VFAST Transactions on Software Engineering
https://vfast.org/journals/index.php/VTSE@ 2026, ISSN(e): 2309-3978, ISSN(p): 2411-6246
Volume 14, Number 2, April-June 2026 pp: 17-32

A Two-Stage Noisy Pre-training and Fine-tuning Pipeline for Low-Resource Named Entity Recognition in Shahmukhi Punjabi

Nazish Basir [1], **Mumtaz Qabulio** [2], **Muhammad Suleman Memon** [3*], **Danish Nazir Arain** [4], **Sehrish Basir Nizamani** [5]

[1]Department of Information Technology, FET, University of Sindh, Jamshoro, Pakistan; [2]Department of Software Engineering, FET, University of Sindh, Jamshoro, Pakistan; [3]Department of Information Technology Dadu Campus University of Sindh, Pakistan; [4][Dr. A. H. S. Bukhari Postgraduate Centre Of ICT, University of Sindh, Pakistan]; [5]Department of Computer Science, Virginia Tech, Blacksburg, United States

Keywords: *Named Entity Recognition, Shahmukhi Punjabi, mBERT, XLM-R, RemBERT, mmBERT, mDeBERTa-V3.*

Journal Info:
Submitted:
March 05, 2026
Accepted:
April 05, 2026
Published:
April 10, 2026

Abstract Named Entity Recognition (NER) for low-resource languages remains a critical challenge in natural language processing, particularly for scripts with limited annotated corpora. This paper addresses this challenge for Shahmukhi Punjabi, an underrepresented Perso-Arabic script used by millions in Pakistan. We propose a two-stage training pipeline that leverages a large-scale machine-labeled corpus generated by a Bagging-CRF ensemble to warm-start multilingual transformer models before fine-tuning on a small, gold-standard human-annotated dataset. We evaluate five state-of-the-art multilingual transformers, mBERT, XLM-R, mmBERT, RemBERT, and mDeBERTa-V3, under two experimental settings: (A) direct supervised fine-tuning on the human-labeled dataset, and (B) the proposed two-stage pipeline. The human-labeled dataset comprises 979 sentences and 25,221 tokens, while the larger machine-labeled corpus having 16,586 sentences and 336,502, both tokens covering 13 entity types. Experimental results demonstrate consistent improvements across all five models, mmBERT and RemBERT achieve the highest weighted F1 scores of 0.85 and 0.86 respectively. The most striking gains are observed for mDeBERTa-V3 (+0.21 F1, 39.6% relative) and XLM-R (+0.20 F1, 33.3% relative), demonstrating that the two-stage pipeline provides the greatest benefit to models with limited baseline performance on low-resource scripts. These results validate the effectiveness of noisy domain adaptation as a data augmentation strategy for low-resource NER in morphologically rich, right-to-left scripts.

***Correspondence author email address:** msuleman@usindh.edu.pk
DOI: 10.21015/vtse.v14i2.2371

1 Introduction

Natural Language Processing (NLP) has witnessed transformative advances driven by large-scale pre-trained transformer models. However, these advances have predominantly benefited high-resource languages such as English, Chinese, and French, while hundreds of millions of speakers of low-resource languages remain underserved [1]. Punjabi, spoken by over 125 million people worldwide, is one such language. In Pakistan, Punjabi is written using the Shahmukhi script, a right-to-left Perso-Arabic writing system that presents unique computational challenges due to its orthographic complexity, shared Unicode blocks with Urdu and Arabic, and severe scarcity of annotated resources [2].

Named Entity Recognition, the task of identifying and classifying named entities such as persons, locations, organizations, and dates within text, is a foundational NLP component that underpins applications ranging from information extraction and machine translation to question answering and knowledge graph construction. In the context of Punjabi written in the Shahmukhi script, Named Entity Recognition (NER) plays a crucial role in a variety of real-world applications, including regional news analysis, social media monitoring, and the digital archiving of cultural and historical content. By automatically identifying and classifying entities such as people, locations, organizations, and dates, NER enables more efficient search, better content organization, and the generation of actionable insights. However, for Shahmukhi Punjabi, NER development is severely constrained by the limited availability of gold-standard annotated corpora, making it difficult to train the large transformer models that have come to define state-of-the-art performance [3]. However, existing approaches for low-resource NER primarily rely on direct fine-tuning or cross-lingual transfer, which remain insufficient when annotated data is extremely scarce and do not fully exploit the potential of large-scale pseudo-labeled data. In this paper, we use machine-labeled and pseudo-labeled interchangeably.

The central challenge of low-resource NER is the mismatch between the data requirements of high-capacity neural models and the practical impossibility of constructing large human-annotated datasets for every language. To address this limitation, this paper introduces a novel two-stage training framework: a two-stage training pipeline that generates a large pseudo-labeled corpus through a Bagging-CRF ensemble which is combining the Bagging (Bootstrap Aggregating) technique with Conditional Random Fields (CRF), used to warm-start multilingual transformer models through noisy domain pre-training, and subsequently refines these models on the small but precise human-labeled dataset. To the best of our knowledge, this is among the first works to apply a machine-labeling pipeline for Shahmukhi Punjabi NER and systematically integrate it with transformer-based pre-training.

We conduct the first systematic evaluation of five prominent multilingual transformer architectures for Shahmukhi Punjabi NER: mBERT (Multilingual Bidirectional Encoder Representations from Transformers) [4], XLM-R (Cross-lingual Language Model – RoBERTa) [5], RemBERT (Refined Multilingual Bidirectional Encoder Representations from Transformers) [6], mDeBERTa-V3 (Multilingual Disentangled attention-enhanced Bidirectional Encoder Representations from Transformers) [7], and mmBERT (Modern Multilingual Bidirectional Encoder Representations from Transformers) [8] under both direct fine-tuning (Experiment A) and the proposed two-stage pipeline (Experiment B). While mBERT, XLM-R, RemBERT, and mmBERT share a standard MLM-based transformer backbone but differ in pre-training data, model size, and language coverage, mDeBERTa-V3 differs additionally in architecture, employing disentangled attention and an ELECTRA-style Replaced Token Detection objective. Our contributions are:

- Development of a human-annotated Shahmukhi Punjabi NER corpus comprising 979 sentences and 25,221 tokens across 13 fine-grained entity categories (PER, LOC, ORG, NORP, DAT, NUM, POS, FOOD, MEA, PRD, EVT, COL, SPT).
- A scalable Bagging-CRF machine-labeling pipeline that generates a high-consensus pseudo-labeled corpus approximately 17$\times$ larger in terms of sentences, comprising 16,586 sentences and 336,502 tokens, with the same 13 entity types.
- A two-stage training strategy combining noisy domain pre-training with label smoothing regularization and clean supervised fine-tuning.
- A comprehensive empirical comparison of five multilingual transformers for Shahmukhi Punjabi NER across 13 entity types using strict IOB (Inside-Outside-Beginning) evaluation.

Overall, this work provides a comprehensive framework for leveraging weak supervision and multilingual transformers to advance NER in low-resource languages. The remainder of this paper is structured as follows: Section 2 reviews related work. Section 3 describes the dataset and machine-labeling pipeline. Section 4 presents the experimental methodology. Section 5 reports and analyzes the results. Section 6 discusses limitations and future directions, and Section 7 concludes the paper.

2 Related Work

2.1 Shahmukhi NER and NLP Resources

Shahmukhi is a Persian-Arabic script that has been modified to represent Punjabi. It includes additional letters to represent the retroflex, nasal and aspirated sounds of Punjabi. Shahmukhi uses the same Unicode blocks as Urdu in the range U+0600–U+06FF and the Arabic Supplement block (U+0750–U+077F), however it needs special processing to remove vocabulary explo sion from different orthographic representations [9, 10]. Such scripts require careful preprocessing due to the fact that tokenization quality and later entity boundary detection can be affected by any inconsistency at the character level.

Ahmad et al. [11] were among the first researchers to investigate the use of natural language processing techniques to identify named entities in Punjabi Shahmukhi texts. They used classical machine learning methods, including support vector machines and conditional random fields to identify named entities in their training corpus of 318,275 tokens that contained 16,300 named entities, achieving an F-score of 85.20% for person, location and organization type entities. Khalid et al. [3] built upon this work by developing BERT-based models using data augmentation on a much larger training corpus of over 1.1 million tokens and 125,789 labeled entities covering five different entity types, achieving an F-score of 93%. Tehseen et al. [12] examined the use of bi-directional LSTM models with contextual embeddings of both ELMo and Word2Vec, resulting in an accuracy of 98.60% and an F-score of 83.75 when using the IO tagging scheme. Ehsan et al. [13] proposed a cluster-based, cross-lingual data augmentation framework for NER across four Pakistani low-resource languages using XLM-R as the primary encoder. Their work included Panjabi Shahmukhi script, and they achieved an F-score of 88.06 in multilingual settings using the BIO annotation scheme.

Additional NLP resources beyond NER for Shahmukhi include a lexical database by Hasan et al. [14], diacritical resources by Hashmi et al. [15], and a BiLSTM-based POS tagger by Tehseen et al. [16] achieving an F1 score of 96.11%. These efforts highlight the gradual progress being made toward developing computational support for Shahmukhi Punjabi. However, compared to high-resource languages, research is limited in terms of large-scale annotated datasets, standardized benchmarks, and publicly available tools. This scarcity poses challenges for building robust NLP systems and highlights the need for further resource development and annotation efforts.

2.2 Multilingual Transformer Models for Low-Resource NER

Devlin et al. [17] introduced BERT, whose multilingual variant (mBERT) covers 104 languages and demonstrates zero-shot cross-lingual transfer capabilities [4]. Conneau et al. [5] introduced XLM-R, trained on 2TB of CommonCrawl data, outperforming mBERT for low-resource languages. Chung et al. [6] proposed RemBERT with decoupled embedding parameters and deeper architectures. He et al. [7] introduced DeBERTa, which employs disentangled attention to separately encode content and positional information. Its multilingual extension, mDeBERTa-V3, uses an ELECTRA-style pre-training objective on multilingual CommonCrawl data. Finally, mmBERT [8] is a modern multilingual encoder from Johns Hopkins University with broad language coverage. Antoun et al. [18] conducted a controlled study demonstrating that DeBERTa-v3 achieves superior NER performance and sample efficiency compared to ModernBERT under matched training conditions.

Adelani et al. [19] created the largest human-annotated NER dataset for 20 African low-resource languages and conducted a comprehensive evaluation of mBERT, XLM-R, and RemBERT, finding that RemBERT achieves performance comparable to XLM-R-large while consistently outperforming mBERT. Fetahu et al. [20] introduced MultiCoNER v2, a large-scale multilingual NER benchmark spanning 33 fine-grained entity classes across 12 languages. In the Perso-Arabic script domain, Ahmed et al. [21] enriched Urdu NER using mBERT embeddings combined with data augmentation and hybrid encoder architectures. Bouabdallaoui et al. [22] proposed FewTopNER integrating XLM-R with few-shot learning and topic-aware contextual modeling.

2.3 Machine-Labeling and Weak Supervision for Low-Resource NER

Conditional Random Fields have a well-established role in sequence labeling and pseudo-label generation for low-resource settings. Arkhipov et al. [24] demonstrated that CRF-augmented mBERT significantly improves labeling consistency for low-resource Slavic languages. Kim et al. [27] applied CRF in a bootstrapping framework for machine-labeled biomedical corpora. Kim et al. [26] demonstrated a 3.9% F1 improvement on Korean corpora using bagging with voting consensus.

The two-stage Bagging-CRF pipeline used in this study has been previously validated on three additional closely related Perso-Arabic script languages: Pashto [29], Urdu [30], and Sindhi [31]. In this work, we extend the pipeline to Shahmukhi Punjabi, demonstrating its applicability to a new language and script while generating a large-scale pseudo-labeled corpus suitable for low-resource NER.

3 Datasets and Machine-Labeling Pipeline

This study utilizes both human-labeled and machine-labeled datasets for Shahmukhi Punjabi NER. The data is collected from diverse sources, including Punjabi news websites, online blogs, and publicly available web content written in Shahmukhi script, such as Pakistan Point (Punjabi) [33] and Punjabi Wikipedia [34]. The human-labeled dataset provides gold-standard annotations, while the machine-labeled dataset is constructed using a proposed labeling pipeline to expand the training data for multilingual transformer models. To ensure reproducibility while respecting data usage constraints, the human-annotated dataset and machine-labeled corpus will be made available upon reasonable request to the corresponding author.

3.1 Human-Labeled Dataset

In this study, we use the gold-standard annotated dataset for Shahmukhi Punjabi NER in IOB format. The 13 types of entities included in the dataset are listed in Table 1. This dataset includes texts from many different domains and contains fine-grained categories of entities (FOOD, SPORT, COLOR, MEASUREMENT) which typically do not appear in most multilingual NER benchmarks.

Prior to all experiments, the dataset underwent systematic pre-processing. Unicode NFC normalization was applied using Python's `unicodedata` library. Script-specific character unification was performed: Arabic 'ي' (U+064A) and 'ى' (U+0649) were mapped to 'ی' (U+06CC), and 'ك' (U+0643) was mapped to 'ک' (U+06A9). These normalizations prevent tokenizers from generating divergent subword representations for orthographically equivalent words.

The human-labeled dataset (gold-standard) has 979 sentences with 25,221 tokens. The machine-labeled dataset has many more sentences (16,586), and also contains many more tokens (336,502). In comparison to the human-labeled dataset, there is a scale difference of approximately 16.9 times as many sentences and 13.3 times as many tokens. Each of these datasets includes the same thirteen entity types.

Therefore, the data within each of the two datasets are annotated based on the same entity type schema. However, the way that the data was annotated differed. The Human-Labeled dataset was carefully annotated by language experts to ensure gold-standard quality, while the machine-labeled dataset was created by an ensemble-based CRF-Bagging method to provide large-scale, fast, and automatically annotated data..

3.2 Machine-Labeling Pipeline

To bridge this gap between the small gold corpus and the data needed for use of large multilingual transformers, we developed a robust machine-labeling pipeline illustrated in Figure 1. The pipeline comprises four stages: feature engineering, ensemble training with cross-validation, consensus-based label assignment, and Machine-Labeled Corpus generation. We provide comprehensive details of preprocessing, feature engineering, CRF configuration, and training pipeline to facilitate replication.

3.2.1 Feature Engineering

Each token is represented by a high-dimensional hybrid feature vector that combines handcrafted linguistic features with distributed semantic representations. The handcrafted component includes:

- Character prefix and suffix n-grams of lengths 1, 2, and 3
- Word length
- Binary flags for digit presence and Perso-Arabic punctuation marks (., !?; □)
- Normalized sentence position (token index divided by sentence length)
- Bilateral context window capturing the immediately preceding and following token's surface form, bigram prefix, and bigram suffix
- Beginning-of-sentence (BOS) and end-of-sentence (EOS) markers

The distributional component uses a 300-dimensional FastText model pre-trained on Urdu CommonCrawl data (cc.ur.300) [32]. The choice of Urdu embeddings is motivated by the substantial lexical overlap between Urdu and Shahmukhi Punjabi in the Perso-Arabic script family, providing reasonable coverage of character n-gram patterns even in the absence of a publicly available Punjabi-specific FastText model. For each token, the 300-dimensional FastText embedding is retrieved via subword averaging and then compressed to 50 dimensions using Principal Component Analysis (PCA). PCA is fitted independently per dataset on a sample of 1,000 unique token types, ensuring that the principal components capture the dominant variance of in-domain vocabulary. The reduced embedding for a token w_i is denoted as $v'(w_i) \in \mathbb{R}^{50}$.

For a sentence $S = [w_1, w_2, \ldots, w_n]$, the token embeddings form a matrix:

$$V'(S) = \Big[v'(w_1), \quad v'(w_2), \quad \ldots, \quad v'(w_n)\Big] \in \mathbb{R}^{n \times 50}. \tag{1}$$

3.2.2 Ensemble Training with 5-Fold Cross-Validation

An ensemble of $N = 10$ independent CRFs is trained. Each model is independently tested using 5-fold cross validation. In addition to testing the model within each of the 5-folds, the training set is also further reduced by sampling: each model is trained on a random 80% sample of each fold's training sentences. This introduces diversity among ensemble members. Each CRF uses the same hyperparameters: the L-BFGS optimization method with $c_1 = c_2 = 0.1$; a maximum of 100 iterations; and all possible transition labels enabled. The choice of N = 10 ensemble members was motivated by prior work in bootstrap aggregation for NER [26], which showed that ensemble sizes between 8 and 15 yield stable label consensus with diminishing returns beyond 15 models. A smaller ensemble (e.g., N = 5) risks higher label variance due to limited voting resolution at integer confidence levels, while larger ensembles (e.g., N = 20) provide no statistically significant improvement in consensus quality but substantially increase training time. N = 10 provides integer-valued confidence increments of 0.10, allowing the τ = 0.80 threshold to correspond exactly to agreement among 8 out of 10 models, a natural and interpretable decision boundary.

3.2.3 Consensus Labeling and Confidence Scoring

After training, each token in the unlabeled corpus receives predictions from all ensemble members. To ensure reliability, only labels agreed upon by at least 80% of the models are assigned; otherwise, the token is labeled as Outside (O). This threshold balances corpus size and label

Table 1. Named Entity Types and Definitions

Tag	Type	Description	Example (Shahmukhi)	Example (English)
PER	Person	Names of individuals	عمران خان	Imran Khan
LOC	Location	Geographical locations	لاھور	Lahore
ORG	Organization	Organizations, institutions	پاکستان کبڈی فیڈریشن	Pakistan Kabaddi Federation
NORP	Nationality/Religion/Political	Groups, affiliations	پاکستانی	Pakistani
DAT	Date	Calendar dates and periods	جنوری	January
NUM	Number	Numerical expressions	پنج	Five
POS	Position/Title	Roles and designations	وزیر اعظم	Prime Minister
FOOD	Food	Food items and cuisine	بریانی	Biryani
MEA	Measurement	Quantities and units	دس کلو	10 kg
PRD	Product	Products and artifacts	موبائل فون	Mobile Phone
EVT	Event	Named events	یوم آزادی	Independence Day
COL	Color	Color expressions	سفید	White
SPT	Sport	Sports and games	کبڈی	Kabaddi

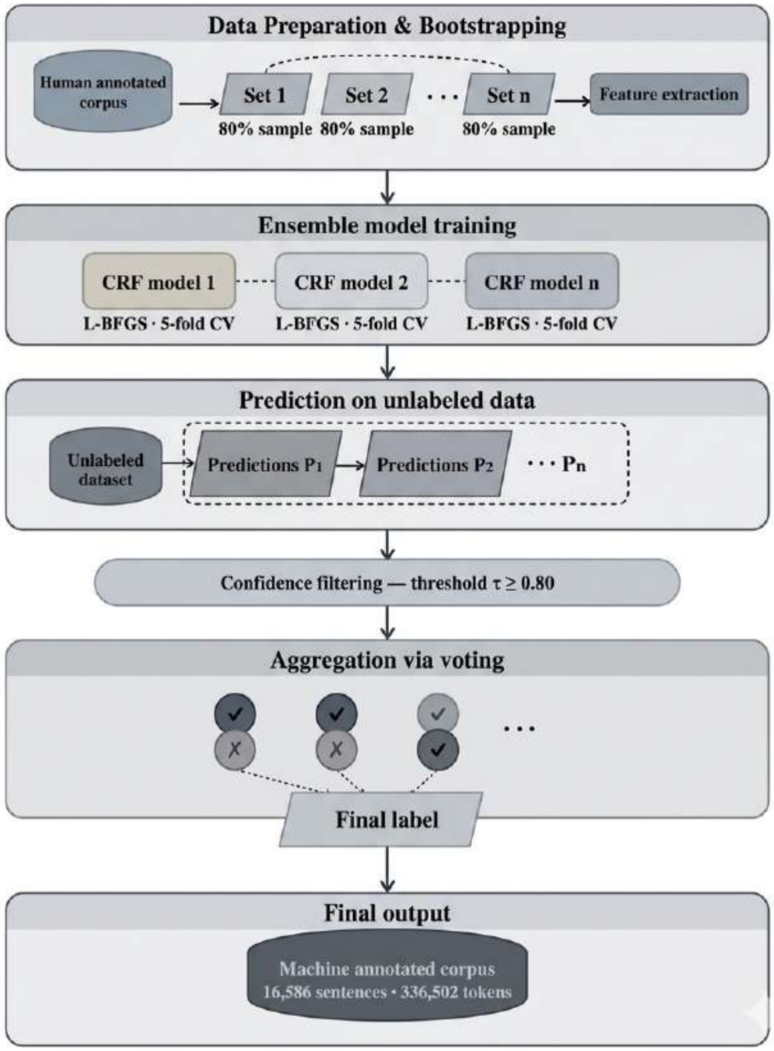

Figure 1. Machine-labeling pipeline. Bootstrap subsets train N independent CRF models; tokens with $\geq$80% label consensus form the pseudo-labeled pre-training corpus.

quality, consistent with prior Bagging-CRF pipelines for Pashto [29], Urdu [30], and Sindhi [30]. The final label for each token w_i is determined using majority voting across the ensemble:

$$\hat{y}_i = \arg\max_{y \in Y} \sum_{j=1}^{N} I(\hat{y}_{j,i} = y) \tag{2}$$

Here, Y is the set of possible labels (e.g., B-PER, I-PER, O), $\hat{y}_{j,i}$ is the label predicted by the j-th model, and I is the indicator function. This procedure ensures that the most consistent predictions are selected, producing a reliable machine-labeled corpus suitable for downstream tasks.

3.2.4 Generated Machine-Labeled Corpus

The final corpus has a total of 16,586 sentences and 336,502 tokens, approximately 17$\times$ larger than the gold corpus. The high proportion of the "O" class (91%) is consistent with standard Named Entity Recognition (NER) datasets, where non-entity tokens naturally dominate. While this reflects realistic language distributions, it can introduce a bias toward the majority class and affect recall for low-frequency entity types. To mitigate this, label smoothing is applied during pre-training to reduce overconfidence in dominant labels. Nevertheless, handling class imbalance for rare entities remains an inherent challenge in NER tasks and is identified as a direction for future improvement. The Noisy Domain Pre-Training Corpus was used exclusively for pre-training and never combined with gold data during fine-tuning.

4 Experimental Methodology

4.1 Model Architectures

We tested five different multilingual transformer models. All models were loaded with the standard settings found in HuggingFace. When testing mDeBERTa-V3, we added `low_cpu_mem_usage=True` and explicit float32 casting to avoid corruption of the model's weights. All of the models shared the same learning rates and training configuration as outlined in Table 2.

4.2 Experimental Design

In this research study, we compared two training paradigms across five multilingual transformer architectures. We used an identical 80/20% split on train/validation sets for all experiments. The random seed was set to 42, and an identical evaluation protocol was employed throughout. All experiments were implemented using standard library of HuggingFace Transformers. Hyperparameters, training configurations, and data processing steps are explicitly described to support reproducibility.

4.2.1 Experiment A, Baseline Supervised Fine-tuning

We fine-tuned each of the five models on the gold-standard training split for ten epochs. The learning rate for all five models was set uniformly at 2×10^{-5}. We chose to apply the same hyperparameters to all models to allow for a controlled and fair comparison.

4.2.2 Experiment B, Two-Stage Noisy Pre-training Pipeline

During Stage 1, each model is trained on the machine-labeled dataset for five epochs using an initial learning rate of 1×10^{-5}, while utilizing Label Smoothing ($\varepsilon = 0.1$) to prevent models from becoming overly confident in the quality of machine-generated labels. In Stage 2, each pre-trained model is fine-tuned on the gold corpus for another five epochs using a learning rate of 2×10^{-5} and applying best checkpoint selection by validation loss.

4.3 Training Configuration and Hardware

Table 3 gives the full hyperparameter configuration. The same effective batch size of 16 was used for each model. For RemBERT, a physical batch size of 4 combined with gradient accumulation of 4 achieved this. Gradient clipping was applied throughout training with max norm of 1.0.

4.4 Evaluation Protocol

Weighted F1 score is the major metric, calculated using strict IOB evaluation through the `seqeval` package, requiring correct prediction of both entity type and exact boundary matching. Here, the weighted F1 is weighted by support, i.e., the number of true instances per class.Precision, recall and per-entity F1 score are also reported for the 13 entity types. A custom confusion matrix for span-based evaluation including the "Outside" class allows us to evaluate Hallucinations (O predicted as some other type) and Missed Detection (an entity predicted as O).

5 Results and Discussion

5.1 Overall Performance Comparison

Table 4 contains weighted average precision, recall and F1 values for all 10 experiments. It demonstrates that the two-stage pipeline (Experiment B) has better results than direct fine-tuning (Experiment A) for each of the five models, validating the central thesis of the paper.

The biggest changes occurred in mDeBERTa-V3 (+0.21 F1, 39.6% relative) and XLM-R (+0.20 F1, 33.3% relative). For mDeBERTa-V3's Experiment A, the lowest baseline of F1 = 0.53 is due to how its disentangled attention mechanism has trouble learning Shahmukhi-specific entity boundaries based on only 979 gold sentences. The two-stage pipeline raises it to F1 = 0.74, demonstrating that noisy domain pre-training provides essential Punjabi-specific distributional grounding.

XLM-R rises from F1 = 0.60 (Exp A) to 0.80 (Exp B), confirming that despite strong performance on standard multilingual benchmarks, it is particularly susceptible to low-resource data scarcity and benefits most from Punjabi-specific domain exposure in Stage 1.

mmBERT and RemBERT both achieve the highest F1 scores in Experiment B at 0.85 and 0.86 respectively, representing gains of +0.13 and +0.05 over their baselines. mmBERT benefits from a more balanced multilingual pre-training that partially overlaps with Punjabi vocabulary, so

Table 2. Multilingual Transformer Models Evaluated

Model	Year	HuggingFace ID	Params	Layers	Languages / Coverage
mBERT	2018	bert-base-multilingual-cased	∼178M	12	104 languages (Wikipedia)
XLM-R	2019	xlm-roberta-base	∼278M	12	100 languages (CommonCrawl)
RemBERT	2021	google/rembert	∼575M	32	110 languages (Wikipedia + mC4)
mDeBERTa-V3	2023	microsoft/mdeberta-v3-base	∼278M	12	100+ languages (CC100)
mmBERT	2025	jhu-clsp/mmbert-base	∼178M	12	∼1,800+ languages

Table 3. Hyperparameter Configuration

Hyperparameter	Value
Optimizer	AdamW
Weight Decay	0.01 (L2 Regularization)
LR, Pre-training (Stage 1)	1×10^{-5} (all models)
LR, Fine-tuning (Exp A & Stage 2)	2×10^{-5} (all models)
LR Scheduler	Linear Warmup + Cosine Decay
Warmup Ratio	10% of total training steps
Effective Batch Size	16 (all models)
Physical Batch / Grad. Accumulation	4/4 (RemBERT); 16/1 (others)
Gradient Clipping	Max Norm = 1.0
Loss, Pre-training	Cross-Entropy + Label Smoothing ($\varepsilon = 0.1$)
Loss, Fine-tuning	Cross-Entropy
Max Sequence Length	256 tokens
Pre-training Epochs	5
Fine-tuning Epochs	10 (Exp A) / 5 (Exp B Stage 2)
Best Checkpoint Selection	Validation loss (Exp B Stage 2 only)
Hardware	L4 Tensor Core GPU (FP32)
Random Seed	42

Stage 1 machine-labeled pretraining amplifies this effect. While RemBERT's strong baseline indicates that its pretraining already provides robust multilingual entity representations, leaving less room for improvement from the two-stage pipeline. RemBERT achieves the lowest validation loss of 0.1781, reflecting superior calibration. mBERT shows a moderate improvement from 0.78 to 0.82 (+0.04), consistent with its multilingual vocabulary already providing partial Perso-Arabic script coverage. Analyzing precision and recall separately reveals interesting patterns. For most models, the two-stage pipeline improves recall more than precision, indicating that Stage 1 pretraining helps the models recognize a broader set of entities that were previously missed. For example, XLM-R's recall rises from 0.66 to 0.82, while precision increases from 0.58 to 0.79, showing that the pipeline primarily enhances the model's coverage of Shahmukhi Punjabi entities without introducing excessive false positives. In contrast, RemBERT maintains high precision and moderate recall gains, reflecting its already strong baseline calibration. These differences highlight how pretraining on machine-labeled data can differentially affect coverage (recall) and correctness (precision) depending on the model's initial exposure to the target language. Overall, models with weaker initial exposure to Shahmukhi/Perso-Arabic script show the largest relative gains, demonstrating the effectiveness of the two-stage pipeline in providing domain-specific distributional grounding.

5.2 Per-Entity Analysis

Table 5 reports per-entity F1 scores for both Experiments A and B across all five models. Structurally well-defined entity types (LOC, PER, NUM, COL) tend to achieve high F1 across both experiments. RemBERT under Experiment B achieved F1 = 1.00 on COL, and high scores for PER (F1 = 0.93) and MEA (F1 = 0.95). One notable anomaly

Table 4. Weighted NER Performance, Experiment A vs. Experiment B. Green shading indicates best-performing models in Experiment B.

Model	Experiment A: Fine-tune Only			Experiment B: Two-Stage Pipeline			△ F1 (A→B)	
	P	R	F1	P	R	F1	Abs.	Rel.%
mBERT	0.76	0.80	0.78	0.81	0.83	0.82	+0.04	5.1%
XLM-R	0.58	0.66	0.60	0.79	0.82	0.80	+0.20	33.3%
RemBERT	0.81	0.81	0.81	0.85	0.86	**0.86**	+0.05	6.2%
mDeBERTa-V3	0.50	0.61	0.53	0.71	0.78	0.74	+0.21	39.6%
mmBERT	0.71	0.73	0.72	0.84	0.85	**0.85**	+0.13	18.1%

is mDeBERTa-V3 under Experiment B: it achieved F1 = 0.00 for both COL and SPT. Its confusion matrix showed complete misclassification of COL tokens (8 as FOOD, 4 as NORP) and all SPT instances were missed. EVT demonstrated the worst F1 across all models due to only 7 validation examples. ORG showed significant improvement under the two-stage pipeline for all architectures except mDeBERTa-V3.

5.3 Learning Dynamics and Training Stability

Figure 2 presents the training and validation loss curves for all five models under both experimental settings, illustrating the convergence behavior and the warm-start effect of the two-stage pipeline.

Several key observations emerge from the loss curves. First, the most visible benefit of two-stage pre-training is the substantial reduction in the starting loss at the first fine-tuning epoch F1 of Experiment B compared to Epoch 1 of Experiment A. This warm-start effect is most pronounced for mDeBERTa-V3 and XLM-R, which begin Experiment B fine-tuning at a validation loss already 30 to 40% lower than their Experiment A Epoch 1 values, consistent with their large overall F1 gains of +0.21 and +0.20 respectively. For RemBERT and mmBERT, which already had competitive Experiment A baselines, the warm-start advantage is more modest but still clearly visible. Second, in Experiment A, mDeBERTa-V3 exhibits an unusually slow and erratic descent in both training and validation loss across all 10 epochs. Its validation loss does not converge smoothly and remains substantially above that of all other models. This behavior is consistent with the known difficulty of cold-starting disentangled attention mechanisms on small gold corpora [7], and explains why mDeBERTa-V3 achieves the lowest Experiment A F1 of 0.53.

Third, in Experiment B Stage 1 (pre-training epochs P1 to P5), all five models show rapid initial loss reduction followed by plateau-like stabilization. This stabilization is expected given that the machine-labeled corpus contains approximately 91% O-labeled tokens, limiting the gradient signal from entity-bearing tokens. The label smoothing regularization ($\varepsilon = 0.1$) applied during Stage 1 visibly reduces the sharpness of loss descent for models such as mBERT and mmBERT, which would otherwise overfit to the dominant O class distribution in the noisy corpus. Fourth, no model exhibits a clear increase in validation loss during Stage 1 pre-training, which would be a sign of catastrophic forgetting of the multilingual representations acquired during upstream pre-training. This con-

Table 5. Per-Entity F1 Scores, Experiment A vs. Experiment B

Model	Exp.	COL	DAT	EVT	FOOD	LOC	MEA	NORP	NUM	ORG	PER	POS	PRD	SPT
mBERT	Exp A	0.86	0.87	0.56	0.82	0.90	0.70	0.83	0.85	0.65	0.87	0.68	0.50	0.52
	Exp B	0.97	0.82	0.47	0.80	0.94	0.87	0.82	0.91	0.71	0.86	0.83	0.36	0.70
XLM-R	Exp A	0.12	0.37	0.12	0.49	0.73	0.62	0.80	0.72	0.64	0.83	0.45	0.46	0.00
	Exp B	0.81	0.83	0.75	0.64	0.95	0.82	0.85	0.88	0.77	0.91	0.82	0.49	0.12
RemBERT	Exp A	0.93	0.80	0.45	0.78	0.92	0.83	0.79	0.90	0.71	0.93	0.70	0.67	0.69
	Exp B	1.00	0.89	0.50	0.81	0.93	0.95	0.95	0.93	0.72	0.93	0.79	0.70	0.85
mDeBERTa-V3	Exp A	0.67	0.12	0.00	0.56	0.64	0.47	0.50	0.61	0.52	0.83	0.47	0.00	0.00
	Exp B	0.00	0.84	0.27	0.56	0.88	0.84	0.82	0.91	0.67	0.89	0.80	0.59	0.00
mmBERT	Exp A	0.90	0.64	0.29	0.71	0.82	0.86	0.74	0.84	0.68	0.82	0.54	0.42	0.56
	Exp B	0.97	0.83	0.56	0.79	0.95	0.90	0.84	0.94	0.81	0.91	0.77	0.72	0.71

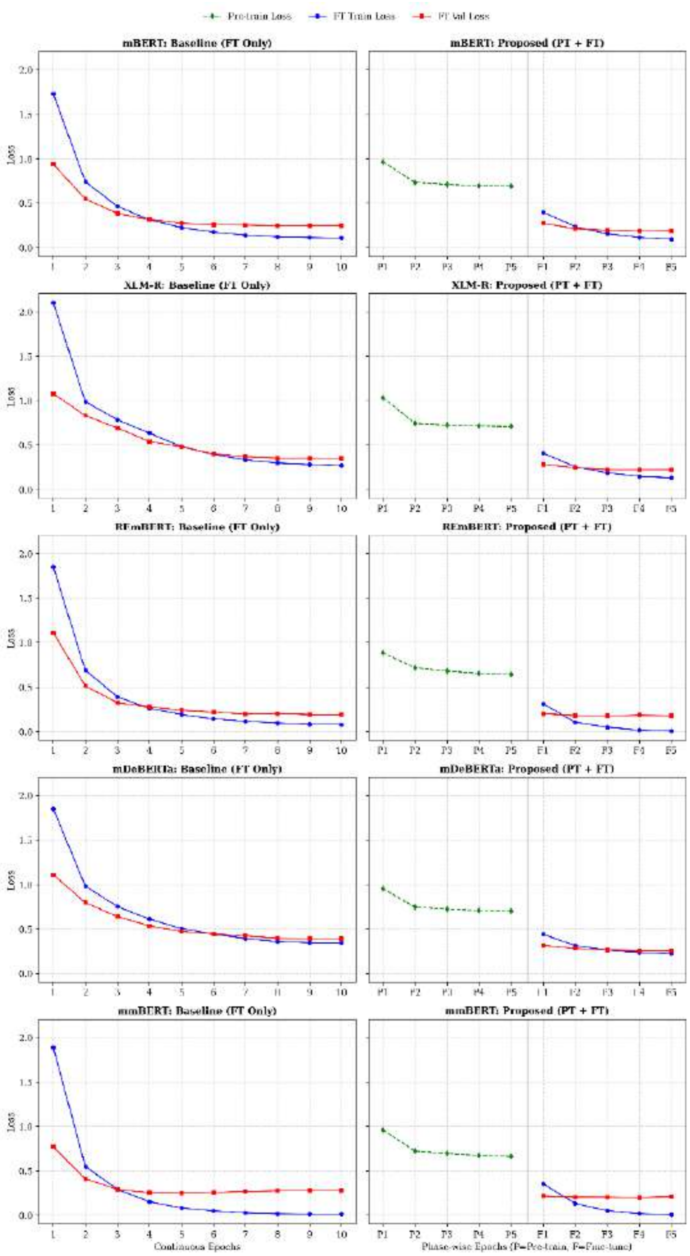

Figure 2. Training and validation loss curves for all five models under Experiment A (direct fine-tuning, 10 epochs) and Experiment B (two-stage pipeline, P1–P5 = pre-training, F1–F5 = fine-tuning).

firms that the lower learning rate of 1×10^{-5} used in Stage 1 is appropriate for preserving the general-purpose multilingual representations while adapting to Shahmukhi-specific distributional patterns.

5.4 Error Analysis: Confusion Matrices, Hallucination, and Missed Detection

We analyze three types of errors: missed detection, cross-entity confusion, and hallucination, to provide a detailed understanding of model performance beyond overall F1 scores. To develop a detailed diagnostic beyond the overall F1 score, we analyzed the span-by-span entity confusion matrices for the five models under both experimental conditions shown in Figure 3. The `get_entities_with_O` evaluation function expands upon the typical SeqEval-based evaluation by adding the "Outside" (O) class into an additional 14$\times$14 confusion matrix, enabling independent quantification of Missed Detection (a true span given the O label) and Hallucination (a true O token misclassified as a named entity).

To ensure consistent and interpretable reporting, all missed detection counts below are expressed as X out of Y total validation instances for that entity class. The total instance counts per class in the 20% validation split are: PER = 92, LOC = 104, ORG = 85, NORP = 40, DAT = 56, NUM = 78, POS = 94, FOOD = 99, MEA = 24, PRD = 26, EVT = 7, COL = 16, SPT = 15. These totals are consistent with the validation set statistics implied by the confusion matrices in Figure 3.

5.4.1 Experiment A, Missed Detection

Missed Detection is the dominant error mode across all five models in Experiment A, consistent with the low-resource setting. XLM-R shows the most severe entity blindness: 37 of 99 FOOD instances, 47 of 94 POS instances, and 21 of 85 ORG instances are misclassified as O. mDeBERTa-V3 misses 46 of 94 POS instances (48.9%) and 27 of 99 FOOD instances (27.3%), the highest POS miss count among all five models. mBERT and mmBERT show moderate missed detection: mBERT misses approximately 11 of 94 POS instances (11.7%) and 3 of 99 FOOD instances (3.0%), while mmBERT misses 20 of 94 POS instances (21.3%) and 37 of 99 FOOD instances (37.4%). RemBERT records the lowest overall missed detection counts in Experiment A, missing approximately 14 of 94 POS instances (14.9%) and 25 of 99 FOOD instances (25.3%), consistent with its superior Experiment A F1 of 0.81.

5.4.2 Experiment A, Cross-Entity Confusion

Beyond missed detections, several linguistically motivated cross-entity confusions are observed. The LOC→NORP error (3 instances in mBERT, 6 in mmBERT, 1 in RemBERT) reflects the morphological relationship between place names and demonyms in Punjabi. SPT (Sport) presents a distinct challenge: XLM-R classifies zero of 15 SPT instances correctly, and mDeBERTa-V3 likewise achieves F1 = 0.00 for SPT in Experiment A. COL is well-handled by most models; however, XLM-R collapses COL largely into NUM (5 instances) and O (5 instances).

5.4.3 Experiment A, Hallucination

The most severe hallucination profiles in Experiment A belong to mDeBERTa-V3 (71 O→FOOD, 62 O→ORG, 55 O→POS) and XLM-R (77 O→FOOD, 39 O→ORG, 53 O→POS), reflecting cold-start difficulties in the absence of domain pre-training.

5.4.4 Experiment B, Missed Detection

The multi-stage pre-training process generates significant decreases in missed detection counts for all four non-exceptional models. XLM-R saw the largest reductions: FOOD from 37 to 24 (–35%), POS from 47 to 15 (–68%), and ORG from 21 to 15 (–29%). For mBERT, POS missed detections dropped from 11 of 94 (11.7%) to approximately 8 of 94 (8.5%), while FOOD improved from 3 of 99 (3.0%) to 2 of 99 (2.0%). For mmBERT, POS missed detections fell from 20 of 94 (21.3%) to 19 of 94 (20.2%), and FOOD from 37 of 99 (37.4%) to approximately 14 of 99 (14.1%). For RemBERT, missed detections approach near zero for MEA (0 of 24, 0.0% vs. 4 of 24, 16.7%), NORP (2 of 40, 5.0% vs. 8 of 40, 20.0%), and COL (0 of 16, 0% in both experiments).

5.4.5 Experiment B, Cross-Entity Confusion and mDeBERTa-V3 Regression

Although mDeBERTa-V3 had a large overall increase in F1 (+0.21), it still scored F1 = 0.00 for both COL and SPT in Experiment B. All 16 COL instances were mis-classified (8 as FOOD, 4 as NORP, 4 as O) and all 15 SPT instances were missed. This selective failure indicates continued sensitivity of mDeBERTa-V3's disentangled attention to rare entity distributional noise.

Since the Stage 1 machine-labeled training data has very few COL and SPT instances due to the stringent $\tau = 0.80$ agreement threshold, there is effectively no signal for these two classes in Stage 1, making them impossible to recover in Stage 2 fine-tuning with such small validation sets.

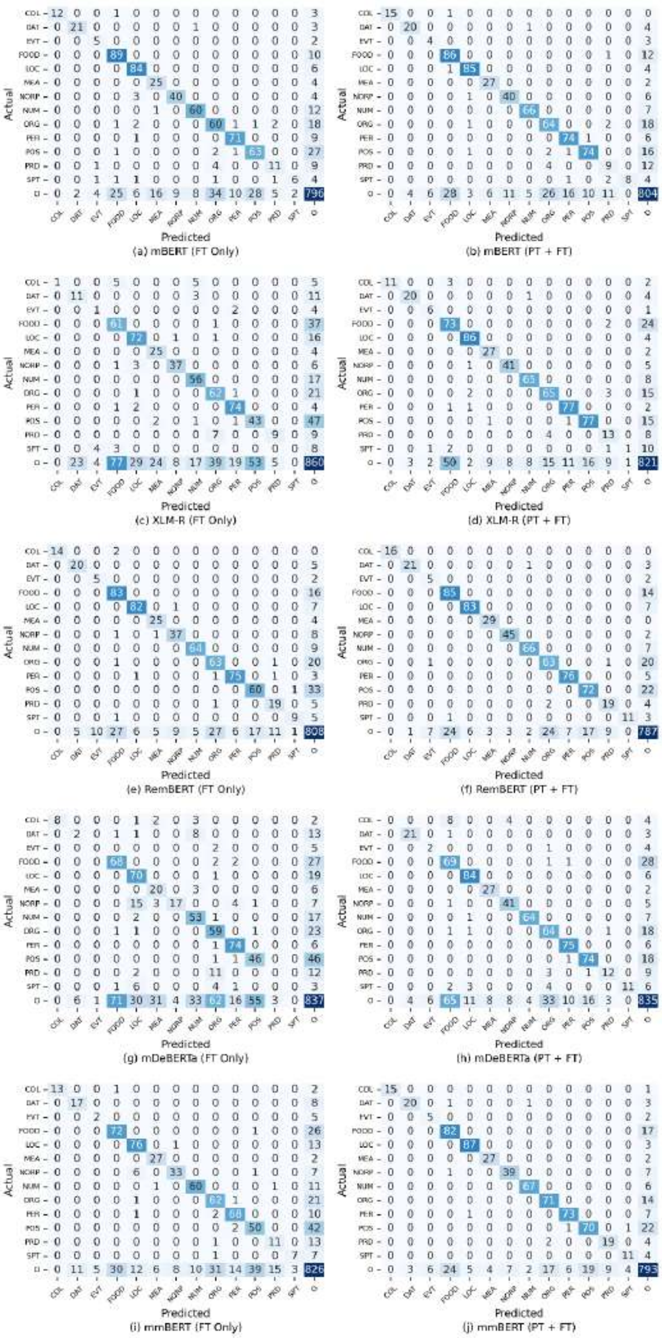

Figure 3. Confusion matrices for all five models under Experiment A (left column) and Experiment B (right column). Rows are true labels, columns are predicted labels, evaluated at the span level over the 20% fixed validation split.

5.4.6 Experiment B, Hallucination

Hallucination error rates were significantly reduced across all models in Experiment B, particularly for XLM-R (O→FOOD from 77 to 50; O→POS from 53 to 16) and mmBERT (O→POS from 39 to 19). Expressed as reduction ratios: XLM-R O□FOOD fell by 35.1% and O□POS by 69.8%, representing the largest absolute hallucination reductions in the experimental matrix. mDeBERTa-V3 also showed improvement: O□FOOD from 71 to 65 (−8.5%), O□ORG from 62 to 33 (−46.8%), and O□POS from 55 to 16 (−70.9%). Taken together, the confusion matrix analysis confirms that the proposed two-stage pipeline reduces hallucination errors systematically across all five architectures.

5.4.7 mDeBERTa-V3 COL/SPT Complete Failure: A Focused Analysis

The most striking finding in this paper is the simultaneous occurrence of a +0.21 overall F1 improvement and F1 = 0.00 on two entity types (COL and SPT) for mDeBERTa-V3 under the two-stage pipeline. This is the only instance in the entire experimental matrix where a model's entity-level F1 score does not improve, or even remain non-zero, despite a large overall gain. We offer the following explanation for this phenomenon. The root cause is a combination of three factors acting in concert. First, COL (Color) and SPT (Sport) are the two rarest entity types in the validation split, with only 16 and 15 instances respectively, representing 1.2% and 1.1% of all validation tokens. Second, due to the τ = 0.80 consensus threshold, the machine-labeled Stage 1 corpus contains very few COL and SPT labels, since the CRF ensemble has insufficient training exposure to these rare categories to consistently agree on their boundaries. This means Stage 1 provides essentially zero distributional grounding for COL and SPT in mDeBERTa-V3. Third, mDeBERTa-V3's disentangled attention mechanism encodes content and position separately, which makes it more reliant on fine-grained positional distributional patterns than standard dot-product attention models such as mBERT or mmBERT. When a rare entity type receives no signal in Stage 1, the disentangled attention heads allocated to that region during Stage 1 adapt to other high-frequency patterns (particularly FOOD and NORP, which have similar surface forms in Shahmukhi), and this adaptation is not reversed during the 5-epoch Stage 2 fine-tuning phase. The result is complete misclassification of COL instances as FOOD (8 of 16) or NORP (4 of 16), and complete missed detection of all SPT instances. This finding has a direct practical implication: when deploying mDeBERTa-V3 for low-resource NER using a two-stage pipeline, practitioners should not rely on noisy pre-training alone to recover performance on rare entity types. Targeted data augmentation for rare classes, class-weighted loss functions in Stage 2, or a higher-quality Stage 1 pseudo-label generation strategy specifically for underrepresented entities would be required before mDeBERTa-V3 could be used reliably in a production NER system covering entity types like COL and SPT.

5.5 Comparison with Prior Shahmukhi NER Work

Table 6 puts our results in context with respect to all prior NER research on Shahmukhi Punjabi. All prior Shahmukhi NER studies, Ahmad et al. [11] (F1 = 85.2), Tehseen et al. [12] (F1 = 83.75), Khalid et al. [3] (F1 = 93), and Ehsan et al. [13] (F1 = 88.06), evaluated their systems exclusively on the three or five most lexically salient entity categories: PER, LOC, and ORG for three and additionally, Language and Religion in one study of Khalid et al. [3]. These categories are the easiest to annotate consistently, appear most frequently in news-domain text, and benefit most from the large annotated corpora used in those studies (up to 1.1 million tokens in Khalid et al. [3]).

The highest prior F1 of 0.93 reported by Khalid et al. [3] was achieved on this five-class subset using over 1.1 million tokens of training data, more than 44 times the 25,221 tokens in our gold corpus. A direct numerical comparison with our reported F1 of 0.86 is therefore not appropriate, as the two systems are not evaluated on the same task. Our system is evaluated on 13 entity types including semantically ambiguous, contextually difficult categories such as EVT (Named Events, F1 = 0.47–0.75), SPT (Sport, F1 = 0.52–0.85), PRD (Product, F1 = 0.36–0.72), and COL (Color, F1 = 0.00–1.00 depending on model), none of which appear in any prior Shahmukhi NER benchmark. When restricted to the common PER, LOC, and ORG entity types, our best model (RemBERT, Experiment B) achieves F1 scores of 0.93, 0.93, and 0.72 respectively, indicating performance at parity with the prior state of the art for the well-defined entity types, while simultaneously cov-

Table 6. Comparison with Prior Shahmukhi Punjabi NER Studies

Study	Year	Tokens	Entity	Entities Covered	Best Model	F1
Ahmad et al. [11]	2020	318,275	3	PER, LOC, ORG	SVM / CRF	0.85†
Tehseen et al. [13]	2023	∼318,275	3	PER, LOC, ORG	BiLSTM+ELMo	0.84†
Khalid et al. [3]	2023	1,131,509	5	PER, LOC, ORG, Lang, REL	BERT+Augment.	0.93
Ehsan et al. [13]	2025	318,275	3	PER, LOC, ORG	Cluster-based Cross-lingual Data Augmentation + XLM-RoBERTa	0.88
Ours, Exp A	2025	25,221	13	All 13 types	RemBERT	0.81
Ours, Exp B	2025	25,221 + 336,502*	13	All 13 types	RemBERT	**0.86**

† IO annotation scheme (no B-/I- distinction), inflates F1 relative to strict IOB.

* Machine-labeled tokens used for Stage 1 pre-training only; not mixed with gold data.

ering 10 additional entity categories with no equivalent benchmark. The lower weighted F1 compared to Khalid et al. [3] therefore reflects the substantially harder and broader evaluation protocol rather than inferior model performance on the core entity types.

Furthermore, prior studies that used IO annotation (without B-/I- prefix distinction) report inflated F1 scores relative to strict IOB evaluation: adjacent entities of the same type cannot be disambiguated without the B- prefix, and IO systems therefore overcount correctly identified spans. Our evaluation uses strict IOB evaluation through the `seqeval` package, which requires exact boundary matching and penalizes all span segmentation errors. This methodological difference alone accounts for a systematic downward adjustment of approximately 1–3 F1 points when comparing to Ahmad et al. [11] and Tehseen et al. [12], further narrowing the apparent gap.

6 Limitations and Future Work

First, all results are reported from single training runs per model due to computational constraints. We acknowledge that this does not capture variance across random seeds. Performing multiple runs and statistical significance testing (e.g., standard deviation, paired tests) is an important direction for future work to further validate the robustness of the observed improvements.

Second, the gold corpus is limited to 979 sentences. With as few as 7 validation instances for EVT and 15 for SPT, per-class F1 scores for rare entities are statistically unreliable. Expanding the annotated corpus is the most critical direction for future work.

Third, the machine-labeled corpus inevitably contains labeling errors despite the 80% consensus threshold. An independent evaluation of machine-labeled data quality against a held-out gold-standard would more rigorously characterize noise levels.

Fourth, mDeBERTa-V3's persistent collapse on COL and SPT in Experiment B highlights a remaining challenge for rare entity types. Targeted data augmentation, class-weighted loss, or higher-quality pseudo-labels for rare classes represent promising directions.

Fifth, FastText embeddings used for CRF feature extraction were pre-trained on Urdu data rather than Punjabi-specific data. Language-specific embeddings would likely yield higher-quality pseudo-labels.

Sixth, the study is limited to a single language and script. Cross-lingual transfer experiments incorporating Urdu, Sindhi, or Pashto data would provide deeper insights. Future work will also explore transformer-based pseudo-labelers, active learning for targeted annotation, and extension to relation extraction and coreference resolution.

In summary, these limitations highlight the need for larger and more diverse annotated corpora, improved pseudo-labeling quality, language-specific resources, and more extensive experimentation to strengthen the reliability and generalizability of Shahmukhi Punjabi NER models.

7 Conclusion

This paper presented a two-stage training pipeline for Named Entity Recognition in Shahmukhi Punjabi, a low-resource Perso-Arabic script with limited annotated data. The pipeline generates a large-scale pseudo-labeled corpus through a Bagging-CRF ensemble with 80% consensus threshold, uses it for noisy domain pre-training with label smoothing, and subsequently fine-tunes on a high-precision human-labeled gold corpus.

Experiments across five state-of-the-art multilingual transformers consistently demonstrate the superiority of the two-stage approach over direct supervised fine-tuning. The most substantial gains were observed for mDeBERTa-V3 (+0.21 F1, 39.6% relative) and XLM-R (+0.20 F1, 33.3% relative). RemBERT achieved the highest weighted F1 of 0.86 under the two-stage pipeline. Notably, all five models were trained under a uniform hyperparameter configuration, demonstrating that the pipeline is robust across diverse transformer architectures without requiring model-specific tuning.

The proposed pipeline is language-agnostic and requires only a small gold corpus, an unlabeled in-domain text collection, and access to pre-trained embeddings. As such, it provides a practical and scalable framework that can be readily adapted to other low-resource languages, enabling high-quality NER with minimal annotated data. Combined with prior published results for Pashto [29], Urdu [30], and Sindhi [31], this study establishes the two-stage Bagging-CRF pipeline as a consistent and reproducible approach for low-resource NER across the Perso-Arabic script family, covering four distinct languages under a unified experimental framework.

Author Contributions

Nazish Basir: Conceptualization, Writing and Reviewing; **Mumtaz Qabulio**: Human annotation data, Writing – Original Draft; **Muhammad Suleman Memon**: Machine-labeling pipeline, Methodology; **Danish Nazir Arain**: Experiments, Fine-tuning; **Sehrish Basir Nizamani**: Visualization, Analysis.

Compliance with Ethical Standards

It is declared that all authors do not have any conflict of interest. This article does not contain any studies with human participants or animals performed by any of the authors.

Funding Information

No external funding was received for this study.

References

[1] P. Pakray, A. Gelbukh, S. Bandyopadhyay, “Natural language processing applications for low-resource languages,” *Natural Language Processing*, vol. 31, no. 2, pp. 183–197, 2025.

[2] M. S. Tahir, M. Ahmad, S. M. Zahra, “Adaptation and Development of Universal Dependencies for Punjabi (Shahmukhi) Script: Challenges and Linguistic Insights,” *Pakistan Research Journal of Social Sciences*, vol. 3, no. 3, 2024.

[3] H. Khalid, G. Murtaza, Q. Abbas, “Using data augmentation and bidirectional encoder representations from transformers for improving Punjabi named entity recognition,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 6, pp. 1–13, 2023.

[4] T. Pires, E. Schlinger, D. Garrette, “How multilingual is multilingual BERT?,” in *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4996–5001, 2019.

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

[6] H. W. Chung, T. Fevry, H. Tsai, M. Johnson, S. Ruder, “Rethinking embedding coupling in pre-trained language models,” *arXiv preprint arXiv:2010.12821*, 2020.

[7] P. He, J. Gao, W. Chen, “Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing,” *arXiv preprint arXiv:2111.09543*, 2021.

[8] M. Marone, O. Weller, W. Fleshman, E. Yang, D. Lawrie, B. Van Durme, “mmbert: A modern multilingual encoder with annealed language learning,” *arXiv preprint arXiv:2509.06888*, 2025.

[9] R. Doctor, A. Gutkin, C. Johny, B. Roark, R. Sproat, “Graphemic normalization of the Perso-Arabic script,” *arXiv preprint arXiv:2210.12273*, 2022.

[10] A. Gutkin, C. Johny, R. Doctor, B. Roark, R. Sproat, “Beyond Arabic: Software for Perso-Arabic Script Manipulation,” in *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*, pp. 381–387, 2022.

[11] M. T. Ahmad, M. K. Malik, K. Shahzad, F. Aslam, A. Iqbal, Z. Nawaz, F. Bukhari, “Named entity recognition and classification for Punjabi Shahmukhi,” *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, vol. 19, no. 4, pp. 1–13, 2020.

[12] A. Tehseen, T. Ehsan, H. B. Liaqat, X. Kong, A. Ali, A. Al-Fuqaha, "Shahmukhi named entity recognition by using contextualized word embeddings," *Expert Systems with Applications*, vol. 229, pp. 120489, 2023.

[13] T. Ehsan, T. Solorio, "Enhancing NER Performance in Low-Resource Pakistani Languages using Cross-Lingual Data Augmentation," in *Proceedings of the Tenth Workshop on Noisy and User-generated Text*, pp. 117–132, 2025.

[14] E. Hasan, M. M. Iqbal, Q. R. Azeemi, A. Javeed, "An online Punjabi Shahmukhi lexical resource," *Sci. Int (Lahore)*, vol. 27, pp. 2529–2535, 2015.

[15] M. A. Hashmi, M. A. Mahmood, and M. I. Mahmood, "Desarrollo de marcas diacríticas para los nombres y verbos de Punjabi Shahmukhi.," *Dilemas contemporáneos: Educación, Política y Valores*, 2019.

[16] A. Tehseen, T. Ehsan, H. B. Liaqat, A. Ali, A. Al-Fuqaha, "Neural POS tagging of shahmukhi by using contextualized word representations," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 1, pp. 335–356, 2023.

[17] J. Devlin, M. Chang, K. Lee, K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

[18] W. Antoun, B. Sagot, "ModernBERT or DeBERTaV3? examining architecture and data influence on transformer encoder models performance," in *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 3061–3074, 2025.

[19] D. I. Adelani et al., "MasakhaNER 2.0: Africa-centric transfer learning for named entity recognition," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 4488–4508, 2022.

[20] B. Fetahu, Z. Chen, S. Kar, O. Rokhlenko, S. Malmasi, "Multiconer v2: a large multilingual dataset for fine-grained and noisy named entity recognition," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 2027–2051, 2023.

[21] A. Ahmed, D. Huang, S. Y. Arafat, I. Hameed, "Enriching Urdu NER with BERT embedding, data augmentation, and hybrid encoder-CNN architecture," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 23, no. 4, pp. 1–38, 2024.

[22] I. Bouabdallaoui, F. Guerouate, S. Bouhaddour, C. Saadi, M. Sbihi, "FewTopNER: integrating few-shot learning with topic modeling and named entity recognition in a multilingual framework," *arXiv preprint arXiv:2502.02391*, 2025.

[23] M. Sabane, A. Ranade, O. Litake, P. Patil, R. Joshi, D. Kadam, "Enhancing low resource NER using assisting language and transfer learning," in *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, pp. 1666–1671, 2023.

[24] M. Arkhipov, M. Trofimova, Y. Kuratov, A. Sorokin, "Tuning multilingual transformers for named entity recognition on slavic languages," in *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2019)*, pp. 89–93, 2019.

[25] J. Kim, Y. Ko, J. Seo, "Construction of machine-labeled data for improving named entity recognition by transfer learning," *IEEE Access*, vol. 8, pp. 59684–59693, 2020.

[26] K. Qian, P. C. Raman, Y. Li, L. Popa, "Learning structured representations of entity names using active learning and weak supervision," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6376–6383, 2020.

[27] J. Kim, Y. Ko, J. Seo, "A bootstrapping approach with CRF and deep learning models for improving the biomedical named entity recognition in multi-domains," *IEEE access*, vol. 7, pp. 70308–70318, 2019.

[28] L. Gligic, A. Kormilitzin, P. Goldberg, A. Nevado-Holgado, "Named entity recognition in electronic health records using transfer learning bootstrapped neural networks," *Neural Networks*, vol. 121, pp. 132–139, 2020.

[29] N. Basir, G. Haider, D. N. Arain, S. B. Nizamani, S. Nizamani, "Enhancing Pashto NER Using Machine-Labeled Data and Transformer-Based Models," in *2025 20th International Conference on Emerging Technologies (ICET)*, pp. 1–8, 2025.

[30] N. Basir, M. Qabulio, M. S. Memon, D. N. Arain, S. B. Nizamani, S. Nizamani, "Expanded Entity Coverage and Machine-Annotated Pre-Training for Urdu Named Entity Recognition," *The Asian Bulletin of Big Data Management*, vol. 6, no. 1, pp. 77–93, 2026.

[31] N. Basir, M. S. Memon, M. Qabulio, D. N. Arain, and R. A. Vighio, "Bridging data scarcity in Sindhi NER using machine-labeled corpora and multilingual transformers," *Spectrum of Engineering Sciences*, vol. 4, no. 3, pp. 179–194, 2026.

[32] FastText, “Pre-trained Urdu embeddings (cc.ur.300),” [Online]. Available: https://dl.fbaipublicfiles.com/fasttext/vectors-crawl/cc.ur.300.bin.gz

[33] Pakistan Point, “Pakistan Point Punjabi News,” [Online]. Available: https://www.pakistanpoint.com/pn/national.html

[34] Punjabi Wikipedia, “Punjabi Wikipedia,” [Online]. Available: https://pnb.wikipedia.org/