Enhancing Pattern Recognition with Learnable Rejection and Top-Rank Learning

冀, 晓彤

Enhancing Pattern Recognition with Learnable Rejection and Top-Rank Learning

Xiaotong Ji

Department of Advanced Information Technology
Doctor of Philosophy

KYUSHU UNIVERSITY

February 2024

Enhancing Pattern Recognition with Learnable Rejection and Top-Rank Learning

Xiaotong Ji

Department of Advanced Information Technology
February 2024
Doctor of Philosophy

Abstract

This thesis aims to enhance the reliability of decision-making processes within prevailing machine learning models. Ensuring such reliability is especially crucial in applications with minimal error tolerance, necessitating high model performance reliability. To address this critical aspect, this thesis adopts two distinct perspectives: Learning with Rejection (LwR) and top-rank learning.

Firstly, the thesis addresses the problem of decision ambiguity by adopting the LwR approach. LwR excels in concurrently learning two independent functions: one for traditional classification and another for rejection, which evaluates and confirms the decisions made by the classification function. These two functions are co-optimized through the minimization of a common loss function. LwR provides a solid theoretical foundation for optimizing decision-making processes and demonstrates its potential utility in tasks requiring highly reliable recognition results. Experimental validation of the character recognition task substantiates the effectiveness of this approach.

Secondly, the thesis leverages the top-rank learning approach to obtain only the positive samples with extremely high ranking scores while ignoring any other samples. The top-rank learning method is a variant of learning-to-rank, which aims to rank the samples according to the rule to maximize the Area Under the Curve (AUC). The learning mechanism of top-rank learning involves identifying the highest-ranked negative samples and optimizing the ranking process to prioritize placing more positive samples ahead of them.

Furthermore, this thesis introduces the concept of Paired Contrastive Feature (PCF), which treats paired inputs to a single model as an intra-related integration. This concept complements current contrastive learning models and enhances their performance by combining them with the previously mentioned LwR and top-rank learning methods. Experiments conducted on three publicly accessible signature verification datasets demonstrate a noticeable improvement in prediction reliability. Additionally, the thesis discusses limitations and outlines future avenues for research in the concluding section.

Acknowledgments

I would like to express my heartfelt gratitude to Professor Seiichi Uchida, my mentor and guide throughout this Ph.D. journey. Professor Uchida's patience and unwavering support have been invaluable to me and all the students in our lab. Beyond his academic guidance, he has always been approachable and willing to address not only our academic questions but also our personal concerns. In many ways, he has become a fatherly figure to me here in Japan, and his presence has consistently provided me with a sense of comfort and encouragement.

Professor Uchida's relentless dedication to fostering our intellectual growth is commendable. He consistently encourages us to think more critically, learn more deeply, and maintain an unquenchable curiosity about all things. He sets an exemplary standard for all of us with his boundless energy and genuine enthusiasm for his work, even as his schedule becomes increasingly hectic. His leadership style is one that inspires and motivates, never pushing but always guiding.

I am particularly grateful for Professor Uchida's emphasis on English language proficiency in our presentations. While seemingly trivial, this has been of immense importance to me. In Japan, opportunities to use English in daily life are limited, and maintaining fluency can be challenging. Thanks to his encouragement, I have been able to sustain and enhance my English skills. His unwavering support has been instrumental not just for me but for everyone in our lab, creating an environment where we can thrive.

I must also extend my gratitude to our lab members, including the associate professors and young researchers. Professors Suehiro and Hayashi have been a source of inspiration and provided significant assistance throughout my research. Professors Bise, Iwana, and Ohyama have served as role models since my early days in academia. I am also indebted to Yuchen and Yan, my seniors, who have offered both academic and personal support. Their presence has added color and companionship to my years in Japan, and I consider myself fortunate to have crossed paths with them.

I would be remiss not to thank my family for their unwavering emotional and financial support. Their comforting presence during my moments of doubt has been my anchor, and I know that I always have a place to call home. It is their encouragement and belief in me that have propelled me forward.

Lastly, I wish to express my deep appreciation for the financial support provided by the Japan Society for the Promotion of Science (JSPS) scholarship. This support has not only eased my financial burdens but has also afforded me more time to dedicate to my research, making my academic journey significantly smoother.

My sincere thanks go out to all these individuals and organizations who have been instrumental in shaping my academic and personal growth during this Ph.D. journey. Your contributions are deeply appreciated.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Pattern recognition [2, 3, 4, 5] is a fundamental concept that underpins the core principles of machine learning, essential for both human cognition and artificial intelligence. It refers to the process of identifying, interpreting, and categorizing patterns within data, spanning various formats of information. In the realm of machine learning, pattern recognition is the fundamental mechanism through which algorithms are trained to extract meaningful insights from data, generalize their knowledge, and make predictions or decisions.

Pattern recognition tasks often manifest in diverse forms, with two prevalent scenarios being (1) assigning a class label to each individual sample [6, 7, 5] and (2) providing prediction scores for pairs of samples [8, 9, 10]. In the first scenario, each sample is associated with a specific class label, indicating its categorization within a predefined set of classes or categories. This type of pattern recognition is commonly employed in applications such as image classification, where images are labeled as "cat," "dog," or "car." This fundamental form of pattern recognition is the basis for most real-world applications. In the second prevalent scenario, pairs of samples are evaluated to determine their relationship, often through output scores to pairs of

data samples. This approach is widely used in applications like signature verification, where the system assesses whether two signature images belong to the same person.

Reliability in pattern recognition refers to the robustness and trustworthiness with which a system can correctly identify and classify patterns in data inputs [11, 12]. It is a critical aspect of any pattern recognition system, directly impacting its performance and applicability in real-world settings. A reliable pattern recognition system consistently produces accurate and consistent results, even in the presence of noise, variations, or unseen data. This reliability is vital because it ensures that the system can make trustworthy predictions or classifications, reducing the chances of errors or misinterpretations. In real-world applications, particularly in fields where incorrect results can have serious consequences, the reliability of pattern recognition systems is paramount. In domains such as signature verification [13, 14, 15] in financial fraud detection, dependable results are not merely a preference; it's a fundamental requirement for ensuring security and effective decision-making. To secure this level of reliability, pattern recognition systems must undergo rigorous training and stringent validation procedures, incorporate well-designed features engineering, and employ effective algorithms.

The reliability of predictions in pattern recognition can be compromised by various factors, even when state-of-the-art models are employed. One significant challenge arises when dealing with overlapping class distributions, a scenario where it becomes theoretically impossible to guarantee error-free and dependable recognition [16, 17]. In such cases, the model's inability to clearly distinguish between similar patterns can lead to erroneous classifications and unreliable outcomes, as exemplified by instances of overlapping class distributions in Fig. 1-1.

Conventional machine learning approaches frequently prioritize performance optimization through feature engineering techniques [18, 19], often neglecting ambiguity within the training data. Ambiguity in the dataset, as illustrated in Fig. 1-1, not only affects predictive results in the test set but also influences the learning process

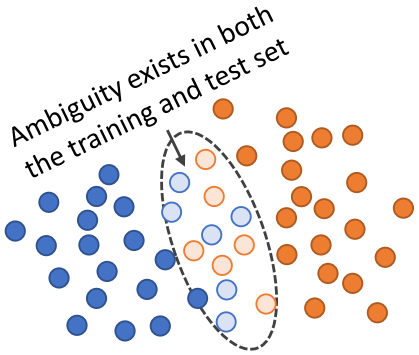


Figure 1-1: The presence of inherent class ambiguity is frequently observed within the distribution of two classes surrounding their classification boundary, posing a potential threat to reliable classification performance. Addressing this class ambiguity by removing such instances could enhance model training and improve predictive capabilities. One of the most effective strategies for mitigating this challenge is to reject ambiguous samples selectively.

during training when such ambiguity is present in the training data. The presence of ambiguity underscores the critical importance of strategically mitigating ambiguity in the training process to achieve dependable prediction results.

Even the most advanced machine learning models require ongoing refinement to achieve higher levels of reliability. While state-of-the-art models exhibit remarkable capabilities, they are not immune to the inherent challenges that persist in real-world data. One main reason lies in the persistent ambiguity within the training set data. Regardless of a model's sophistication, ambiguous or overlapping samples pose a significant challenge. Striving to capture every nuance in the training data can lead to overfitting, where the model becomes excessively tailored to the intricacies of the training data, resulting in decreased generalization and reliability.

1.2 Purpose

The primary purpose of this thesis is to enhance the predictive reliability of machine learning models in the context of pattern recognition. This purpose arises from the

inherent ambiguity and uncertainty often found within real-world data, a challenge that even the most advanced machine learning models grapple with. In scenarios where misclassifications are unacceptable, the need for improved reliability becomes paramount. Current machine learning practices, which often focus on dataset quality enhancement, pre-training, and cross-validation, have indeed made significant strides in performance optimization. However, they sometimes fail to address misclassifications that persist, even after these standard remedies.

The core strategy involves selectively retaining unambiguous samples during both training and testing. By training models on those samples, greater reliability and generality can be achieved in predictions. This thesis explores innovative techniques tailored to enhance decision-making accuracy in applications such as image classification and signature verification.

Enhancing the predictive reliability in pattern recognition carries profound implications for a multitude of real-world applications, particularly those where incorrect predictions can lead to severe consequences. This, in turn, facilitates transformative advancements in artificial intelligence applications across diverse domains, ensuring their practical performance. Consequently, this work aims to empower machine learning models to make more trustworthy decisions, fostering advancements in artificial intelligence applications across diverse domains and unlocking the full potential of machine learning.

1.3 Proposal 1: Introducing Rejection Option for Ambiguous Sample Discard

In the realm of machine learning, addressing ambiguous samples that hinder the recognition performance of models has been a persistent challenge. Existing solutions have often resorted to recognition with a rejection option [20, 21, 22]. Figure 1-1 outlines the primary targets for rejection: those located within overlapping regions

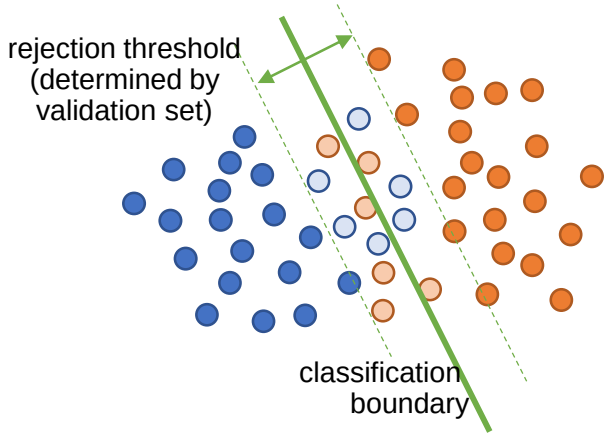


Figure 1-2: Typical heuristic strategy for rejecting the samples in the overlapping area of class distributions. This method assumes the presence of an accurate classification boundary. However, when training the classification boundary, a question arises: how can we effectively reject ambiguous samples?

of class distributions who have instigated the incorporation of a rejection mechanism both during the training and testing phases. These samples lay in overlapping areas, are inherently challenging to classify, and are better treated as "don't-care" instances. That is, they are better to be rejected from the prediction candidates.

Traditionally, heuristic rejection strategies have been employed to address ambiguous samples. These strategies often involve discarding samples close to the classification boundary during the testing phase, as depicted in Fig. 1-2. This figure portrays a straightforward rejection method for a Support Vector Machine (SVM), wherein samples within the overlapping region are rejected if their proximity to the classification boundary falls below a certain threshold [23].

However, the aforementioned heuristic rejection strategy has a notable limitation; it presupposes the presence of a precise classification boundary to enable rejection. In essence, it can only reject test samples, while the existence of ambiguous samples within the training set adversely affects the determination of the classification boundary. This underscores the necessity of introducing a more sophisticated rejection mechanism, where the rejection criteria are co-optimized alongside the classification

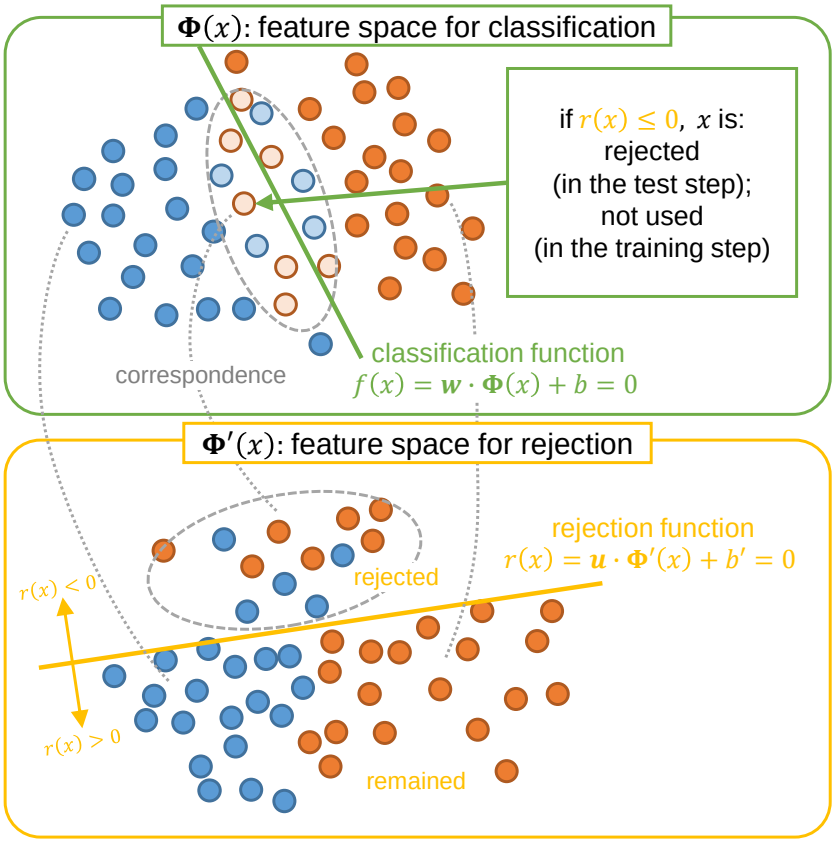


Figure 1-3: An overview of *learning with rejection* for a two-class problem. The rejection function $r(x)$ is co-trained with the classification function $f(x)$. Note that *different feature spaces* optimize $r(x)$ and $f(x)$.

boundary during the training phase.

To address these issues, this thesis proposes the adoption of the Learning with Rejection (LwR) [24] strategy in recognition tasks. LwR is designed to identify and exclude ambiguous samples, both during the training and testing phases. The fundamental concept behind LwR is to learn the classification function and a separate rejection function simultaneously, co-optimizing both functions by distinct features during the training phase.

Figure 1-3 provides an overview of the LwR framework. During training, the rejection function $r(x)$ identifies and excludes ($r(x) < 0$) several samples, which are then omitted from the classifier's training process. Notably, this co-optimization

represents a pivotal characteristic, where both the rejection function $r(x)$ and the classification function $f(x)$ are optimized using different features of the inputs through a unified optimization framework. This approach removes the need to presuppose a precise classification boundary, enabling the rejection of ambiguous samples during both training and testing.

Additionally, to effectively handle high-dimensional data such as images, an adaptation of LwR techniques is proposed, incorporating deep Convolutional Neural Network (CNN) features known for their robust feature extraction capabilities. The involvement of deep features offers an improved representation of high-dimensional input data, surpassing the traditional LwR approach with kernel-based techniques. The proposed LwR with deep features offers several advantages, including robust theoretical underpinnings against overfitting, the flexibility to employ different feature spaces for the rejection function, and the ability to tackle challenging character recognition tasks.

1.4 Proposal 2: Introducing Rejection Option for Ambiguous Sample Pairs Discard

In pattern recognition tasks involving paired inputs, the rejection process is more intricate than it initially seems. Firstly, rejecting one of the two inputs in a manner similar to how a single sample is rejected in the previous section does not make sense. This is because heuristic or learnable rejection techniques can reject the result of a single input, as it represents class likelihood. However, the output of a dual-input model can carry various meanings, such as representing the distance between the two inputs or serving as a likelihood indicator of whether both inputs belong to the same class. It cannot definitively indicate whether one of the two inputs should be rejected.

Moreover, even if considering rejecting two paired inputs simultaneously, relying solely on the output score for this determination is insufficient. This is primarily

because the model may use intricate computations to assess and manipulate the inherent relationship between the two inputs. A single numerical value alone may struggle to provide a sound basis for decision-making in such a context. Furthermore, in the case of LwR, it is imperative to prepare distinct inputs for classification and the rejection function. Ideally, both of these inputs should result from an effective feature extraction process rather than being mere numerical values. Nonetheless, it's important to note that LwR cannot be directly applied to models dealing with paired inputs that yield only a single output score.

Consequently, utilizing a unified feature for a pair of samples is practical. It essentially entails combining the individual features of two samples to create a representation that encapsulates the essence of the pair as a whole, as illustrated in Fig. 1-4 (a). This united feature is anticipated not to transcend a mere concatenation of features but a carefully crafted combination that captures the interplay and relationships between the two samples. Once the united feature is established, addressing ambiguity equates to rejecting the united features of the paired inputs, as depicted in Fig. 1-4 (b). Subsequently, the LwR technique can be directly applied to the unified feature.

Determining the appropriate feature representation for a pair of samples is critical to this thesis. Several strategies will be considered to address this challenge effectively. Initially, one might straightforwardly contemplate the direct concatenation of the original features from both samples in the pair. However, this approach may prove insufficient in capturing the nuanced relationships between the samples despite its apparent simplicity, as elucidated in preceding sections. Alternatively, one can delve into advanced methodologies wherein individual features undergo an intra-related transformative learning process and excel in capturing the nuanced relationships between samples. This results in generating more informative and intra-related representations, ultimately contributing to forming a unified feature.

In the pursuit of achieving an optimal united feature representation for paired

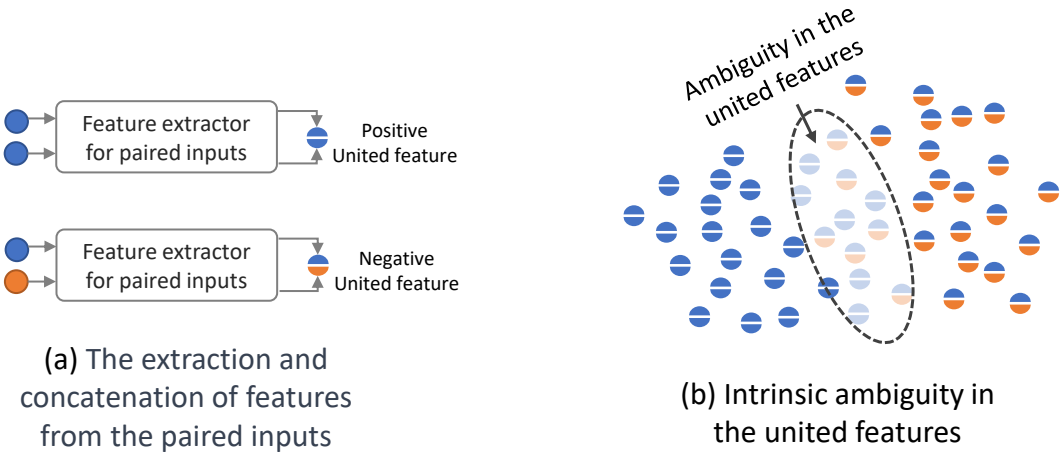


Figure 1-4: (a) Illustrates a concise process for generating united features from a pair of input data points (blue=positive, orange=negative) by feeding them into the same feature extractor. Following this procedure, the inherent ambiguity appears among the positive and negative united features, as depicted in (b). Rejecting this ambiguity is equivalent to rejecting the paired inputs with ambiguity.

samples, a more advanced and promising technique involves leveraging the feature representation derived from contrastive learning [25, 26, 27]. It operates by encouraging the model to distinguish between pairs of samples. It essentially teaches the model to pull similar pairs closer while pushing dissimilar pairs farther apart in the feature space. Its significance lies in its ability to unlock a multitude of applications, ranging from computer vision to natural language processing, where it has demonstrated remarkable performance improvements. Notably, contrastive learning allows models to capture intricate patterns and semantic relationships in data, ultimately paving the way for more robust and efficient machine learning systems.

The application of contrastive learning to feature representation of paired samples can have transformative effects. By encouraging the model to create discriminative representations of sample pairs, it can be expected that the united feature will capture intricate patterns and semantic relationships between the samples more effectively. This, in turn, can lead to more robust and efficient machine learning systems for specific tasks.

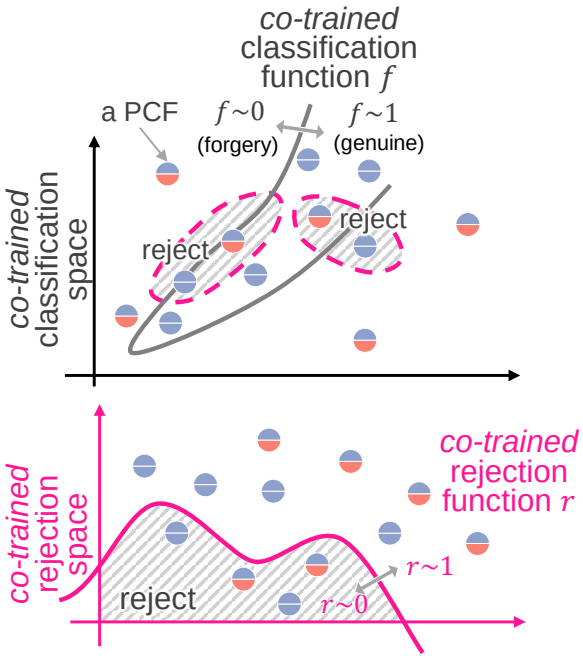


Figure 1-5: With PCF, LwR can handle the writer-independent signature verification task. The coordinate axis represents a two-dimensional feature space in which the data is distributed.

Given the substantial advantages of contrastive learning, I introduce the term Paired Contrastive Feature (PCF) to characterize this novel feature representation technique. PCF is a cutting-edge approach that harnesses the power of contrastive learning to effectively capture intricate patterns and semantic relationships within paired input data. Its application can indeed have transformative effects, establishing it as a valuable tool for machine learning practitioners grappling with complex tasks that necessitate paired input analysis.

The framework that combines PCF with LwR is depicted in Fig. 1-5. As introduced in section 1.3, LwR employs a rejection function to assess whether a given sample should be accepted or rejected. Specifically, it embodies a co-training procedure that simultaneously trains a rejection function and a rejection feature space alongside a classifier and a classification feature space. This co-training procedure of two functions and two feature spaces within LwR guarantees accurate and efficient

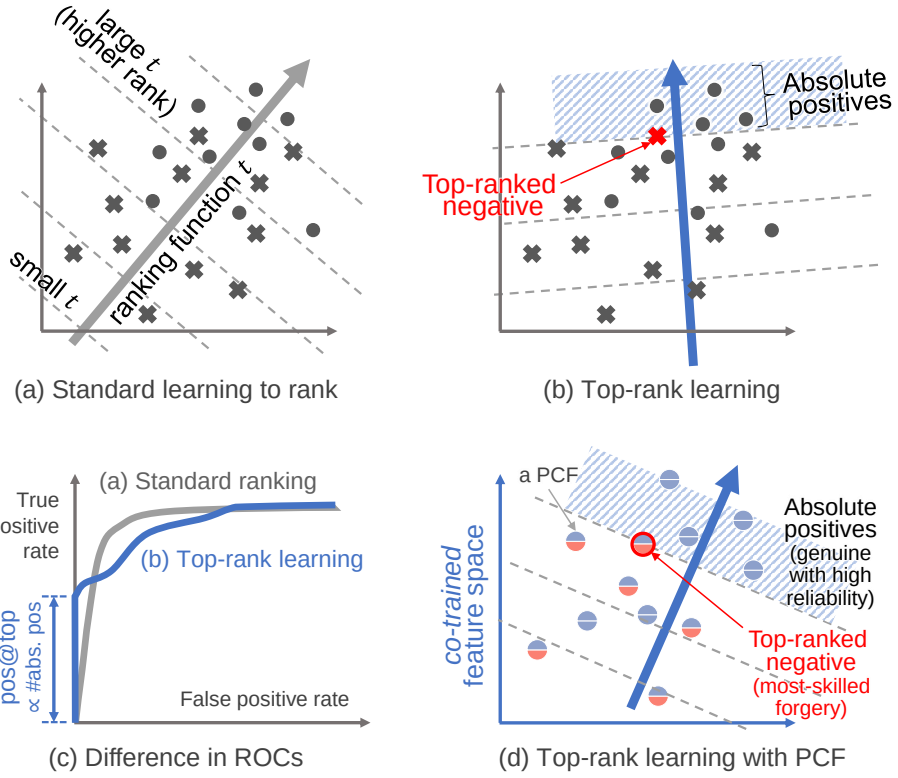


Figure 1-6: (a) Standard learning-to-rank. The ranking function is trained to give a higher rank value for positive samples (depicted as •) than negatives (×). (b) Top-rank learning, whose objective is to have more absolute positives. (c) The difference between (a) and (b) in Receiver Operating Characteristic (ROC) curves. (d) Reliable signature verification by top-rank learning with PCF.

rejection. Besides, deviating from the model employed in section 1.3, here I adopt a one-step CNN-based model to conduct LwR. This variation aims to provide better control over the rejection rate, and the detailed structure of this model will be explained in Chapter 4.

1.5 Proposal 3: Introducing Top-rank Learning for Reliable Sample Pair Detection

In addition to LwR, another approach aimed at enhancing the reliability of machine learning model predictions is explored in this thesis. Termed top-rank learning, I propose its integration with PCF to address the intricacies associated with paired inputs effectively. Diverging from the operational mechanism of LwR, which seeks to identify and mitigate the influence of ambiguity, top-rank learning adopts a distinct strategy by proactively disregarding ambiguity from the outset. Its focus is exclusively directed toward positive samples that surpass any negative samples in ranking scores, thereby prioritizing unambiguous instances for consideration. Both LwR and top-rank learning are crucial methodologies examined in this thesis, and it is imperative to emphasize and maintain awareness of their distinctive characteristics throughout the thesis.

Given that top-rank learning constitutes a specialized branch within the broader category of learning-to-rank methods, initiating the exposition with an overview of the conventional learning-to-rank approach is pertinent. learning-to-rank is a significant subfield of machine learning that addresses the challenge of ranking items in a manner that optimally captures the underlying patterns and relationships within data. The illustrative representation of this concept is featured in Fig. 1-6 (a). learning-to-rank finds application across a diverse spectrum of scenarios, encompassing web search, recommendation systems, content retrieval, and other domains where the prioritization of items based on their relevance or desirability holds paramount significance.

Standard learning-to-rank techniques primarily focus on ranking positive samples higher than negative ones, which is equivalent to maximizing the Area Under the Curve (AUC), whose curve is depicted in gray color in Fig. 1-6 (c). Although this methodology is valuable for many ranking scenarios, it might not provide the level of reliability required in situations where the utmost precision is essential. In particular,

standard learning-to-rank falls short when seeking to maximize the ranking of positive instances above negative ones. This limitation becomes apparent in tasks where the reliability of rankings is a primary concern, as seen in domains like medical diagnosis, where misclassification can have dire consequences.

To address the abovementioned limitation of standard learning-to-rank, top-rank learning is distinguished by its unique objective: maximizing absolute positives, which refer to positives that rank higher than the top-ranked negatives. This means not just ranking positive samples higher than negative ones but ensuring that many positive instances ranked higher than any negatives. To provide a clear contrast with standard learning-to-rank, Fig. 1-6 (b) illustrates the fundamental concept of top-rank learning. This distinctive strategy of top-rank learning makes it powerful for enhancing the identification of top positive cases, particularly suited for challenging tasks that demand a high degree of reliability.

Top-rank learning introduces a novel paradigm to prioritize the absolute positives instead of maximizing the AUC in traditional learning-to-rank. This is equal to maximizing the pos@top (the ratio of absolute top positives to all positives) as shown in the blue line in Fig. 1-6 (c). By doing so, it becomes possible to draw samples that best meet the requirement. This shift in focus is particularly advantageous in applications that require a high level of reliable prediction with extremely low tolerance to errors.

In essence, top-rank learning presents a novel paradigm that emphasizes prioritizing absolute positives (the instances ranked higher than any negatives) over the conventional approach of maximizing the AUC in traditional learning-to-rank methods. This emphasis corresponds to the maximization of pos@top, representing the ratio of absolute top positives to all positives. This optimization objective is visualized in the blue line in Fig. 1-6 (c), in contrast to the traditional learning-to-rank objective.

The top-rank learning approach empowers the selection of samples that closely

conform to specific criteria, making it exceptionally advantageous for applications where precision and error avoidance are of utmost importance. Top-rank learning is anticipated to produce remarkably reliable outcomes when implemented in pattern recognition tasks. This advantage is particularly evident in challenging scenarios, such as medical diagnosis, where the need for highly reliable predictions is critical.

In tackling tasks that require highly reliable predictions involving paired inputs, integrating the top-rank learning strategy with PCF presents a seamless and effective approach. Unlike LwR with PCF, top-rank learning enhances prediction reliability by maximizing absolute positive samples. However, like LwR, the top-rank learning approach also avoids predicting ambiguous samples.

When combining top-rank learning with PCF, these techniques can be applied to a broader range of tasks involving paired inputs, such as signature verification. As detailed in section 1.4, a PCF compiles the concatenated features with intrinsic relationships between paired inputs, offering sufficient information for top-rank learning to make predictions. Figure 1-6 (d) illustrates how, by utilizing PCF, a highly reliable signature verification system can be realized, one that only accepts definitively genuine samples via absolute positive PCFs.

1.6 Contributions

This thesis introduces several novel contributions:

- This work represents the first practical application of LwR to the field of pattern recognition. While the concept of LwR holds significant promise for a wide range of pattern recognition tasks, including character recognition and document processing, there has been a notable absence of prior research applying this concept to practical applications.

- To design features for classification and rejection, this study leverages a CNN as a versatile multi-feature extractor. The classification function is trained using

the output from the final convolutional layer, while the rejection function is trained from a different layer of the CNN.

- Two comprehensive classification experiments are conducted, demonstrating the superior performance of LwR over alternatives. Specifically, LwR outperforms CNN with confidence-based rejection and traditional SVM using the rejection strategy illustrated in Fig. 1-2 (b).

- The proposal of PCF is introduced as a mechanism for transforming a pairwise prediction task into a sample-wise prediction task, which is especially valuable for writer-independent signature verification.

- To the best of the authors' knowledge, this work represents the first application of two machine-learning frameworks, LwR, and top-rank learning, to achieve highly reliable signature verification. The unique properties of PCF enable these applications.

- Extensive experiments, employing multiple evaluation metrics, quantitatively and qualitatively validate the high reliability of the proposed methods. These experimental results provide robust evidence of the effectiveness and practical applicability of the contributions.

1.7 Organization

This thesis is structured to provide a comprehensive understanding of the proposed methods, their related work, and their application in the real world. The following chapters delineate the progression of the research:

- Chapter 1 serves as a foundational backdrop to the thesis. It initially contextualizes the pressing need for reliability in the field of machine learning, presenting approaches and methods employed to attain this imperative. It then outlines the

motivations, propositions, and objectives underpinning the proposed methods' real-world applications. Finally, it succinctly summarizes this thesis's novelty and significant contributions.

- Chapter 2 conducts an extensive review of the relevant literature pertaining to the proposed methods. It encompasses a comprehensive examination of LwR, top-rank learning, contrastive learning, and their applications within character recognition and signature verification domains.

- Chapter 3 delves into the proposed method of LwR in tandem with deep features. This comprehensive section provides an overview of the LwR framework and an in-depth exploration of its methodology and the intricate details of experiment implementation.

- Chapter 4 is dedicated to presenting the details of the proposed PCF. It outlines the method's overarching structure, underlying methodology, and the collaborative utilization of PCF with top-rank learning and LwR. Furthermore, it offers an extensive look into its practical implementation in the realm of character recognition and signature verification.

- Chapter 5 encapsulates the culmination of this thesis. It offers a conclusive assessment of the proposed methods, elucidates their limitations, and provides valuable insights into potential future research avenues and developments.

Chapter 2

Related Work

2.1 Machine Learning Methods for Reliable Decision Making

This section outlines the related works that form the foundation of LwR and Top-rank learning. These two approaches have distinct historical backgrounds and demonstrate remarkable capabilities in diverse application scenarios, each contributing to enhancing the reliability of recognition results.

2.1.1 Learning with Rejection (LwR)

Reject options in machine learning have gained significance in improving prediction reliability. They are being utilized in various tasks, including handwriting recognition [28, 29, 30] and image classification [31, 21, 32]. By enabling the rejection of ambiguous predictions, particularly around the classification boundary, some methods contribute to overall performance improvement. Among the earliest works in rejection-based algorithms, Fumera et al.[33] presented a rejection approach based on ROC curves in binary classification, yielding an optimal threshold and providing theoretical guarantees. Remarkably, even in recent studies, the threshold-based rejection

method remains popular and widely used[20, 34, 35].

LwR, in contrast to the threshold-based rejection approach, has emerged to offer promising advantages. LwR involves learning an independent rejection function alongside the classification function, introducing a flexible and independent rejection mechanism. Cortes et al. [24] proposed a learning algorithm that simultaneously obtains both the classification and rejection function, grounded in statistical learning theory. Moreover, Mozannar et al.[36] introduced an LwR system that not only incorporates a rejection function but also considers the decision of a human expert, enhancing its practicality in real-world scenarios.

Another notable contribution is SelectiveNet [37], which follows the concept of LwR and adopts a CNN-based structure. SelectiveNet has demonstrated remarkable application performance by integrating a highly representative classifier and rejector, showcasing the potential of LwR in modern deep-learning architectures. Notably, SelectiveNet excels in accurately controlling the rejection rate compared to the original LwR model, as per its unique learning scheme.

This paper has explored the mechanism of LwR and its role in improving the reliability of machine learning model decision-making processes. I have discussed the application of LwR in character recognition tasks, building upon the traditional machine learning method introduced in [24], which strong theoretical foundations support. Furthermore, I have highlighted the utilization of LwR with a representational learning mechanism in [37] for signature verification tasks. These two distinct methods hold different advantages and exhibit potential in diverse scenarios, further underscoring the versatility and practicality of LwR.

2.1.2 learning-to-rank

The learning-to-rank strategy [38, 39] proves to be a highly effective approach when the demand for identifying the most relevant targets is paramount. This is particularly evident in tasks like signature verification [40, 41] and medical image recognition [42].

In scenarios where obtaining highly reliable genuine signatures is crucial, learning-to-rank is more reasonable than learning to classify. This is because learning-to-rank allows signatures to be arranged in order of genuineness, as its key objective is to rank queries in a supervised manner.

Among the existing ranking methods, bipartite ranking [43] is a typical learning-to-rank approach. The main learning objective of bipartite ranking is to obtain a scoring function that assigns higher values to positive samples (genuine signatures, for instance) compared to negative ones (e.g., forged signatures). Since the objective of the bipartite ranking is equivalent to maximizing the Area Under the Curve (AUC), it has found applications in various domains where a high AUC is important. For instance, the bipartite ranking has been effectively utilized in tasks such as product recommendation systems [44], information retrieval [45], and ad click-through rate prediction [46], where ranking plays a crucial role in optimizing user experience and system performance.

Recent research has explored incorporating deep learning models into ranking frameworks to enhance the learning-to-rank approach. Deep learning techniques, with their ability to automatically extract intricate features, have shown promising results in various ranking tasks. For instance, the work of Pang et al. [47] introduced DeepRank, a deep learning-based method that effectively combines query-document matching and learning-to-rank principles. Moreover, list-wise ranking algorithms have also emerged as a popular direction in learning-to-rank research. ListNet [48], proposed by Cao et al., is a notable example that aims to optimize the ranking list instead of individual pairs of documents.

2.1.3 Top-rank Learning

Top-rank learning is a specialized type of bipartite ranking that focuses on optimizing ranking performance at the top rather than considering the overall ranking [49]. The primary objective of this approach is to maximize the number of samples that can

be determined with an extremely high degree of certainty as positive (i.e., absolute positives), ensuring that positive samples are ranked higher than any negative sample. This emphasis on reliable predictions makes top-rank learning particularly suitable for highly reliable tasks like signature verification.

A notable contribution to top-rank learning comes from Zheng et al. [42]. In this work, the authors introduced a representation learning approach for top-rank learning using Neural Network to minimize the loss of p-norm push [50]. The parameter p allows control over the concentration of the objective, enabling a focus on the upper ranks of the ranking with varying degrees of intensity. Unlike previous methods based on linear or non-linear kernels [49, 51], the model proposed in [42] achieves high performance even for complex recognition tasks. This demonstrates the applicability and effectiveness of top-rank learning in demanding domains that require high reliability, such as medical diagnosis [52].

Furthermore, research in top-rank learning has also seen advancements in understanding the theoretical foundations and developing efficient algorithms. Li et al.[49] provided foundational work on top-rank learning and established the theoretical underpinnings of this approach. Boyd et al.[51] explored the use of top-rank learning in the context of classification tasks, showing its potential to improve accuracy at the top of the ranking. Moreover, the work of Frery et al. [53] contributes to the practical aspects of top-rank learning by proposing efficient algorithms that scale well to large datasets.

2.2 Contrastive Learning

This section provides an in-depth exploration of contrastive learning, elucidating its fundamental concept and principle. Contrastive learning has emerged as a potent paradigm in the realm of self-supervised representation learning and has garnered remarkable success across a spectrum of domains, spanning computer vision and

natural language processing [54, 55]. At its core, contrastive learning operates on the principle of training a model through the contrast between positive pairs of samples and negative pairs, compelling the model to map similar samples in closer proximity within the learned representation space while simultaneously driving apart dissimilar samples.

One of the foundational methodologies within contrastive learning is Siamese Networks, a concept initially introduced by Bromley et al. [56]. Siamese Networks adopt a neural network architecture structured like twins, sharing weights to compute embeddings for pairs of samples. The network's training objective revolves around the minimization of the distance between embeddings for positive pairs and the maximization of distance for negative pairs. This mechanism allows the network to learn to effectively discriminate between samples that exhibit similarity and those that are distinct from one another.

2.3 Character and Signature Recognition

In this section, I explore two pivotal scenarios integral to this thesis: character recognition and signature verification. The following subsections offer a comprehensive introduction to the critical aspects of these two application domains, shedding light on the key challenges and methodologies that underpin our investigation. This thesis will also elaborate on the novel contributions to these fields, further enhancing their practical applications and theoretical foundations.

2.3.1 Character Recognition

Character recognition aims to identify characters from scanned images or handwritten documents automatically [57]. Early systems heavily relied on handcrafted features and rule-based algorithms; techniques like Template Matching [58] involved matching characters to predefined templates, which worked well for specific fonts but strug-

gled with variations in writing styles, different fonts, and noisy images. As character recognition evolved, feature engineering played a pivotal role in improving performance. Notable techniques include Histogram of Oriented Gradients (HOG) [59], Scale-Invariant Feature Transform (SIFT) [60], and Local Binary Patterns (LBP) [61]. Feature engineering approaches significantly enhanced recognition accuracy, although they demanded domain expertise and manual tuning, limiting their adaptability to different datasets.

With the advent of machine learning, character recognition systems shifted towards data-driven approaches. Supervised learning algorithms, such as Support Vector Machines (SVM) [62] and k-Nearest Neighbors (k-NN) [63], gained prominence for classifying characters based on labeled training data [64]. These algorithms achieved impressive results, particularly with well-segmented and clean character images.

Further, CNNs have demonstrated their prowess in character recognition by automatically learning hierarchical features from raw pixel data, significantly reducing the reliance on handcrafted feature engineering. In an early work by Bai et al. [65], they introduced a shared-hidden-layer deep CNN architecture for image character recognition. Remarkably, this CNN design employed common hidden layers across characters from different languages, facilitating a universal feature extraction process aimed at capturing shared character traits among various languages. Furthermore, Aneja et al. proposed the application of transfer learning using pre-trained CNNs to recognize handwritten Devanagari alphabets, achieving promising results [66]. By leveraging knowledge from pre-trained models, the transfer learning technique significantly improved recognition accuracy for the intricate Devanagari script.

In enhancing the reliability of character recognition, an intrinsic challenge is encountered in distinguishing inherently indistinguishable characters, such as the handwritten uppercase "I" and lowercase "l." Traditional methods may achieve high accuracy, but they struggle with the forced division of these ambiguous characters. To address this issue, I propose applying LwR in this manuscript, which allows us to

exclude such uncertain samples rather than attempting to classify them forcibly.

2.3.2 Signature Verification

Signature verification tasks can be categorized into two main scenarios: writer-dependent and writer-independent [67], as explained in Chapter 1 and Fig. 4-4 (a) and (b). Additionally, signature verification tasks can be classified into online and offline scenarios from another perspective [68]. Online signature verification approaches [69, 70] utilize signature information such as pressure and stroke, typically involving time-series analysis. On the other hand, offline signature verification [71, 72] relies solely on signature image information, which is more practical but presents greater challenges in verification. In this paper, the offline writer-independent scenario is the focal point due to its practicality and the heightened demand for reliability.

Various methods have been proposed to extract discriminative features from signature images. For instance, Zois et al. [73] introduced a method based on asymmetric pixel relations designed to capture the static signature image's intricate details. In another work, Banerjee et al. proposed a wrapper feature selection method that combines and selects features from multiple perspectives [74]. Parcham et al. presented a novel approach [75], which utilizes a combination of CNN and Capsule Neural Networks to capture spatial properties of signature features, leading to improved verification performance.

SigNet [1] is one of the state-of-the-art methods for writer-independent skilled-forgery-only signature verification based on contrastive learning mechanisms. This model employs a Siamese network [56] for signature representation, leveraging metric learning with contrastive loss. Empirical results demonstrated the superiority of metric learning with contrastive loss over standard classification approaches.

However, even in the writer-independent scenario, existing signature verification methods focus primarily on the feature mapping of each signature image while giving little attention to the contrastive relation within signature pairs. As previously

mentioned, the writer-independent scenario involves a pairwise input consisting of a query signature and a genuine reference signature. Therefore, it is more reasonable to utilize the information from pairs of signatures rather than individual signatures. Motivated by this, I introduce PCF, which leverages contrastive feature extraction from signature pairs to enhance the verification process.

Chapter 3

Learning with Rejection with Deep Features

3.1 Overview

This thesis focuses on enhancing the decision-making reliability of machine learning models. The primary emphasis is on improving their performance in character recognition tasks, which often involve distinguishing between characters, especially in challenging handwritten scenarios. To achieve this goal, I explore applying the LwR method, a technique for identifying and rejecting ambiguous samples that could impede recognition. Moreover, LwR is underpinned by a strong theoretical foundation, rendering it a highly promising approach.

In this chapter, the primary objective is to harness the potential of the LwR method to enhance the reliability and accuracy of decision-making in character recognition applications. The proposed approach entails the development of an optimized rejection function that works in tandem with the classification function. This collaboration enables the learned rejection function to reasonably manage the rate of sample rejections, thereby facilitating the segregation of recognition candidates with greater uncertainty or ambiguity. By embracing the synergy between classification and re-

jection, this approach empowers the recognition model to minimize its overall loss, achieving optimal recognition performance within a controllable range of rejections.

This chapter unfolds as follows: Firstly, it introduces the methodology of the LwR approach employed in this study, offering a comprehensive view of its intricacies and underpinnings. Subsequently, the experimental settings utilized in the research are detailed. Finally, the chapter presents the results from the experiments, followed by a comprehensive discussion and analysis of the outcomes.

3.2 Methodology of Learning with Rejection

In this section, I present the problem formulation of LwR that involves introducing key components. The training dataset is denoted as (x_1, y_1), ..., (x_m, y_m), with x_i denoting the input data and y_i corresponding to the output or target value. I establish two distinct functions, $\boldsymbol{\Phi}$ and $\boldsymbol{\Phi}'$, each mapping input data x into separate feature spaces. Specifically, I define the linear classification function, $f = \boldsymbol{w} \cdot \boldsymbol{\Phi}(x) + b$, which operates within feature space $\boldsymbol{\Phi}$, alongside the linear rejection function $r = \boldsymbol{u} \cdot \boldsymbol{\Phi}'(x) + b'$, operating within feature space $\boldsymbol{\Phi}'$. Consequently, the LwR problem centers on the optimization of parameters $\boldsymbol{w}$, b, $\boldsymbol{u}$, and b', aiming to discover the optimal values for these parameters to effectively model the given training data and attain the desired regression or classification outcomes. Refer to Fig. **??** to better illustrate the overall process.

In the context of a two-class classification problem with a reject option, the decision rule employed is as follows:

$$(f, r)(x) = \begin{cases} +1 \ (x \text{ is class1}) & \text{if } (f(x) > 0) \wedge (r(x) > 0), \\ -1 \ (x \text{ is class2}) & \text{if } (f(x) \leq 0) \wedge (r(x) > 0), \\ \text{reject} & \text{if } r(x) \leq 0. \end{cases} \tag{3.1}$$

To strike the optimal balance between classification accuracy and rejection rate, the objective is to minimize the following composite risks:

$$\mathcal{R}(f,r) = \sum_{i=1}^{m} \left(I(\text{sign}(f(x_i)) \neq y_i) \wedge (r(x_i) > 0)) + cI(r(x_i) \leq 0)) \right), \tag{3.2}$$

where $I(\cdot)$ denotes an indicator function. Here, the parameter c assumes the role of a weight factor, influencing the significance of the rejection cost and reflecting the penalty associated with the reject operation. Further, the initial portion of this summation equation signifies that in cases of non-rejection, the cost associated with an incorrect classification is equal to 1. To illustrate, consider document processing tasks such as digit recognition scenarios for bank bills, where the consequences of errors are exceedingly undesirable. In such cases, a lower value of c can prove more appropriate, actively encouraging the rejection of suspicious patterns to mitigate the potential for costly mistakes. On the other hand, opting for a larger value of c can be appropriate in scenarios like character recognition for personal diaries. This choice fosters a higher tolerance for ambiguous characters and makes accepting more of them bearable, given their relatively lower impact on the overall task.

Given the intricate nature of directly minimizing the risk R as described in Equation 3.2, an alternative approach was introduced by Cortes et al. [24]. This alternative method relies on the utilization of a convex surrogate loss function, denoted as L, defined as follows:

$$L = \max\left(1 + \frac{\alpha}{2}(r(x) - yf(x)), c(1 - \beta r(x)), 0\right),$$

where $\alpha = 1$ and $\beta = 1/(1-2c)$. In the work of Cortes et al. [24], it has been demonstrated that the minimization problem of L, combined with a customary regularization applied to $\boldsymbol{w}$ and $\boldsymbol{u}$, leads to the following optimization problem, akin to

SVMs:

$$\begin{aligned}
&\min_{\boldsymbol{w},\boldsymbol{u},b,b'\boldsymbol{\xi}} \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{\lambda'}{2}\|\boldsymbol{u}\|^2 + \sum_{i=1}^{m}\xi_i \\
&\text{sub. to } \xi_i \geq c\left(1 - \beta\left(\boldsymbol{u}\cdot\boldsymbol{\Phi}'\left(x_i\right) + b'\right)\right), \\
&\qquad \xi_i \geq 1 + \frac{\alpha}{2}\left(\boldsymbol{u}\cdot\boldsymbol{\Phi}'\left(x_i\right) + b' - y_i(\boldsymbol{w}\cdot\boldsymbol{\Phi}\left(x_i\right) + b)\right), \\
&\qquad \xi_i \geq 0 \ (i = 1, \ldots, m),
\end{aligned} \tag{3.3}$$

where λ and λ' are regularization parameters and ξ_i is a slack variable. Indeed, the problem described can be characterized as a quadratic programming (QP) problem featuring a quadratic objective function subject to linear constraints. Leveraging the properties of QP, the optimal solution can be efficiently computed using a specialized quadratic programming solver. This approach ensures the effective and computationally tractable optimization of the LwR model.

The optimal solution to (3.3) carries a theoretical guarantee regarding generalization performance, as postulated by [24]. Let us contemplate the selection of a classification function, denoted as f, from a set of functions F, and a rejection function, denoted as r, from another set of functions G. Both F and G represent arbitrary sets of functions that yield values within the set {-1, +1}. The risk on test samples is represented by $\mathcal{R}_{\text{test}}$, while the risk on training samples is denoted as $\mathcal{R}_{\text{train}}$. Subsequently, the following assertions can be made:

$$\mathcal{R}_{\text{test}}(f, r) \leq \mathcal{R}_{\text{train}}(f, r) + \mathfrak{R}(F) + (1 + c)\mathfrak{R}(G). \tag{3.4}$$

Here, $\mathfrak{R}$ represents the Rademacher complexity [76], gauging the theoretical overfitting risk of a function set. Essentially, this theorem suggests that a judiciously chosen feature space is not excessively complex and is mapped by $\boldsymbol{\Phi}$ and $\boldsymbol{\Phi}'$, which can yield test sample performance comparable to that achieved on training samples.

By solving (3.3) to minimize $\mathcal{R}_{\text{train}}$, we obtain theoretically optimal classification and rejection functions. These strike an optimal balance between the classification error for non-rejected samples and the rejection cost, as elaborated by [77].

A robust theoretical foundation underpins the LwR solution obtained through the aforementioned formulation. Analogous to the standard SVM solution, accompanied by a theoretical guarantee regarding its generalization performance, the optimal solution to the optimization problem (see Eq.3.3) in LwR also benefits from theoretical justifications. Notably, the original LwR research primarily focused on theoretical analysis and had not been practically applied previously. To the author's knowledge, this proposal represents the inaugural practical application of LwR to a real-world task.

3.3 Application to Character Recognition

Methods for comparison

Through the experiments, I compare three methods: The proposed LwR approach, the CNN classifier, and the SVM classifier[1]. The deep CNN features employed, which already contain nonlinear compositions, make it possible for the SVM classifier set to operate linearly.

The performances of these three classification models are compared by varying the rejection cost parameter c in the range of 0.1 to 0.4. The CNN and SVM output class probabilities $p(y|x)$, and I determine their final output, denoted as $f(x)$, based on a confidence threshold θ as follows:

$$f(x) = \begin{cases} +1 \ (x \text{ is class1}) & \text{if } p(+1|x) > \theta, \\ -1 \ (x \text{ is class2}) & \text{if } p(-1|x) > \theta, \\ \text{reject} & \text{otherwise.} \end{cases}$$

[1] An implementation of SVM is employed to estimate posterior class probabilities.

The optimal confidence threshold θ for rejection is determined based on the risk $\mathcal{R}$ using the *trained* $f(x)$ and a validation set, where the $\mathcal{R}$ of CNN and SVM can both be defined as:

$$\mathcal{R}(f) = \sum_{i=1}^{m} \left(I((f(x_i)) \neq y_i) \wedge (f(x) \neq \text{reject}) + cI(f(x) = \text{reject}) \right),$$

for the LwR model, validation samples are used especially for tuning hyperparameters λ and λ' because LwR could determine its rejection rule by using only training samples.

The learning scenario

In these experiments, I follow this learning scenario. For each binary classification task, I divide the binary-labeled training samples into three sets: (1) A sample set for training a CNN. This CNN is used not only as the feature extractor for LwR and SVM but also as a CNN classifier. Note that this sample set will not be used in the following steps. (2) A sample set for training SVM and LwR with the features extracted by the trained CNN. (3) A sample set for validating the parameters of each method.

Feature Extraction from a Pre-trained CNN

As mentioned above, one of the key ideas of LwR is to use different feature spaces to construct a rejection function independent of the classification function. In this regard, the outputs of two different layers of a pre-trained CNN are utilized for the classification function $f(x)$ and the rejection function $r(x)$, as depicted in Fig. 3-1. Specifically, I employ the final convolutional layer of the CNN for classification. Notably, the SVM classifier uses exactly the same layer as its feature extractor. For the training of the rejection function, the second-to-last layer of the CNN is utilized, as it exhibited superior performance in our preliminary experiments. It has to be

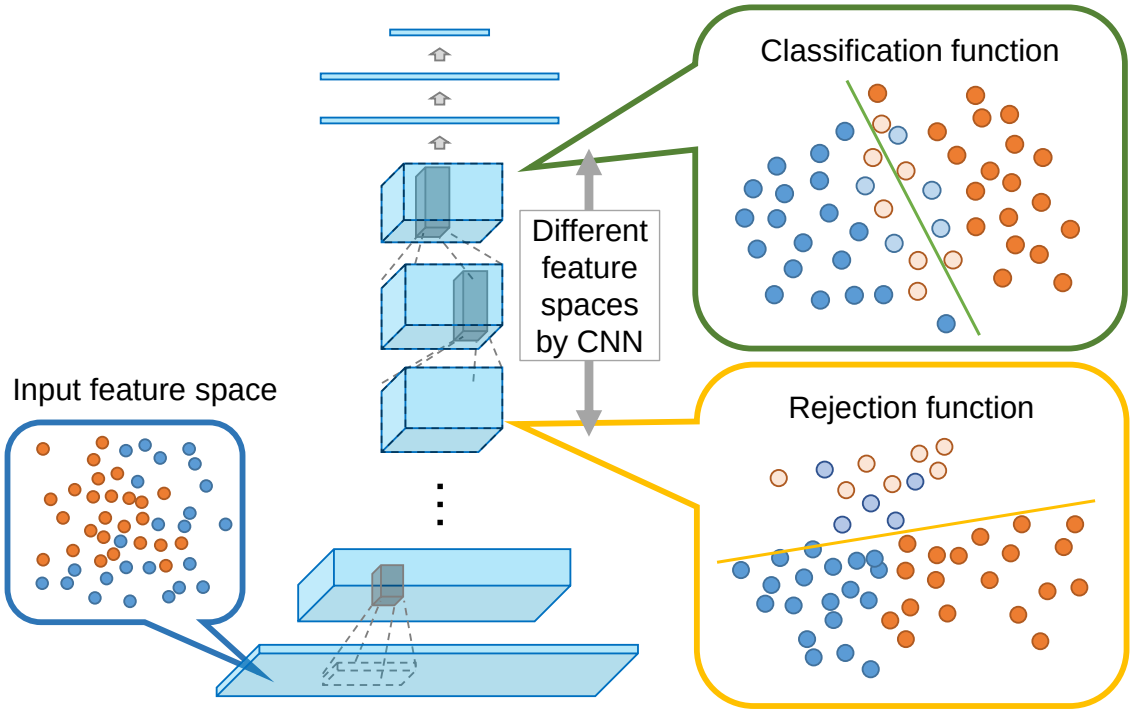


Figure 3-1: Extracting different features from CNN for LwR.

underlined that using different layers for classification and rejection functions prevents the LwR model from becoming an SVM-like model relying solely on one feature space.

Evaluation criteria

To assess the classification performance with the reject option for these three models using the test set, I consider the evaluation metric classification accuracy for *non-rejected* test samples:

$$\text{accuracy} = \frac{\sum_{x_i \in \{S - S_{rej}\}} I(f(x_i) = y_i)}{|S| - |S_{\text{rej}}|},$$

and the rejection rate:

$$\text{rejection rate} = \frac{|S_{\text{rej}}|}{|S|}.$$

Here, S represents the test set, S_{rej} denotes the set of rejected samples in the test set, f indicates the classification function of each method, and $f(x_i)$ represents the

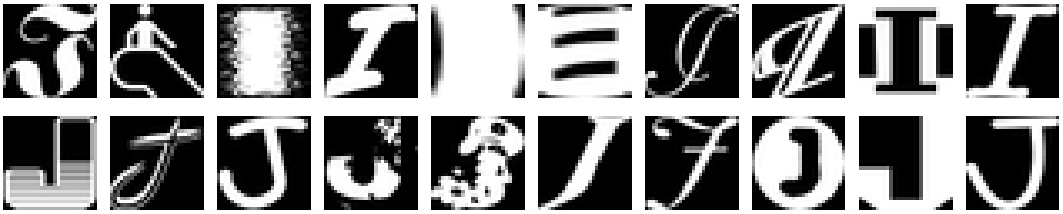


Figure 3-2: Examples of 'I' (upper row) and 'J' (lower row) form notMNIST.

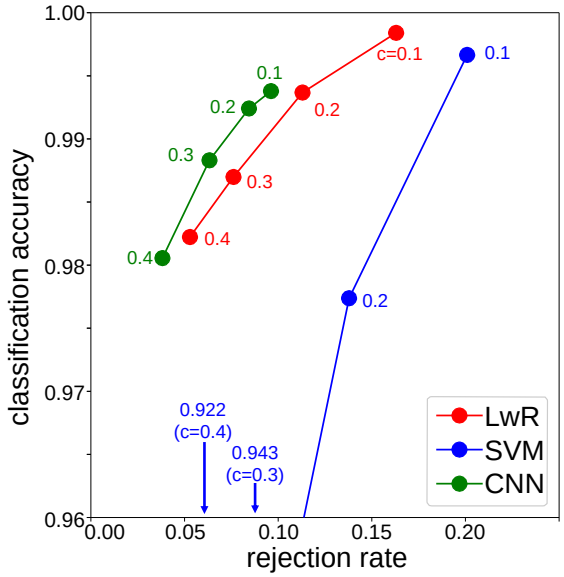


Figure 3-3: The relationship between accuracy and reject rate for the classification task between 'I' and 'J' from the notMNIST dataset.

classification result of sample x_i.

3.3.1 Classification of Similar Characters with Reject Option

In this section, I employ the dataset named notMNIST[2] for both training and test purposes. The notMNIST dataset comprises images of alphabets 'A' to 'J' rendered in various fonts. To challenge the proposed recognition models, I select a pair of highly similar letters, 'I' and 'J,' setting the stage for a demanding two-class recognition task where the option to reject ambiguous samples becomes crucial. The examples of

[2]https://www.kaggle.com/lubaroli/notmnist/

these characters are depicted in Fig. 3-2.

For the experiment, I randomly sampled 3,000 training instances each for the letters 'I' and 'J' from the notMNIST dataset. Among these, 2,000 samples, with an equal distribution of I and J, are designated for CNN training, another 2,000 for SVM and LwR training, and another 2,000 for validation purposes. Using these datasets, I created three classifiers: CNN with a reject option, SVM with a reject option, and LwR.

The architecture of the CNN used in this experiment is described as follows:

- Two consecutive convolutional layers, each equipped with three kernels of size 3×3 and a stride of 1.
- Two pooling layers with a kernel size of 2×2 and a stride of 2, each following a convolutional layer.
- Rectified Linear Units (ReLU) activation function after each pooling layer.
- At the end of the network, there are three consecutive fully connected layers with 3,136, 1,568, and 784 hidden units, respectively.

The training process involves the minimization of the cross-entropy loss. Figure 3-3 illustrates the relationship between the recognition accuracy and rejection rate, as evaluated on the test dataset. Notably, LwR outperforms the standard SVM with a reject option by a heuristically determined rejection threshold. Interestingly, CNN exhibits a slight advantage over LwR when the rejection rate is extremely low, potentially attributed to the CNN classifier, specifically the fully connected layers, being jointly trained and tuned alongside the feature extraction part of the CNN. However, the graph suggests that LwR might be more capable of rejecting potentially threatening samples than the CNN method. As showcased in the following section, it anticipates LwR to shine in a more realistic experiment.

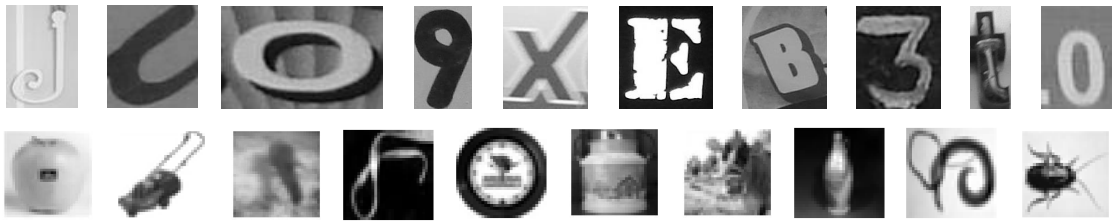


Figure 3-4: Examples from Chars74k (upper row) and CIFAR100 (lower row) datasets.

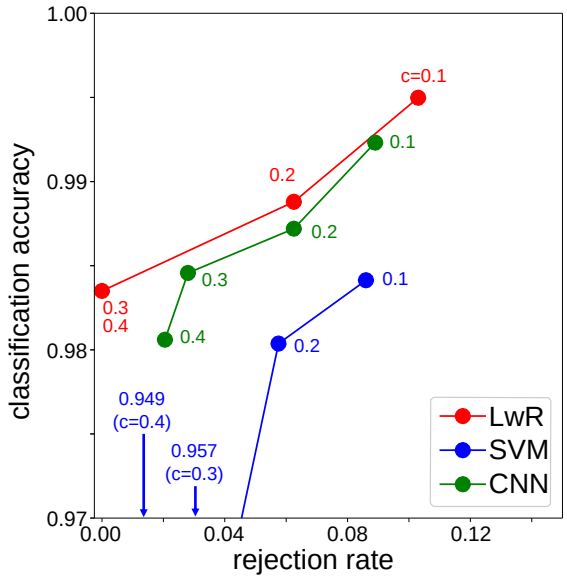


Figure 3-5: The relationship between accuracy and reject rate for character and non-character classification using Chars74k and CIFAR100.

3.3.2 Classification of Character and Non-character with Reject Option

In this section, I delve into a more practical application involving the classification of characters and non-characters—a task inherently challenging due to the ambiguity between these categories. Notably, this classification task plays a pivotal role in scene text detection [78], where distinguishing between characters and non-characters remains a formidable challenge due to the ambiguity between characters and non-characters.

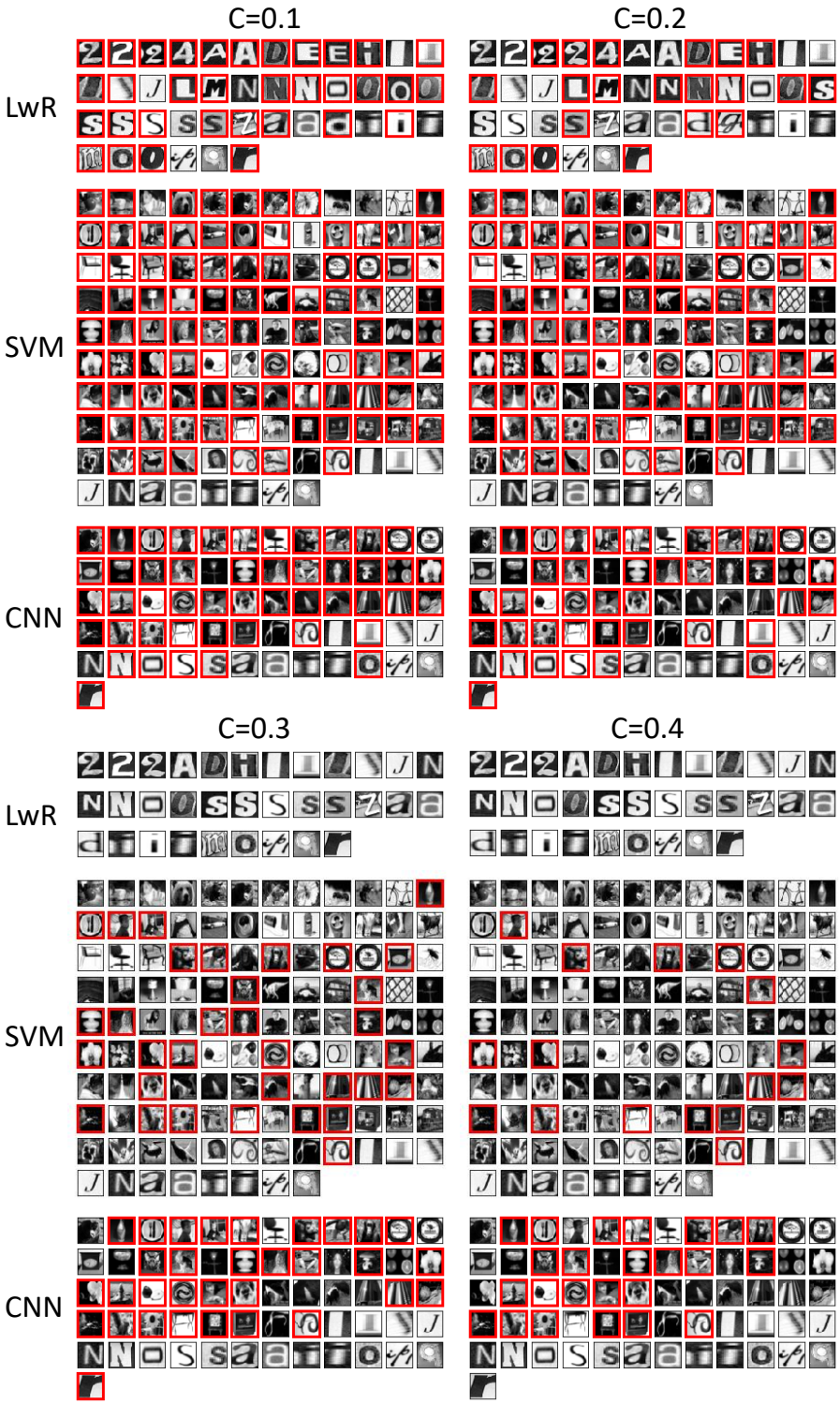


Figure 3-6: Test samples misclassified by the LwR, the standard SVM, and the CNN at different $c \in \{0.1, 0.2, 0.3, 0.4\}$. Rejected samples are highlighted in red.

For the experimentation, I turn to the Chars74k dataset[3], serving as the source for scene character images. This dataset encompasses a substantial collection of character images captured in natural scenes. In opposition, as the non-character dataset, I utilize the CIFAR100 dataset[4], known for its diverse array of objects commonly found in natural environments as well. All image samples from these datasets undergo grayscale conversion and resizing to dimensions of 32×32 pixels, as depicted in Fig. 3-4.

From both the CIFAR100 and Chars74k datasets, I randomly selected 5,000 images each for training of the experimentation. Among these, with an equal distribution of CIFAR100 and Chars74k datasets, 6,000 are allocated for CNN training, while 2,000 are reserved for SVM and LwR training. Moreover, an additional set of 2,000 samples is earmarked for validation purposes. After training, I obtained a CNN classifier, SVM with rejection thresholds, and the LwR model, and the CNN with the following architecture was used for this task:

- Convolutional layers with 32, 64, and 128 kernels, featuring kernel sizes of 3×3, 5×5, and 5×5, with stride values of 1, 2, and 2, respectively. Rectified Linear Units (ReLU) serve as the activation function.

- One max-pooling layer with a 2×2 sized kernel and stride 2.

- Two fully connected layers, boasting 512 and 2 units, with ReLU and softmax activations at the network's end.

The training process revolves around minimizing the cross-entropy loss. In Fig. 3-5, I present the test accuracy achieved using 2,000 random samples from the Chars74k and CIFAR100 datasets. Here, it becomes evident that LwR strikes an optimal balance between accuracy and rejection rate among the three methods. Surprisingly, for rejection cost parameters $c = 0.3$ and $c = 0.4$, LwR predicts labels for all test samples,

[3]http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/
[4]https://www.cs.toronto.edu/ kriz/cifar.html

meaning that no samples are rejected. Despite this, LwR maintains higher accuracy than CNN. Since excessive rejections result in higher losses, LwR avoids making unnecessary rejections. This outcome underscores the potency of the theoretically formulated LwR, which is designed to minimize risk (R) through its classification and rejection functions, while the reject option for CNN and SVM leans toward a heuristic approach.

Figure 3-6 provides visual examples of image samples that would be misclassified without the aid of the rejection function. These are the instances where relying solely on the classification function $f(x)$ would result in misclassification. Remarkably, the samples highlighted within red boxes represent those successfully rejected by the rejection function $r(x)$. These highlighted samples underscore the complementary nature of the classification function $f(x)$ and the rejection function $r(x)$. Additionally, it is worth noting that LwR rejects a distinct set of samples compared to SVM and CNN, as LwR employs different feature spaces for classification and rejection. This insight reveals the unique capabilities of LwR in enhancing classification accuracy by effectively managing the reject option.

3.3.3 Summary

In this chapter, I introduce a novel approach that harnesses the power of LwR. It devises a learnable optimal rejection method to filter out ambiguous samples through an independently learned rejection function. Notably, this rejection function is co-trained alongside a classification function equipped with deep features within the overarching LwR framework. Several key highlights of LwR are elucidated as follows. The first one is the theoretical foundation; unlike conventional heuristic rejection strategies, the LwR is grounded in machine learning theory, lending it a robust and principled foundation. The next one is its versatility of feature spaces, where training the rejection function in an arbitrary feature space is possible, distinct from the one used for classification. This flexibility empowers the LwR model to select a feature

space that is optimally suited for rejection purposes, a novel aspect that enhances its practical applicability.

Traditionally, LwR research has predominantly focused on its theoretical underpinnings. However, I propose a shift towards leveraging LwR for practical pattern classification tasks. Furthermore, I advocate for the use of features extracted from various CNN layers for both classification and rejection.

The primary objective of this chapter is to unveil the efficacy of the proposed rejection function, which confers the distinct advantage of bolstering decision-making reliability, especially in intricate character recognition tasks. I delve into the intricacies of optimizing the rejection function, providing a comprehensive exposition of its theoretical foundations. Encompassing notMNIST classification and character/non-character classification experiments, the investigations in this work firmly substantiate the superiority of the proposed method over conventional rejection strategies. This underscores the practical viability and tangible performance gains the proposed approach offers.

Chapter 4

Paired Contrastive Features (PCF)

4.1 Overview

This chapter introduces the concept of PCF within the context of highly reliable writer-independent signature verification tasks. As mentioned in section 1.4, PCF is represented as the concatenation of two contrasting features. Since in scenarios where the authentication of signatures is crucial and often demanding, the extraction of signature features needs to be treated with caution. Extracting contrastive features from the paired signatures being compared becomes paramount, especially in writer-independent scenarios where new writers lack sufficient training data. These contrastive features are the basis for determining whether the two signatures originate from the same writer. In this chapter, I leverage the power of contrastive features and propose the concept of PCF as a means to train more reliable models, ultimately enhancing the performance and reliability of signature recognition.

The primary objective of this chapter is to elucidate the principles underlying PCF, highlight its advantages, and explore its two distinct applications within various mechanisms. To begin, a comprehensive definition of PCF, accompanied by an explanation of its constituent elements, is provided. Subsequently, I delve into the methodology for generating PCFs, where I introduce the Siamese architecture as a

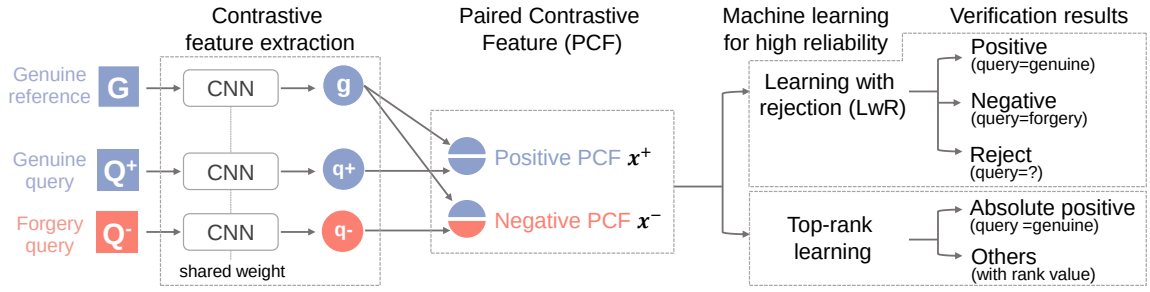


Figure 4-1: Overview of proposals of two machine learning frameworks with PCF: *learning with rejection* and *top-rank learning*, for highly reliable signature verification.

pivotal component of this process.

Following this, I discuss the integration of the proposed PCF with two reliable machine-learning frameworks: *Learning with Rejection (LwR)* and *Top-rank learning*. An overview of this process is illustrated in Fig.4-1.

Each PCF is constructed by leveraging the distinctive features of two signatures: the genuine reference signature G and the query signature Q. The underlying principle of PCF revolves around the training of these features to be contrastive, effectively amplifying the differences between genuine and forged signatures. Through this approach, PCF capitalizes on the inherent advantages of contrastive features to bolster the reliability and precision of signature verification.

Subsequently, a detailed account of the experimental applications of PCF within the context of signature verification tasks is presented. Particular emphasis is placed on its compatibility with writer-independent scenarios. To conclude, I thoroughly discuss the findings and acknowledge the limitations of this work.

4.2 Definition of PCF

In section 1, I concisely introduced the PCF, a foundational element within the proposed methodologies designed for tasks involving contrastive learning processes. PCF is structured as a feature vector comprising two contrasting components, as illustrated in Fig. 4-1. The subsequent elucidation defines PCF using the following components:

G represents a genuine reference signature while k denotes its true writer. Q^+ corresponds to a genuine query signature authored by writer k. Q^-, on the other hand, represents a forgery query signature, which is not authored by writer k. I denote the contrastive feature vectors of these components as g, q^+, and q^-, respectively.

Next, detailed information regarding extracting these contrastive features will be explained. As illustrated in Fig. 4-1, I mathematically define a positive PCF, denoted as $\boldsymbol{x}^+$, as the vector concatenation ($\oplus$) of g and q^+. Similarly, a negative PCF denoted as $\boldsymbol{x}^-$ is defined as the vector concatenation of g and q^-. It is important to note that, in this context, a portion of the genuine signatures is randomly selected to serve as reference signature G, while the remaining subset is utilized as Q^+, and notably, there is no overlapping between these two sets.

Given that the dataset originally does not categorize genuine signatures into G and Q^+, the formation of PCF necessitates reconsidering the new grouping's calculation for the number of training data. When there is a training dataset consisting of L_0 genuine references and L_g genuine queries for each of the K writers, a positive PCF set $\boldsymbol{\Omega}^+ = \{\boldsymbol{x}_i^+\}_{i=1}^m$, where $m = L_0 L_g K$ can be obtained. Likewise, when there are L_f forgery queries for each writer, I create a negative PCF set $\boldsymbol{\Omega}^- = \{\boldsymbol{x}_j^-\}_{j=1}^n$, where $n = L_0 L_f K$. It is crucial to highlight that the division of the data into training, validation, and test sets is writer-centric. This implies that a writer whose signatures are incorporated into the training dataset will not be represented in either the validation or test sets.

The essence of PCF generation lies in contrastive feature extraction, which aims to accentuate inter-writer dissimilarities while mitigating intra-writer similarities. In other words, the features extracted from g and q^+ should exhibit similarity, whereas those from g and q^- should manifest dissimilarity. This preprocessing step emphasizes the differences between different writers and reduces the influence of individual writing styles within PCF creation.

PCF is a critical foundation for enabling writer-independent signature verification

and facilitating the application of various machine-learning models. A straightforward approach involves using a standard binary classification model such as the SVM. In this approach, if a PCF is classified as positive, it indicates that the same writer authored both the query and the reference signatures. However, as explained in the next two sections, this work employs more advanced machine learning techniques to achieve a highly dependable signature verification system with PCF.

The overarching verification process with PCF follows these steps: First, I train a verification model using the combined dataset $\mathbf{\Omega} = \mathbf{\Omega}^+ \cup \mathbf{\Omega}^-$. Then, given a PCF $\boldsymbol{x} = g \oplus q$, the trained model determines whether the writer of g authored q. In the proposed writer-independent scenario, the writers of g and q are not part of the set of K writers from the training signatures. Consequently, the trained model is applicable to PCFs from arbitrary writers.

It is essential to emphasize that the proposed approach employs PCF in the "skilled-forgery-only" condition, wherein the forgery query Q^- is intentionally crafted to resemble the genuine reference G. This contrasts with previous signature verification approaches that often assume a more lenient "random forgery" condition, where any arbitrary forgery is considered. However, the "skilled-forgery-only" condition solely focuses on forgeries of the same signature as the genuine reference, rendering it significantly more challenging. Additionally, the forgeries in this condition must be visually similar to the genuine reference, further increasing the difficulty. Based on the above, this challenging condition is more practical and suitable for achieving reliable signature verification.

To implement contrastive feature extraction for PCF, I employ a Siamese network, which has also been utilized in SigNet [1], a state-of-the-art signature verification model in the skilled-forgery-only condition. The Siamese network is based on CNNs and utilizes the contrastive loss function:

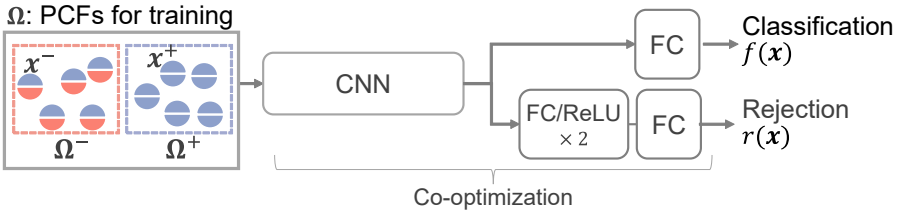


Figure 4-2: Reliable signature verification by LwR with PCF.

$$\mathcal{L}_{contrastive} = (1-z)D^2 + z\max(0, margin - D)^2. \tag{4.1}$$

The variable z in the function serves as a binary indicator, denoting whether two signatures belong to the same class or not. Meanwhile, D represents the Euclidean distance computed in the embedded feature space between two signatures, and the margin controls the acceptable range of distance for positive pairs, which is set to 1 in our case. This loss function encourages the reduction of feature distance between samples from the same class (i.e., the same writer in the proposed context) while maximizing the distance between samples from different classes. This network architecture effectively captures the discriminative features required for writer-independent signature verification.

4.3 PCF with Rejection Model

As introduced in section 1.4 and Fig. 1-5, LwR learns a rejection function and its rejection feature space simultaneously with a classification function and its classification feature space. Consequently, ambiguous samples are rejected to guarantee that the non-rejected samples are classified with high reliability.

As detailed in section 2.1.1, LwR offers multiple approaches for its formulation, distinct from the QP learning strategy [24] employed in the previous chapter. In this context, I employ the model structure of SelectiveNet [37], which differs from

the network used in Chapter 3. SelectiveNet follows a one-step learning approach, optimizing feature mapping for both classification and rejection. It utilizes a softmax activation function at the final layer to transform negative values into positive ones. This modification shifts the threshold for determining whether a prediction should be rejected in Eq. 3.1 from 0 to 0.5.

The proposed rejection model incorporates two distinct output channels for classification and rejection, denoted as $f_{\text{pair}}(\boldsymbol{x})$ and $r_{\text{pair}}(\boldsymbol{x})$, respectively. Specifically, $f_{\text{pair}}(\boldsymbol{x})$ is a binary classification function with values in the set $\{0, 1\}$, while $r_{\text{pair}}(\boldsymbol{x})$ represents a continuous rejection function with values within the interval $[0, 1]$. A fundamental aspect of the proposed methodology is the following decision rule:

$$(f_{\text{pair}}, r_{\text{pair}})(\boldsymbol{x}) = \begin{cases} f_{\text{pair}}(\boldsymbol{x}) & \text{if } r_{\text{pair}}(\boldsymbol{x}) \geq 0.5, \\ \text{reject} & \text{if } r_{\text{pair}}(\boldsymbol{x}) < 0.5. \end{cases} \tag{4.2}$$

Figure 4-2 shows the network structure for co-training two functions $f_{\text{pair}}(\boldsymbol{x})$ and $r_{\text{pair}}(\boldsymbol{x})$ with PCF. Given a PCF input $\boldsymbol{x}$, the network outputs the predictions of $f_{\text{pair}}(\boldsymbol{x})$ and $r_{\text{pair}}(\boldsymbol{x})$. These two functions are co-trained with the representations for $\boldsymbol{x}$ by the CNN in Fig. 4-2. As shown in Fig. 4-2, $r_{\text{pair}}(\boldsymbol{x})$ employs extra fully connected layers with ReLU. Thus, the feature space of $r_{\text{pair}}(\boldsymbol{x})$ is different from the classification feature space of $f_{\text{pair}}(\boldsymbol{x})$.

In accordance with the training methodology employed in SelectiveNet [37], I incorporate a dual loss function strategy, consisting of $\mathcal{L}_1$ and $\mathcal{L}_2$, to facilitate the optimization of the neural network model. The primary loss function, denoted as $\mathcal{L}_1$, pertains to the concept of rejection and is formally defined as:

$$\mathcal{L}_1 = \mathcal{R}_{\text{pair}} + \kappa \max(0, v - \mathcal{V})^2, \tag{4.3}$$

where κ is a hyper-parameter that controls the strength of the soft-constraint term

$\max(0, v-\mathcal{V})$, which in the above equation, this term is squared for strictly penalizing the case of $v > \mathcal{V}$.

Furthermore, the parameter v represents the desired coverage level the rejection function aims to maintain under ideal circumstances. In contrast, the actual *coverage* value $\mathcal{V}$ quantifies the proportion of samples that are not subject to rejection:

$$\mathcal{V} = \frac{1}{m+n} \sum_{i=1}^{m+n} r_{\text{pair}}(\boldsymbol{x}_i). \tag{4.4}$$

The left-hand side of Eq. 4.3 signifies the "rejection risk," denoted as $\mathcal{R}_{\text{pair}}$, which is defined as follows:

$$\mathcal{R}_{\text{pair}} = \frac{1}{m+n} \sum_{i=1}^{m+n} l_{CE}(f_{\text{pair}}(\boldsymbol{x}_i), y_i) r_{\text{pair}}(\boldsymbol{x}_i)/\mathcal{V}. \tag{4.5}$$

In this equation, m corresponds to the count of positive PCF ($\boldsymbol{\Omega}^+$), and n denotes the count of negative PCF ($\boldsymbol{\Omega}^-$). The term $l_{CE}(z)$ signifies a cross-entropy loss, and consequently, $l_{CE}(f_{\text{pair}}(\boldsymbol{x}_i), y_i) r_{\text{pair}}(\boldsymbol{x}_i)$ quantifies the loss associated with retaining a misclassified sample $\boldsymbol{x}_i$. Further, the division by $\mathcal{V}$ is essential to prevent a trivial solution in which $r_{\text{pair}}(\boldsymbol{x}_i) \equiv 0$, signifying the rejection of all samples.

Equation 4.3 is derived by transforming the following hard-constrained minimization problem into a soft-constrained problem:

$$\min \mathcal{R}_{\text{pair}} \text{ subject to } \mathcal{V} \geq v. \tag{4.6}$$

This transformation is introduced to circumvent excessive rejections, which could lead to the actual coverage falling below the desired target coverage.

The other loss function, denoted as $\mathcal{L}_2$, is introduced to independently optimize the classifier f over all samples and is defined as:

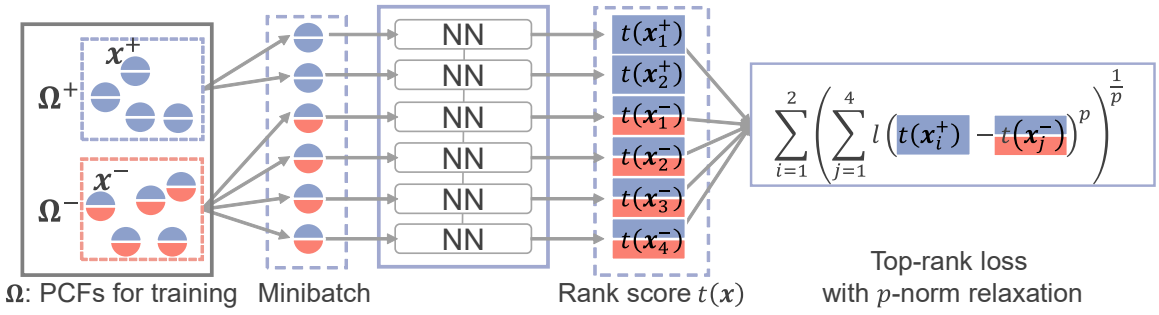


Figure 4-3: Training process of top-rank learning with PCF.

$$\mathcal{L}_2 = \frac{1}{m+n}\sum_{i=1}^{m+n} l_{CE}(f_{\text{pair}}(\boldsymbol{x}_i), y_i). \tag{4.7}$$

Finally, by combining $\mathcal{L}_1$ and $\mathcal{L}_2$, the overall loss is formulated as:

$$\mathcal{L}_{\text{overall}} = \rho\mathcal{L}_1 + (1-\rho)\mathcal{L}_2, \tag{4.8}$$

where $\rho \in [0, 1]$ serves as a hyper-parameter to balance the two loss functions, this comprehensive loss function, in comparison to the loss function introduced in the previous chapter, offers more precise control over the rejection coverage by adjusting the parameter v instead of the cost parameter v. Additionally, the inclusion of $\mathcal{L}_2$ places greater emphasis on the general performance of the classification function, thus preventing the model from converging to a local minimum prematurely during the learning of the rejection function.

4.4 PCF with Top-rank Learning

In this section, I introduce the concept of top-rank learning as a framework for enhancing the reliability of signature verification within the PCF context. As highlighted in Chapter 1 and illustrated in Fig. 1-6, the primary objective of top-rank learning can

be defined as follows:

$$\text{pos@top} = \frac{1}{m}\sum_{i=1}^{m} I\left(t(\boldsymbol{x}_i^+) > \max_{1\le j\le n} t(\boldsymbol{x}_j^-)\right), \tag{4.9}$$

where $I(\cdot)$ denotes an indicator function. Pos@top represents the "absolute positives" ratio to the total number of positive samples (m). Absolute positives are positive samples whose ranking scores exceed those of any negative samples. In other words, a sample $\boldsymbol{x}_i^+$ is classified as an absolute positive if it satisfies the condition specified in the indicator function of Equation (4.9). These absolute positives are considered highly reliable positive samples since they outrank even the "top-ranked negative," defined as $\max_{1\le j\le n} t(\boldsymbol{x}_j^-)$.

One of the notable advantages of top-rank learning with PCF is its capability to enable stringent and highly reliable signature verification. The procedure involves first obtaining a ranking function $t(\boldsymbol{x})$ that maximizes pos@top. Subsequently, a top-ranked negative sample is identified, and its rank score, denoted as t_{topneg}, is determined for a given test dataset. Finally, when a PCF $\boldsymbol{x} = g \oplus q$ qualifies as an absolute positive (i.e., $t(\boldsymbol{x}) > t_{\text{topneg}}$), I confidently verify $\boldsymbol{x}$ as a positive PCF and its corresponding query Q as genuine.

This verification scheme offers two crucial benefits regarding the threshold t_{topneg}:

- Automated Threshold Determination: The value of t_{topneg} is automatically determined during the pos@top maximization process with respect to the ranking function t. This eliminates the need for heuristic or naive threshold-setting methods.

- Dependency on a Single Negative Sample: t_{topneg} relies on a single negative sample, simplifying the threshold determination process. Consequently, even highly skilled forgeries that closely mimic genuine signatures may result in hard-negative samples with high t_{topneg} scores, making it challenging for genuine pairs to be classified as absolute positives. This property reinforces the system's

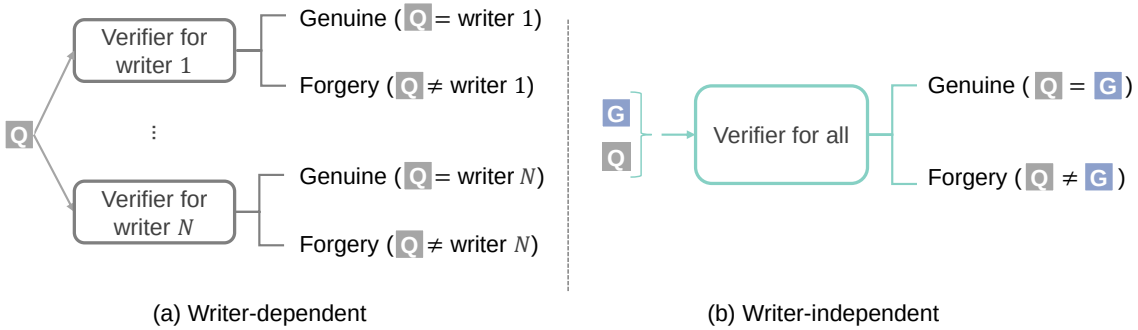
reliability, even in conditions dominated by skilled forgeries. Additionally, this property also addresses situations with significant intra-writer variability, where positive PCFs may not exhibit distinctly higher rank values compared to hard-negative PCFs. Such scenarios will be explored further in the later experiments with UTSig.

Figure 4-3 illustrates the training process of top-rank learning with PCF. Each PCF in a minibatch undergoes non-linear transformation and rank score calculation within a network model. While the primary goal of top-rank learning is to maximize pos@top as per Equation (4.9), direct maximization often leads to overfitting. To mitigate this issue, I employ the p-norm relaxation technique (as described in [50, 42]) to convert Equation (4.9) into a more suitable loss function:

$$\mathcal{L}_{\text{TopRank}} = \frac{1}{m}\sum_{i=1}^{m}\left(\sum_{j=1}^{n}\left(l(t(\boldsymbol{x}_i^+) - t(\boldsymbol{x}_j^-))\right)^p\right)^{\frac{1}{p}}, \tag{4.10}$$

where $l(z) = \log(1 + e^{-z})$ represents a surrogate loss function.

By setting $p = \infty$, Equation (4.10) reduces to Equation (4.9). However, selecting a relatively smaller value for p (e.g., 4, 8, or 16) ensures that Equation (4.10) exerts a milder influence on the maximization of pos@top, thereby mitigating the risk of overfitting. Another key technique for stable training involves organizing mini-batches to address sample imbalances. During training, positive samples in each minibatch should be ranked higher than the top negative samples within that minibatch. Consequently, each minibatch should include a sufficient number of negative samples to ensure the presence of a negative sample that closely approximates the top-ranked negative sample for effective training.



(a) Writer-dependent (b) Writer-independent

Figure 4-4: Two scenarios of signature verification. 'Q' and 'G' refer to query and genuine reference signatures, respectively.

4.5 Application to Signature Verification

The application of PCF is particularly compelling in the domain of signature verification, given its prevalent use in financial transactions and the handling of personal information. In this context, the need for the utmost reliability is paramount [68, 79]. In the context of signature verification, two primary scenarios are commonly encountered, each presenting distinct requirements and challenges, as visually represented in Fig. 4-4.

As shown in Fig. 4-4 (a), the writer-dependent scenario mandates the development of individual models for each unique writer, necessitating the maintenance of substantial databases containing writer-specific signatures. In contrast, the writer-independent scenario, as illustrated in Fig. 4-4 (b), seeks to establish a universal model applicable to all writers, offering a practical and desirable approach. From this comparison, it becomes evident that writer-independent methods demand less effort in terms of training models for individual users and do not require additional data when new users join the system. These advantages render writer-independent methods more practical despite their intrinsic challenges compared to writer-dependent methods. In this thesis, my primary focus is on investigating the utilization of PCF in conjunction with reliable pattern recognition strategies within writer-independent scenarios.

4.5.1 Datasets

To evaluate the performance of the reliable writer-independent signature verification with PCF, I applied them to three commonly used signature datasets: BHSig-B, BHSig-H, and UTSig. Figure 4-5 shows examples of genuine and forgery from these three datasets.

BHSig-B and **BHSig-H** represent two subsets within the BHSig260 dataset, which is available for downloading[1]. BHSig-B contains Bengali signature images, while BHSig-H contains Hindi signature images. Each subset includes 24 genuine signatures and 30 skilled forgery signatures for each writer. In total, BHSig-B comprises 100 writers, while BHSig-H consists of 160 writers.

The **UTSig** dataset is a collection of Persian signatures obtained from university students, also accessible for downloading[2]. This dataset encompasses 115 distinct writers, each contributing 27 genuine signatures, three opposite-hand signatures, six specially crafted skilled forgeries by experts, and 36 skilled forgeries by individuals without expert training. As per precedent in the literature [80], I considered all 45 forged signatures as skilled forgery samples.

The selection of these datasets is motivated by the emphasis on the "skilled-forgery-only" condition, necessitating datasets with a substantial number of skilled-forgery images. In contrast to prior studies, where the "random-forgery" condition often prevails, involving the mixing of several relatively small datasets to generate random forgeries, our approach adheres to the "skilled-forgery-only" condition. This choice aligns with our primary focus on achieving high reliability in tackling a challenging verification task.

The dataset samples are used in the following manner. Initially, an equal number of positive pairs (G, Q^+) and negative pairs (G, Q^-) were generated to derive $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$, respectively. Specifically, in the cases of BHSig-B and BHSig-H, where there are

[1] `http://www.gpds.ulpgc.es/download`
[2] `http://mlcm.ut.ac.ir/Datasets.html`

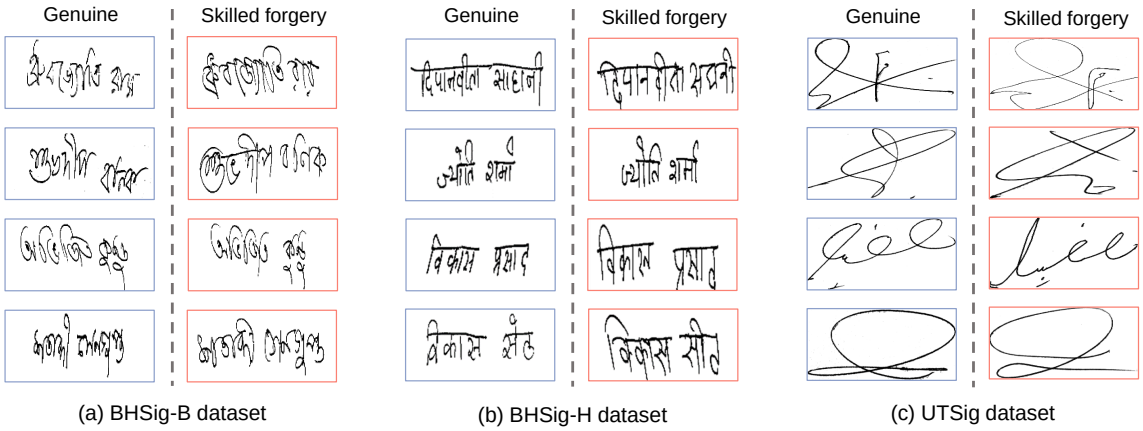


Figure 4-5: Examples of genuine and forgery signatures from each dataset.

276 positive and 720 negative pairs for each writer, I randomly selected 276 pairs from the 720, ensuring no duplication. Subsequently, these pairs were divided into training, validation, and test sets at an 8:1:1 ratio based on writers, with the validation set being utilized for early stopping and hyper-parameter tuning. Consequently, the sets are writer-disjoint, meaning no writer's samples appear in more than one of these sets. It is worth noting that the performance evaluation was conducted by averaging the results over ten random data splits. Further, before the training process, each sample was resized to dimensions of 155×220 pixels.

4.5.2 Implementation details

Baseline metric leaning with contrastive loss

This study adopted SigNet [1] as the baseline metric learning model, utilizing the contrastive loss function as originally proposed. I closely followed the setup described in the SigNet paper, with one minor alteration. Instead of employing a fixed number of training epochs, as was done in the original setup, I utilized the validation set to determine the optimal number of training epochs dynamically. This modification slightly improved baseline accuracy, for example, increasing the accuracy from 0.806 to 0.861 for the BHSig-B dataset.

Preparing PCFs

The trained baseline, i.e., the SigNet model, was also used to generate contrastive features for PCFs. Specifically, I extracted features (g, q^+, and q^-) from the second-to-last layer of the SigNet architecture. These features were then concatenated to form positive and negative PCFs ($\boldsymbol{x}^+$ and $\boldsymbol{x}^-$), as illustrated in Fig. 4-1. Each feature vector had a dimensionality of 1,024, resulting in PCFs that were 2,048-dimensional vectors.

Preparing the LwR model

For the LwR model, a configuration as depicted in Fig. 4-2 was employed. In this setup, SelectiveNet [37] was integrated with PCFs, where the CNN utilized in Fig. 4-2 was based on the VGG-16 architecture. I leveraged the validation set to determine key hyper-parameters during the training process. Among those hyper-parameters, the parameter ρ, chosen from candidates $\{0.5, 0.6, 0.7\}$, controlled the SelectiveNet's behavior. Additionally, I experimented with different values for the hyper-parameter v, which governed the minimum ratio of non-rejected samples. Specifically, I considered values from the set $\{0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.0\}$. The network's training used stochastic gradient descent (SGD).

For evaluation, I compared the performance of the proposed model with SigNet [1], which represents the state-of-the-art in writer-independent verification under skilled-forgery conditions. I also introduced a variation of SigNet, denoted as "SigNet+thre," which incorporates a threshold-based rejection mechanism. This method rejects a sample pair if the output score of SigNet falls within an interval $[\theta_1, \theta_2]$. These two thresholds, θ_1 and θ_2, were determined using a grid search approach. Two criteria guided our threshold selection: first, ensuring that the ratio of non-rejected samples was greater than the target coverage v, and second, achieving the highest accuracy for the non-rejected samples.

Table 4.1: Quantitative evaluation results are presented for SigNet [1], SigNet with the reject option ("SigNet+thre"), and our LwR model with PCFs. Averages and standard deviations are calculated from ten random data splits. Notably, SigNet is a leading method in skilled-forgery-only conditions. The "Coverage $\mathcal{V}$ ($v = 0.7$)" column indicates the percentage of coverage in the test set when v is set to 0.7 in the validation set.

Dataset	Methods	Accuracy (↑)	AUC (↑)	EER (↓)	Coverage $\mathcal{V}$ ($v = 0.7$)
BHSig-B	SigNet	0.861±0.040	0.935±0.031	0.137±0.042	no rejection
	SigNet+thre	0.929±0.037	0.960±0.026	0.079±0.038	0.748±0.053
	LwR	**0.945±0.036**	**0.976±0.024**	**0.061±0.047**	0.735±0.076
BHSig-H	SigNet	0.835±0.028	0.916±0.024	0.164±0.029	no rejection
	SigNet+thre	0.912±0.028	0.952±0.024	0.096±0.025	0.692±0.077
	LwR	**0.926±0.027**	**0.959±0.018**	**0.088±0.031**	0.631±0.121
UTSig	SigNet	0.670±0.014	0.744±0.021	0.323±0.017	no rejection
	SigNet+thre	0.715±0.025	0.776±0.023	**0.285±0.024**	0.757±0.029
	LwR	**0.727±0.030**	**0.778±0.038**	0.297±0.036	0.700±0.094

Preparing the top-rank learning model

As shown in Fig. 4-3, the inputs of the top-rank learning model are the PCFs. As for the network model (depicted as “NN” in Fig. 4-3), four fully connected layers with ReLU are employed. During the training, hyper-parameter p was determined by the validation set from candidates {2, 4, 8, 16, 32}. Furthermore, each minibatch contains five positive and 40 negative samples because each minibatch should be imbalanced to incorporate as many negative samples as possible to approximate the top-ranked negative. Here, SGD was used to train the top-rank learning network.

Evaluation metrics

In this section, I detail the evaluation metrics employed to assess the performance of the verification models. These metrics provide a comprehensive overview of the models’ effectiveness.

Accuracy measures the ratio of correctly classified samples. It serves as a fundamental indicator of the model’s overall performance. **AUC** quantifies the area under

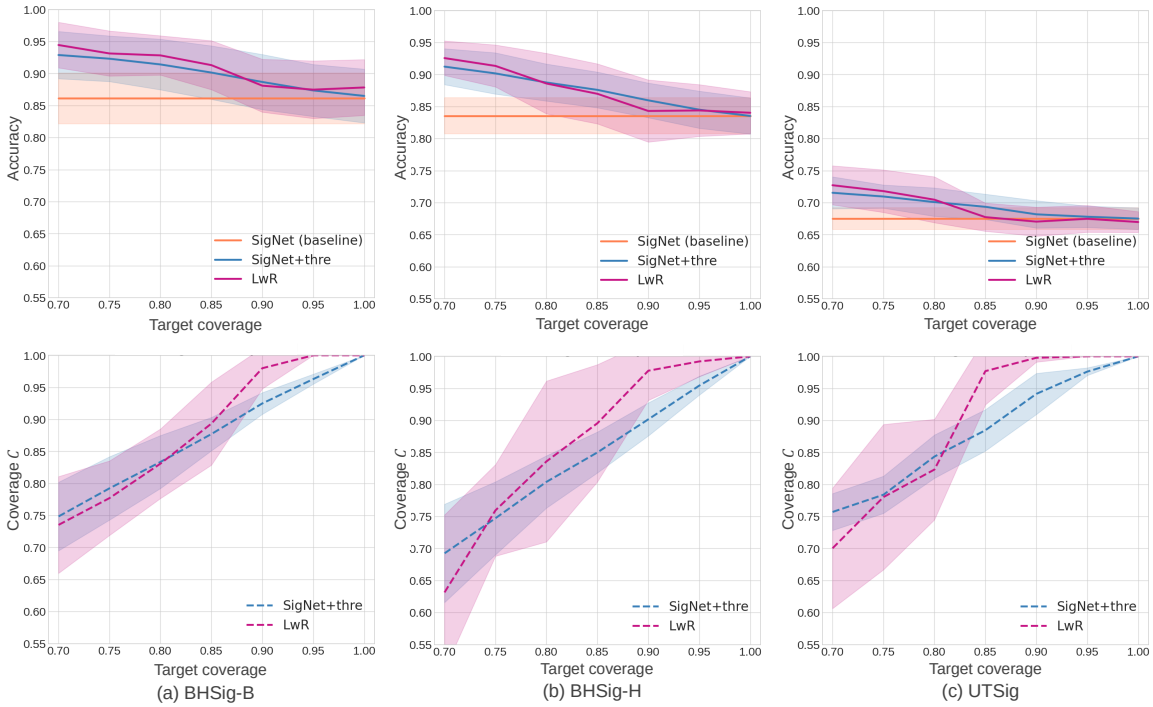


Figure 4-6: Average accuracy and the actual coverage $\mathcal{V}$ at different target coverage on (a) BHSig-B, (b) BHSig-H, and (c) UTSig datasets in ten-time random data splittings. The ranges within their standard deviations are filled up with lighter colors.

the Receiver Operating Characteristic (ROC) curve. It offers insights into the models' discrimination capabilities across different classification thresholds. **EER** is defined as the point at which the False Acceptance Rate (FAR) and False Rejection Rate (FRR) are equal. It critically assesses the models' balance between false acceptances and false rejections. **pos@top** represents the ratio of the absolute positives to all positives in the test set. It's important to note that the top-ranked negative sample within the test set defines the absolute positives.

For the specific models "LwR" and "SigNet+thre," the following considerations apply: Accuracy, AUC, and EER were computed while excluding rejected samples. This exclusion is particularly relevant for these models because they handle rejected cases.

4.5.3 Results with LwR

Quantitative evaluation of LwR

Table 4.1 presents the quantitative evaluation results for SigNet, "SigNet+thre," and the proposed LwR model with PCFs across various datasets. In this table, I maintained a target coverage of $v = 0.7$. For "SigNet+thre," the target coverage was determined as explained in section 4.5.2.

Table 4.1 confirms several noteworthy findings: First, as expected, reject options can increase the reliability of the verification result. By rejecting around 30% samples, both LwR and "SigNet+thre" achieved markedly higher accuracies than the original SigNet. Notably, the positive impact of rejection is most pronounced in the BHSig-B and BHSig-H datasets. Second, LwR achieved higher accuracies than "SigNet+thre". LwR consistently outperformed "SigNet+thre" in terms of accuracy. This finding underscores the effectiveness of the proposed LwR framework with PCFs, which can efficiently reject ambiguous samples by leveraging its learnability of the rejection function[3].

The usefulness of the rejection function $r(\boldsymbol{x})$ in our LwR model was further confirmed by other detailed observations as follows. First, $r(\boldsymbol{x})$ could reject more samples that were better off being rejected. Specifically in the case of BHSig-B, 61% of the misclassified samples of SigNet were rejected by $r(\boldsymbol{x})$, whereas 56% by "SigNet+thre." This indicates that the proposed $r(\boldsymbol{x})$ effectively identifies and rejects ambiguous samples (i.e., PCFs) to improve the reliability of the verification results. The second observation relates to the collaboration between the classification function $f(\boldsymbol{x})$ and the rejection function $r(\boldsymbol{x})$. When decisions were solely based on the classification function $f(\boldsymbol{x})$ (instead of Eq.(4.2)), a certain number of misclassified samples per-

[3]To evaluate the proposed methods thoroughly, an additional experiment on dataset BHSig-B was conducted following the ICDAR2021 competition guidelines [81], where skilled forgery and random forgery signatures were simultaneously included during training and evaluated separately. The results show a high accuracy of 0.976 for skilled forgery (coverage=0.7) and a notable accuracy of 0.927 for random forgery. Importantly, there was no significant degradation by mixing skilled and random forgeries for training.

sisted. However, $r(\boldsymbol{x})$ successfully rejected approximately 71% of these misclassified samples, highlighting the effectiveness of co-training f and r.

Figures 4-6 (a), (b), and (c) depict the accuracy and actual coverage $\mathcal{V}$ at different target coverage levels $v \in 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.0$ for BHSig-B, BHSig-H, and UTSig, respectively. Several key observations emerge from these graphs: First, the graphs clearly demonstrate that our LwR model consistently achieves equivalent or superior accuracy compared to SigNet, represented by the orange horizontal line, at any given v.

Second, for lower values of v ($c = 0.7 \sim 0.85$), the proposed LwR model consistently outperforms "SigNet+thre" in accuracy, despite having similar actual coverages $\mathcal{V}$. Remarkably, as v increases ($c = 0.85 \sim 1.0$), our model rejects fewer samples than specified while maintaining nearly the same accuracy as "SigNet+thre." These trends underscore the meaningful contribution of our rejection function r to achieving reliable verification results.

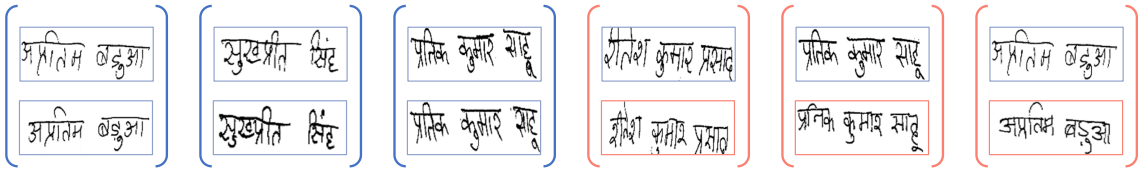
Qualitative evaluation of LwR

Figure 4-7 provides insights into the performance of the proposed LwR model by showcasing rejected pairs that were initially misclassified by SigNet and not rejected by "SigNet+thre." These pairs represent cases where SigNet failed to reject with a threshold-based approach. From the figure, I can deduce the following key observations:

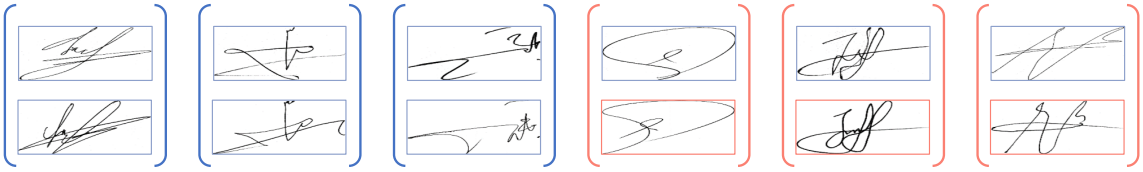
- The proposed LwR with PCF model appropriately rejects negative pairs $\boldsymbol{x}^-$ (specifically those composed of genuine (G) and skilled-forgery (Q^-) signatures, in the red brackets) even when Q^- shows high similarity with G. Note again that "SigNet+thre" does not possess the capability to reject these negative pairs. This proves the superiority of the rejection ability of LwR, whose rejection function is co-trained to minimize misclassifications.

- The proposed LwR with PCF model could also reject positive pairs $\boldsymbol{x}^+$ (in

(a) Examples from BHSig-B



(b) Examples from BHSig-H



(c) Examples from UTSig

Figure 4-7: Examples of signature pairs that LwR rejects. Like Fig. 4-5, blue signatures are genuine (G or Q^+) and red are skilled-forgery (Q^-). Accordingly, the blue bracket indicates a positive pair for $\boldsymbol{x}^+$, and the red indicates a negative pair for $\boldsymbol{x}^-$. All these signature pairs are "unwanted survivors" from "SigNet+thre;" they were misclassified by SigNet but not rejected by "SigNet+thre."

the blue brackets) to keep the high reliability of its verification results. In other words, the proposed model is sensitive to intra-writer differences and thus willing to reject positive pairs when the intra-writer differences are close to the differences with skilled forgeries. (Note again that these positive pairs were misclassified as negative pairs by SigNet.)

4.5.4 Results with top-rank learning

Quantitative evaluation of top-rank learning

Table 4.2 shows the quantitative evaluation results of the original SigNet and the proposed top-rank learning model with PCFs. The proposed model with PCFs consistently achieved higher pos@top than SigNet across all three datasets, emphasizing

Table 4.2: The quantitative evaluation results of our top-rank learning with PCFs. The fact that UTSig shows a low pos@top proves the signatures in UTSig are very unstable and, thus, difficult to achieve highly reliable verification. Note again that SigNet is the state-of-the-art method in the skilled-forgery-only condition. All values are the average of ten random data splittings with their standard deviation.

Dataset	Methods	pos@top (↑)	Accuracy (↑)	AUC (↑)	EER (↓)
BHSig-B	SigNet	0.147±0.135	0.861±0.040	0.935±0.031	0.137±0.042
	Top	**0.390±0.128**	**0.882±0.042**	**0.956±0.024**	**0.112±0.041**
BHSig-H	SigNet	0.091±0.072	0.835±0.028	0.916±0.024	0.164±0.029
	Top	**0.092±0.086**	**0.838±0.034**	**0.925±0.026**	**0.153±0.032**
UTSig	SigNet	0.001±0.001	**0.670±0.014**	**0.744±0.021**	**0.323±0.017**
	Top	**0.005±0.005**	0.642±0.028	0.697±0.075	0.345±0.027

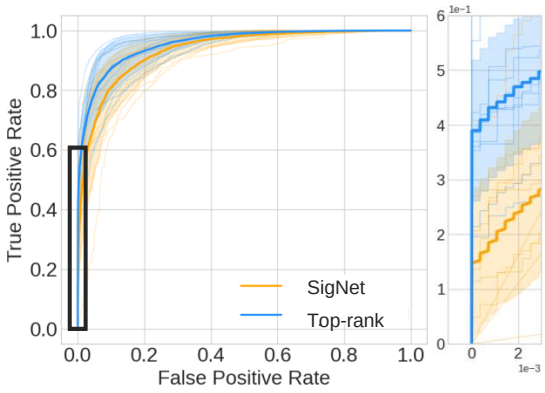
its ability to identify highly reliable genuine signatures as absolute positives[4]. I will show examples of absolute positive samples in a later section.

Figure 4-8 shows ROC curves for the three datasets, focusing on the leftmost vertical position, representing the number of absolute positives to all positives (pos@top). The beginning part of each curve is zoomed in to observe its leftmost vertical position. The proposed model exhibits s a keener rise at the beginning of the ROC curve than SigNet for all datasets. This observation also proves its effectiveness in identifying more highly reliable signatures as absolute positives.

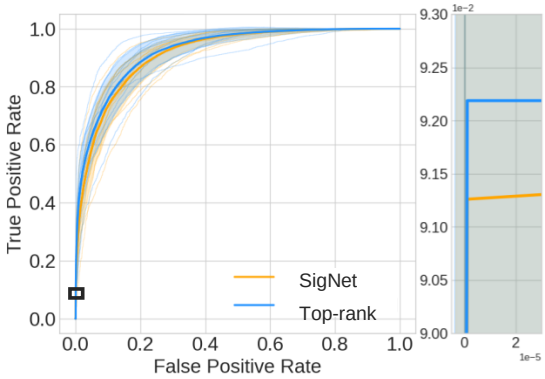
Importantly, the proposed top-rank learning model not only excels in pos@top but also outperforms SigNet in accuracy, AUC, and EER on both the BHSig-B and BHSig-H datasets.

This accomplishment is noteworthy due to its success in optimizing top-rank learning models, which primarily aim to maximize the count of absolute positives.

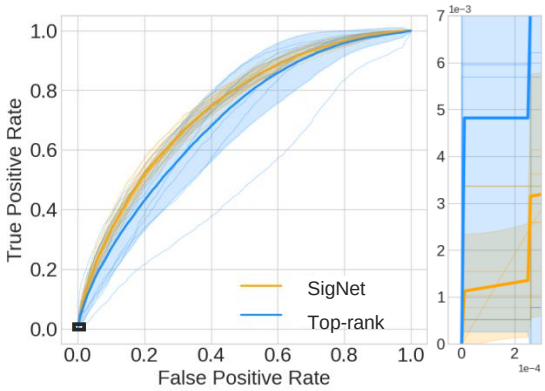
[4]To ensure a fair comparison, a normal learning-to-rank method was also evaluated using ranking scores for optimization. Using the same model structure, this comparative method achieved a preliminary pos@top of 0.09. Moreover, In accordance with the training protocol outlined in the ICDAR2021 competition guidelines [81], I conducted an additional experiment on dataset BHSig-B that incorporates a combination of skilled and random forgeries during the training phase, followed by their respective evaluation. The pos@top values obtained are as follows: skilled forgery achieved 0.435, while random forgery achieved 0.005. Due to the feature extractor's specialization for skilled forgery, achieving a high pos@top in random forgery scenarios without modifying the feature extractor is challenging.



(a) BHSig-B dataset



(b) BHSig-H dataset



(c) UTSig dataset

Figure 4-8: The ROC curves on (a) BHSig-B, (b) BHSig-H, and (c) USTig datasets. Ten random data splittings (the thin lines) and their average (the bold line) of both SigNet and top-rank learning are shown. The ranges of standard deviations are filled up with lighter colors around the average ROC curves. The beginning parts of the ROC curves are zoomed in to observe their vertical parts that correspond to absolute positives.

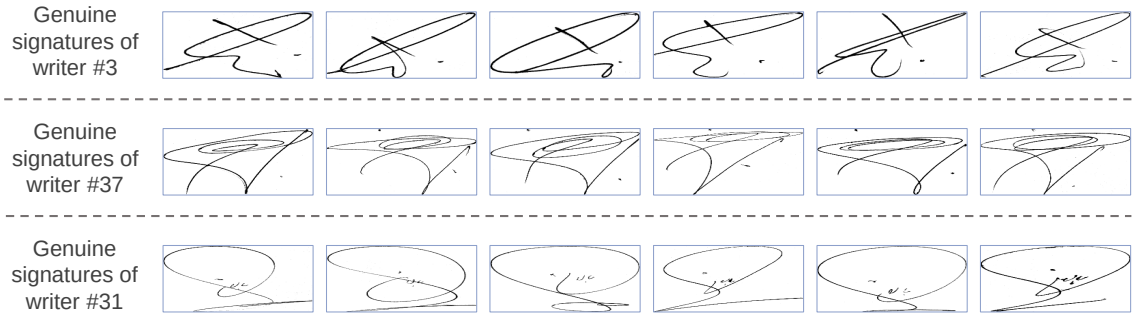


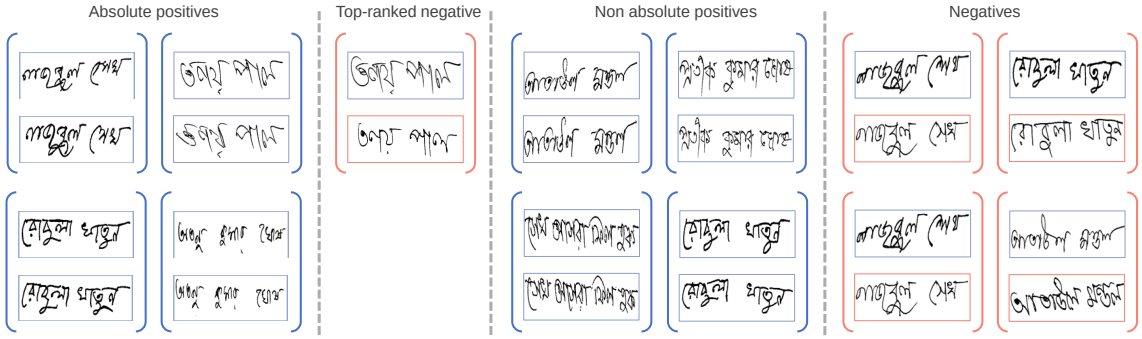
Figure 4-9: The reason that UTSig cannot achieve higher pos@top. It is possible to observe how the genuine signatures of three randomly chosen writers fluctuated largely.

Although there was a potential concern regarding performance degradation in other evaluation metrics, the proposed model not only mitigated these concerns but also surpassed SigNet in these diverse metrics, demonstrating its state-of-the-art performance.

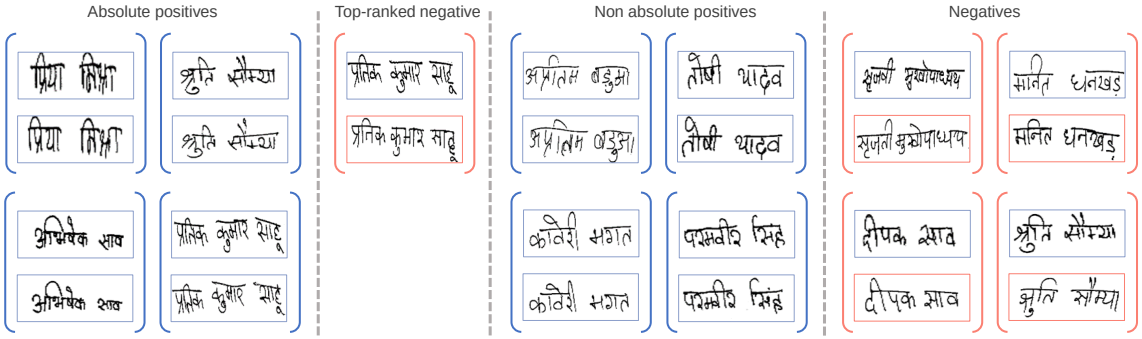
Exploring Low pos@top in UTSig

The low pos@top value observed in UTSig (Table 4.2) highlights the inherent challenges of this dataset in achieving high reliability. UTSig's pos@top was a mere 0.5%, although this is still an improvement compared to SigNet's 0.1%. This result suggests significant intra-writer variability in UTSig, where genuine signatures tend to exhibit similar or even greater variability than the differences between genuine and skilled-forgery signatures. This suggestion is confirmed by Fig. 4-9, where genuine signatures from three writers show significant intra-writer variability.

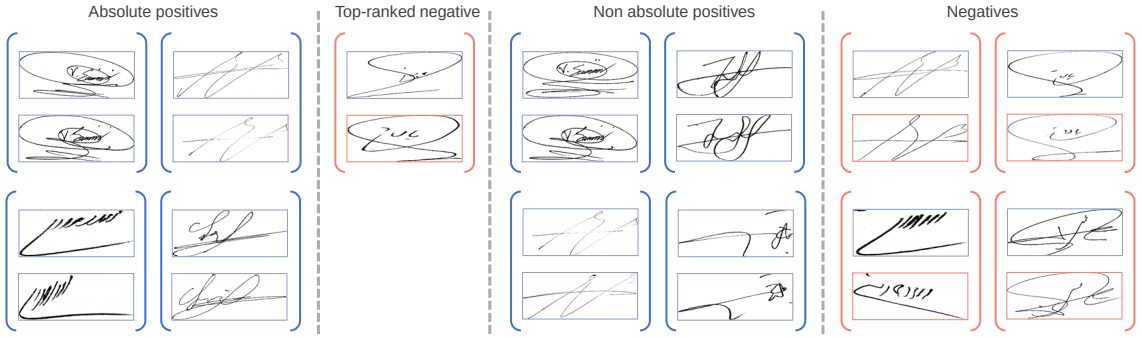
Consequently, the insights gained from the low pos@top value in UTSig shed light on the unique characteristics of this dataset. Traditional metrics like AUC and accuracy might not reveal the underlying challenges associated with Persian signatures. Instead, evaluating the pos@top provides a clearer understanding of the limited number of highly reliable signatures in UTSig, which are difficult to forge even by skilled forgers.



(a) BHSig-B



(b) BHSig-H



(c) UTSig

Figure 4-10: Examples of the absolute positive, top-ranked negative, non-absolute positive, and negative signature pairs from three datasets ranked by our top-rank learning model.

Qualitative evaluation of top-rank learning

Figure 4-10 presents absolute positives, top-ranked negatives, non-absolute positives, and negatives for each dataset. (Except for the top-ranked negative, all those examples are randomly chosen.) This figure shows that the signature pairs in absolute positives are far more similar than those ranked behind them. This high similarity demonstrates the high reliability of our PCF-based top-rank learning model. In our skilled-forgery-only condition, skilled-forgery signatures that are very similar to genuine signatures are selected as the top-ranked negatives. The absolute positive pairs achieve higher similarity than these top-ranked negative pairs (i.e., hard-negatives) and thus are very reliable.

Comparing absolute positives with non-absolute positive pairs also reveals the reliability of the absolute positives. Especially for BHSig-B and BHSig-H, the paired genuine samples of non-absolute positives are also very similar; they do not show large intra-writer differences. As shown in Table 4.2, only 39% and 9.2% of positive pairs are selected as absolute positives for BHSig-B and BHSig-H. This severe selection for absolute positives indicates that our method is useful for deriving a limited number of highly reliable signatures that will not be disturbed even by "very skilled" forgeries.

4.6 Limitations

In this section, I discuss the limitations of the proposed work and outline potential avenues for future research. While the proposed study has demonstrated the effectiveness of PCFs for reliable signature verification, it is essential to acknowledge the following limitations.

Firstly, the application of PCFs has been confined to signature verification. However, the versatility of PCFs extends beyond this domain. For instance, PCFs can be applied to tasks demanding high reliability, such as person reidentification. The adoption of PCFs in diverse applications warrants exploration.

Secondly, this work has utilized a simple contrastive metric learning method to derive PCFs, primarily for the purpose of a fair comparison with the state-of-the-art method. i.t., SigNet [1]. However, future research can consider employing more advanced feature extraction techniques, as demonstrated in [74], and leveraging frameworks like CBCapsNet [75] for enhanced feature representation. These enhancements hold the potential to improve prediction performance further.

Thirdly, this study primarily concentrated on skilled forgery scenarios and did not address random ones. This decision was driven by our emphasis on enhancing reliability, capturing subtle features, and signature authenticity. However, future investigations could involve adapting our methods to address random forgery by integrating specialized feature extraction techniques tailored to this specific challenge.

4.7 Summary

This chapter introduces the concept of PCF along with two highly reliable machine learning frameworks that can be effectively combined with PCF. These frameworks are instrumental in mitigating ambiguous results and providing trustworthy predictions, thereby enhancing the reliability of various tasks. In this context, the focus is on writer-independent signature verification tasks, where high reliability is of paramount importance. PCF entails the fusion of features extracted from distinct samples through a single contrastive learning machine learning model. It draws its inspiration primarily from Siamese-formed models like SigNet, which are specifically applied in this work to capture the intra-similarity between pairs of inputs.

As the first approach in conjunction with PCF, the adoption of Top-rank learning is proposed to achieve highly reliable predictions. Top-rank learning is a specialized variant of the learning-to-rank method, emphasizing the top-ranked positive samples that exhibit unequivocal authenticity. This unique characteristic of Top-rank learning makes it a well-suited approach for highly reliable prediction models. This chapter

delves into the mechanisms and optimization processes of top-rank learning, focusing on its practical application in signature verification tasks.

The second proposed method incorporating PCF is LwR. Unlike the specific approach outlined in Chapter 3, this model employs a fully convolutional neural network (CNN) for enhanced integration and control of the rejection ratio. Through comprehensive performance evaluations on three publicly available datasets, the effectiveness of these methods is rigorously demonstrated, accompanied by in-depth discussions.

It is crucial to emphasize once more that the fusion of PCF with existing highly reliable prediction methods has the potential to significantly enhance the authenticity and reliability of tasks involving paired inputs, such as signature verification. This is especially pertinent in the context of the writer-independent scenario with the skilled-forgery-only condition, where achieving high reliability presents a formidable challenge of paramount importance.

In this chapter, two machine learning frameworks, learning with rejection and top-rank learning, are proposed for application to this task due to their capacity to suppress ambiguous results, thereby ensuring the generation of reliable verification outcomes. As these frameworks operate with a single input, they facilitate the transformation of a pair of genuine and query signatures into a single feature vector known as PCF, which inherently represents the similarity or dissimilarity between the paired signatures, enabling reliable machine learning frameworks to make trustworthy decisions using PCF. Through extensive experiments conducted on three public signature datasets within the offline skilled-forgery-only writer-independent scenario, the effectiveness and reliability of the proposed models are meticulously assessed and validated by comparing their performance with a state-of-the-art model.

Chapter 5

Conclusion and Future Work

This thesis presents novel approaches (including LwR, top-rank learning, and PCF) aimed at enhancing the reliability of machine learning models in pattern recognition tasks. The specific focus of applications has been on character recognition and writer-independent signature verification. These proposed methods leverage LwR with deep features and introduce the concept of PCF integrated into highly reliable machine learning frameworks. In this concluding section, the key findings and contributions of this thesis are summarized, while potential avenues for future work are outlined.

5.1 Conclusion

The LwR framework, as introduced in detail in Chapter 3, incorporates a learnable rejection mechanism co-optimized with a classification function. It is grounded in robust machine learning theory, distinguishing it from conventional heuristic rejection strategies and contributing to its reliability. By LwR, the rejection function can be trained in a feature space different from the one used for classification, enhancing practical applicability through flexibility.

Deep features extracted from various CNN layers have been employed for both classification and rejection in this thesis. This approach has granted improved feature

learning of high-dimensional data, such as images, resulting in superior performance compared to the traditional LwR method. Comprehensive experiments on character recognition tasks, including notMNIST classification and character/non-character classification, have established the superior performance of LwR over conventional rejection strategies, highlighting its practical viability and tangible performance gains.

In Chapter 4, PCF has been introduced and integrated into two highly reliable machine learning frameworks: LwR and Top-Rank Learning. PCF is obtained by fusing features extracted from distinct samples using a single contrastive learning machine learning model. This approach draws inspiration from Siamese-formed models known for their contrastive feature learning capacity.

The combination of PCF with top-rank learning, a variant of learning-to-rank methods, emphasizes top-ranked positive samples as highly reliable predictions. Experiments have demonstrated the effectiveness of this approach in enhancing the reliability of signature verification tasks. Regarding the combination of PCF with the LwR model, in contrast to the approach outlined in Chapter 3, this model employed a CNN for better integration and control of the rejection ratio. Performance evaluations on publicly available datasets have underscored the effectiveness and reliability of this proposed model. Nevertheless, the fusion of PCF with highly reliable machine learning methods has significantly improved the authenticity and reliability of tasks involving paired inputs, such as signature verification. This is particularly pertinent in the context of the writer-independent scenario with the skilled-forgery-only condition.

Quantitative and qualitative experimental results have verified that the proposed models (i.e., LwR with PCF and top-rank learning with PCF) have achieved reliable verification. Significantly, these methods have improved the reliability of SigNet [1], which represents the state-of-the-art baseline of the skilled-forgery-only verification condition. Specifically, LwR with PCF has been capable of rejecting samples that harm classification reliability while automatically adjusting the rejection function.

The top-rank learning model with PCF has been beneficial in certifying highly reliable verification results as absolute positive samples, denoting positive samples ranked higher than all negative samples. The number of absolute positives also serves as an indicator of the reliability of a given dataset; a small number indicates that the signatures in the dataset are challenging to certify due to large intra-writer variability.

5.2 Future Work

While this thesis has made valuable contributions to enhancing reliability in pattern recognition tasks, several avenues for potential future work and research are listed as follows:

- There is a need to investigate more advanced and adaptive rejection mechanisms capable of dynamically adjusting the rejection feature space during the training phases. Exploring reinforcement learning-based approaches for optimizing rejection mechanisms could be a promising direction. These mechanisms would enhance the reliability of machine learning models and improve their adaptability to changing data distributions.

- Advanced contrastive learning techniques should be investigated beyond the Siamese network paradigm. Methods like Momentum Contrast and SimCLR have shown promise in enhancing the quality of features and representations. Incorporating these advanced techniques could further improve the performance of PCF and related applications.

- Extending the proposed methods, particularly LwR with deep features, to real-world document processing tasks holds great potential. Tasks such as text extraction, handwriting recognition, and document classification often require high reliability. Evaluating the applicability and practicality of these methods in document analysis can open new avenues for their utilization.

- Developing human-in-the-loop systems that leverage the proposed methods' reliability could significantly impact critical decision-making domains. Specifically, in the context of this thesis, the focus on eliminating ambiguity suggests a role for human experts in identifying samples better suited for rejection in the LwR model—those prone to misclassification—and in identifying peculiar yet genuinely positive samples for top-rank learning models. Furthermore, post-prediction, human experts can contribute additional insights for samples that were rejected or ranked below the top-ranked negatives. Integrating human expertise with the reliability of machine learning models can enhance overall system performance and trust.

- To ensure the broader applicability of the proposed methods, it is essential to incorporate diverse datasets with varying authenticity constraints. This approach will provide valuable insights into how well these methods perform in real-world scenarios and diverse contexts. It will be crucial to evaluate their robustness and reliability across different data distributions and domains.

- To deal with the special case in top-rank learning scenarios when negative outliers appear, causing difficulty finding no accessible absolute top. In the situation that a negative sample is an outlier or given a wrong label, the training process of top-rank learning will be fatally impacted because there is the possibility that no matter how hard the model learns, there are no positive samples that go beyond the top-ranked negative. In this case, eliminating the influence of the negative outlier in the training set becomes extremely important. As a rough idea, top-rank learning with rejection can be considered to learn a rejection function at the same time as the top-rank function.

In conclusion, valuable contributions have been made to advancing the field of pattern recognition by enhancing the reliability of machine learning models. Significant improvements in the authenticity and trustworthiness of predictions have been

demonstrated by the introduction of LwR with deep features and PCF integrated into highly reliable machine learning frameworks. The future research directions outlined above offer exciting opportunities to strengthen further the field's reliability and its practical applications in various domains.

Bibliography

[1] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, "Signet: Convolutional siamese network for writer independent offline signature verification," *arXiv preprint arXiv:1707.02131*, 2017.

[2] S. Theodoridis and K. Koutroumbas, *Pattern recognition*. Elsevier, 2006.

[3] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 1, pp. 4–37, 2000.

[4] K. S. Fu, "Syntactic pattern recognition," in *Applications of Pattern Recognition*, 2019, pp. 37–64.

[5] D. Li and M. Wu, "Pattern recognition receptors in health and diseases," *Signal transduction and targeted therapy*, vol. 6, no. 1, p. 291, 2021.

[6] X.-Y. Zhang, C.-L. Liu, and C. Y. Suen, "Towards robust pattern recognition: A review," *Proceedings of the IEEE*, vol. 108, no. 6, pp. 894–922, 2020.

[7] A. N. Tarekegn, M. Giacobini, and K. Michalak, "A review of methods for imbalanced multi-label classification," *Pattern Recognition*, vol. 118, p. 107965, 2021.

[8] M. Li, Y.-M. Zhai, Y.-W. Luo, P.-F. Ge, and C.-X. Ren, "Enhanced transport distance for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 13 936–13 944.

[9] J. S. Chung, J. Huh, S. Mun, M. Lee, H.-S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *Proc. Interspeech 2020*, 2020, pp. 2977–2981.

[10] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6398–6407.

[11] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 427–436.

[12] Z. Hao, S. Liu, Y. Zhang, C. Ying, Y. Feng, H. Su, and J. Zhu, "Physics-informed machine learning: A survey on problems, methods and applications," *arXiv preprint arXiv:2211.08064*, 2022.

[13] D. Impedovo and G. Pirlo, "Automatic signature verification: The state of the art," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 5, pp. 609–635, 2008.

[14] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, vol. 70, pp. 163–176, 2017.

[15] W. Hussein, M. A. Salama, and O. Ibrahim, "Image processing based signature verification technique to reduce fraud in financial institutions," in *MATEC Web of Conferences*, vol. 76, 2016, p. 05004.

[16] M. M. Nwe and K. T. Lynn, "Knn-based overlapping samples filter approach for classification of imbalanced data," *Software Engineering Research, Management and Applications*, pp. 55–73, 2020.

[17] N. Otani, Y. Otsubo, T. Koike, and M. Sugiyama, "Binary classification with ambiguous training data," *Machine Learning*, vol. 109, no. 12, pp. 2369–2388, 2020.

[18] P. Dhal and C. Azad, "A comprehensive survey on feature selection in the various fields of machine learning," *Applied Intelligence*, pp. 1–39, 2022.

[19] F. Nargesian, H. Samulowitz, U. Khurana, E. B. Khalil, and D. S. Turaga, "Learning feature engineering for classification," in *Ijcai*, vol. 17, 2017, pp. 2529–2535.

[20] K. Hendrickx, L. Perini, D. V. der Plas, W. Meert, and J. Davis, "Machine learning with a reject option: A survey," *arXiv preprint arXiv:2107.11277*, 2021.

[21] F. Condessa, J. Bioucas-Dias, and J. Kovačević, "Supervised hyperspectral image classification with rejection," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 6, pp. 2321–2332, 2016.

[22] L. B. Marinho, J. S. Almeida, J. W. M. Souza, V. H. C. Albuquerque, and P. P. Rebouças Filho, "A novel mobile robot localization approach based on topological maps using classification with reject option in omnidirectional images," *Expert Systems with Applications*, vol. 72, pp. 1–17, 2017.

[23] P. L. Bartlett and M. H. Wegkamp, "Classification with a reject option using a hinge loss," *Journal of Machine Learning Research*, vol. 9, no. 8, 2008.

[24] C. Cortes, G. DeSalvo, and M. Mohri, "Learning with rejection," in *Proceedings of the Algorithmic Learning Theory*, vol. 9925, 2016, pp. 67–82.

[25] C.-Y. Chuang, J. Robinson, Y.-C. Lin, A. Torralba, and S. Jegelka, "Debiased contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8765–8775, 2020.

[26] J. Cui, Z. Zhong, S. Liu, B. Yu, and J. Jia, "Parametric contrastive learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 715–724.

[27] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 661–18 673, 2020.

[28] A. Sotgiu, A. Demontis, M. Melis, B. Biggio, G. Fumera, X. Feng, and F. Roli, "Deep neural rejection against adversarial examples," *EURASIP Journal on Information Security*, vol. 2020, p. 5, 2020.

[29] Y. Kessentini, T. Burger, and T. Paquet, "A dempster–shafer theory based combination of handwriting recognition systems with multiple rejection strategies," *Pattern recognition*, vol. 48, no. 2, pp. 534–544, 2015.

[30] F. JulcaAguilar, N. S. Hirata, C. ViardGaudin, H. Mouchère, and S. Medjkoune, "Mathematical symbol hypothesis recognition with rejection option," in *2014 14th International Conference on Frontiers in Handwriting Recognition*, 2014, pp. 500–505.

[31] Z. Zhao, C. Liu, Y. Li, Y. Li, J. Wang, B. Lin, and J. Li, "Noise rejection for wearable ecgs using modified frequency slice wavelet transform and convolutional nural networks," *IEEE Access*, vol. 7, pp. 34 060–34 067, 2019.

[32] D. Lin, L. Sun, K.-A. Toh, J. B. Zhang, and Z. Lin, "Biomedical image classification based on a cascade of an svm with a reject option and subspace analysis," *Computers in Biology and Medicine*, vol. 96, pp. 128–140, 2018.

[33] G. Fumera, F. Roli, and G. Giacinto, "Multiple reject thresholds for improving classification reliability," in *Advances in Pattern Recognition: Joint IAPR International Workshops SSPR 2000 and SPR 2000 Alicante*, vol. 1876, 2000, pp. 863–871.

[34] X. Yin, S. Kolouri, and G. K. Rohde, "GAT: generative adversarial training for adversarial example detection and robust classification," 2019.

[35] E. Giusti, S. Ghio, A. H. Oveis, and M. Martorella, "Transfer learning-based fully-polarimetric radar image classification with a rejection option," in *Proceedings of the European Radar Conference*, 2022, pp. 357–360.

[36] H. Mozannar and D. A. Sontag, "Consistent estimators for learning to defer to an expert," in *Proceedings of the International Conference on Machine Learning*, vol. 119, 2020, pp. 7076–7087.

[37] Y. Geifman and R. El-Yaniv, "Selectivenet: A deep neural network with an integrated reject option," in *Proceedings of the International Conference on Machine Learning*, vol. 97, 2019, pp. 2151–2159.

[38] A. Trotman, "Learning to rank," *Information Retrieval*, vol. 8, pp. 359–381, 2005.

[39] C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. N. Hullender, "Learning to rank using gradient descent," in *Proceedings of the 22nd International Conference on Machine learning*, vol. 119, 2005, pp. 89–96.

[40] S. Lai, L. Jin, L. Lin, Y. Zhu, and H. Mao, "Synsig2vec: Learning representations from synthetic dynamic signatures for real-world verification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 735–742.

[41] Y. Zheng, Y. Zheng, W. Ohyama, D. Suehiro, and S. Uchida, "Ranksvm for offline signature verification," in *Proceedings of the ICDAR*, 2019, pp. 928–933.

[42] Y. Zheng, Y. Zheng, D. Suehiro, and S. Uchida, "Top-rank convolutional neural network and its application to medical image-based diagnosis," *Pattern Recognition*, vol. 120, p. 108138, 2021.

[43] S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, and D. Roth, "Generalization bounds for the area under the roc curve," *Journal of Machine Learning Research*, vol. 6, pp. 393–425, 2005.

[44] N. Charoenphakdee, J. Lee, Y. Jin, D. Wanvarie, and M. Sugiyama, "Learning only from relevant keywords and unlabeled documents," in *Proceedings of the EMNLP-IJCNLP*, 2019, pp. 3991–4000.

[45] B. Liu, J. Chen, and X. Wang, "Application of learning to rank to protein remote homology detection," *Bioinformatics*, vol. 31, pp. 3492–3498, 2015.

[46] S. Mehta, R. Pimplikar, A. Singh, L. R. Varshney, and K. Visweswariah, "Efficient multifaceted screening of job applicants," in *Proceedings of the EDBT/ICDT*, 2013, pp. 661–671.

[47] L. Pang, Y. Lan, J. Guo, J. Xu, J. Xu, and X. Cheng, "Deeprank: A new deep architecture for relevance ranking in information retrieval," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 257–266.

[48] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: from pairwise approach to listwise approach," in *Proceedings of the International Conference on Machine Learning*, 2007, pp. 129–136.

[49] N. Li, R. Jin, and Z. Zhou, "Top rank optimization in linear time," in *Advances in Neural Information Processing Systems*, 2014, pp. 1502–1510.

[50] C. Rudin, "The p-norm push: A simple convex ranking algorithm that concentrates at the top of the list," *Journal of Machine Learning Research*, vol. 10, pp. 2233–2271, 2009.

[51] S. P. Boyd, C. Cortes, M. Mohri, and A. Radovanovic, "Accuracy at the top," in *Advances in Neural Information Processing Systems*, 2012, pp. 962–970.

[52] M. Masud, N. Sikder, A.-A. Nahid, A. K. Bairagi, and M. A. AlZain, "A machine learning approach to diagnosing lung and colon cancer using a deep learning-based classification framework," *Sensors*, vol. 21, p. 748, 2021.

[53] J. Frery, A. Habrard, M. Sebban, O. Caelen, and L. He-Guelton, "Efficient top rank optimization with gradient boosting for supervised anomaly detection," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD*, 2017, pp. 20–35.

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

[55] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, 2020.

[56] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," *Advances in Neural Information Processing Systems*, vol. 6, 1993.

[57] J. Memon, M. Sami, R. A. Khan, and M. Uddin, "Handwritten optical character recognition (ocr): A comprehensive systematic literature review (slr)," *IEEE Access*, vol. 8, pp. 142 642–142 668, 2020.

[58] R. Brunelli and T. Poggiot, "Template matching: Matched spatial filters and beyond," *Pattern recognition*, vol. 30, pp. 751–768, 1997.

[59] S. Tian, U. Bhattacharya, S. Lu, B. Su, Q. Wang, X. Wei, Y. Lu, and C. L. Tan, "Multilingual scene character recognition with co-occurrence of histogram of oriented gradients," *Pattern Recognition*, vol. 51, pp. 125–134, 2016.

[60] W. Burger and M. J. Burge, "Scale-invariant feature transform (sift)," in *Digital Image Processing: An Algorithmic Introduction*, 2022, pp. 709–763.

[61] C.-S. Yang and Y.-H. Yang, "Improved local binary pattern for real scene optical character recognition," *Pattern Recognition Letters*, vol. 100, pp. 14–21, 2017.

[62] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, pp. 18–28, 1998.

[63] L. Jiang, Z. Cai, D. Wang, and S. Jiang, "Survey of improving k-nearest-neighbor for classification," in *Fourth international Conference on Fuzzy Systems and Knowledge Discovery*, vol. 1, 2007, pp. 679–683.

[64] V. A. Naik and A. A. Desai, "Online handwritten gujarati character recognition using svm, mlp, and k-nn," in *Proceedings of the International Conference on Computing, Communication and Networking Technologies*, 2017, pp. 1–6.

[65] J. Bai, Z. Chen, B. Feng, and B. Xu, "Image character recognition using deep convolutional neural network learned from different languages," in *Proceedings of the IEEE International Conference on Image Processing*, 2014, pp. 2560–2564.

[66] N. Aneja and S. Aneja, "Transfer learning using cnn for handwritten devanagari character recognition," in *Proceedings of the International Conference on Advances in Information Technology*, 2019, pp. 293–296.

[67] A. Kumar and K. Bhatia, "A survey on offline handwritten signature verification system using writer dependent and independent approaches," in *Proceedings of the International Conference on Advanced Computing Applications*, 2016, pp. 1–6.

[68] M. Diaz, M. A. Ferrer, D. Impedovo, M. I. Malik, G. Pirlo, and R. Plamondon, "A perspective analysis of handwritten signature technology," *ACM Computing Surveys*, vol. 51, pp. 1–39, 2019.

[69] S. Lai and L. Jin, "Recurrent adaptation networks for online signature verification," *IEEE Transactions on Information Forensics and Security*, vol. 14, pp. 1624–1637, 2019.

[70] N. Sae-Bae and N. Memon, "Online signature verification on mobile devices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 6, pp. 933–947, 2014.

[71] M. Stauffer, P. Maergner, A. Fischer, and K. Riesen, "A survey of state of the art methods employed in the offline signature verification process," *New Trends in Business Information Systems and Technology*, pp. 17–30, 2021.

[72] M. Sharif, M. A. Khan, M. Faisal, M. Yasmin, and S. L. Fernandes, "A framework for offline signature verification system: Best features selection approach," *Pattern Recognition Letters*, vol. 139, pp. 50–59, 2020.

[73] E. N. Zois, A. Alexandridis, and G. Economou, "Writer independent offline signature verification based on asymmetric pixel relations and unrelated training-testing datasets," *Expert Systems with Applications*, vol. 125, pp. 14–32, 2019.

[74] D. Banerjee, B. Chatterjee, P. Bhowal, T. Bhattacharyya, S. Malakar, and R. Sarkar, "A new wrapper feature selection method for language-invariant offline signature verification," *Expert Systems with Applications*, vol. 186, p. 115756, 2021.

[75] E. Parcham, M. Ilbeygi, and M. Amini, "Cbcapsnet: A novel writer-independent offline signature verification model using a cnn-based architecture and capsule neural networks," *Expert Systems with Applications*, vol. 185, p. 115649, 2021.

[76] P. L. Bartlett and S. Mendelson, "Rademacher and gaussian complexities: Risk bounds and structural results," *Journal of Machine Learning Research*, vol. 3, no. Nov, pp. 463–482, 2002.

[77] D. S. Maitra, U. Bhattacharya, and S. K. Parui, "Cnn based common approach to handwritten character recognition of multiple scripts," in *Proceedings of the ICDAR*, 2015, pp. 1021–1025.

[78] X. Bai, B. Shi, C. Zhang, X. Cai, and L. Qi, "Text/non-text image classification in the wild with convolutional neural networks," *Pattern Recognition*, vol. 66, pp. 437–446, 2017.

[79] E. Alajrami, B. A. M. Ashqar, B. S. Abu-Nasser, A. J. Khalil, M. M. Musleh, A. M. Barhoom, and S. S. Abu-Naser, "Handwritten signature verification using deep learning," *International Journal of Advanced Multidisciplinary Research*, vol. 3, pp. 39–44, 2020.

[80] A. Soleimani, K. Fouladi, and B. N. Araabi, "Utsig: A persian offline signature dataset," *IET Biometrics*, vol. 6, pp. 1–8, 2017.

[81] R. Tolosana, R. Vera-Rodriguez, C. Gonzalez-Garcia, J. Fierrez, S. Rengifo, A. Morales, J. Ortega-Garcia, J. Carlos Ruiz-Garcia, S. Romero-Tapiador, J. Jiang *et al.*, "Icdar 2021 competition on on-line signature verification," in *Proceedings of the ICDAR*, 2021, pp. 723–737.