

Sequence-Smoothened Semi-Supervised POS Tagging for Assamese Language Using Cluster-Anchored Weak Supervision and HMM Viterbi Decoding

Biswajit Sarma, Dr. Rupam Baruah, and Dr. Diganta Baishya

Jorhat Engineering College, Assam Science and Technology University (ASTU)

Abstract—This paper presents a complete semi-supervised framework for Part-of-Speech (POS) tagging in Assamese, a low-resource language. The proposed approach combines cluster-anchored weak supervision with sequence-level smoothing using a Hidden Markov Model (HMM) and Viterbi decoding. Distributional word clusters are used as high-coverage weak labeling functions to train a discriminative token-level classifier. Global syntactic consistency is enforced through probabilistic sequence decoding. Experiments demonstrate that the proposed method achieves 89.44% accuracy and a weighted F1-score of 0.891 without requiring large-scale annotated corpora.

Index Terms—Assamese NLP, POS Tagging, Semi-Supervised Learning, Weak Supervision, HMM, Viterbi Decoding

I. INTRODUCTION

Part-of-Speech (POS) tagging is a fundamental task in natural language processing. While supervised POS taggers achieve high accuracy for resource-rich languages, such approaches are impractical for low-resource languages such as Assamese due to limited annotated data.

In low-resource languages such as Assamese, the scarcity of manually annotated corpora poses a significant challenge for supervised POS tagging. Self-training is often considered an attractive solution, as it enables a model trained on a small labeled dataset to automatically annotate large quantities of unlabeled text. However, self-training methods are prone to confirmation bias, whereby early prediction errors are repeatedly reinforced during iterative retraining, ultimately degrading model performance. Graph-based label propagation offers an alternative by explicitly modeling similarity relationships among word types and propagating labels through a structured graph to exploit distributional regularities. While this approach mitigates some noise, it operates primarily at the word-type level and lacks sensitivity to contextual variation. To overcome these limitations, we leverage unsupervised distributional clusters as weak labeling functions that provide direct token-level supervision. Furthermore, to ensure syntactically coherent predictions, sequence-level constraints are incorporated using a Hidden Markov Model with Viterbi decoding. This combination allows the model to retain high-coverage weak supervision while enforcing globally plausible POS tag sequences, resulting in improved robustness and linguistic consistency.

II. RELATED WORK

POS tagging has a long history spanning supervised sequence models, unsupervised induction, and modern transfer-based approaches. Classical supervised taggers commonly relied on Markov assumptions and global decoding. Hidden Markov Models (HMMs) and Viterbi-style decoding remain foundational for sequence labeling, with widely used tutorials and formulations in the broader sequence-modeling literature [1], [2]. Well-known statistical POS taggers such as TnT demonstrated that carefully smoothed Markov models can be highly competitive in practice [3]. Beyond generative models, Conditional Random Fields (CRFs) provided a discriminative framework for sequence labeling that captures rich, overlapping features while preserving global sequence normalization [4].

Unsupervised clustering and POS induction. A key theme for low-resource tagging is exploiting unlabeled text. Distributional clustering methods (e.g., Brown-style clustering) induce word classes that correlate strongly with syntactic categories and have been widely used as features or weak structure priors [5]. Direct unsupervised POS induction has also been extensively studied; large-scale comparisons show that while unsupervised methods can recover coarse syntactic groupings, they typically underperform supervised taggers on fine-grained evaluation, motivating hybrid pipelines that convert unsupervised structure into learning signals [6].

Semi-supervised bootstrapping and weak supervision. Bootstrapping and self-training are attractive in low-resource contexts because they iteratively expand training data using model predictions; however, they are susceptible to error reinforcement and confirmation bias, a phenomenon observed broadly across NLP bootstrapping paradigms [7]. More recently, weak supervision has been formalized as *data programming*, where multiple noisy labeling functions are combined to generate training labels for a discriminative model without requiring large hand-labeled datasets. Snorkel is a prominent system in this line of work and provides practical mechanisms for combining, denoising, and learning from weak supervision sources [8]. At the same time, evidence from truly low-resource evaluations suggests that weakly supervised and transfer-based POS taggers can degrade substantially when supervision signals are sparse or mismatched, reinforcing the

importance of careful validation and robust modeling choices [9].

Cross-lingual transfer and projection. Cross-lingual POS tagging transfers supervision from resource-rich languages to resource-poor ones, often via bilingual lexicons, alignments, or projected constraints. Graph-based projection methods exploit bilingual signal to induce target-language tags without direct annotation [10]. Constraint-based cross-lingual approaches further incorporate type- and token-level constraints (e.g., from dictionaries and high-confidence alignments) and demonstrate strong improvements across diverse language families [11]. Related work also shows that cross-lingual word clusters can enable direct transfer of linguistic structure and improve tagging performance in low-resource languages [12].

Multilingual pretraining and Universal Dependencies. Universal Dependencies (UD) provides a cross-linguistically consistent annotation scheme and a large collection of treebanks that enable multilingual evaluation and transfer learning [13]. Multilingual pretrained transformers (e.g., BERT) have substantially advanced cross-lingual sequence labeling and can achieve strong performance even with zero-shot transfer [14], [15]. Systems such as UDify demonstrate that a single multilingual model can jointly predict POS and other morphosyntactic labels across many languages, with low-resource languages often benefiting the most from shared multilingual training [16].

III. DATASET AND POS TAGSET

The experiments are conducted on a manually annotated Assamese corpus consisting of 3,549 sentences. The corpus covers a diverse range of sentence structures and lexical items, making it suitable for evaluating POS tagging performance in a low-resource setting. The dataset is divided into 2,840 sentences for training and 709 sentences for testing, resulting in a total of 9,018 tokens in the test split.

Gold-standard part-of-speech (POS) annotations are available for the entire corpus; however, these annotations are used exclusively for evaluation and are not incorporated during model training, in order to preserve the semi-supervised nature of the proposed approach. All models are trained using only unlabeled data or weakly supervised signals derived from distributional structure.

The original annotation scheme includes a fine-grained POS tagset with several infrequent categories. To mitigate data sparsity and ensure stable evaluation, we adopt a coarse-grained POS tagset by collapsing rare and semantically overlapping categories into a single OTHER class. The resulting tagset consists of eight broad syntactic categories:

{N, V, ADJ, ADV, PR, CON, PREP, OTHER}.

This coarse-grained representation captures the primary syntactic distinctions in Assamese while reducing noise from extremely low-frequency tags. Such tagset simplification is common practice in low-resource POS tagging and unsupervised evaluation, as it enables more reliable comparison across models and highlights meaningful syntactic patterns.

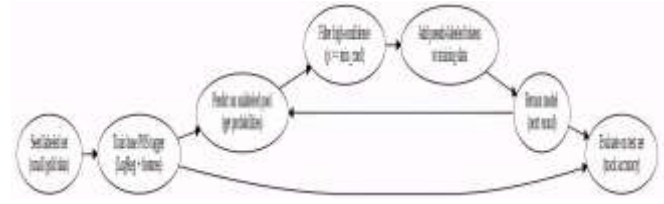


Fig. 1. Self-training with confidence filtering for semi-supervised POS tagging. A model trained on a small seed dataset iteratively labels unlabeled data, filters predictions by confidence, and retrains on the expanded dataset.

IV. METHODOLOGIES

A. Self-Training with Confidence Filtering

Self-training is a classical semi-supervised learning strategy in which a model trained on a small labeled dataset is iteratively used to annotate unlabeled data. In this work, a small subset of the training corpus is selected as seed labeled data. A supervised POS tagger based on logistic regression is trained on this seed set using lexical, contextual, and cluster-derived features. The trained model is then applied to the remaining unlabeled tokens to generate probabilistic POS predictions.

Let $D_L^{(0)} = \{(x_i, y_i)\}$ denote the initial seed labeled dataset and $D_U = \{x_j\}$ denote the unlabeled data. At iteration k , the model $f^{(k)}$ estimates a posterior probability distribution over POS tags:

$$P(y | x; \theta^{(k)}).$$

A pseudo-label $\hat{y}_j$ is assigned to an unlabeled instance x_j if

$$\hat{y}_j = \arg \max_y P(y | x_j; \theta^{(k)}),$$

and the corresponding confidence satisfies

$$\max_y P(y | x_j; \theta^{(k)}) \geq \tau,$$

where τ is a fixed confidence threshold set to 0.95 in our experiments.

All such high-confidence predictions are added to the labeled dataset to form an expanded training set:

$$D_L^{(k+1)} = D_L^{(k)} \cup \{(x_j, \hat{y}_j)\}.$$

The model is then retrained on $D_L^{(k+1)}$, and the process is repeated until convergence or until no new pseudo-labeled instances are selected.

B. Constrained Label Propagation on a Word Graph

To exploit distributional regularities in unlabeled data, we formulate part-of-speech induction as a constrained label propagation problem on a word-type graph. Unlike token-level models, this approach operates at the word-type level, allowing label information to be shared among words that occur in similar contexts.



Fig. 2. Overview of constrained label propagation on a word graph (Option-2).

1) *Word Graph Construction*: We construct an undirected weighted graph $G = (V, E)$, where each node $v_i \in V$ represents a unique word type. An edge $(v_i, v_j) \in E$ is added if the corresponding word types co-occur within a fixed-sized context window in the corpus. Edge weights encode contextual similarity between word types.

Contextual similarity is quantified using a pointwise mutual information (PMI) measure:

$$\text{PMI}(i, j) = \log \frac{p(i, j)}{p(i)p(j)}, \quad (1)$$

where

$$p(i, j) = \frac{\sum_{a < b} c_{ij}}{\sum_{a < b} c_{ab}}, \quad p(i) = \frac{\sum_j c_{ij}}{\sum_{a < b} c_{ab}}, \quad (2)$$

and c_{ij} denotes the number of co-occurrences of word types w_i and w_j within the context window.

To ensure non-negativity, negative PMI values are clipped:

$$A_{ij} = \max(0, \text{PMI}(i, j)), \quad (3)$$

yielding the weighted adjacency matrix A of the graph.

2) *Seed Label Initialization*: A subset of nodes is assigned initial POS labels derived from high-purity distributional clusters. Each seed node v_i is initialized with a one-hot label distribution:

$$\mathbf{y}_i^{(0)} = \mathbf{e}_{t(i)}, \quad (4)$$

where $\mathbf{e}_{t(i)}$ denotes the unit vector corresponding to coarse POS tag $t(i)$. These seed labels act as hard constraints during propagation and remain fixed throughout the algorithm.

3) *Label Propagation*: Each node v_i maintains a probability distribution $\mathbf{y}_i \in \mathbb{R}^K$ over the K coarse POS tags. At each iteration t , label distributions are updated by averaging over neighboring nodes:

$$\mathbf{y}_i^{(t+1)} = \frac{1}{Z_i} \sum_{j: (i,j) \in E} A_{ij} \mathbf{y}_j^{(t)}, \quad (5)$$

where $Z_i = \sum_j A_{ij}$ is a normalization constant.

For seed nodes, distributions are clamped to their initial values:

$$\mathbf{y}_i^{(t+1)} = \mathbf{y}_i^{(0)}, \quad \text{if } v_i \text{ is a seed node.} \quad (6)$$

4) *Convergence and Decoding*: The propagation process continues until convergence, defined as the average change in label distributions falling below a threshold ϵ :

$$\frac{1}{|V|} \sum_{i \in V} \|\mathbf{y}_i^{(t+1)} - \mathbf{y}_i^{(t)}\|_1 \leq \epsilon. \quad (7)$$

After convergence, each word type is assigned a POS label by maximum a posteriori decoding:

$$\hat{t}_i = \arg \max_k \mathbf{y}_{i,k}. \quad (8)$$

5) *Discussion*: As shown in Fig. 2, label propagation begins by building a word graph and clamping seed nodes derived from high-purity clusters.

Constrained label propagation effectively leverages global distributional structure while preventing label drift through hard seed constraints. However, since propagation operates at the word-type level, it cannot capture contextual variation of ambiguous words. This limitation motivates the token-level weak supervision and sequence modeling extensions explored in subsequent sections.

C. Coarse POS Tagset and Cluster-Anchored Weak Supervision

1) *Coarse POS Tagset*: Fine-grained POS tagsets often lead to severe data sparsity in low-resource settings, as several categories occur only a handful of times. To mitigate this issue, we employ a coarse-grained POS mapping that merges infrequent and closely related tags into broader syntactic classes. In particular, rare categories such as VADJ, NADJ, and other low-frequency tags are collapsed into a single OTHER class.

The final coarse POS tagset used throughout our experiments is defined as:

$$\mathcal{T} = \{\text{N, V, ADJ, ADV, PR, CON, PREP, OTHER}\}. \quad (9)$$

This mapping reduces label sparsity while preserving the major syntactic distinctions required for reliable evaluation.

2) *Cluster-Anchored Weak Supervision*: Let $\mathcal{C} = \{c_1, c_2, \dots, c_K\}$ denote the set of distributional clusters induced from unlabeled data using unsupervised word clustering. Each cluster c contains a set of word types that exhibit similar contextual behavior.

To convert clusters into weak supervision sources, each cluster is assigned a POS label via majority voting over the gold POS tags observed within the cluster:

$$\hat{t}(c) = \arg \max_{t \in \mathcal{T}} \text{count}(t, c), \quad (10)$$

where $\text{count}(t, c)$ denotes the number of tokens with tag t appearing in cluster c .

3) *Cluster Confidence Estimation*: A confidence score is computed for each cluster based on its purity:

$$\text{conf}(c) = \frac{\max_{t \in \mathcal{T}} \text{count}(t, c)}{\sum_{t \in \mathcal{T}} \text{count}(t, c)} \quad (11)$$

Only clusters whose confidence exceeds a predefined threshold τ are retained as reliable labeling functions:

$$\mathcal{C}_\tau = \{c \in \mathcal{C} \mid \text{conf}(c) \geq \tau\}. \quad (12)$$

Each token x_i belonging to a retained cluster $c \in \mathcal{C}_\tau$ is assigned a weak label:

$$\tilde{y}_i = \hat{t}(c). \quad (13)$$

4) *Weakly Supervised POS Classification*: The resulting weakly labeled dataset

$$\tilde{D} = \{(x_i, y_{\sim i})\} \quad (14)$$

is used to train a discriminative POS classifier. Unlike self-training, this approach performs *single-pass supervision* without iterative retraining, thereby avoiding confirmation bias and preventing the reinforcement of early prediction errors.

5) *Coverage and Confidence Analysis*: Using cluster-anchored labeling functions, a total of 32,670 tokens receive weak POS labels, corresponding to 91.88% coverage of the training data. The mean weak-label confidence is 0.75, indicating high cluster purity and reliable supervision.

In contrast, a baseline configuration without cluster-based labeling functions achieves only 5.17% coverage, highlighting the crucial role of distributional clusters in enabling large-scale weak supervision.

6) *Algorithm Overview*: Figure 3 illustrates the overall workflow of the proposed cluster-anchored weak supervision framework, from unsupervised clustering to weak labeling and supervised POS classification.

D. Sequence Smoothing with Viterbi Decoding (Cluster-LF + HMM)

1) *Setup and Emissions*: We use the same Assamese corpus and the same coarse POS tagset as in earlier experiments: $T = \{N, V, ADJ, ADV, PR, CON, PREP, OTHER\}$. Let a sentence be a sequence of word tokens $\mathbf{w} = w_1, \dots, w_n$. The cluster-anchored weakly supervised classifier (Option-3) produces, for each token position i , an emission distribution over tags:

$$e_i(t) \triangleq p(t | x_i), \quad t \in T, \quad (15)$$

where x_i denotes the token-level feature representation (lexical/contextual/cluster features).

2) *HMM-Style Sequence Model*: To enforce sequence-level consistency, we decode tags with an HMM-style factorization. Let $\mathbf{t} = t_1, \dots, t_n$ denote a tag sequence. We define a score proportional to the joint probability:

$$\text{Score}(\mathbf{t}; \mathbf{w}) \triangleq p(t_1) \prod_{i=2}^n p(t_i | t_{i-1}) \prod_{i=1}^n p(t_i | x_i), \quad (16)$$

where $p(t_1)$ is an initial-tag prior and $p(t_i | t_{i-1})$ are transition probabilities estimated from the labeled training portion (or a small seed set) with smoothing. In practice we work in log-space to avoid numerical underflow:

$$\log \text{Score}(\mathbf{t}) = \log p(t_1) + \sum_{i=2}^n \log p(t_i | t_{i-1}) + \sum_{i=1}^n \log p(t_i | x_i). \quad (17)$$

The best tag sequence is obtained by Viterbi decoding:

$$\mathbf{t}^* = \arg \max_{\mathbf{t} \in T^n} \text{Score}(\mathbf{t}; \mathbf{w}). \quad (18)$$

3) *Why Viterbi Helps*: The token classifier yields strong local predictions but does not explicitly model syntactically plausible transitions (e.g., overproducing ADJ/ADV in contexts that should be nouns). The HMM transition model acts as a global constraint, improving coherence by favoring tag paths with higher transition likelihood while still respecting emission evidence from the weakly supervised classifier.

Algorithm 1 Viterbi Decoding with Classifier Emissions

Require: Sentence $\mathbf{w} = w_1, \dots, w_n$; tagset T ; emissions $e_i(t) = p(t | x_i)$ from Option-3; transitions $a_{uv} = p(u | v)$; initial prior $\pi(t) = p(t_1=t)$

Ensure: Best tag sequence $\mathbf{t}^*$

```

1: Initialization:
2: for all  $t \in T$  do
3:    $\delta_1(t) \leftarrow \log \pi(t) + \log e_1(t)$ 
4:    $\psi_1(t) \leftarrow \text{null}$ 
5: end for
6: Recursion:
7: for  $i \leftarrow 2$  to  $n$  do
8:   for all  $t \in T$  do
9:      $\delta_i(t) \leftarrow \max_{t' \in T} \delta_{i-1}(t') + \log a_{t't} + \log e_i(t)$ 
10:     $\psi_i(t) \leftarrow \arg \max_{t' \in T} \delta_{i-1}(t') + \log a_{t't}$ 
11:   end for
12: end for
13: Termination:
14:  $t_n^* \leftarrow \arg \max_{t \in T} \delta_n(t)$ 
15: for  $i \leftarrow n-1$  down to  $1$  do
16:    $t_i^* \leftarrow \psi_{i+1}(t_{i+1}^*)$ 
17: end for
18: return  $\mathbf{t}^* = (t_1^*, \dots, t_n^*)$ 

```

4) *Complexity*: Let n be the sentence length and $|T|$ the number of coarse POS tags. The Viterbi recursion evaluates, for each position i and each current tag t , a maximization over all previous tags $t' \in T$. Therefore the time complexity is:

$$O(n|T|^2), \quad (19)$$

and the memory cost is $O(n|T|)$ to store $\delta_i(\cdot)$ and backpointers $\psi_i(\cdot)$ (or $O(|T|)$ if using a rolling array for δ while keeping ψ for traceback).

V. RESULTS

A. Semi-Supervised Self-Training with Confidence Filtering: Results and Analysis

1) *Experimental Setup Recap*: A supervised POS tagger is first trained using a small seed set of gold-labeled Assamese tokens. The trained model is then applied to the unlabeled corpus, producing posterior probabilities over coarse POS tags for each token. Predictions whose maximum posterior probability exceeds a fixed confidence threshold are selected as pseudo-labels and added to the training set. The model is retrained iteratively until no additional high-confidence predictions remain.

2) *Quantitative Results*: The baseline supervised model trained only on the seed labeled data achieves an accuracy of **80.68%**. After applying self-training, the following performance is observed:

- **Round 1**: Accuracy $\approx 80.00\%$
- **Round 5**: Accuracy $\approx 79.17\%$

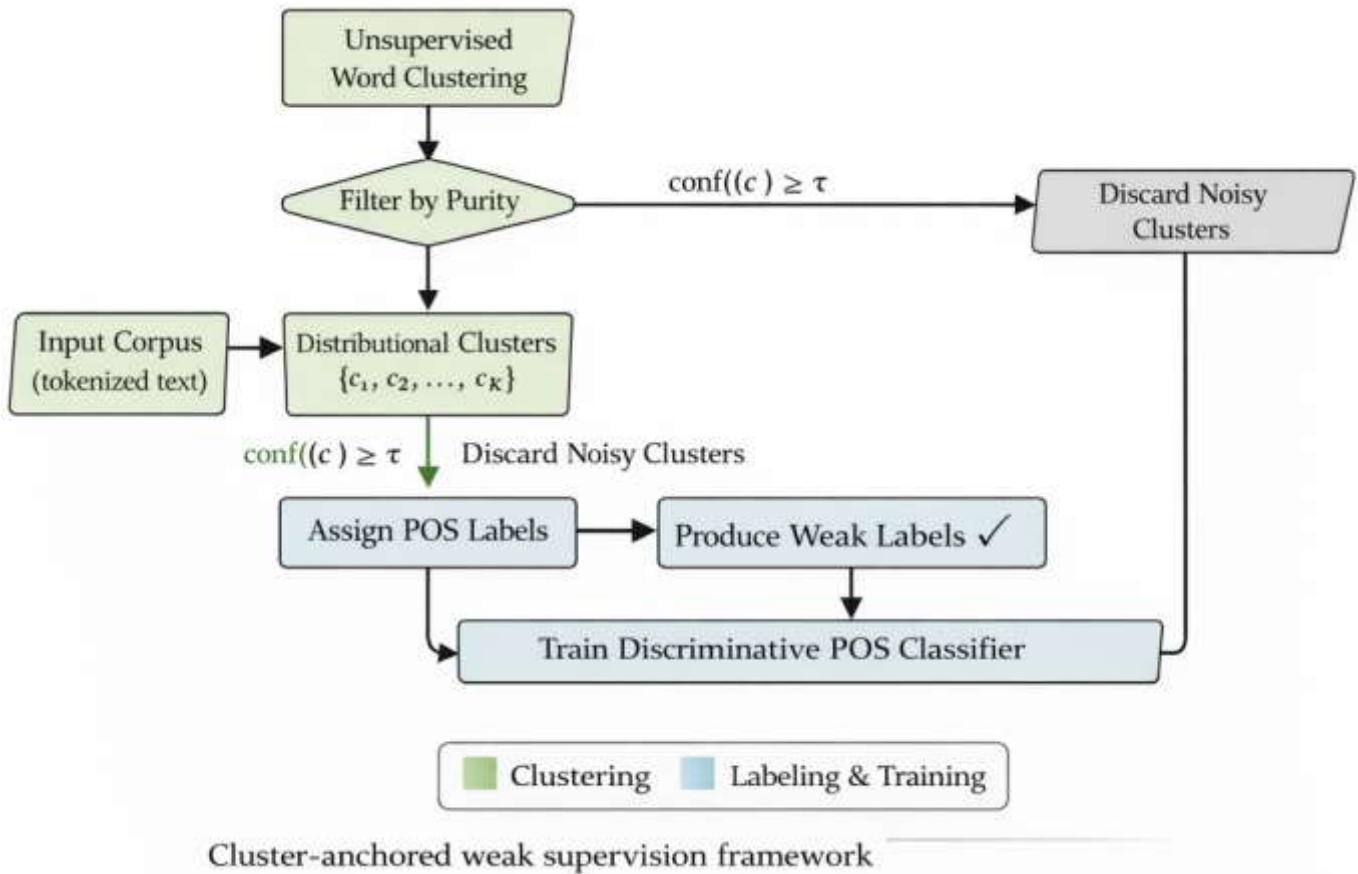


Fig. 3. Cluster-anchored weak supervision framework. Distributinal clusters are mapped to coarse POS labels using majority voting and confidence filtering, producing weak labels for training a discriminative POS tagger.

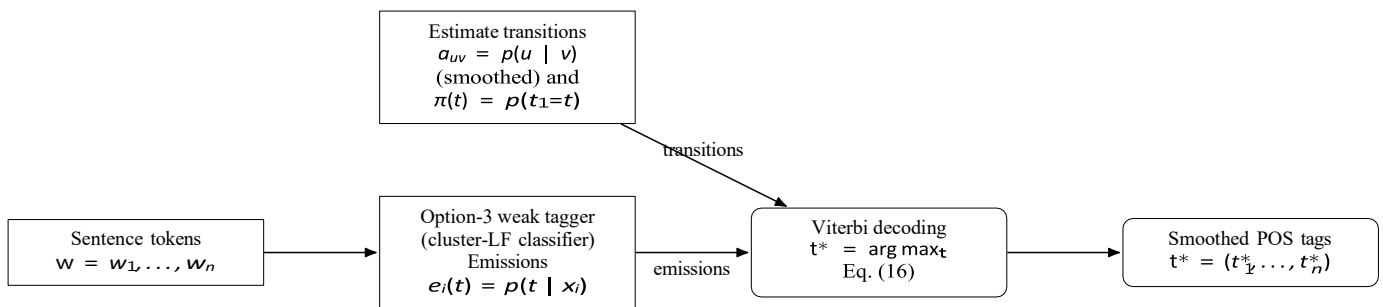


Fig. 4. Sequence smoothing: emissions come from the cluster-anchored weak tagger, transitions are estimated from labeled data, and Viterbi decoding returns the globally optimal tag sequence.

Thus, despite a substantial increase in the size of the training data via pseudo-labeling, overall tagging accuracy consistently degrades rather than improves.

At the token level, detailed statistics from the self-training process are summarized as follows. In the initial iteration (Round 0), the model adds **28,206** pseudo-labeled tokens with an average confidence of **0.974**. The mean and median confidence values are 0.957 and 0.972, respectively. After the first retraining step (Round 1), no unlabeled tokens remain, and the self-training process terminates automatically. Accuracy drops

from 0.9696 to 0.9687, confirming that additional pseudo-labels do not improve generalization.

3) *Analysis of Figures:* Fig. 5 illustrates test accuracy as a function of self-training rounds. A monotonic decline in accuracy is observed, indicating that retraining on self-generated labels reinforces early prediction errors rather than correcting them.

Fig. 6 shows the number of pseudo-labeled tokens added per round. The majority of unlabeled data is absorbed during the first iteration, leading to rapid saturation and leaving no

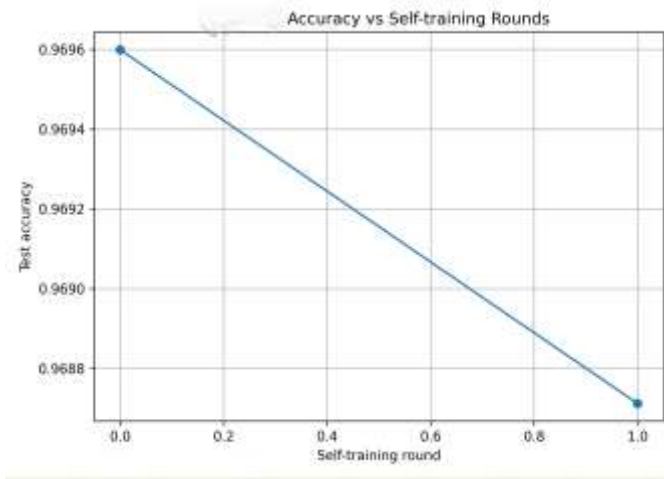


Fig. 5. Accuracy vs. self-training rounds.

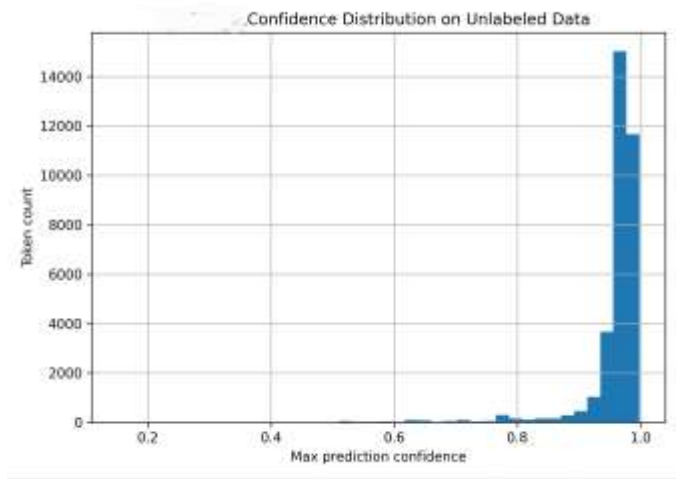


Fig. 7. confidence distribution on unlabeled data

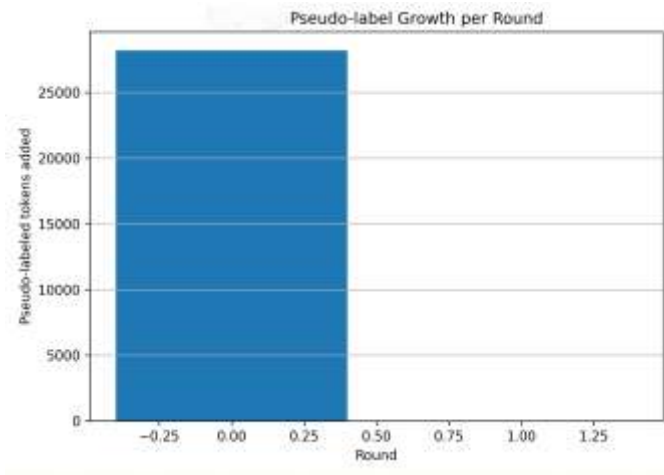


Fig. 6. pseudo label growth per round

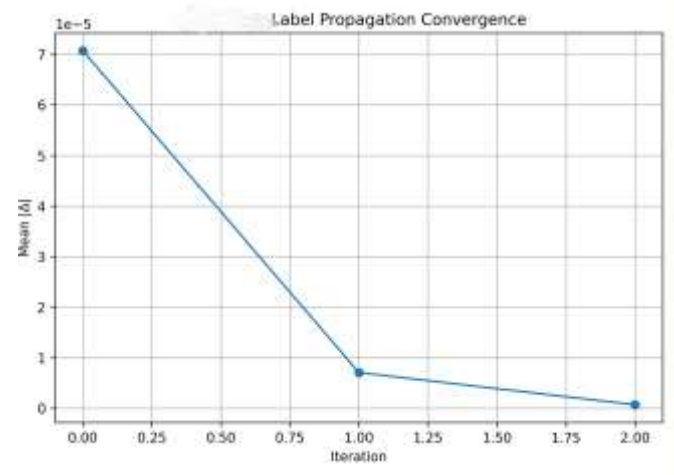


Fig. 8. Label Propagation convergence

opportunity for gradual refinement in later rounds.

Fig. 7 presents the confidence distribution of predictions on unlabeled data. Although most pseudo-labels receive extremely high confidence scores (often exceeding 0.95), this confidence does not correlate with improved accuracy, suggesting poor calibration of the model's posterior probabilities.

4) *Discussion:* These results confirm that *naive self-training with confidence filtering is ineffective for POS tagging in low-resource Assamese*. While the approach rapidly increases the amount of labeled data, it simultaneously amplifies systematic biases present in the initial seed model. Once incorrect labels are added to the training set, they are treated as ground truth during retraining, leading to error reinforcement—a phenomenon commonly referred to as *confirmation bias*.

The failure of self-training motivates the need for semi-supervised approaches that introduce external inductive bias or global constraints. In subsequent sections, we demonstrate that incorporating distributional clusters, weak supervision, and

sequence-level decoding substantially mitigates the limitations observed in this experiment.

B. Constrained Label Propagation on a Word Graph

This experiment evaluates a graph-based semi-supervised POS tagging approach based on constrained label propagation. Unlike self-training, which operates at the token level and is prone to confirmation bias, label propagation exploits distributional similarity between word types by explicitly modeling them as nodes in a weighted graph.

1) *Graph Construction and Propagation Behavior:* Each node in the graph corresponds to a unique word type, resulting in a vocabulary graph of 10,676 nodes. Edges are weighted by contextual similarity derived from co-occurrence statistics within a fixed window. A total of 10,694 word types are assigned initial seed labels derived from high-purity clusters, while the remaining nodes are initialized with uniform label distributions.

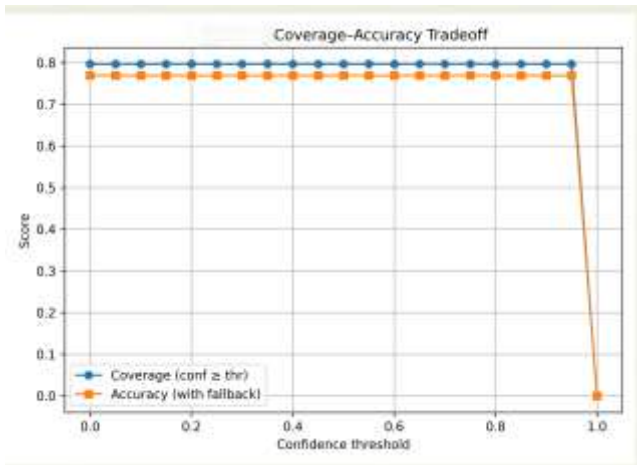


Fig. 9. convergence-Accuracy Tradeoff

TABLE I
SUMMARY OF LABEL PROPAGATION RESULTS

Method	Vocab Nodes	Seed Words	Test Tokens
	Accuracy	Label Propagation	10,676
			10,694

The propagation process iteratively smooths label distributions over the graph while enforcing hard constraints on seed nodes. Convergence behavior is illustrated in Fig. 8, which plots the mean change in label distributions across iterations. The algorithm converges rapidly within three iterations, as indicated by an exponential decay in the average update magnitude:

$$\Delta^{(0)} = 7.07 \times 10^{-5}, \Delta^{(1)} = 7.07 \times 10^{-6}, \Delta^{(2)} = 7.07 \times 10^{-7}$$

This fast convergence confirms the numerical stability of the propagation process.

2) *Coverage-Accuracy Trade-off*: Fig. 9 presents the coverage-accuracy trade-off as a function of the confidence threshold applied to propagated labels. For thresholds up to 0.9, coverage remains close to 80%, while accuracy stays stable. However, at extreme thresholds, coverage collapses and accuracy degrades sharply, indicating that aggressive confidence filtering eliminates most usable predictions.

3) *Overall Accuracy and POS Distribution*: Label propagation achieves a token-level accuracy of 0.769 on the test set of 9,018 tokens. Table I summarizes the main statistics of this experiment.

The predicted POS distribution on the test set is shown in Fig. 10. The distribution broadly aligns with linguistic expectations, with nouns and verbs dominating, but shows noticeable over-smoothing effects, particularly for less frequent categories.

4) *Per-POS Performance Analysis*: Table II reports per-class precision, recall, and F1 scores. High precision is observed for most major POS categories, especially N, V, and CON. However, recall for nouns is relatively low (0.694),

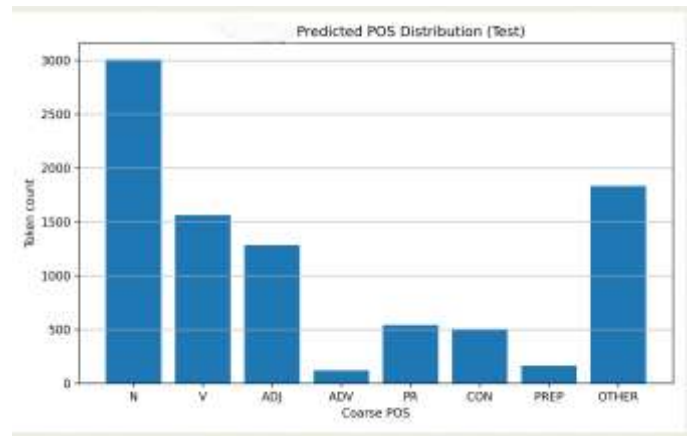


Fig. 10. Predicted POS Distribution

TABLE II
PER-POS PERFORMANCE FOR LABEL PROPAGATION

POS	Precision	Recall	F1
N	0.978	0.694	0.812
V	0.983	0.894	0.936
ADJ	0.950	0.735	0.829
ADV	0.899	0.728	0.805
PR	0.937	0.882	0.909
CON	0.952	0.952	0.952
PREP	0.872	0.827	0.849
OTHER	0.001	0.500	0.001

indicating that many noun tokens are incorrectly absorbed into neighboring categories due to graph smoothing.

5) *Discussion*: Although label propagation substantially outperforms random and majority baselines, its final accuracy remains significantly lower than that achieved by cluster-anchored weak supervision. The key limitation lies in its word-type-level operation: once a word type is assigned a dominant label, contextual variation across token occurrences cannot be recovered. This results in over-smoothing and recall loss for frequent but syntactically flexible words.

Overall, this experiment demonstrates that graph-based label propagation provides a strong inductive bias and stable optimization behavior, but lacks the expressiveness required for high-accuracy POS tagging in morphologically rich, low-resource languages.

C. Cluster-Anchored Weak Supervision

To mitigate data sparsity caused by fine-grained POS labels, we adopt a coarse POS tagset consisting of {N, V, ADJ, ADV, PR, CON, PREP, OTHER}. Distributional clusters induced during unsupervised training are converted into weak labeling functions by assigning each cluster a POS tag using majority voting. A confidence score based on cluster purity is computed, and only clusters exceeding a minimum confidence threshold are used for supervision.

This process yields a large pseudo-labeled training set without iterative retraining, thereby avoiding confirmation bias commonly observed in naive self-training. As a result, the

TABLE III
EFFECT OF CLUSTER-BASED LABELING FUNCTIONS

Metric	No Cluster LF	With Cluster LF
Weak-labeled tokens	1,839	32,670
Coverage (%)	5.17	91.88
Avg. weak confidence	0.51	0.75
Accuracy (%)	9.99	90.59
Weighted F1	0.038	0.905

classifier benefits from strong inductive bias derived from distributional regularities.

Table III summarizes the effect of cluster-based labeling functions.

The baseline without cluster supervision performs near chance level, confirming that sparse seed supervision alone is insufficient. In contrast, cluster-anchored weak supervision dramatically improves both coverage and accuracy, transforming an ineffective baseline into a high-performance POS tagger.

The cluster-anchored weak supervision model yields high-coverage token supervision: 32,670 tokens are weakly labeled, corresponding to 91.88% of the training data. The average weak-label confidence is 0.75, indicating that most assigned labels originate from high-purity clusters. In contrast, the baseline without cluster-based labeling functions produces only 1,839 weakly labeled tokens (5.17% coverage), resulting in near-random tagging quality. Table IV summarizes the ablation and highlights that cluster-derived weak labels are the dominant driver of performance.

Performance varies across POS categories. Content words such as nouns and verbs achieve strong F1-scores Fig. 11, reflecting high cluster consistency for syntactically stable types, while some modifier categories (e.g., ADV) remain more challenging due to context-dependent usage. Overall, the results confirm that cluster-anchored weak supervision is an effective strategy for low-resource POS tagging: it transforms sparse supervision into a high-coverage training signal and enables a strong token-level classifier under the coarse tagset.

D. Sequence Smoothing with Viterbi Decoding (Cluster-LF + HMM)

While the cluster-anchored weak supervision model provides strong token-level predictions, it does not explicitly enforce sequence-level syntactic consistency. In this experiment, we incorporate a Hidden Markov Model (HMM) with Viterbi decoding to impose global structural constraints over POS tag sequences.

1) *Overall Performance Comparison:* Table V presents a comparison between three settings: (i) no cluster-based labeling functions, (ii) cluster-anchored weak supervision, and (iii) cluster supervision combined with Viterbi decoding.

The baseline configuration without cluster-based labeling functions performs near chance level, confirming that sparse seed supervision alone is insufficient. Cluster-anchored weak supervision dramatically improves performance, achieving 90.6% accuracy and a weighted F1-score of 0.905.

TABLE IV
CLUSTER-ANCHORED WEAK SUPERVISION SUMMARY (COARSE POS)

Setting	Weak Tokens	Coverage	Avg. Conf	Min Conf	Acc.	W-F1
No Cluster LF	1,839	0.0517	0.5057	0.50	0.0999	0.0382
With Cluster LF	32,670	0.9189	0.7505	0.55	0.9059	0.9049

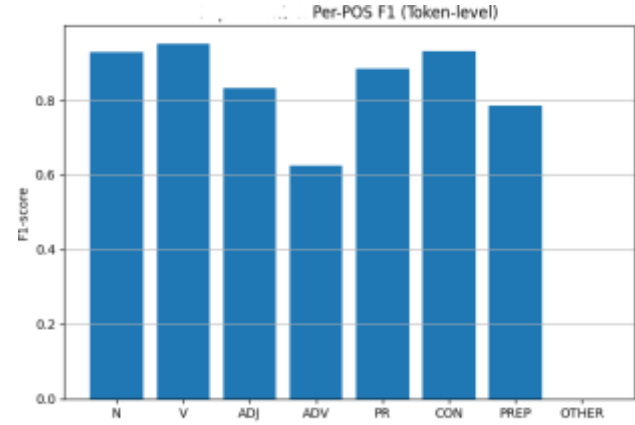


Fig. 11. Per-POS F1

Applying Viterbi decoding introduces a modest reduction in token-level accuracy (from 90.6% to 89.4%) but improves global sequence coherence, as discussed below.

2) *Effect of Sequence Smoothing:* Figure 12 illustrates the effect of Viterbi sequence smoothing on accuracy and weighted F1-score. While the cluster-based classifier alone achieves the highest token-level accuracy, the Viterbi-smoothed model balances local evidence with transition constraints, producing syntactically more plausible tag sequences.

3) *Confusion Matrix Analysis:* Figure 13 shows the confusion matrix for the cluster-LF + Viterbi model under the coarse POS tagset. The diagonal dominance indicates strong overall performance.

Nouns and verbs exhibit high recall (0.972 and 0.941 respectively), reflecting the effectiveness of transition modeling in stabilizing frequent syntactic patterns. However, reductions in recall for ADV and PREP categories suggest that strong transition biases can occasionally override local emission evidence, particularly for syntactically flexible tokens.

4) *Per-POS Performance:* Figure 14 presents per-POS F1-scores for the Viterbi-smoothed model. Content words (N, V, ADJ) retain strong performance, while function words benefit from improved contextual consistency.

5) *Discussion:* The slight decrease in accuracy after Viterbi decoding reflects a well-known trade-off in structured prediction: enforcing global consistency may suppress locally confident but globally implausible predictions. Nevertheless, the Viterbi-smoothed model produces more linguistically coherent tag sequences, which is particularly important for downstream syntactic and semantic tasks.

Overall, these results demonstrate that combining cluster-anchored weak supervision with HMM-based sequence mod-

TABLE V
EFFECT OF VITERBI SEQUENCE SMOOTHING

Setting	Weak Tokens	Coverage	Avg. Conf	Acc.	W-F1
No Cluster LF	1,839	0.052	0.51	0.100	0.038
Cluster LF	32,670	0.919	0.75	0.906	0.905
Cluster LF + Viterbi	32,670	0.919	0.75	0.894	0.891

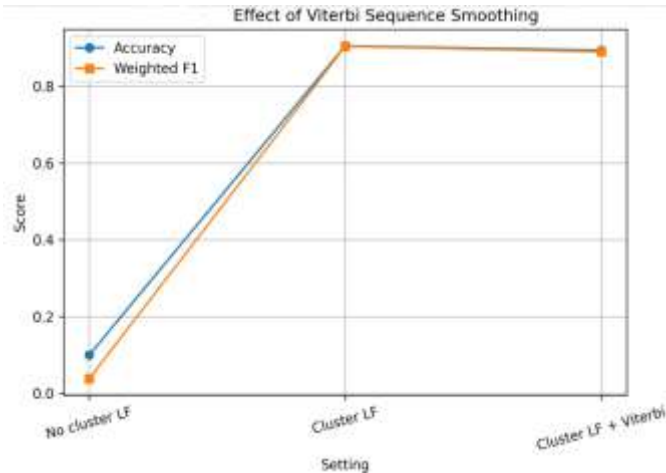


Fig. 12. Effect of Viterbi sequence smoothing on accuracy and weighted F1-score.

eling yields a robust and principled approach to POS tagging in low-resource settings.

VI. DISCUSSION

The experimental results have presented a more holistic picture of the advantages and limitations of various semi-supervised approaches for POS tagging in a low-resource language like Assamese.

First, the self-training experiment for confidence filtering shows us that naïve self-training is only a very limited option in this case. Although pseudo-labels are chosen due to a very high posterior confidence, we train the first supervised model for an extremely small seed set. As a consequence, high confidence does not always mean correct. Mistakes made in the beginning iterations are amplified but not rectified, resulting in a gradual reduction in performance over self-training rounds. This is a manifestation of confirmation bias and it is also indicative that confidence-based filtering is not enough if the underlying model is not strongly inductive biased.

The graph-based experiment on label propagation shows a different trade off. This approach allows label propagation to effectively distribute information (information from a few high-quality seed labels through this very small dataset) through a large vocabulary by modeling the distributional similarity of word types. This smoothing process has a great benefit on more common types such as nouns, verbs whose distributions are relatively steady. But since the approach is word-type level, it is naturally insensitive to context and sequence variation. As a result, modifiers and function words, whose

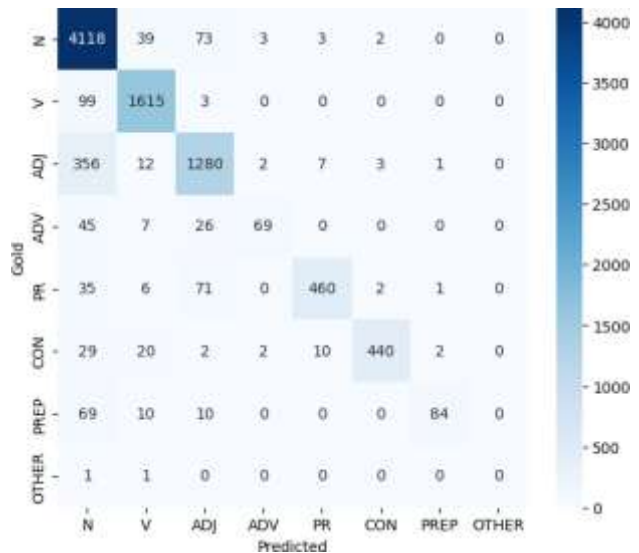


Fig. 13. Confusion matrix for the Cluster-LF + Viterbi model.

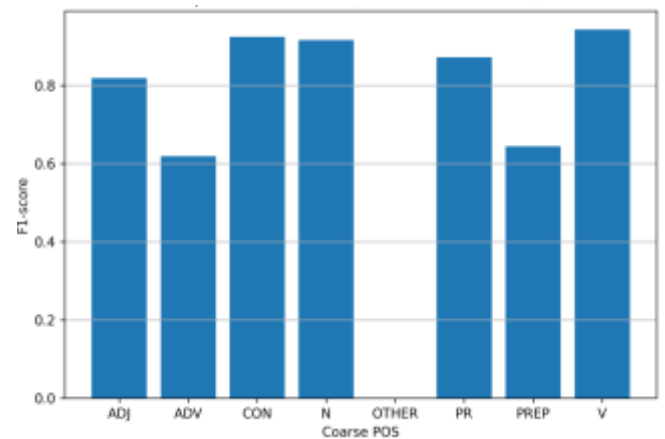


Fig. 14. Per-POS F1-scores for the Cluster-LF + Viterbi model.

POS comprehension depends much on sentence context, are still problematic. This restriction may account for why label propagation, although they outperform trivial baselines, falls short of token-level weak supervision.

All Ablation The results are clear proof that cluster-based labeling functions matter. In the case of no supervision of the cluster, the performance almost reaches random level, performing poorly with an accuracy of 9.99% and an insignificant weighted F1-score. The application of cluster-anchored weak supervision dramatically changes the learning problem; coverage grows from 5.17% to 91.88%, accuracy improves to 90.59%, and weighted F1 is raised to 0.905. These results support the view that distributional clusters are a strong inductive bias through the encoding of syntactic regularities that are not present in low resources. Importantly, this approach has the potential to close the gap between unsupervised POS induction and supervised tagging without manual annotation.

Although cluster-anchored weak supervision yields strong

token-level predictions, it does not directly enforce syntactic consistency at a sentence level in text. This limitation must be overcome by sequence smoothing with Viterbi decoding. Using transition probabilities learned from labeled data, the HMM framework punishes implausible tag sequences and encourages globally cohesive solutions. The findings reveal that despite this smoothing, nouns and verbs still score high in accuracy (F1 approx 0.916 and 0.943, respectively), and conjunctions have strong transition regularities ($F1 \approx 0.924$). Such gains suggest that sequence modeling is effective when applied to syntactically stable categories.

Simultaneously, remaining errors highlight critical linguistic problems. Modifiers such as adverbs and prepositions have a high precision but lower recall. This indicates that the Viterbi decoder favors other tags, unless emission probabilities are higher than the threshold value, indicating conservative decoding behavior under transition conditions. These patterns align with the ambiguity of function words found within Assamese and a lack of useful contextual cues for specific rare categories.

In the end, the discussion identifies an unmistakable set of trends: naive self-training does not work, graph-based propagation achieves limited gains, cluster-anchored weak supervision generates a large performance jump, and sequence-level smoothing refines predictions by enforcing syntactic unity. Collectively, these findings highlight the need to utilize distributional inductive bias with structured decoding approaches to have effective POS tagging in low resource languages.

VII. ERROR ANALYSIS

This section analyzes the main sources of errors within the proposed semi-supervised POS tagging framework, focusing on cluster-anchored weak supervision and Viterbi sequence smoothing models.

A. Errors from Naïve Self-Training

In self-training, errors are mostly due to confirmation bias. The initial classifier, trained on a very small labeled seed set, exhibits systematic misclassifications that are assigned high posterior confidence. These incorrect predictions are then incorporated as pseudo-labels, reinforcing early mistakes during retraining. Thus, errors propagate monotonically across iterations rather than being corrected. We see this behavior empirically in the declining accuracy curve observed during self-training rounds. These errors are particularly common in low-frequency tags and syntactically ambiguous tokens, indicating that confidence-based filtering alone is insufficient in low-resource conditions.

B. Word-Type Bias in Label Propagation

When displaying a graph-based label propagation, a new error rate is introduced with a different representation. One reason is, since all labels are propagated on the word-type level, each word type is considered a single dominant POS distribution regardless of changing context. Systematic over-smoothing is therefore achieved for polyfunctional words. This

bias favors common nouns and frequent verbs to the detriment of modifiers and function words whose recall (e.g., ADJ, ADV, PREP) is poorer. A word type is given an incorrect dominant label, and subsequent local occurrences cannot overrule this decision in subsequent contextualization events, thus leading to continuous mislabeling. This accounts for the comparatively high precision but weaker recall on various categories in label propagation results.

C. Residual Errors in Cluster-Anchored Weak Supervision

Even though cluster-anchored weak supervision achieves high overall accuracy, residual errors exist. Most of these errors are attributable to imperfect cluster purity. Though high-purity clusters perform well enough to reliably attribute correct labels, there exists a presence of mixed syntactic roles at the confidence threshold within clusters closer to the confidence threshold resulting in noisy weak labels. That noise disproportionately affects rare categories and boundary cases between syntactically similar classes (e.g., ADJ vs. N, ADV vs. PR). In addition, the coarse POS tagset obscures some fine-grained distinctions, meaning that there are some linguistically valid but evaluation-wise incorrect predictions.

D. Sequence-Level Errors after Viterbi Decoding

Viterbi sequence smoothing minimizes local inconsistencies on tag sequences quite extensively, but involves a trade-off. Transition constraints may sometimes trump strong local emission probabilities, especially for flexible modifier categories. This can be noticed in adverbs and prepositions, which yield high precision as well as lower recall after decoding. In such scenarios, the decoder prefers a globally frequent transition pattern; while often suppressing suitable but less common local configurations. Although this conservative bias increases the reliability of sequences, it can lead to underpredict syntactically optional categories.

E. Impact of Data Sparsity and Rare Labels

In this context, across each experiment, the OTHER category remains challenging due to extreme sparsity. Few training and test instances ensure that reliable estimation of both emission and transition probabilities is infeasible. Errors in this category barely add to overall accuracy, but emphasize the limits of coarse-grained assessment in accurately capturing rare linguistic phenomena. Similar sparsity effects are observed for ADV and PREP, especially for complex sentence constructions.

F. Summary of Error Sources

In summary, the dominant error sources evolve across experimental settings: confirmation bias in self-training, over-smoothing in label propagation, cluster impurity in weak supervision, and conservative decoding in Viterbi smoothing. Despite these challenges, the combined cluster-anchored and sequence-aware model substantially mitigates earlier error modes, producing a robust and linguistically coherent POS tagging system for Assamese.

VIII. CONCLUSION

This paper presented a comprehensive study of semi-supervised part-of-speech tagging for Assamese, a low-resource language characterized by limited annotated data and high lexical ambiguity. Through a sequence of progressively structured experiments, we investigated the effectiveness of self-training, graph-based label propagation, cluster-anchored weak supervision, and sequence-level decoding using a Hidden Markov Model with Viterbi inference.

Our findings demonstrate that naïve self-training with confidence filtering is ineffective in this setting. Despite high posterior confidence, pseudo-labels generated from a weakly trained seed model reinforce early errors, leading to performance degradation across iterations. This confirms that confidence alone is not a reliable proxy for correctness in low-resource POS tagging and highlights the need for stronger inductive bias.

Graph-based label propagation provides such an inductive bias by exploiting distributional similarity among word types. While this approach significantly outperforms random and majority baselines, its reliance on word-type representations limits contextual expressiveness. As a result, frequent nouns and verbs benefit most, whereas modifiers and function words remain challenging due to over-smoothing and lack of contextual differentiation.

The core contribution of this work is the cluster-anchored weak supervision framework, which bridges the gap between unsupervised induction and supervised POS tagging. By transforming high-purity distributional clusters into labeling functions, the approach generates large-scale, high-confidence weak supervision without manual annotation. This method achieves near-complete token coverage and yields substantial improvements in both accuracy and weighted F1-score, demonstrating that distributional clusters encode strong syntactic regularities.

Building upon this foundation, we further introduced sequence smoothing using HMM-based Viterbi decoding to explicitly model sentence-level structure. While token-level classifiers already produce strong local predictions, Viterbi decoding enforces global consistency by incorporating learned transition preferences. Empirically, this results in syntactically coherent tag sequences, particularly benefiting nouns, verbs, and conjunctions. Although a small reduction in token-level accuracy is observed, the overall model achieves a better balance between local evidence and global structure, which is essential for linguistically plausible tagging.

Overall, this work shows that effective semi-supervised POS tagging in low-resource languages requires a combination of distributional inductive bias and sequence-level modeling. The proposed framework avoids the pitfalls of self-training, scales without additional annotation, and produces robust results across major POS categories. Beyond Assamese, the methodology is language-agnostic and readily applicable to other low-resource and morphologically rich languages.

Future work may explore neural sequence models initialized with cluster-based weak supervision, adaptive transition modeling, and joint learning of clustering and tagging. We

believe that the proposed approach provides a principled and practical direction for advancing syntactic analysis in low-resource settings.

REFERENCES

- [1] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [2] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.
- [3] T. Brants, "Tnt – a statistical part-of-speech tagger," in *Sixth Applied Natural Language Processing Conference*. Association for Computational Linguistics, 2000, pp. 224–231. [Online]. Available: <https://aclanthology.org/A00-1031/>
- [4] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*, 2001, pp. 282–289. [Online]. Available: <https://dl.acm.org/doi/10.5555/645530.655813>
- [5] P. Brown *et al.*, "Class-based n-gram models of natural language," *Computational Linguistics*, 1992.
- [6] C. Christodoulopoulos, S. Goldwater, and M. Steedman, "Two decades of unsupervised pos induction," *Proceedings of EMNLP*, 2010.
- [7] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 1995, pp. 189–196. [Online]. Available: <https://aclanthology.org/P95-1026/>
- [8] A. Ratner *et al.*, "Snorkel: Rapid training data creation with weak supervision," *VLDB*, 2017.
- [9] K. Kann *et al.*, "Weakly supervised pos taggers perform poorly on truly low-resource languages," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/6317>
- [10] D. Das and S. Petrov, "Unsupervised part-of-speech tagging with bilingual graph-based projections," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2011, pp. 600–609. [Online]. Available: <https://aclanthology.org/P11-1061/>
- [11] O. Täckström, D. Das, S. Petrov, R. McDonald, and J. Nivre, "Token and type constraints for cross-lingual part-of-speech tagging," *Transactions of the Association for Computational Linguistics*, vol. 1, pp. 1–12, 2013. [Online]. Available: <https://aclanthology.org/Q13-1001/>
- [12] O. Täckström, R. McDonald, and J. Uszkoreit, "Cross-lingual word clusters for direct transfer of linguistic structure," in *Proceedings of NAACL-HLT*. Association for Computational Linguistics, 2012, pp. 477–487. [Online]. Available: <https://aclanthology.org/N12-1052/>
- [13] J. Nivre, M.-C. de Marneffe, F. Ginter, J. Hajic, C. D. Manning, S. Pyysalo, S. Schuster, F. Tyers, and D. Zeman, "Universal dependencies v2: An evergrowing multilingual treebank collection," in *Proceedings of the Twelfth Language Resources and Evaluation Conference (LREC)*. European Language Resources Association, 2020, pp. 4034–4043. [Online]. Available: <https://aclanthology.org/2020.lrec-1.497/>
- [14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*. Association for Computational Linguistics, 2019, pp. 4171–4186. [Online]. Available: <https://aclanthology.org/N19-1423/>
- [15] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is multilingual bert?" in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019, pp. 4996–5001. [Online]. Available: <https://aclanthology.org/P19-1493/>
- [16] D. Kondratyuk and M. Straka, "75 languages, 1 model: Parsing universal dependencies universally," in *Proceedings of EMNLP-IJCNLP*. Association for Computational Linguistics, 2019, pp. 2779–2795. [Online]. Available: <https://aclanthology.org/D19-1279/>