DARKIN-FT with synthetic phosphosites from *ProGen2–Phospho* using either loss function (Cross Entropy, CE, and BLOSUM) and three kinase sets (Train, Train + Val, Train + Val + Test).

Augmentation Method	**Kinase Set**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
–	–	**0.2250**	**0.1751**	**0.3977**	**0.5228**	**0.3321**
ProGen BLOSUM	Train	0.1713	0.1570	0.3302	0.3995	0.2291
	Train + Val	0.1740	0.1591	0.3281	0.4399	0.2313
	Train + Val + Test	0.1565	0.1478	0.2765	0.4526	0.2039
ProGen CE	Train	0.1924	0.1711	0.3670	0.4759	0.2509
	Train + Val	0.1887	0.1704	0.3642	0.4710	0.2473
	Train + Val + Test	0.1731	0.1598	0.3090	0.4201	0.2276

Table 4.13 Zero-shot effect of augmenting DARKIN-Interact with synthetic phosphosites from *ProGen2–Phospho* using either loss function (Cross Entropy, CE, and BLOSUM) and three kinase sets (Train, Train + Val, Train + Val + Test).

Augmentation Method	**Kinase Set**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
–	–	**0.2078**	0.1669	**0.3501**	**0.5035**	**0.3109**
ProGen BLOSUM	Train	0.1997	0.1612	0.3359	0.4936	0.3046
	Train + Val	0.1782	0.1591	0.2998	0.4392	0.2866
	Train + Val + Test	0.1506	0.1393	0.2609	0.3564	0.2469
ProGen CE	Train	0.1959	**0.1676**	0.3218	0.4760	0.3013
	Train + Val	0.1282	0.1117	0.2461	0.3239	0.2198
	Train + Val + Test	0.1397	0.1202	0.2298	0.3232	0.2282

4.5.2 Incorporating ProGen2-Phospho Generated Phosphosites into the DARKIN Dataset

After fine-tuning, we used the *ProGen2–Phospho* models to generate 100 phosphosite sequences for each kinase. These sequences were then filtered to remove invalid or low-quality samples. One of the key filtering steps involved the use of a phosphorylation predictor model based on ESM1B, (detailed in 3.5.1). We retained only those generated phosphosites that received a predicted phosphorylation probability greater than 90%. As the generation process was performed for all kinases, including the unseen test kinases from the DARKIN dataset, we were able to explore the effect of adding kinases from different splits into the training set.

Contrary to expectations, we observed a decrease in Macro AP following the application of the augmentation for both approaches (refer to Table 4.12 for DARKIN-FT and Table 4.13 for DARKIN-Interact). This decline is more apparent in the detailed attribute-level results, presented in Appendix Tables A.11 and A.12 for DARKIN-

FT and DARKIN-Interact, respectively.

We interpret this outcome in three ways. First, the inclusion of synthetic examples with potentially low confidence may have introduced noise into the training set, hindering the model's ability to learn effectively. These synthetic samples may have disrupted the modeling process, leading to confusion that ultimately degraded test-time performance. Second, we hypothesize that the increased diversity of training samples caused the model to shift its focus from memorizing protein-specific patterns to attempting to generalize across a broader structural space. While this could be beneficial in theory, it may have diluted the model's ability to specialize in certain kinases, resulting in reduced performance. Since Macro AP treats all kinase classes equally, strong performance on a few specific kinases can improve the overall score. A redistribution of attention across many kinases, without learning them effectively, may therefore lead to a drop in this metric. Lastly, the quality and biological plausibility of the synthetic phosphosites generated by *ProGen2–Phospho* inevitably affect the utility of the augmented data. While current performance is limited, we believe that continued improvements in protein sequence generation models will make them valuable for data augmentation pipelines in phosphosite–kinase prediction tasks.

4.5.3 KSSA Evaluation on DARKIN

Before using the KSSA to label unlabeled phosphosite sequences obtained from PhosphoSitePlus, we first evaluated the performance of this method on the DARKIN dataset to benchmark it against the models developed in this work.

One important distinction of the KSSA method is that it separately handles S/T kinases and Y kinases. Moreover, predictions must be made using only the kinase list originally included in the KSSA. The two versions of the KSSA, developed in studies by Johnson et al. (2023) and Yaron-Barir et al. (2024), restrict scoring such that S/T phosphosites are evaluated only with S/T kinases, and similarly for Y phosphosites. To perform a fair comparison, we excluded any kinases not present in the KSSA from the DARKIN dataset. The resulting subsets were divided into two separate datasets: DARKIN-KL-ST and DARKIN-KL-Y. Figure 4.3 illustrates the difference in data quantity resulting from this split. The only kinase that is not shared with the DARKIN test kinases is Q38SD2, which belongs to the serine/threonine (S/T) kinase family.

To maintain consistency, the ESM1B model (the best-performing model from Table 4.1) was evaluated separately on DARKIN-KL-ST and DARKIN-KL-Y. Predictions were scored using both models and the KSSA. To assess KSSA predictions, the

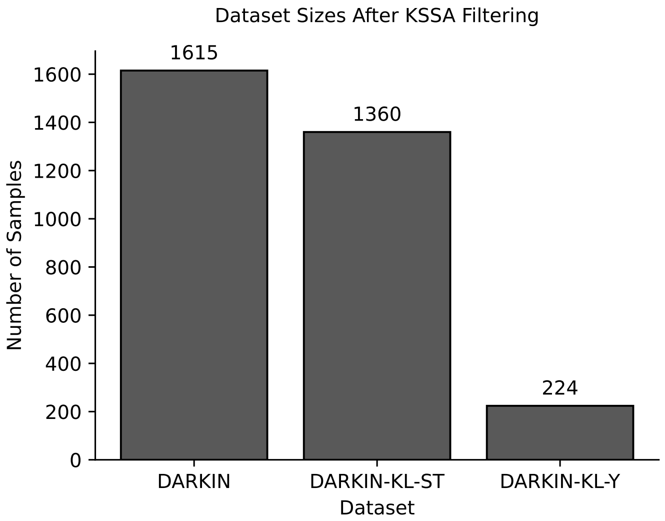

Figure 4.3 Dataset sizes before and after KSSA filtering, with serine/threonine kinases (S/T) separated from tyrosine kinases (Y).

Table 4.14 Performance of DARKIN-FT, DARKIN-Interact, and Kinase Substrate Specificity Atlas (KSSA) on the filtered DARKIN-KL-ST and DARKIN-KL-Y subsets.

Kinase Type	**Method**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
S/T	DARKIN-FT	**0.2240**	0.1760	0.3410	0.4700	0.3101
	DARKIN-Interact	0.2073	0.1772	0.3339	0.4666	0.3097
	KSSA	0.1600	**0.2320**	**0.4521**	**0.5908**	**0.3870**
Y	DARKIN-FT	**0.3810**	0.1800	0.7730	1.0000	0.4915
	DARKIN-Interact	0.3389	0.1342	0.6991	1.0000	0.4259
	KSSA	0.1259	**0.4213**	**0.8056**	1.0000	**0.6338**

log-scores produced were treated as logits and evaluated with ranking-based metrics like AP and accuracy.

Table 4.14 compares the performance of the KSSA with the compatibility-based and binary classification models. The KSSA achieved lower AP scores than the models introduced in this work, but surprisingly delivered better Accuracy. This discrepancy likely stems from the reduced complexity of the problem: the KSSA operates over smaller candidate sets, making the task inherently easier. Nevertheless, even when evaluating KSSA on these smaller datasets, the models developed in this work, applied to the more challenging standard DARKIN dataset, demonstrated superior performance in terms of AP, as shown in Table 4.1 and 4.7. Despite this, we opted to use the KSSA to label the unlabeled phosphosites due to its experimental knowledge and demonstrated strength in the phosphosite-kinase association problem.

Table 4.15 Impact of KSSA-labeled unlabeled phosphosite sequences on DARKIN-FT and DARKIN-Interact.

Model	Augmentation Method	Kinase Set	Macro AP	Top 1 Acc	Top 3 Acc	Top 5 Acc	Phosphosite AP
DARKIN-FT	–	–	**0.2250**	**0.1751**	**0.3977**	**0.5228**	**0.3321**
	KSSA	Train (Filtered)	0.1676	0.1739	0.3522	0.4957	0.3167
DARKIN-Interact	–	–	**0.2078**	0.1669	0.3501	0.5035	0.3109
	KSSA	Train (Filtered)	0.1993	**0.2008**	**0.3925**	**0.5283**	**0.3481**

4.5.4 Augmenting DARKIN with KSSA–Labeled Unlabeled Data

We began by filtering the unlabeled phosphosite sequences obtained from PhosphoSitePlus to remove those that did not meet basic criteria, such as having a central residue other than S, T, or Y or containing invalid amino acids. This filtering reduced the unlabeled dataset size from 379.029 to 311.081 sequences. The KSSA's integrated data preprocessing tool performed this filtering.

The remaining sequences were grouped based on the central residue and scored using the appropriate kinase subset from the intersection of the KSSA and DARKIN kinase classes. Unlike the *ProGen2–Phospho* experiments, we did not perform a special split-based selection for the kinase subsets. The primary concern is that, during the construction of the KSSA model, some phosphosite–kinase interactions from the test set were included in its training, leading to potential data leakage if not properly controlled. This issue was not present in *ProGen2–Phospho*, which is why we were able to explore various split generations in that setting.

For each phosphosite, the top two kinases with scores above the 99th percentile were selected, as shown in Figure 3.7. These multilabel predictions were compatible with the structure of the DARKIN dataset, which already contains multilabel examples. During training, these multilabel examples are decomposed into multiple rows, allowing the model to see diverse pairings and label combinations.

Table 4.15 reports the performance of the compatibility-based and binary classification models after augmenting DARKIN with kinase-labeled synthetic data. The corresponding attribute-level results are also provided in Appendix Table A.13. These results show that, although we significantly increased the size of the dataset by labeling previously unlabeled data using a model that had already demonstrated acceptable performance (see Table 4.14), this augmentation did not improve overall model performance.

While the decrease was less pronounced in the binary classification approach, the results once again highlight the importance of learning from real, high-quality data. Unlike the phosphosites generated by *ProGen2–Phospho*, the samples used in this augmentation were real phosphosites, so we expected to observe a more notable performance gain. We believe the limited improvement may stem from the fact that learning from a much larger and more diverse dataset makes it harder for the model to focus on key discriminative patterns. When trained on kinase classes with only a few phosphosite examples, the model tends to focus more effectively on specific patterns. While this behavior could be interpreted as overfitting, it also reflects the model's ability to capture structural similarities between phosphosites and kinases. Although increasing the number of training samples allows the model to learn from a broader range of examples, it appears to struggle more with unseen data. Additionally, due to the similarities among embeddings produced by protein language models, the new information introduced by the augmented data may not be sufficiently distinguishable to the model, limiting its contribution to training.

4.5.5 Homologous Sequences and Integration into the DARKIN Dataset

The key distinction between the homologous sequences added to the dataset and the synthetic sequences generated by *ProGen2–Phospho* or weakly labeled unlabeled data lies in their origin: homologous sequences are derived from alignments with real phosphosite sequences. As a result, they closely resemble actual phosphosites but contain small variations. This makes the added data both biologically plausible and high-quality, as it primarily reflects minor sequence differences while preserving core phosphosite characteristics. Additionally, this augmentation strategy increases the number of samples for certain kinase classes that originally had very few training instances. Introducing homologous sequences with subtle modifications encourages the model to learn discriminative patterns for these underrepresented classes better.

As shown in Table 4.16, among the three data augmentation strategies tested, using homologous sequences produced the most effective results. In the binary classification setting, this method led to an improvement in Macro AP. While kinase-level performance metrics improved, we observed a drop in phosphosite-level accuracy and AP scores, suggesting that the model became better at distinguishing among kinase classes but struggled more in retrieving the correct kinase for a given phosphosite. For the DARKIN-FT approach, even though there was a minor decline in overall performance, the Macro AP stayed above 0.20. Appendix Table A.14 further illustrates the effect of adding homologous sequences on the attribute-level results.

Table 4.16 Effect of augmentation with homologous sequences on DARKIN-FT and DARKIN-Interact.

Model	**Augmentation Method**	**Kinase Set**	**Macro AP**	**Top 1 Acc**	**Top 3 Acc**	**Top 5 Acc**	**Phosphosite AP**
DARKIN-FT	–	–	**0.2250**	**0.1751**	**0.3977**	**0.5228**	**0.3321**
	Homologous	Train	0.2060	0.1648	0.3614	0.4872	0.2687
DARKIN-Interact	–	–	0.2078	**0.1669**	**0.3501**	**0.5035**	**0.3109**
	Homologous	Train	**0.2131**	0.1640	0.3048	0.4165	0.2901

Compared to the other two approaches, the synthetic data produced via homologous sequences exhibited stronger biological relevance and greater similarity to real data, resulting in better model performance. In contrast, *ProGen2–Phospho* generates entirely new phosphosite sequences, and the KSSA strategy relies on labeling real phosphosites using a predictive model, which may or may not assign biologically accurate kinase labels. The homologous sequence approach is conceptually similar to data augmentation techniques in natural language processing, where parts of the input are randomly altered to improve generalization. Overall, this method yielded more stable and reliable improvements compared to the other augmentation strategies.

Chapter 5

CONCLUSION

While it is possible to detect phosphosites through high-througput experiments, it is a challenge to figure out which kinase is the catalyzing kinase. Therefore, computational models to associate phosphosites with their cognate kinases is useful. The available data for the kinases-phosphosites association pairs show that there are many kinases that are understudied, with no or few phosphosites known from them. This problem was casted as a zero-shot learning problem earlier in Deznabi et al., 2020. This thesis extends this set up to few-shot learning as well. In the zero-few shot learning set ups, we developed new deep learning models and tested several data augmentation methods designed for this specific biological problem to deal with the lack of labeled data for a large number of classes.

This work has three main contributions. First, we built two new zero-shot learning models, DARKIN-FT and DARKIN-Interact. DARKIN-FT improves on older models Deznabi et al., 2020; Sunar et al., 2024 by finetuning the protein language model embeddings for phosphosites from end to end. DARKIN-Interact changes the problem to a binary classification task to assess the compatibility of kinase and phosphosite interaction. Both models performed much better than the baselines on the DARKIN benchmark. Second, we designed and tested three data augmentation strategies: (i) generating new phosphosites for a given kinase using a finetuned ProGen2 model, (ii) labeling unlabeled data using scores from the experimental Kinase Substrate Specificity Atlas (KSSA), and (iii) creating new training examples using homologous sequences. Third, through many experiments and ablation studies, we analyzed the strengths and weaknesses of the models and augmentation methods, providing valuable insights for this research area.

Experiments in this study showed a few important things. The DARKIN-FT model proved that finetuning phosphosite embeddings for this task works much better than using static, pre-trained embeddings. This illustrates the importance of task-specific representations. Interestingly, models that started with random weights often learned the task better than models that were pre-trained on general protein data. The second main model of this study, DARKIN-Interact, showed that using cross-sequence attention to compare a kinase and a phosphosite directly is a very effective strategy, with performance close to the DARKIN-FT model. This demonstrates the power of modern transformers to find the small but important patterns in these sequence pairs. Ablation studies also showed that while a learned compatibility matrix $\mathbf{W}$ works well, it can be replaced with simpler projection layers, and its performance can be slightly improved with regularization techniques like spectral norm regularization.

Data augmentation methodologies yielded mixed results. The strategies of generating entirely new phosphosites with ProGen2 or labeling data with KSSA predictions made the models perform worse. Adding this synthetic or weakly-labeled data, even after filtering, introduced noise that confused the models. This prevented them from learning the specific features needed to distinguish between very similar kinases. In contrast, using homologous sequences for augmentation proved to be the most successful strategy. These sequences are very similar to real ones and introduced useful variety into the training data, improving the Macro AP for the DARKIN-Interact model. This suggests that for this problem, the most effective data augmentation methods are those that closely adhere to real biological examples, such as making meaningful minor changes to known positives.

While the results are promising, several limitations remain. The quality of the synthetic data is still a challenge, as today's generative models may not fully capture what makes a phosphosite functional. Also, all models still struggle to distinguish between kinases from the same family or group since their sequences are often very similar. This suggests that sequence data alone might not be enough to solve this problem completely.

Looking forward, this research opens up several possibilities for future work. Future efforts should focus on improving the quality of generated phosphosites by using models that consider sequence and 3D structure to create more realistic synthetic data. Adding 3D structural information to current sequence-only models, using graph neural networks or geometric deep learning, could provide the extra information needed to distinguish closely related kinases. We could also explore more advanced data augmentation techniques, such as contrastive learning, which may help the model learn more effective representations. A self-supervised task explicitly designed for kinase-substrate pairs could also be very effective. Finally, given how hard it is to tell kinases apart at the family level, a hierarchical model that works in steps, such as first predicting the group, then the family, and finally the specific kinase, might perform better, as could a multi-task model that also learns to predict other site properties.

In summary, this thesis presents a deep learning framework for addressing the dark kinome problem and provides a practical examination of how data augmentation is applied to low-data biological tasks. By showing the promise of both new model architectures and smart, biologically-informed data augmentation, this thesis helps pave the way for future tools that can speed up the mapping of phosphorylation networks and, in the long run, improve the understanding of human health.

BIBLIOGRAPHY

Ardito, Fatima, Michele Giuliani, Donatella Perrone, Giuseppe Troiano, and Lorenzo Lo Muzio (2017). “The crucial role of protein phosphorylation in cell signalling and its use as targeted therapy”. In: *International Journal of Molecular Medicine* 40.2, pp. 271–280. DOI: 10.3892/ijmm.2017.3036.

Betts, MatthewJ. and Robert B. Russell (2003). “Amino Acid Properties and Consequences of Substitutions”. In: *Bioinformatics for Geneticists.* Ed. by Michael R. Barnes and Ian C. Gray. John Wiley & Sons, pp. 289–316. DOI: 10.1002/0470867302.ch14.

Blom, Nikolaj, Steen Gammeltoft, and Søren Brunak (1999). “Sequence and structure-based prediction of eukaryotic protein phosphorylation sites1”. In: *Journal of molecular biology* 294.5, pp. 1351–1362.

Brown, Tom et al. (2020). “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems.* Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., pp. 1877–1901. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Chen, Jingbang et al. (2023). *Mixup-Augmented Meta-Learning for Sample-Efficient Fine-Tuning of Protein Simulators.* URL: https://arxiv.org/abs/2308.15116.

Cock, Peter JA et al. (2009). “Biopython: freely available Python tools for computational molecular biology and bioinformatics”. In: *Bioinformatics* 25.11, pp. 1422–1423.

Cohen, Philip (2002). “Protein kinases — the major drug targets of the twenty-first century?” In: *Nature Reviews Drug Discovery* 1.4, pp. 309–315. DOI: 10.1038/nrd773.

Das, Payel et al. (2018). *PepCVAE: Semi-Supervised Targeted Design of Antimicrobial Peptide Sequences.* URL: https://arxiv.org/abs/1810.07743.

Deguchi, Teppei, Yoichi Kurumida, Shinji Iida, Kaito Kobayashi, and Yutaka Saito (2025). “Data-efficient protein mutational effect prediction with weak supervision by molecular simulation and protein language models”. In: *bioRxiv.* DOI: 10.1101/2025.04.08.647800. URL: https://www.biorxiv.org/content/early/2025/04/15/2025.04.08.647800.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* URL: https://arxiv.org/abs/1810.04805.

Deznabi, Iman, Busra Arabaci, Mehmet Koyutürk, and Oznur Tastan (2020). "DeepKinZero: zero-shot learning for predicting kinase–phosphosite associations involving understudied kinases". In: *Bioinformatics* 36.12, pp. 3652–3661.

Elnaggar, Ahmed et al. (2021). "ProtTrans: towards cracking the language of lifes code through self-supervised deep learning and high performance computing". In: *IEEE transactions on pattern analysis and machine intelligence* 44.10, pp. 7112–7127.

Ferruz, Noelia, Sabrina Schmidt, and Birte Höcker (2022). "ProtGPT2 is a deep unsupervised language model for protein design". In: *Nature Communications* 13, p. 4348. DOI: 10.1038/s41467-022-32007-7.

Gao, Jianjiong, Jay J Thelen, A Keith Dunker, and Dong Xu (2010). "Musite: a tool for global prediction of general and kinase-specific phosphorylation sites". In: *Molecular & Cellular Proteomics*, mcp–M110.

Gomez, Aidan N., Ivan Zhang, Kevin Swersky, Yarin Gal, and Geoffrey E. Hinton (2019). "Learning Sparse Networks Using Targeted Dropout". In: *CoRR* abs/1905.13678. URL: http://arxiv.org/abs/1905.13678.

Greener, Joe G., Lewis Moffat, and David T. Jones (2018). "Design of metalloproteins and novel protein folds using variational autoencoders". In: *Scientific Reports* 8, p. 16189. DOI: 10.1038/s41598-018-34533-1.

Guo, Lei et al. (2021). "DeepPSP: A Global–Local Information-Based Deep Neural Network for the Prediction of Protein Phosphorylation Sites". In: *Journal of Proteome Research* 20.1. PMID: 33241931, pp. 346–356. DOI: 10.1021/acs.jproteome.0c00431. URL: https://doi.org/10.1021/acs.jproteome.0c00431.

Hayes, Tyler et al. (2025). "Simulating 500 million years of evolution with a language model". In: *Science (New York, N.Y.)* 387.6736, pp. 850–858. DOI: 10.1126/science.ads0018.

Henikoff, Steven and Jorja G. Henikoff (1992). "Amino acid substitution matrices from protein blocks". In: *Proceedings of the National Academy of Sciences of the United States of America* 89.22, pp. 10915–10919. DOI: 10.1073/pnas.89.22.10915. URL: https://doi.org/10.1073/pnas.89.22.10915.

Hie, Brian L. and Kevin K. Yang (2022). "Adaptive machine learning for protein engineering". In: *Current Opinion in Structural Biology* 72, pp. 145–152. DOI: https://doi.org/10.1016/j.sbi.2021.11.002. URL: https://www.sciencedirect.com/science/article/pii/S0959440X21001457.

Horn, Heiko et al. (2014). "KinomeXplorer: an integrated platform for kinome biology studies". In: *Nature methods* 11.6, p. 603.

Hornbeck, Peter V et al. (2012). "PhosphoSitePlus: a comprehensive resource for investigating the structure and function of experimentally determined post-translational modifications in man and mouse". In: *Nucleic acids research* 40.D1, pp. D261–D270.

Hu, Edward J et al. (2022). "LoRA: Low-Rank Adaptation of Large Language Models". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=nZeVKeeFYf9.

Hunter, Tony (1995). "Protein kinases and phosphatases: the yin and yang of protein phosphorylation and signaling". In: *Cell* 80.2, pp. 225–236.

Johnson, Jared L., Tomer M. Yaron, Emily M. Huntsman, et al. (2023). "An atlas of substrate specificities for the human serine/threonine kinome". In: *Nature* 613, pp. 759–766. DOI: 10.1038/s41586-022-05575-3.

Jumper, John et al. (2021). "Highly accurate protein structure prediction with AlphaFold". In: *Nature* 596.7873, pp. 583–589.

Katoh, Kazutaka and Daron M Standley (2013). "MAFFT multiple sequence alignment software version 7: improvements in performance and usability". In: *Molecular Biology and Evolution* 30.4, pp. 772–780. DOI: 10.1093/molbev/mst010.

KinBase (2024). http://kinase.com/human/kinome/phylogeny.html. [Online; accessed 1-November-2024].

Koenig, Matthias and Niels Grabe (2004). "Highly specific prediction of phosphorylation sites in proteins". In: *Bioinformatics* 20.18, pp. 3620–3627.

Kucera, Tim, Matteo Togninalli, and Laetitia Meng-Papaxanthos (2022). "Conditional generative modeling for de novo protein design with hierarchical functions". In: *Bioinformatics* 38.13, pp. 3454–3461. DOI: 10.1093/bioinformatics/btac353. URL: https://doi.org/10.1093/bioinformatics/btac353.

Kuru, Nursena et al. (2022). "PHACT: Phylogeny-Aware Computing of Tolerance for Missense Mutations". In: *Molecular Biology and Evolution* 39.6, msac114. DOI: 10.1093/molbev/msac114.

Li, Tingting, Fei Li, and Xuegong Zhang (2008). "Prediction of kinase-specific phosphorylation sites with sequence features by a log-odds ratio approach". In: *Proteins: Structure, Function, and Bioinformatics* 70.2, pp. 404–414.

Lin, Zeming et al. (2023). "Evolutionary-scale prediction of atomic-level protein structure with a language model". In: *Science* 379.6637, pp. 1123–1130. DOI: 10.1126/science.ade2574.

Liu, Yinhan et al. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. URL: https://arxiv.org/abs/1907.11692.

Luo, Fenglin, Minghui Wang, Yu Liu, Xing-Ming Zhao, and Ao Li (2019). "DeepPhos: prediction of protein phosphorylation sites with deep learning". In: *Bioinformatics* 35.16, pp. 2766–2773. DOI: 10.1093/bioinformatics/bty1051. URL: https://doi.org/10.1093/bioinformatics/bty1051.

Lv, Liuzhenghao et al. (2025). “ProLLaMA: A Protein Large Language Model for Multi-Task Protein Language Processing”. In: *IEEE Transactions on Artificial Intelligence*, pp. 1–12. DOI: 10.1109/TAI.2025.3564914.

Madani, Ali et al. (2023). “Large language models generate functional protein sequences across diverse families”. In: *Nature Biotechnology* 41, pp. 1099–1106. DOI: 10.1038/s41587-022-01618-2.

Manning, Gerard, David B Whyte, Ricardo Martinez, Tony Hunter, and Sucha Sudarsanam (2002). “The protein kinase complement of the human genome”. In: *Science* 298.5600, pp. 1912–1934.

Moret, Nienke et al. (2020). “A resource for exploring the understudied human kinome for research and therapeutic opportunities”. In: *BioRxiv*, pp. 2020–04.

Needham, Elise J, Benjamin L Parker, Timur Burykin, David E James, and Sean J Humphrey (2019). “Illuminating the dark phosphoproteome”. In: *Sci. Signal.* 12.565, eaau8645.

Nijkamp, Erik, Justin A. Ruffolo, Ethan N. Weinstein, Neel Naik, and Ali Madani (2023). “ProGen2: Exploring the boundaries of protein language models”. In: *Cell Systems* 14.11, 968–978.e3. DOI: 10.1016/j.cels.2023.10.002.

Obenauer, John C., Lewis C. Cantley, and Michael B. Yaffe (2003). “Scansite 2.0: Proteome-wide prediction of cell signaling interactions using short sequence motifs”. In: *Nucleic Acids Research* 31.13, pp. 3635–3641. DOI: 10.1093/nar/gkg584.

Patrick, Ralph, Kim-Anh Lê Cao, Bostjan Kobe, and Mikael Bodén (2014). “PhosphoPICK: modelling cellular context to map kinase-substrate phosphorylation events”. In: *Bioinformatics* 31.3, pp. 382–389.

Perez, Luis and Jason Wang (2017). “The Effectiveness of Data Augmentation in Image Classification using Deep Learning”. In: *CoRR* abs/1712.04621. URL: http://arxiv.org/abs/1712.04621.

Qin, Gui-Min, Rui-Yi Li, and Xing-Ming Zhao (2016). “PhosD: inferring kinase–substrate interactions based on protein domains”. In: *Bioinformatics* 33.8, pp. 1197–1204.

Radford, Alec and Karthik Narasimhan (2018). “Improving Language Understanding by Generative Pre-Training”. In: URL: https://api.semanticscholar.org/CorpusID:49313245.

Radford, Alec, Jeffrey Wu, et al. (2019). “Language Models are Unsupervised Multitask Learners”. In: *OpenAI.*

Raffel, Colin et al. (2020). “Exploring the limits of transfer learning with a unified text-to-text transformer”. In: *J. Mach. Learn. Res.* 21.1.

Repecka, Donatas, Vykintas Jauniskis, Lukas Karpus, et al. (2021). “Expanding functional protein sequence spaces using generative adversarial networks”. In: *Nature Machine Intelligence* 3, pp. 324–333. DOI: 10.1038/s42256-021-00310-5. URL: https://doi.org/10.1038/s42256-021-00310-5.

Riesselman, Adam J., Jake B. Ingraham, and Debora S. Marks (2018). “Deep generative models of genetic variation capture the effects of mutations”. In: *Nature Methods* 15.10, pp. 816–822. DOI: 10.1038/s41592-018-0138-4.

Rives, Alexander et al. (2021). “Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences”. In: *Proceedings of the National Academy of Sciences* 118.15. DOI: 10.1073/pnas.2016239118. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2016239118.

Saunders, Neil FW, Ross I Brinkworth, Thomas Huber, Bruce E Kemp, and Bostjan Kobe (2008). “Predikin and PredikinDB: a computational framework for the prediction of protein kinase peptide specificity and an associated database of phosphorylation sites”. In: *BMC bioinformatics* 9.1, p. 245.

Song, Jiangning et al. (2017). “PhosphoPredict: A bioinformatics tool for prediction of human kinase-specific phosphorylation substrates and sites by integrating heterogeneous feature selection”. In: *Scientific Reports* 7.1, p. 6862.

Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov (2014). “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”. In: *Journal of Machine Learning Research* 15, pp. 1929–1958.

Sunar, Emine Ayşe, Zeynep Işık, Mert Pekey, Ramazan Gokberk Cinbis, and Oznur Tastan (2024). “DARKIN: A zero-shot classification benchmark and an evaluation of protein language models”. In: *ICLR 2024 Workshop on Machine Learning for Genomics Explorations.*

The UniProt Consortium (2021). “UniProt: the universal protein knowledgebase in 2021”. In: *Nucleic Acids Research* 49, pp. D480–D489. DOI: 10.1093/nar/gkaa1100.

Vasconcelos, Cristina Nader and Bárbara Nader Vasconcelos (2017). “Increasing Deep Learning Melanoma Classification by Classical And Expert Knowledge Based Image Transforms”. In: *CoRR* abs/1702.07025. URL: http://arxiv.org/abs/1702.07025.

Vaswani, Ashish et al. (2017). “Attention is all you need”. In: *Advances in neural information processing systems* 30.

Vella, Viviana, Georgios Giamas, and Angeliki Ditsiou (2022). “Diving into the dark kinome: lessons learned from LMTK3”. In: *Cancer Gene Therapy* 29.8, pp. 1077–1079.

Walsh, Gary and Roy Jefferis (2006). “Post-translational modifications in the context of therapeutic proteins”. In: *Nature Biotechnology* 24.10, pp. 1241–1252. DOI: 10.1038/nbt1252.

Wang, Duolin et al. (2017). “MusiteDeep: a deep-learning framework for general and kinase-specific phosphorylation site prediction”. In: *Bioinformatics* 33.24, pp. 3909–3916.

Wang, Xun et al. (2022). “TransPhos: A Deep-Learning Model for General Phosphorylation Site Prediction Based on Transformer-Encoder Architecture”. In:

International Journal of Molecular Sciences 23.8. DOI: 10.3390/ijms23084263. URL: https://www.mdpi.com/1422-0067/23/8/4263.

Wei, Jason and Kai Zou (2019). “EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks”. 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, pp. 6383–6389. URL: https://www.aclweb.org/anthology/D19-1670.

Wong, Yung-Hao et al. (2007). “KinasePhos 2.0: a web server for identifying protein kinase-specific phosphorylation sites based on sequences and coupling patterns”. In: *Nucleic acids research* 35.suppl_2, W588–W594.

Xian, Yongqin, Bernt Schiele, and Zeynep Akata (2017). “Zero-shot learning-the good, the bad and the ugly”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4582–4591.

Xue, Yu, Zexian Liu, et al. (2010). “GPS 2.1: enhanced prediction of kinase-specific phosphorylation sites with an algorithm of motif length selection”. In: *Protein Engineering, Design & Selection* 24.3, pp. 255–260.

Xue, Yu, Jing Ren, et al. (2008). “GPS 2.0, a tool to predict kinase-specific phosphorylation sites in hierarchy”. In: *Molecular & Cellular Proteomics* 7.9, pp. 1598–1608. DOI: 10.1074/mcp.M700574-MCP200.

Yaffe, Michael B et al. (2001). “A motif-based profile scanning approach for genome-wide prediction of signaling pathways”. In: *Nature biotechnology* 19.4, p. 348.

Yaron-Barir, Tomer M., Brian A. Joughin, Emily M. Huntsman, et al. (2024). “The intrinsic substrate specificity of the human tyrosine kinome”. In: *Nature* 629.8014, pp. 1174–1181. DOI: 10.1038/s41586-024-07407-y.

Yoshida, Yuichi and Takeru Miyato (2017). “Spectral Norm Regularization for Improving the Generalizability of Deep Learning”. In: *arXiv preprint arXiv:1705.10941*. URL: https://arxiv.org/abs/1705.10941.

Zaheer, Manzil, Guru Guruganesh, Aviral Dubey, and et al. (2020). “Big Bird: Transformers for Longer Sequences”. In: *Advances in Neural Information Processing Systems* 33, pp. 17283–17297.

Zhang, Hongyi, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz (2018). “mixup: Beyond Empirical Risk Minimization”. In: *6th International Conference on Learning Representations (ICLR)*. URL: https://openreview.net/forum?id=r1Ddp1-Rb.

Zhou, Zhongliang, Wayland Yeung, Nathan Gravel, et al. (2023). “Phosformer: an explainable transformer model for protein kinase-specific phosphorylation predictions”. In: *Bioinformatics* 39.2. DOI: 10.1093/bioinformatics/btad046. URL: https://doi.org/10.1093/bioinformatics/btad046.

Zhou, Zhongliang, Wayland Yeung, Saber Soleymani, et al. (2024). “Using explainable machine learning to uncover the kinase–substrate interaction landscape”.

In: *Bioinformatics* 40.2. DOI: 10.1093/bioinformatics/btae033. URL: https://doi.org/10.1093/bioinformatics/btae033.

Zhu, Yiheng et al. (2024). “Generative AI for Controllable Protein Sequence Design: A Survey”. In: *CoRR* abs/2402.10516. URL: https://doi.org/10.48550/arXiv.2402.10516.

Zou, Liang et al. (2013). “PKIS: computational identification of protein kinases for experimentally discovered protein phosphorylation sites”. In: *BMC bioinformatics* 14.1, p. 247.

APPENDIX A

Table A.1 Zero-shot performance of four pLM backbones under three strategies: BZSM (DARKIN benchmark), DARKIN-FT, and DARKIN-FT–PT (the phosphosite encoder is initialized with pretrained weights), evaluated with clustered attribute-level metrics.

Model	Method	Family AP	Family Acc	Group AP	Group Acc	F.grain AP	F.grain Acc
ESM1B	BZSM	0.2198	0.2854	0.4096	0.5620	0.1957	0.2192
	DARKIN-FT	0.2945	0.3180	0.4762	0.6065	0.2558	0.2536
	DARKIN-FT - PT	0.2398	0.3041	0.4182	0.5962	0.2072	0.2022
ProtT5XL	BZSM	0.2217	0.2779	0.3840	0.5532	0.1890	0.1944
	DARKIN-FT	0.2243	0.2430	0.3603	0.4957	0.1934	0.1664
	DARKIN-FT - PT	0.2207	0.2906	0.4001	0.6018	0.1840	0.1492
ESM2-650M	BZSM	0.2159	0.2857	0.3909	0.5473	0.1867	0.1893
	DARKIN-FT	0.2621	0.3258	0.4324	0.5870	0.2238	0.263
	DARKIN-FT - PT	0.2443	0.3069	0.3996	0.5686	0.2128	0.2687
ESM2-150M	BZSM	0.1758	0.2678	0.3595	0.5245	0.1500	0.2138
	DARKIN-FT	0.2449	0.2857	0.4160	0.5674	0.2131	0.2173
	DARKIN-FT - PT	0.2195	0.2496	0.3749	0.5643	0.1914	0.1952

Table A.2 Few-shot (5-shot) performance of four pLM backbones under three strategies: BFSM (DARKIN benchmark BZSM, renamed for few-shot, DARKIN-FT, and DARKIN-FT–PT (the phosphosite encoder is initialized with pretrained weights), evaluated with clustered attribute-level metrics.

Model	Method	Family AP	Family Acc	Group AP	Group Acc	F.grain AP	F.grain Acc
ESM1B	BFSM	0.2374	0.3115	0.4277	0.5804	0.2122	0.2027
	DARKIN-FT	0.3238	0.3375	0.4956	0.6285	0.2844	0.2516
	DARKIN-FT - PT	0.2571	0.3068	0.4222	0.6073	0.2229	0.2082
ProtT5XL	BFSM	0.2316	0.2902	0.4087	0.5725	0.2012	0.1853
	DARKIN-FT	0.2404	0.2784	0.3751	0.5126	0.2047	0.1956
	DARKIN-FT - PT	0.2374	0.3020	0.4186	0.6246	0.1999	0.1569
ESM2-650M	BFSM	0.2408	0.2957	0.4155	0.5576	0.2129	0.1845
	DARKIN-FT	0.3076	0.3446	0.4825	0.6301	0.2716	0.2650
	DARKIN-FT - PT	0.2452	0.2800	0.4206	0.6073	0.2158	0.2263
ESM2-150M	BFSM	0.1858	0.2792	0.3832	0.5323	0.1639	0.2019
	DARKIN-FT	0.2722	0.3115	0.4481	0.5962	0.2424	0.2484
	DARKIN-FT - PT	0.2323	0.2516	0.3954	0.5757	0.1976	0.1837

Table A.3 Zero-shot performance of DARKIN-Interact with different pLM encoders fine-tuned via LoRA, evaluated with clustered attribute-level metrics.

Model	n	**Layers**	**Family AP**	**Family Acc**	**Group AP**	**Group Acc**	**F.grain AP**	**F.grain Acc**
ESM1B	4	LoRA	0.2431	0.2468	0.4240	0.5785	0.2089	0.1541
ESM2-650M	4	LoRA	0.2198	0.2659	0.4167	0.5658	0.1901	0.1839
ESM2-150M	4	LoRA	0.2001	0.1846	0.4045	0.5382	0.1793	0.1301
ESM2-35M	4	LoRA	0.1704	0.2906	0.3462	0.5530	0.1464	0.1365

Table A.4 Few-shot (5-shot) performance of DARKIN-Interact with different pLM encoders fine-tuned via LoRA, evaluated with clustered attribute-level metrics.

Model	n	**Layers**	**Family AP**	**Family Acc**	**Group AP**	**Group Acc**	**F.grain AP**	**F.grain Acc**
ESM1B	4	LoRA	0.2181	0.3194	0.3738	0.5410	0.1964	0.2224
ESM2-650M	4	LoRA	0.2315	0.3099	0.4177	0.5552	0.2115	0.2397
ESM2-150M	4	LoRA	0.1923	0.2839	0.3405	0.5063	0.1705	0.1940
ESM2-35M	4	LoRA	0.1566	0.2389	0.2984	0.4700	0.1431	0.1822

Table A.5 Effect of the number of negative samples per positive (n) on DARKIN-Interact's zero-shot performance. ESM1B encoder is utilized and fine-tuned via LoRA, evaluated with clustered attribute-level metrics.

Model	n	**Layers**	**Family AP**	**Family Acc**	**Group AP**	**Group Acc**	**F.grain AP**	**F.grain Acc**
ESM1B	1	LoRA	0.2337	0.3112	0.4278	0.5799	0.2076	0.2383
ESM1B	2	LoRA	0.2424	0.2857	0.4496	0.5792	0.2140	0.1754
ESM1B	4	LoRA	0.2431	0.2468	0.4240	0.5785	0.2089	0.1541
ESM1B	8	LoRA	0.2483	0.2510	0.4244	0.5657	0.2150	0.1846
ESM1B	12	LoRA	0.2417	0.2814	0.3837	0.5353	0.2115	0.1520

Table A.6 Effect of the number of negative samples per positive (n) on DARKIN-Interact's few-shot (5-shot) performance. ESM1B encoder is utilized and fine-tuned via LoRA, evaluated with clustered attribute-level metrics.

Model	n	**Layers**	**Family AP**	**Family Acc**	**Group AP**	**Group Acc**	**F.grain AP**	**F.grain Acc**
ESM1B	1	LoRA	0.2400	0.3225	0.4032	0.5662	0.2094	0.2303
ESM1B	2	LoRA	0.2232	0.3201	0.3712	0.5126	0.2052	0.1956
ESM1B	4	LoRA	0.2181	0.3194	0.3738	0.5410	0.1964	0.2224
ESM1B	8	LoRA	0.2324	0.3438	0.3736	0.5465	0.2115	0.2461
ESM1B	12	LoRA	0.2303	0.2918	0.3968	0.5165	0.2105	0.2232

Table A.7 Zero-shot impact of unfreezing the last l ESM1B encoder layers versus LoRA on DARKIN-Interact, evaluated with clustered attribute-level metrics.

Model	n	**Layers**	**Family AP**	**Family Acc**	**Group AP**	**Group Acc**	**F.grain AP**	**F.grain Acc**
ESM1B	4	LoRA	0.2431	0.2468	0.4240	0.5785	0.2089	0.1541
ESM1B	4	Last 1	0.1961	0.2723	0.3787	0.5870	0.1740	0.2079
ESM1B	4	Last 2	0.2536	0.2744	0.4526	0.5969	0.2234	0.2136
ESM1B	4	Last 4	0.2636	0.3048	0.4556	0.6011	0.2338	0.2001
ESM1B	4	Last 8	0.2594	0.2843	0.4462	0.6117	0.2320	0.2008
ESM1B	4	Last 12	0.2663	0.2744	0.4486	0.5948	0.2322	0.1959

Table A.8 Few-shot (5-shot) impact of unfreezing the last l ESM1B encoder layers versus LoRA on DARKIN-Interact, evaluated with clustered attribute-level metrics.

Model	n	Layers	Family AP	Family Acc	Group AP	Group Acc	F.grain AP	F.grain Acc
ESM1B	4	LoRA	0.2181	0.3194	0.3738	0.5410	0.1964	0.2224
ESM1B	4	Last 1	0.2055	0.3131	0.3559	0.5300	0.1871	0.2334
ESM1B	4	Last 2	0.2409	0.3351	0.3906	0.5363	0.2261	0.2429
ESM1B	4	Last 4	0.2406	0.3494	0.3871	0.5229	0.2289	0.2823
ESM1B	4	Last 8	0.2738	0.3691	0.4452	0.5923	0.2490	0.2594
ESM1B	4	Last 12	0.2802	0.3856	0.4606	0.6372	0.2616	0.2792

Table A.9 Ablation study of zero-shot DARKIN-FT exploring alternative compatibility projections, merging strategies (concatenation vs gated fusion), kinase-pooler designs, and optional attribute projections, evaluated with clustered attribute-level metrics.

Phosphosite Pooler	Kinase Pooler	Merging	Attribute Projection	Family AP	Family Acc	Group Ap	Group Acc	F.grain AP	F.grain Acc
Default Model Setting				0.2945	0.318	0.4762	0.6065	0.2558	0.2536
✓	–	–	–	0.2844	0.3097	0.4737	0.6259	0.2469	0.2482
✓	Single Stage	Concat	–	0.2945	0.3331	0.4719	0.6148	0.2514	0.2588
			✓	0.2725	0.3034	0.4673	0.6336	0.2376	0.2312
		Gated	–	0.2619	0.2998	0.4583	0.5813	0.2241	0.2093
			✓	0.2817	0.3161	0.4628	0.6308	0.2427	0.2503
	Two Stage	Concat	–	0.2843	0.3097	0.4714	0.6237	0.2416	0.2249
			✓	0.2824	0.3076	0.473	0.6174	0.2416	0.2383
		Gated	–	0.2789	0.3133	0.4531	0.6089	0.2416	0.2546
			✓	0.2726	0.3083	0.4583	0.6216	0.2316	0.2454

Table A.10 Regularization strategies for the compatibility matrix **W** in zero-shot DARKIN-FT, evaluated with clustered attribute-level metrics.

Regularization	Family AP	Family Acc	Group Ap	Group Acc	F.grain AP	F.grain Acc
L2 (Default)	0.2945	0.318	0.4762	0.6065	0.2558	0.2536
L2 + L1	0.2204	0.2935	0.4151	0.6089	0.1891	0.2334
L2 + Ranked Dropout	0.2957	0.326	0.4766	0.6138	0.2534	0.2715
L2 + Spectral Norm	0.2973	0.3267	0.4701	0.6138	0.2585	0.2575

Table A.11 Zero-shot effect of augmenting DARKIN-FT with synthetic phosphosites from *ProGen2–Phospho* using either loss function (Cross Entropy, CE, and BLOSUM) and three kinase sets (Train, Train + Val, Train + Val + Test), evaluated with clustered attribute-level metrics.

Augmentation Method	**Kinase Set**	**Family AP**	**Family Acc**	**Group Ap**	**Group Acc**	**F.grain AP**	**F.grain Acc**
–	–	0.2945	0.3180	0.4762	0.6065	0.2558	0.2536
ProGen BLOSUM	Train	0.2291	0.2807	0.4288	0.5693	0.1963	0.2242
	Train + Val	0.2313	0.2722	0.4149	0.5558	0.2000	0.2213
	Train + Val + Test	0.2039	0.2539	0.4067	0.5353	0.1800	0.1909
ProGen CE	Train	0.2509	0.2984	0.4333	0.5735	0.2182	0.2468
	Train + Val	0.2473	0.2977	0.4253	0.5672	0.2144	0.2376
	Train + Val + Test	0.2276	0.2638	0.4074	0.5495	0.1970	0.2015

Table A.12 Zero-shot effect of augmenting DARKIN-Interact with synthetic phosphosites from *ProGen2–Phospho* using either loss function (Cross Entropy, CE, and BLOSUM) and three kinase sets (Train, Train + Val, Train + Val + Test), evaluated with clustered attribute-level metrics.

Augmentation Method	**Kinase Set**	**Family AP**	**Family Acc**	**Group Ap**	**Group Acc**	**F.grain AP**	**F.grain Acc**
–	–	0.2663	0.2744	0.4486	0.5948	0.2322	0.1959
ProGen BLOSUM	Train	0.2541	0.2885	0.4351	0.5912	0.2264	0.1945
	Train + Val	0.2284	0.2730	0.3973	0.5834	0.2016	0.1832
	Train + Val + Test	0.1872	0.2425	0.3248	0.4583	0.1741	0.1711
ProGen CE	Train	0.2494	0.2829	0.4270	0.6011	0.2184	0.1888
	Train + Val	0.1649	0.2475	0.3469	0.5495	0.1426	0.1732
	Train + Val + Test	0.1840	0.2143	0.3270	0.4767	0.1610	0.1457

Table A.13 Impact of KSSA-labeled unlabeled phosphosite sequences on DARKIN-FT and DARKIN-Interact, evaluated with clustered attribute-level metrics.

Model	**Augmentation Method**	**Kinase Set**	**Family AP**	**Family Acc**	**Group Ap**	**Group Acc**	**F.grain AP**	**F.grain Acc**
DARKIN-FT	–	–	0.2945	0.318	0.4762	0.6065	0.2558	0.2536
	Kin Lib	Train (Filtered)	0.2586	0.3267	0.4140	0.5764	0.2230	0.2532
DARKIN-Interact	–	–	0.2663	0.2744	0.4486	0.5948	0.2322	0.1959
	Kin Lib	Train (Filtered)	0.2111	0.3232	0.3767	0.5898	0.1835	0.2383

Table A.14 Effect of augmentation with homologous sequences on DARKIN-FT and DARKIN-Interact, evaluated with clustered attribute-level metrics.

Model	**Augmentation Method**	**Kinase Set**	**Family AP**	**Family Acc**	**Group Ap**	**Group Acc**	**F.grain AP**	**F.grain Acc**
DARKIN-FT	–	–	0.2945	0.318	0.4762	0.6065	0.2558	0.2536
	Homologous	Train	0.2687	0.2913	0.4434	0.5780	0.2356	0.2355
DARKIN-Interact	–	–	0.2663	0.2744	0.4486	0.5948	0.2322	0.1959
	Homologous	Train	0.2722	0.2553	0.4584	0.5926	0.2371	0.1754

Table A.15 Hyperparameter settings for the DARKIN-FT, DARKIN-Interact, and *ProGen2–Phospho*.

Method	**Hyperparameters**						
	LR	**Optimizer**	**LR Scheduler**	**Weight Decay**	**Batch Size**	**LoRA r**	**LoRA alpha**
DARKIN-FT	[0.0001, 0.001, **0.01**]	[Adam, **SGD**]	[**Cosine**, Exponential, Step]	[**0.0001**, 0.001, 0.01]	[64, 128, **256**]	-	-
DARKIN-Interact	[**0.0003**, 0.001, 0.01]	[**Adam**, AdamW]	[Exponential, **Step**]	[**0.0001**, 0.001, 0.01]	[8, 16, **24**]	[**-**, 32, 64, 128]	[**-**, 32, 64, 128]
Progen2-Phospho	[**0.0001**, 0.001, 0.005]	[**AdamW**]	[**Cosine**]	[0.001, **0.01**]	[4, 8, **16**]	[**16**, 32, 64]	[**16**, 32, 64]