ACCENTMATCH: ADAPTIVE COMPETENCE CONTROL VIA EXTERNAL NETWORK TEACHING

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper, we present AccentMatch, a novel approach that distills the reasoning capabilities of Large Language Models (LLMs) into efficient Semi-Supervised Learning (SSL) students. Unlike standard distillation, AccentMatch introduces a dynamic "teacher annealing" mechanism based on training dynamics. We utilize an LLM as a teacher to generate high-quality initial pseudo-labels. We then fuse these pseudo-labels with the student's own predictions, dynamically adjusting the weight of the LLM guidance based on the student's learning status of unlabeled examples. This ensures the student relies on the LLM pseudo-label when it has a low learning status for an unlabeled example and transitions to self-training as it stabilizes. We show that our approach outperforms both state-of-the-art SSL baselines and few-shot LLMs across six text classification benchmarks, setting a new standard for efficient, high-accuracy classification in low-resource regimes.

1 INTRODUCTION

Semi-supervised learning (SSL) leverages a model to generate pseudo-labels for unlabeled data, enabling iterative increase of supervision from limited labeled examples (Arazo et al., 2019; Lee et al., 2013). A central challenge in SSL is the pseudo-label quality: with scarce supervision, early predictions are often unreliable, and their errors can accumulate over training. To mitigate this, many methods apply high-confidence thresholding, using only pseudo-labels predicted with confidence above some fixed high threshold (Lee et al., 2013; Xie et al., 2020a; Sohn et al., 2020; Wang et al., 2023; Zhang et al., 2021). While effective at reducing noise (Arazo et al., 2020), such filtering discards a large portion of unlabeled data, limiting the benefits of large-scale unlabeled corpora. More recent approaches replace fixed high-confidence thresholds with continuous weighting, allowing all samples to contribute to training by adaptively scaling their contribution to the loss based on the model confidence (Chen et al., 2023). However, these approaches still rely on the model's own predictions, making them susceptible to overconfident errors, which can reinforce themselves during training, leading to confirmation bias and error accumulation (see Figures 1a and 1b).

Large language models (LLMs), on the other hand, have demonstrated strong performance at inference time, with minimal few-shot supervision (Brown et al., 2020). However, despite their competitive predictions, fully trained SSL models ultimately surpass LLM-based inference (see Figures 1c and 1d). Notably, a closer look at the impurity (errors) of pseudo-labels of unlabeled data reveals a complementary pattern: in the early stages of training, when the SSL model is not well-trained, LLM's predictions exhibit substantially lower impurity rates (see Figure 1e and 1f). This suggests that LLMs provide a more reliable correctness signal when the student model is weak and unstable.

To this end, we introduce AccentMatch—**A**daptive **C**ompetence **C**ontrol via **E**xternal **N**etwork **T**eaching—a novel approach that combines the complementary strengths of an LLM teacher with a lightweight, deployment-ready BERT (Devlin et al., 2019) SSL student, and dynamically fuses pseudo-labels from the student and the LLM. Its main novelty is a new competence-guided, instance-specific teacher-annealing mechanism that determines how authority is transferred from the generalist LLM teacher to the task-specific-tuned SSL student, throughout the entire training, using a continuous dynamic handover: the LLM pseudo-label receives greater weight for an example on which the student remains unstable, while the student progressively takes over only when its predictions across training iterations become consistently reliable for that example. The fusion is dynamically weighted based on the student's *learning status* for each unlabeled example, encapsulated in the average pseudo-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

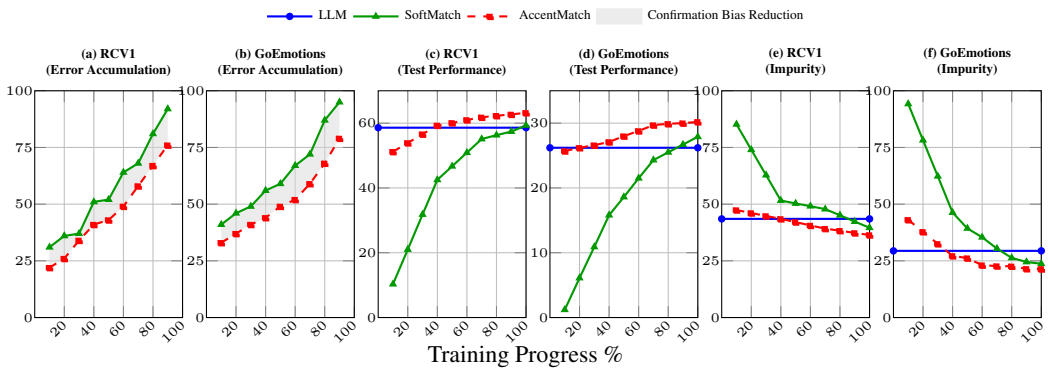


Figure 1: Pseudo-label training dynamics for RCV1 and GoEmotions with 20 labels per class. **(a, b) Error Accumulation:** Quantifies confirmation bias by measuring the percentage of examples that enter the training set with an incorrect pseudo-label (at some iteration) and remain incorrect until the end of training. The shaded region visually highlights the performance gap, demonstrating AccentMatch's superior ability to reduce confirmation bias compared to the SoftMatch baseline; **(c, d) Test Performance:** Evolution of test F1 as the models progress through the training iterations; **(e, f) Impurity:** Tracks the percentage of pseudo-labeled examples that participate in the training at some iteration with a wrong pseudo-label.

margin (APM), i.e., the average prediction margin across training iterations (Bartlett et al., 2017; Pleiss et al., 2020; Elsayed et al., 2018; Jiang et al., 2018; Sosea & Caragea, 2023). Intuitively, APM captures how confidently and consistently the model separates the predicted class from alternatives over time: large positive values indicate stable, correct predictions, while low or negative values reflect uncertainty or systematic errors. This design naturally emphasizes LLM guidance when the student is weak and gradually reduces the teacher's influence as the student becomes more confident.

We carry out comprehensive experiments using various experimental setups on six SSL text classification benchmarks: IMDB (Maas et al., 2011), RCV1 (Lewis et al., 2004), GoEmotions (Demszky et al., 2020), Amazon Review (McAuley & Leskovec, 2013), TREC-6 (Li & Roth, 2002), and Yelp Review (Asghar, 2016). We find that AccentMatch yields significant improvements on all benchmarks, outperforming strong LLM and SSL baselines. Notably, using only 20 labels per class, our method outperforms FlexMatch and SoftMatch by 3.6% and 2.2%, respectively, in accuracy on IMDB and by as much as 5.8% and 5.5%, respectively, on TREC-6.

Our contributions are as follows: **1)** We introduce AccentMatch, a novel approach that addresses the SSL cold-start problem and mitigates confirmation bias by combining the strengths of an LLM teacher with a lightweight SSL student; **2)** We reformulate the temporal Average Pseudo-Margin (APM) as an instance-level competence statistic that reflects the *model's learning status* of each unlabeled example and use it to actively gate the teacher–student fusion. We provide a decision-theoretic motivation for the instance-wise competence gating; **3)** We show that AccentMatch outperforms strong baselines and prior works on six well-established text classification benchmarks, showing larger improvements especially on challenging datasets, while achieving similar or better convergence; **4)** We provide a comprehensive analysis to offer insights into why AccentMatch outperforms existing SSL methods.

2 Related Work

Semi-supervised Learning Many approaches in SSL (Wang et al., 2022) such as Unsupervised Data Augmentation (UDA) (Xie et al., 2020a), FixMatch (Sohn et al., 2020), FlexMatch (Zhang et al., 2021), and FreeMatch (Wang et al., 2023) combine consistency regularization with pseudo-labeling, typically selecting unlabeled examples whose predictions exceed a confidence threshold (either fixed or adaptive). While effective, they often discard a large portion of unlabeled data through the confidence thresholding mechanism, limiting their potential. SoftMatch (Chen et al., 2023) replaces hard thresholds with a continuous weighting mechanism that scales the contribution of each unlabeled example based on confidence statistics, allowing greater utilization of unlabeled data. Specifically, it scales the contribution of each unlabeled example using a weight $\lambda(p)$ derived from the model's confidence. For a prediction p (class distribution), the weight is modeled as a truncated Gaussian:

$$\lambda(p) = \begin{cases} \exp\left(-\frac{(\max(p)-\mu_t)^2}{2\sigma_t^2}\right), & \text{if } \max(p) < \mu_t \\ 1, & \text{otherwise} \end{cases} \tag{1}$$

where μ_t and σ_t track the mean and variance of the confidence distribution over the training iterations.

MarginMatch (Sosea & Caragea, 2023) further incorporates training dynamics by tracking prediction margins over time to capture model uncertainty and design more informed thresholds for improved data utilization. Similarly, Uncertainty-aware Self-Training (UST) (Mukherjee & Awadallah, 2020) uses Bayesian uncertainty estimates to mitigate noisy pseudo-labels. Despite these advances, most SSL methods ultimately rely solely on the student model's own predictions, leaving them vulnerable to unreliable supervision in the early stages of training—particularly in low-resource settings.

Other recent SSL approaches include: FlatMatch (Huang et al., 2023) which minimizes a cross-sharpness measure to ensure consistent learning performance between the labeled and unlabeled sets; InterLUDE (Huang et al., 2024) which explicitly couples labeled and unlabeled examples through embedding interpolation and consistency losses; and RLGSSL (Heidari et al., 2024) which uses reinforcement learning to guide SSL.

Large Language Models as Weak or External Supervisors Large language models (LLMs) have emerged as weak supervisors for training smaller models, either via distillation or direct supervision. Shi et al. (2023) study SSL in conjunction with task-adaptive pre-training (TAPT) using language models and show that TAPT is a robust SSL learner. Gan & Wei (2024) focus on fine-tuning LLMs for SSL. Hybrid LLM and distant supervision approaches use LLMs to pseudo-label large datasets and train task-specific models in a fully supervised or semi-supervised manner (Chung & Chen, 2026; Park & Caragea, 2024). For example, VerifyMatch (Park & Caragea, 2024) introduces a verifier (SSL) model to validate LLM-generated labels, using agreement signal and MixUp-based denoising. ZeroMatch (Chung & Chen, 2026) combines zero-shot foundation-model distillation with SSL through a two-stage pipeline and auxiliary distillation heads. A key limitation of these approaches is that LLM supervision is either applied uniformly, regardless of the student model's evolving competence, or is used as binary signal (the SSL either matches the LLM pseudo-label or not). In contrast, AccentMatch treats the LLM as a dynamic teacher using a continuous handover of authority mechanism: its influence is strongest when the student is uncertain and gradually annealed as the student stabilizes. This adaptive integration is critical for addressing the SSL cold-start problem, where early student predictions are particularly unreliable.

3 Proposed Approach: AccentMatch

Notation Let $L = \{(x_1, y_1), ..., (x_B, y_B)\}$ be a batch of B labeled examples and $U = \{\hat{x}_1, ..., \hat{x}_{\nu B}\}$ be a batch of νB unlabeled examples, where ν is the batch-wise ratio of unlabeled to labeled examples. Let $p_\theta(y|x)$ denote the class distribution produced by a model θ on example x, and let $\hat{p}_\theta(y|x)$ denote the hard pseudo-label (i.e., argmax of this distribution represented as a one-hot vector) or soft pseudo-label (the class distribution itself). Let $\pi(x)$ and $\Pi(x)$ be weak and strong augmentations of x and $H(p, q)$ be the cross-entropy between two distributions p and q.

We propose AccentMatch, a novel approach that addresses the cold start problem in SSL through adaptive teacher annealing, where knowledge from an LLM teacher is distilled into an SSL (BERT) student. Instead of treating the teacher annealing according to a scheduler that transitions from 0 to 1 in small fixed steps, we utilize a dynamic mechanism. Particularly, we automatically anneal (reduce) the teacher's guidance on an unlabeled example when the student's real-time learning status of that example is high, and increase the reliance on the teacher when it is low. We formalize the student's learning status of an unlabeled example using pseudo-margins, averaged over training iterations.

Quantifying Learning Status via Pseudo-Margins The pseudo-margins averaged over training iterations summarize how consistently and confidently a model predicts the true (latent) class of an unlabeled example across training. Large positive values imply robust classification, while small or negative values indicate systematic misclassification (Pleiss et al., 2020; Sosea & Caragea, 2023). Thus, pseudo-margins of unlabeled examples can be used as a proxy to identify potentially mislabeled examples by the student.

Formally, for an unlabeled example $\hat{x}$ for which the student (BERT) model predicts the pseudo-label $\hat{c}$ at some epoch t, we treat $\hat{c}$ as "ground truth" and calculate the pseudo-margin (PM) for $\hat{c}$ at t as the difference between the logit of class $\hat{c}$, $z^t_{\hat{c}}$, and the largest logit among the logits of remaining classes, $z^t_j, j \neq \hat{c}$, as follows:

$$PM^t_{\hat{c}}(\hat{x}) = z^t_{\hat{c}}(\hat{x}) - \max_{j \neq \hat{c}} z^t_j(\hat{x}) \tag{2}$$

Algorithm 1 AccentMatch

Require: Labeled data L; unlabeled data U; maximum number of epochs T; number of classes C; base model θ; LLM model θ^{LLM}; weak augmentations π; strong augmentations Π.
Generate hard or soft pseudo-labels for every $\hat{x}_i \in U$ using θ^{LLM}:
$\hat{y}^{\text{LLM}}(\hat{x}_i) = \hat{p}_{\theta^{\text{LLM}}}(\hat{x}_i)$ ▷ Logit-based or verbalized prediction/confidence
for $t = 1$ to T **do**
while U is not exhausted **do**
Labeled batch $L_b = \{(x_1, y_1), ..., (x_B, y_B)\}$, unlabeled batch $U_b = \{\hat{x}_1, ..., \hat{x}_{\nu B}\}$
for each $\hat{x}_i \in U_b$ **do**
Compute logits z_c using model θ for each class c after applying weak augmentations, $\pi(\hat{x}_i)$.
Obtain $\hat{c} = \arg\max(p_\theta(y|\pi(\hat{x}_i)))$ ▷ Student's predicted pseudo-label
Calculate pseudo-margin $PM^t_{\hat{c}}$ (Eq. 2) and update Average $APM^t_{\hat{c}}$ (Eq. 3) for class $\hat{c}$.
Compute normalized instance-specific learning status for $\hat{x}_i$:
$S_{\hat{c}}(\hat{x}_i) = \frac{1}{1+\exp(-APM^t_{\hat{c}}(\hat{x}_i))}$ ▷ Student's real-time learning status/Stability metric on $\hat{x}_i$
Compute hard or soft pseudo-label by fusing the LLM (θ^{LLM}) and base model (θ) predictions:
$\tilde{y}(\hat{x}_i) = S_{\hat{c}}(\hat{x}_i) \times \hat{p}_\theta(y|\pi(\hat{x}_i)) + (1 - S_{\hat{c}}(\hat{x}_i)) \times \hat{y}^{\text{LLM}}(\hat{x}_i)$ ▷ Fused pseudo-label
end for
Minimize total loss $\mathcal{L} = \mathcal{L}_s + \mathcal{L}_u$
$\mathcal{L}_s = \frac{1}{B}\sum_{i=1}^{B} H(y_i, p_\theta(y|\pi(x_i)))$
$\mathcal{L}_u = \frac{1}{\nu B}\sum_{i=1}^{\nu B} \lambda(p_\theta(y|\pi(\hat{x}_i)))H(\tilde{y}(\hat{x}_i), p_\theta(y|\Pi(\hat{x}_i)))$, where $\lambda(.)$ is as in Eq. 1.
end while
end for

We then average all the pseudo-margins $PM^e_{\hat{c}}(\hat{x})$ across all epochs $e \leq t$ with respect to this $\hat{c}$ to distinguish between robust semantic understanding (consistently predicting the same class $\hat{c}$ across epochs for $\hat{x}$) and stochastic fluctuations (when the model does not know how to predict $\hat{x}$ and may frequently switch from one class to another across epochs). Note that although $PM^t_{\hat{c}}(\hat{x})$ is by design always positive for epoch t since $\hat{c}$ is the predicted pseudo-label at t (argmax at t), $PM^e_{\hat{c}}(\hat{x})$ at some epoch e prior to t need not be positive (if the model predictions at epochs e and t do not match). We compute the Average Pseudo-Margin (APM) using an exponential moving average (instead of a simple average) to place more importance on recent iterations (as older margins may become obsolete). The APM is thus defined as follows:

$$\text{APM}^t_{\hat{c}}(\hat{x}) = \beta \cdot \text{APM}^{t-1}_{\hat{c}}(\hat{x}) + (1 - \beta) \cdot \text{PM}^t_{\hat{c}}(\hat{x}) \tag{3}$$

where β controls the memory (history) length. This temporal averaging acts as a *stability filter*: a high APM indicates that the student has consistently committed to a specific pseudo-label over time, signaling a mature "learning status" capable of self-training. In contrast, a low APM reflects ongoing oscillation in the pseudo-label, triggering our approach to rely on the stable guidance of the LLM teacher. In practice, we compute pseudo-margins for all classes c at each epoch t using Eq. 2 (by calculating the difference between the logit of c and the largest logit among the remaining classes) and maintain a vector of APMs for all classes accumulated over the training epochs. We dynamically retrieve the accumulated pseudo-margin value of the argmax class $\hat{c}$ to obtain $APM^t_{\hat{c}}$ at epoch t.

Instance-level, Competence-guided Adaptive Teacher Annealing We leverage this class-specific stability metric to define an instance-specific annealing schedule. First, we generate a high-quality static pseudo-label $\hat{y}^{\text{LLM}}$ for each unlabeled example using the LLM, which acts as the stable prior. This is done one-time before the SSL training starts. During training, at some iteration t, for a given unlabeled example $\hat{x}$, let $\hat{c} = \arg\max(p_\theta(y|\pi(\hat{x})))$ be the student's current predicted pseudo-label at t. We fuse the teacher prior with the student's prediction via a stability-gated linear combination. We map the unbounded APM of the pseudo-label $\hat{c}$ to a student stability score (learning status or competence) $S_{\hat{c}}(\hat{x}) \in [0, 1]$ using a sigmoid function:

$$S_{\hat{c}}(\hat{x}) = \frac{1}{1 + \exp(-\text{APM}^t_{\hat{c}}(\hat{x}))} \tag{4}$$

This coefficient $S_{\hat{c}}$ governs the annealing rate. The final rectified pseudo-label $\tilde{y}$ is computed as:

$$\tilde{y}(\hat{x}) = \underbrace{S_{\hat{c}}(\hat{x}) \cdot \hat{p}_\theta(y|\pi(\hat{x}))}_{\text{Student Term}} + \underbrace{(1 - S_{\hat{c}}(\hat{x})) \cdot \hat{y}^{\text{LLM}}(\hat{x})}_{\text{Teacher Term}} \tag{5}$$

This formulation creates a smooth, adaptive handover of authority: **1. Teacher-Scaffolding:** During the "cold start" phase or when SSL predictions are ambiguous (negative APMs), the stability score $S_{\hat{c}}$ remains low (close to 0—from the shape of sigmoid). In this regime, AccentMatch relies heavily on the LLM's prior to suppress noise, but crucially, as the student's APMs go from negative towards 0 ($S_{\hat{c}}$ goes towards 0.5), the model smoothly fuses the two signals, allowing the student to gradually assume more responsibility even before it is fully stable. **2. Adaptive Specialization:** As the student gains competence and its APMs become consistently positive, $S_{\hat{c}}$ shifts towards 1. This naturally anneals the teacher's influence, but does so continuously. This transition allows the student to leverage the teacher for cases where it has a low learning status, while increasingly asserting its own predictions on well-learned examples to capture domain-specific nuances that the generalist teacher may miss.

Finally, we minimize the total loss $\mathcal{L} = \mathcal{L}_s + \mathcal{L}_u$. The supervised loss $\mathcal{L}_s$ is computed on labeled data:

$$\mathcal{L}_s = \frac{1}{B} \sum_{i=1}^{B} H(y_i, p_\theta(y|\pi(x_i))) \tag{6}$$

The unsupervised loss $\mathcal{L}_u$ incorporates our pseudo-labels $\tilde{y}$ and the adaptive confidence weight $\lambda(\cdot)$ from Eq. 1:

$$\mathcal{L}_u = \frac{1}{\nu B} \sum_{i=1}^{\nu B} \lambda(p_\theta(y|\pi(\hat{x}_i)))H(\tilde{y}(\hat{x}_i), p_\theta(y|\Pi(\hat{x}_i))) \tag{7}$$

where π and Π represent weak and strong augmentations, respectively.

3.1 Theoretical Motivation for Competence-Guided Teacher Annealing

We provide a decision-theoretic motivation for the competence-guided weighting mechanism in AccentMatch. This design is motivated by the following observation: when the student is unstable, the LLM teacher has lower pseudo-label error; as the student becomes stable, it can surpass the static teacher by adapting to the target domain.

Let $e_S(x)$ and $e_T(x)$ denote the conditional pseudo-label error rates of the student and teacher, respectively:

$$e_S(x) = \Pr(\hat{y}_S(x) \neq y \mid x), \qquad e_T(x) = \Pr(\hat{y}_T(x) \neq y \mid x).$$

Assume that the competence statistic $m(x) = \mathrm{APM}(x)$ is order-preserving with respect to student reliability, and that there exists a crossover value m_0 such that

$$m(x) < m_0 \Rightarrow e_T(x) \leq e_S(x), \qquad m(x) \geq m_0 \Rightarrow e_S(x) \leq e_T(x).$$

This assumption captures the intended behavior of AccentMatch: the teacher is more reliable in the low-competence regime, while the student becomes more reliable once its predictions stabilize.

Proposition 3.1 (Optimality of competence-based gating)**.** *Let $\alpha(x) = S(x) = \sigma(\mathrm{APM}(x))$. Consider a randomized predictor that uses the student prediction with probability $\alpha(x)$ and the teacher prediction with probability $1 - \alpha(x)$. Its conditional risk is $R_\alpha(x) = \alpha(x)e_S(x) + (1 - \alpha(x))e_T(x)$. Under the crossover condition above, the pointwise risk-minimizing gate is $\alpha^\star(x) = \mathbf{1}\{m(x) \geq m_0\}$, and it satisfies $R_{\alpha^\star}(x) = \min\{e_S(x), e_T(x)\} \leq e_S(x), R_{\alpha^\star}(x) \leq e_T(x)$. Consequently, $\mathbb{E}_x[R_{\alpha^\star}(x)] \leq \min\{\mathbb{E}_x[e_S(x)], \mathbb{E}_x[e_T(x)]\}$. We show the proof of this proposition in Appendix L.*

The proposition shows that, when a competence statistic separates regimes in which the teacher or student is more reliable, instance-wise gating dominates either fixed teacher-only or fixed student-only supervision under empirical crossover conditions. Thus, risk optimality is proven for a hard, thresholded step function $\alpha^*(x) = \mathbf{1}\{m(x) \geq m_0\}$ under the crossover assumption. In practice, AccentMatch implements a smooth approximation to this risk-minimizing gate. It replaces the discontinuous oracle gate $\alpha^\star$ with a smooth sigmoid gate $\sigma(\mathrm{APM}(x))$. In Section 5 (Table 4), we test this hard switch (AccentMatch^{binary}) and show that it underperforms the continuous sigmoid gating.

4 Experiments and Results

4.1 Baselines

First, we compare AccentMatch with fully supervised and LLM-based baselines. **BASE** trains BERT base uncased (Devlin et al., 2019) on the small labeled data only. **Zero-Shot** and **Few-Shot** use our

Dataset (Metric)	IMDB (Acc.)		RCV1 (F1)		GoEmotions (F1)		TREC-6 (Acc.)		Amazon (Acc.)		Yelp (Acc.)	
Num Labels	20	50	20	50	20	50	20	50	20	50	20	50
BASE	$69.1_{4.25}$	$78.4_{3.55}$	$22.1_{4.55}$	$24.2_{4.36}$	$9.2_{6.34}$	$20.4_{4.51}$	$80.4_{2.51}$	$85.1_{2.31}$	$47.6_{3.15}$	$48.4_{3.01}$	$38.5_{1.45}$	$46.2_{1.52}$
ZeroShot	82.9	82.9	57.8	57.8	24.6	24.6	78.1	78.1	48.1	48.1	47.2	47.2
FewShot	85.8	85.8	58.6	58.6	26.2	26.2	83.8	83.8	50.4	50.4	49.7	49.7
DS	$80.0_{4.12}$	$82.4_{3.65}$	$55.1_{4.21}$	$57.5_{3.56}$	$24.1_{2.86}$	$24.7_{2.65}$	$75.1_{1.86}$	$78.1_{1.96}$	$46.9_{4.04}$	$50.4_{2.45}$	$46.3_{1.67}$	$46.9_{1.53}$
DS-AUM	$81.4_{3.11}$	$82.2_{1.25}$	$56.3_{2.18}$	$58.1_{2.11}$	$25.3_{4.12}$	$25.7_{4.04}$	$76.8_{2.18}$	$78.6_{2.07}$	$47.2_{1.78}$	$53.6_{2.15}$	$48.1_{1.53}$	$49.2_{2.33}$
NOISY-S	$71.1_{2.41}$	$81.3_{2.35}$	$42.3_{3.56}$	$43.2_{3.41}$	$15.4_{5.18}$	$25.3_{3.51}$	$81.3_{2.42}$	$86.2_{2.21}$	$46.4_{3.12}$	$50.2_{2.85}$	$46.5_{1.55}$	$47.3_{1.53}$
UST	$72.5_{2.71}$	$82.1_{2.51}$	$52.3_{4.25}$	$54.1_{3.68}$	$17.4_{2.41}$	$23.1_{2.23}$	$82.2_{1.54}$	$87.5_{1.94}$	$45.2_{2.36}$	$51.5_{1.78}$	$46.8_{1.31}$	$49.1_{1.73}$
UDA	$77.4_{2.45}$	$83.2_{2.22}$	$54.0_{4.51}$	$56.7_{4.21}$	$24.5_{3.05}$	$25.9_{5.12}$	$84.2_{2.31}$	$88.1_{2.15}$	$47.1_{3.31}$	$52.1_{1.21}$	$45.8_{1.24}$	$49.4_{1.21}$
MarginMatch	$83.6_{2.78}$	$85.8_{2.13}$	$57.2_{1.79}$	$59.2_{1.86}$	$26.9_{3.87}$	$27.3_{3.19}$	$80.7_{1.78}$	$87.1_{1.98}$	$49.1_{1.53}$	$\underline{53.9}_{1.98}$	$49.6_{1.47}$	$52.4_{2.01}$
VerifyMatch	$\underline{86.1}_{1.79}$	$\underline{88.2}_{2.45}$	$\underline{60.4}_{1.87}$	$\mathbf{62.1}_{\mathbf{1.57}}$	$27.2_{2.17}$	$26.6_{1.86}$	$\underline{88.2}_{2.04}$	$\underline{90.2}_{2.17}$	$51.6_{1.86}$	$52.4_{1.75}$	$\underline{51.3}_{1.68}$	$55.1_{1.45}$
FlexMatch	$84.1_{3.96}$	$86.3_{3.52}$	$56.3_{4.56}$	$58.9_{4.21}$	$26.4_{4.61}$	$26.2_{4.51}$	$83.9_{4.22}$	$88.4_{3.58}$	$\underline{51.8}_{3.42}$	$51.3_{2.45}$	$50.1_{1.45}$	$48.4_{1.53}$
FixMatch	$84.2_{2.61}$	$84.2_{2.32}$	$55.2_{3.51}$	$56.3_{3.25}$	$24.1_{4.63}$	$24.9_{4.21}$	$80.1_{1.98}$	$89.1_{1.91}$	$50.6_{2.35}$	$51.1_{2.22}$	$49.7_{1.26}$	$48.1_{1.48}$
SoftMatch	$85.5_{2.31}$	$88.1_{2.15}$	$59.3_{3.19}$	$60.2_{3.27}$	$\underline{27.9}_{4.31}$	$\underline{27.8}_{4.25}$	$84.2_{2.21}$	$90.1_{2.11}$	$51.7_{1.43}$	$53.7_{2.44}$	$50.4_{2.06}$	$\underline{55.9}_{1.78}$
AccentMatch	$\mathbf{87.7}_{\mathbf{2.60}}$	$\mathbf{89.8}_{\mathbf{2.93}}$	$\mathbf{63.2}_{\mathbf{2.17}}$	$\underline{61.5}_{2.54}$	$\mathbf{30.2}_{\mathbf{3.43}}$	$\mathbf{30.6}_{\mathbf{2.27}}$	$\mathbf{89.7}_{\mathbf{2.22}}$	$\mathbf{91.2}_{\mathbf{2.23}}$	$\mathbf{53.2}_{\mathbf{1.72}}$	$\mathbf{54.6}_{\mathbf{1.87}}$	$\mathbf{52.7}_{\mathbf{2.33}}$	$\mathbf{56.9}_{\mathbf{2.34}}$

Table 1: Results on six datasets with 20 and 50 labels per class. Best model is colored, second best is underlined.

LLM with zero-shot and few-shot CoT prompting, respectively; the latter also provides predictions (pseudo-labels) for AccentMatch. **DS** trains BERT on the small labeled data plus LLM-generated pseudo-labels for all unlabeled examples. **DS-AUM** additionally monitors pseudo-label AUM during training, removes low-AUM examples, and re-trains the model (Pleiss et al., 2020).

Second, we compare AccentMatch with teacher-student approaches: **NOISY-S** (Xie et al., 2020b) iteratively trains models using pseudo-labeling, whereas **UST** (Mukherjee & Awadallah, 2020) selects reliable pseudo-labels using uncertainty estimated via Monte Carlo dropout.

Third, we compare AccentMatch with consistency-based approaches. **UDA** (Xie et al., 2020a) enforces prediction consistency under backtranslation (Edunov et al., 2018). **FixMatch** (Sohn et al., 2020) uses a fixed confidence threshold for consistency between weak and strong augmentations, while **FlexMatch** (Zhang et al., 2021) uses adaptive, class-dependent thresholds. **SoftMatch** (Chen et al., 2023) weights pseudo-labels by confidence, whereas **MarginMatch** (Sosea & Caragea, 2023) filters them using training dynamics. Finally, **VerifyMatch** (Park & Caragea, 2024) validates LLM-generated labels with a secondary verifier and uses MixUp to denoise them.

4.2 Datasets and Experimental Setup

We evaluate AccentMatch on six text classification benchmarks: IMDB (Maas et al., 2011), RCV1 (Lewis et al., 2004), GoEmotions (Demszky et al., 2020), TREC-6 (Li & Roth, 2002), Amazon Review (McAuley & Leskovec, 2013), and Yelp Review (Asghar, 2016). We use 20 and 50 labeled examples per class, treating the remaining training data as unlabeled. We report macro-F1 for RCV1 and GoEmotions (as both are class-imbalanced) and accuracy for the remaining datasets, averaged over 3 runs with standard deviations as subscripts. We use BERT base uncased as the student backbone and Qwen (Yang et al., 2025) as the LLM teacher with few-shot CoT prompting. All baselines share the same augmentation pipeline. Further experimental details are provided in Appendix A. We show statistical significance results over 10 runs between AccentMatch and the best baselines in Appendix H.

We set the smoothing parameter β in Eq. 3 to 0.997 and use a 10-epoch warmup for the APM values to stabilize. The student's contribution to the unsupervised loss remains modulated by the confidence weight $\lambda(\cdot)$ in Eq. 1, including during warmup. We provide a sensitivity analysis over β in Appendix E and a computational cost analysis in Appendix B.

4.3 Results

Main Results The complete results across all six datasets are reported in Table 1. Notably, Zero-Shot already provides a strong starting point, substantially outperforming the supervised baseline (BASE). For example, on IMDB with 20 labels per class, Zero-Shot improves over BASE by 13.8 points in accuracy (82.9 vs. 69.1). Similarly, DS remains competitive, occasionally outperforming standard SSL methods (e.g., on RCV1 with 50 labels, DS achieves 57.5 vs. 56.3 for FixMatch). These results highlight the strength of LLM-derived supervision even without iterative refinement.

Among SSL approaches, consistency-based methods are very strong. In particular, FlexMatch and SoftMatch emerge as the most competitive baselines across datasets. For example, on GoEmotions with 20 labels, SoftMatch achieves the best baseline performance (27.9 F1), outperforming methods

Dataset (Metric)	IMDB (Acc.)		RCV1 (F1)		GoEmotions (F1)		TREC-6 (Acc.)		Amazon (Acc.)		Yelp (Acc.)	
Num Labels	20	50	20	50	20	50	20	50	20	50	20	50
AccentMatch$^{\text{no-LLM}}$	$85.5_{2.31}$	$88.1_{2.15}$	$59.3_{3.19}$	$60.2_{3.41}$	$27.9_{4.31}$	$\underline{27.8}_{4.25}$	$84.2_{2.21}$	$90.1_{2.11}$	$51.7_{1.43}$	$53.7_{2.44}$	$50.4_{2.06}$	$55.9_{1.78}$
AccentMatch^{naive}	$87.3_{1.24}$	$\underline{89.7}_{1.30}$	$\underline{61.4}_{2.08}$	$60.6_{3.49}$	$\underline{28.1}_{2.46}$	$27.7_{1.63}$	$85.6_{2.90}$	$\underline{90.9}_{2.07}$	$\underline{52.5}_{1.82}$	$54.0_{1.00}$	$50.7_{2.48}$	$\underline{56.3}_{3.36}$
AccentMatch^{agreement}	$87.2_{3.84}$	$89.5_{1.04}$	$60.7_{3.91}$	$60.7_{1.18}$	$27.6_{3.99}$	$27.8_{3.77}$	$85.6_{2.34}$	$90.4_{1.39}$	$51.9_{3.63}$	$53.9_{3.45}$	$52.2_{2.76}$	$56.0_{3.95}$
AccentMatch^{scheduler}	$\underline{87.4}_{3.13}$	$89.4_{1.12}$	$60.8_{3.29}$	$\underline{61.0}_{1.37}$	$27.4_{3.57}$	$27.8_{3.12}$	$\underline{85.8}_{2.13}$	$90.3_{1.15}$	$51.8_{3.39}$	$\underline{54.4}_{3.47}$	$\underline{52.3}_{2.57}$	$56.1_{3.51}$
AccentMatch	$\mathbf{87.7}_{2.60}$	$\mathbf{89.8}_{2.93}$	$\mathbf{63.2}_{2.17}$	$\mathbf{61.5}_{2.54}$	$\mathbf{30.2}_{3.43}$	$\mathbf{30.6}_{2.27}$	$\mathbf{89.7}_{2.22}$	$\mathbf{91.2}_{2.23}$	$\mathbf{53.2}_{1.72}$	$\mathbf{54.6}_{1.87}$	$\mathbf{52.7}_{2.33}$	$\mathbf{56.9}_{2.34}$

Table 2: Ablation study of our method. The best model is colored and the second best is underlined.

such as Noisy-S and UST by large margins. VerifyMatch is also a strong competitor, achieving the best or second-best performance among baselines on several datasets (e.g., 62.1 F1 on RCV1 with 50 labels per class and 90.2 accuracy on TREC-6). However, its gains are less consistent across tasks, especially in more challenging settings such as GoEmotions, where performance remains below the strongest consistency-based method, SoftMatch.

Remarkably, AccentMatch consistently yields the best overall performance across nearly all datasets and label regimes (11 out of 12 settings). For example, on IMDB with 20 labels per class, AccentMatch achieves 87.7 accuracy, outperforming the strongest baseline (VerifyMatch at 86.1) by $+1.6$ points. On GoEmotions, AccentMatch yields the largest gains, reaching 30.2 F1 with 20 labels per class and 30.6 with 50, substantially surpassing SoftMatch (27.9/27.8) and VerifyMatch (27.2/26.6). Overall, AccentMatch establishes a new state-of-the-art, with consistent gains over both strong SSL and recent LLM-guided methods such as VerifyMatch, demonstrating the effectiveness of competence-guided teacher annealing in extreme low-resource regimes.

Ablation Study We perform an ablation study to isolate the components that lead to the success of our AccentMatch. To this end, we design the following variations of AccentMatch: 1) AccentMatch^{naive} uses an equal weight for both the LLM and the BERT pseudo-labels for each unlabeled example (i.e., $S = 0.5$ in Eq. 5); 2) AccentMatch^{agreement} uses an unlabeled example for training only if both pseudo-labels from the LLM and BERT agree (otherwise, the example is discarded); 3) AccentMatch^{scheduler} employs a linear scheduling for the weighting term S (from 0 to 1 in small steps of 0.1); 4) AccentMatch$^{\text{no-LLM}}$ does not use the LLM at all. We show the results in Table 2.

As can be seen from the table, the full AccentMatch model consistently scores first among all ablated versions, pinpointing the effectiveness of the LLM pseudo-label and the dynamic, instance-specific weighting scheme. For example, on RCV1 using 20 labels per class, the full AccentMatch model achieves an F1 of 63.2 whereas AccentMatch$^{\text{no-LLM}}$ yields an F1 of 59.3, AccentMatch^{naive} yields an F1 of 61.4, AccentMatch^{agreement} obtains an F1 of 60.7, and AccentMatch^{scheduler} achieves an F1 of 60.8, showing a substantial improvement of the full AccentMatch over the ablated versions. This demonstrates the general robustness and value of using LLM pseudo-labels.

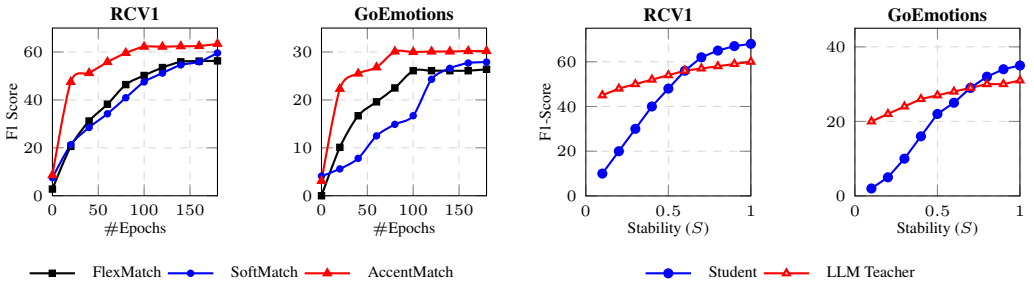
5 Analysis

In this section, we perform a comprehensive analysis to gain deeper insights into the effectiveness of AccentMatch. We examine its generalizability across teacher LLMs and student backbones, its convergence speed, and the relationship between the student's stability score and prediction quality. We also investigate how different strategies for transferring supervision from the LLM to the student affect performance.

Architectural Generalizability: Teacher LLMs and Student Backbones To verify that the effectiveness of our framework is not dependent on a specific LLM architecture or student backbone, we evaluate AccentMatch using three open-source models as teachers: Qwen (Yang et al., 2025), Llama-3-8B (Touvron et al., 2023) and Mistral-7B (Jiang et al., 2023) across two distinct student backbones, BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019). Our analysis in Table 3 demonstrates that teacher quality significantly impacts student performance, as Qwen consistently provides the strongest few-shot guidance and yields the best overall results across all datasets and backbones. The framework exhibits robust generalizability regardless of the student architecture; for example, on the IMDB dataset with 20 labels per class, utilizing a Qwen teacher boosts the BERT student to 87.7 and the RoBERTa student to 88.0, marking substantial gains over their respective SoftMatch baselines of 85.5 and 85.4. Furthermore, AccentMatch successfully extracts valuable signals even from relatively weaker teachers. In the RCV1 50 labels per class regime, while the Llama-3 teacher achieves only a 54.9 F1 score, a RoBERTa student trained via AccentMatch reaches

Dataset (Metric)	IMDB (Accuracy)		RCV1 (F1)		GoEmotions (F1)		Yelp (Acc.)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
LLM								
FewShot (LLama 3)	82.1	82.1	54.9	54.9	24.8	24.8	47.9	47.9
FewShot (Mistral)	83.2	83.2	55.2	55.2	23.9	23.9	48.2	48.2
FewShot (Qwen)	85.8	85.8	58.6	58.6	26.2	26.2	49.7	49.7
BERT								
BERT	$69.1_{4.25}$	$78.4_{3.55}$	$22.1_{4.55}$	$24.2_{4.36}$	$09.2_{6.34}$	$20.4_{4.51}$	$38.5_{1.45}$	$46.2_{1.52}$
SoftMatch	$85.5_{2.31}$	$88.1_{2.15}$	$59.3_{3.19}$	$60.2_{3.41}$	$27.9_{4.31}$	$27.8_{4.25}$	$50.4_{2.06}$	$55.9_{1.78}$
AccentMatch (LLama3)	$86.2_{2.31}$	$88.9_{2.07}$	$61.3_{1.84}$	$60.8_{1.36}$	$28.9_{2.01}$	$29.3_{2.35}$	$51.2_{2.41}$	$53.8_{1.74}$
AccentMatch (Mistral)	$87.2_{1.41}$	$89.3_{2.13}$	$62.4_{1.53}$	$60.9_{1.76}$	$29.4_{2.18}$	$29.9_{2.23}$	$51.9_{1.95}$	$54.5_{2.22}$
AccentMatch (Qwen)	**$87.7_{2.60}$**	**$89.8_{2.93}$**	**$63.2_{2.17}$**	**$61.5_{2.54}$**	**$30.2_{3.43}$**	**$30.6_{2.27}$**	**$52.7_{2.33}$**	**$56.9_{2.34}$**
RoBERTa								
RoBERTa	$68.7_{3.61}$	$78.2_{3.53}$	$22.2_{4.31}$	$24.2_{4.11}$	$10.1_{6.21}$	$21.2_{4.37}$	$39.2_{3.11}$	$46.0_{2.87}$
SoftMatch	$85.4_{2.15}$	$90.4_{2.06}$	$59.7_{1.53}$	$60.3_{2.51}$	$28.1_{2.13}$	$27.7_{1.64}$	$51.2_{2.56}$	$51.6_{1.97}$
AccentMatch (LLama3)	$86.2_{1.54}$	$90.2_{1.76}$	$60.1_{1.51}$	$60.7_{1.65}$	$29.3_{2.31}$	$29.7_{2.06}$	$51.4_{3.44}$	$53.5_{2.67}$
AccentMatch (Mistral)	$86.9_{1.89}$	$90.5_{2.06}$	$61.3_{1.49}$	$61.3_{1.61}$	$29.7_{2.04}$	$29.5_{2.19}$	$52.3_{2.99}$	$54.7_{1.82}$
AccentMatch (Qwen)	**$88.0_{2.15}$**	**$91.3_{2.17}$**	**$63.5_{1.67}$**	**$61.5_{1.65}$**	**$30.4_{2.13}$**	**$30.7_{2.34}$**	**$52.6_{1.75}$**	**$56.6_{1.36}$**

Table 3: Comparisons using different teacher LLMs and student models. The best results are colored.



(a) Convergence speed

(b) Analysis of Stability Score

Figure 2: (a) Convergence speed of AccentMatch vs FlexMatch and SoftMatch on RCV1 and GoEmotions. (b) Analysis of stability score S on the same datasets using 20 labels per class.

60.7, effectively leveraging early-stage guidance to surpass both the static teacher and the standard semi-supervised learning baseline. This crossover in performance confirms that instance-specific stability weighting allows the student to benefit from external knowledge while eventually specializing to domain-specific nuances that the generalist teacher may miss.

Convergence Speed We compare in Figure 2a the F1 of AccentMatch against SoftMatch and FlexMatch during the entire training process on two datasets, RCV1 and GoEmotions, with varying number of classes (103 and 27 classes, respectively) in the 20 labels per class setting to understand the convergence speed of our AccentMatch. We observe common trends across the experiments: AccentMatch achieves high performance much quicker compared to both SoftMatch and FlexMatch, indicating that the LLM predictions boost the capabilities of the model to learn quickly in the early stages of training. Similar results are observed on the other datasets in all settings (data not shown).

Analysis of Stability Score To validate the effectiveness of our adaptive annealing mechanism, we analyze the model performance conditioned on the student's learning status or stability score S. In Figure 2b, we bin unlabeled examples according to the student's stability score S and evaluate the accuracy of both the Student and LLM Teacher on these bins. We observe a strong positive correlation between the stability score S and the student's prediction accuracy, validating S as a reliable proxy for class correctness. As illustrated, the student's performance curve is monotonic: when the stability score is low ($S < 0.2$), the student's accuracy is indeed low, reflecting a state of high uncertainty or oscillation where the model cannot reliably predict the target class. Conversely, as S increases, the student's accuracy rises sharply, indicating that the metric successfully identifies true positives. This correlation justifies our dynamic weighting schedule. In the low-stability regime ($S < 0.6$), where the student is demonstrably prone to error, the LLM Teacher maintains robust performance, providing the necessary external guidance to prevent early-training confirmation bias. However, as the student gains competence, reflected by high stability scores ($S > 0.7$), its performance crosses over and surpasses the static LLM prior. Specifically, we observe crossover points at $S \approx 0.6$ for RCV1 and 0.7 for GoEmotions. This confirms that the Student successfully adapts to the domain-specific nuances of the labeled data, eventually outperforming the generalist teacher on well-learned examples, thereby justifying our dynamic fusion strategy.

Dataset (Metric)	IMDB (Acc.)		RCV1 (F1)		GoEmotions (F1)		TREC-6 (Acc.)		Amazon (Acc.)		Yelp (Acc.)	
Labels per class	20	50	20	50	20	50	20	50	20	50	20	50
AccentMatch$^{\text{no-LLM}}$	$85.5_{2.31}$	$88.1_{2.15}$	$59.3_{3.19}$	$60.2_{3.41}$	$27.9_{4.31}$	$27.8_{4.25}$	$84.2_{2.21}$	$90.1_{2.11}$	$51.7_{1.43}$	$53.7_{2.44}$	$50.4_{2.06}$	$55.9_{1.78}$
AccentMatch$^{20\%\text{-warm}}$	$84.3_{2.14}$	$87.6_{2.05}$	$58.7_{3.21}$	$59.1_{3.15}$	$27.4_{3.85}$	$27.3_{3.62}$	$83.6_{2.41}$	$89.7_{2.18}$	$51.3_{2.05}$	$53.4_{1.88}$	$50.1_{2.32}$	$54.8_{2.45}$
AccentMatch$^{S=0.25}$	$87.1_{2.45}$	$\underline{89.2}_{1.85}$	$\underline{61.1}_{2.78}$	$\underline{60.7}_{2.91}$	$27.6_{3.12}$	$27.3_{3.40}$	$\underline{85.1}_{2.65}$	$\underline{90.2}_{1.92}$	$\underline{52.1}_{2.11}$	$54.0_{1.64}$	$50.5_{2.20}$	$\underline{55.4}_{2.15}$
AccentMatch$^{S=0.75}$	$\underline{87.2}_{1.95}$	$88.7_{1.76}$	$60.2_{3.05}$	$60.1_{2.84}$	$\underline{28.2}_{2.95}$	$27.5_{2.80}$	$84.6_{2.30}$	$90.1_{1.85}$	$51.7_{2.25}$	$\underline{54.2}_{1.55}$	$\underline{50.7}_{2.15}$	$54.3_{2.30}$
AccentMatch^{binary}	$86.4_{2.60}$	$88.3_{2.12}$	$59.8_{3.42}$	$59.7_{3.10}$	$27.4_{3.50}$	$\underline{28.1}_{3.25}$	$84.7_{2.80}$	$89.8_{2.10}$	$52.0_{2.40}$	$54.1_{1.75}$	$50.6_{2.55}$	$54.2_{2.80}$
AccentMatch	$\mathbf{87.7}_{2.60}$	$\mathbf{89.8}_{2.93}$	$\mathbf{63.2}_{2.17}$	$\mathbf{61.5}_{2.54}$	$\mathbf{30.2}_{3.43}$	$\mathbf{30.6}_{2.27}$	$\mathbf{89.7}_{2.22}$	$\mathbf{91.2}_{2.23}$	$\mathbf{53.2}_{1.72}$	$\mathbf{54.6}_{1.87}$	$\mathbf{52.7}_{2.33}$	$\mathbf{56.9}_{2.34}$

Table 4: Additional ablations of teacher-student transfer. Results are means and standard deviations over three runs. The best result is **colored** and the second best is underlined.

Analysis of Teacher-Student Transfer We further analyze how different strategies for transferring supervision from the LLM to the student affect performance, complementing the ablations in Table 2. To this end, AccentMatch$^{20\%\text{-warm}}$ uses LLM pseudo-labels exclusively for the first 20% of training, then switches entirely to student pseudo-labels. AccentMatch$^{S=0.25}$ and AccentMatch$^{S=0.75}$ replace the dynamic weighting term with fixed values favoring the teacher and student, respectively. Finally, AccentMatch^{binary} uses only the LLM pseudo-label when $\text{APM} < 0$ and only the student pseudo-label otherwise, replacing the smooth weighting with a hard switch.

The results in Table 4 show that the 20%-warm-start variant underperforms the no-LLM baseline across all settings, suggesting that providing teacher guidance only at the beginning of training is insufficient under this abrupt switching strategy. Similarly, neither fixed weight (0.25 or 0.75) consistently improves over the equal weighting used by AccentMatch^{naive} in Table 2. We also observe that binary gating underperforms the full AccentMatch across all settings. For example, on GoEmotions with 20 and 50 labels per class, it achieves 27.4 and 28.1 F1, compared to 30.2 and 30.6 for full AccentMatch, respectively. These results further support the effectiveness of continuously adjusting the teacher and student contributions based on the learning status of each unlabeled example.

Additional Analyses We provide extensive supplementary analyses and implementation details in the appendix to further evaluate our approach. Appendix B presents a cost-benefit analysis and token profiling, showing minimal GPU overhead. Appendix C demonstrates that our teacher annealing framework readily generalizes to hard-thresholding methods like FixMatch and FlexMatch. Appendix D shows that hard pseudo-labels consistently outperform soft distributions, while Appendix E reveals that $\beta = 0.997$ balances historical stability against adaptation to recent dynamics. In Appendix F, we find that confidence-based LLM abstention underperforms dynamic stability weighting, and Appendix G shows that mid-training LLM re-prompting degrades accuracy by compounding student confirmation bias. For robustness, Appendix H confirms statistically significant gains ($p < 0.05$) across 10 independent runs, Appendix I demonstrates that dynamic annealing preserves generalization under cross-domain distribution shift, and Appendix J verifies that teacher guidance remains robust across prompt changes. Furthermore, Appendix K analyzes pseudo-label error rates directly against raw APM values, empirically supporting our theoretical motivation from Appendix L. Finally, Appendix M details prompt templates.

6 Conclusion

In this paper, we introduced AccentMatch, a semi-supervised learning approach centered around a novel competence-aware weighting mechanism based on Average Pseudo-Margins (APM), a dynamic statistic that captures the stability of the student's predictions during training. By continuously tracking the student's learning status at the instance level, AccentMatch adaptively regulates the influence of the LLM teacher, relying more on external supervision when the student is uncertain and progressively transitioning to self-training as the student stabilizes. This instance-specific, competence-guided design directly addresses the cold-start problem in SSL and mitigates the accumulation of early pseudo-labeling errors. Our comprehensive evaluation demonstrates that AccentMatch establishes a new SOTA in extremely low-resource settings, consistently outperforming strong baselines and prior works, as well as few-shot LLM prompting. Our analysis of training dynamics confirms that the student successfully specializes to the domain, eventually surpassing the generalization capabilities of the LLM teacher. More broadly, AccentMatch highlights the potential of combining powerful LLM with efficient lightweight students to enable accurate learning in domains where labeled data are scarce or expensive, e.g., healthcare and crisis informatics. An interesting future direction is the exploration of our competence-aware weighting mechanism applied to other domains, e.g., law, health, and finance.

AI USE STATEMENT

We used AI tools lightly to assist with final polishing of the paper and identify potential misspellings. The authors reviewed and revised all AI-assisted text and take full responsibility for the final content, including the accuracy of its claims, citations, and reported results. The use of an LLM teacher for pseudo-label generation is part of the proposed AccentMatch methodology and is documented in Section 3 and Appendices A and M.

ETHICS STATEMENT

Our work focuses on improving semi-supervised text classification through adaptive guidance from an LLM teacher. We conduct experiments using established benchmark datasets and pretrained models, without recruiting human participants or collecting new personal data. We will release our source code to support transparency, reproducibility, and responsible reuse.

REPRODUCIBILITY STATEMENT

We provide detailed information to support reproduction of our results. The AccentMatch procedure, including pseudo-margin estimation, adaptive teacher annealing, pseudo-label fusion, and the training objective, is described in Section 3 and Algorithm 1. The datasets, labeled-data sampling, evaluation metrics, student backbone, augmentation pipeline, training duration, and teacher prompting setup are described in Section 4.2 and Appendix A. Appendix M provides our prompt templates. The APM smoothing and warmup settings are specified in Section 4.2, with sensitivity analysis in Appendix E. We report means and standard deviations over three runs in Section 4 and provide additional ten-run evaluations with paired significance tests in Appendix H. Ablation studies and computational cost analyses appear in Sections 4 and 5, and Appendix B, respectively, while the proof of Proposition 3.1 is provided in Appendix L. We will release the source code upon acceptance of the manuscript to support transparency, reproducibility, and responsible reuse.

REFERENCES

Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. *CoRR*, abs/1908.02983, 2019. URL http://arxiv.org/abs/1908.02983.

Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020.

Nabiha Asghar. Yelp dataset challenge: Review rating prediction. *arXiv preprint arXiv:1605.05362*, 2016.

Peter L. Bartlett, Dylan J. Foster, and Matus Telgarsky. Spectrally-normalized margin bounds for neural networks. *CoRR*, abs/1706.08498, 2017. URL http://arxiv.org/abs/1706.08498.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. *arXiv preprint arXiv:2301.10921*, 2023.

Jichan Chung and Irene Y. Chen. Enhancing semi-supervised learning with zero-shot pseudolabels. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=WB05Doi29V.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. Goemotions: A dataset of fine-grained emotions. *arXiv preprint arXiv:2005.00547*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina 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, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. Understanding back-translation at scale. In *Conference of the Association for Computational Linguistics (ACL)*, 2018.

Gamaleldin Fathy Elsayed, Dilip Krishnan, Hossein Mobahi, Kevin Regan, and Samy Bengio. Large margin deep networks for classification. 2018. URL https://arxiv.org/pdf/1803.05598.pdf.

Kai Gan and Tong Wei. Erasing the bias: fine-tuning foundation models for semi-supervised learning. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

Marzi Heidari, Hanping Zhang, and Yuhong Guo. Reinforcement learning guided semi-supervised learning. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Zhe Huang, Xiaowei Yu, Dajiang Zhu, and Michael C. Hughes. Interlude: interactions between labeled and unlabeled data to enhance semi-supervised learning. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Zhuo Huang, Li Shen, Jun Yu, Bo Han, and Tongliang Liu. Flatmatch: Bridging labeled data and unlabeled data with cross-sharpness for semi-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Yiding Jiang, Dilip Krishnan, Hossein Mobahi, and Samy Bengio. Predicting the generalization gap in deep networks with margin distributions, 2018. URL https://arxiv.org/abs/1810.00113.

Oleksandr Kolomiyets, Steven Bethard, and Marie-Francine Moens. Model-portability experiments for textual temporal analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pp. 271–276, USA, 2011. Association for Computational Linguistics. ISBN 9781932432886.

Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, pp. 896, 2013.

David D Lewis, Yiming Yang, Tony Russell-Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397, 2004.

Xin Li and Dan Roth. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002. URL https://www.aclweb.org/anthology/C02-1150.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1015.

Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pp. 165–172, 2013.

Subhabrata Mukherjee and Ahmed Awadallah. Uncertainty-aware self-training for few-shot text classification. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21199–21212. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/f23d125da1e29e34c552f448610ff25f-Paper.pdf.

Seo Yeon Park and Cornelia Caragea. Verifymatch: A semi-supervised learning paradigm for natural language inference with confidence-aware mixup. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 19319–19335, 2024.

Geoff Pleiss, Tianyi Zhang, Ethan Elenberg, and Kilian Q Weinberger. Identifying mislabeled data using the area under the margin ranking. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 17044–17056. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/c6102b3727b2a7d8b1bb6981147081ef-Paper.pdf.

Zhengxiang Shi, Francesco Tonolini, Nikolaos Aletras, Emine Yilmaz, Gabriella Kazai, and Yunlong Jiao. Rethinking semi-supervised learning with language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 5614–5634, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.347. URL https://aclanthology.org/2023.findings-acl.347.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 596–608. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/06964dce9addb1c5cb5d6e3d9838f733-Paper.pdf.

Tiberiu Sosea and Cornelia Caragea. Marginmatch: Improving semi-supervised learning with pseudo-margins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15773–15782, 2023.

Jörg Tiedemann and Santhosh Thottingal. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal, 2020.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. SwitchOut: an efficient data augmentation algorithm for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 856–861, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1100. URL https://aclanthology.org/D18-1100.

Yidong Wang, Hao Chen, Yue Fan, Wang SUN, Ran Tao, Wenxin Hou, Renjie Wang, Linyi Yang, Zhi Zhou, Lan-Zhe Guo, Heli Qi, Zhen Wu, Yu-Feng Li, Satoshi Nakamura, Wei Ye, Marios Savvides, Bhiksha Raj, Takahiro Shinozaki, Bernt Schiele, Jindong Wang, Xing Xie, and Yue Zhang. USB: A unified semi-supervised learning benchmark for classification. In *Thirty-sixth*

Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2022. URL https://openreview.net/forum?id=QeuwINa96C.

Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Bernt Schiele, and Xing Xie. Freematch: Self-adaptive thresholding for semi-supervised learning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PDrUPTXJI_A.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6256–6268. Curran Associates, Inc., 2020a. URL https://proceedings.neurips.cc/paper/2020/file/44feb0096faa8326192570788b38c1d1-Paper.pdf.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10687–10698, 2020b.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Bowen Zhang, Yidong Wang, Wenxin Hou, HAO WU, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 18408–18419. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/995693c15f439e3d189b06e89d145dd5-Paper.pdf.

A Experimental Setup

Data and Evaluation. Our benchmarks cover sentiment classification (IMDB, Amazon Review, and Yelp Review), topic classification (RCV1), emotion detection (GoEmotions), and question classification (TREC-6). For each dataset, we sample 20 or 50 labeled examples per class without replacement from the training set and treat the remaining examples as unlabeled. We report accuracy for IMDB, TREC-6, Amazon Review, and Yelp Review, and macro-F1 for RCV1 and GoEmotions. Results are reported as means and standard deviations over three runs with different random initializations.

Training and Augmentation. We train BERT base uncased (Devlin et al., 2019) for 200 epochs. Weak augmentations randomly apply synonym replacement (Kolomiyets et al., 2011), SwitchOut (Wang et al., 2018), or both. Strong augmentations combine backtranslation with chains of length > 5, SwitchOut, and synonym replacement, using translation models from Tiedemann & Thottingal (2020). All baselines use the same augmentation pipeline as AccentMatch.

LLM Teacher. We use Qwen (Yang et al., 2025) with few-shot CoT prompting, including up to 5 examples per class drawn from the available labeled set. We reduce the number of demonstrations when necessary to fit the 4096-token limit.

B Cost-Benefit Analysis and Token Profiling

We evaluate the computational cost of AccentMatch compared to SoftMatch in the 20 labels per class setting, tracking GPU-hours, relative slowdown, and token usage. As shown in Table 5, computing and maintaining instance-level Average Pseudo-Margins adds minimal overhead during training, leading to an average relative slowdown of only 5.7% across benchmarks.

Dataset (Metric)	**SoftMatch** (F1/Acc)	**AccentMatch** (F1/Acc)	Δ (Gain)	**Total Added Cost** (GPU-hours)	**Relative Slowdown (%)**	**Gain per 100 Added GPU-hours**
IMDB (Acc)	85.5	87.7	+2.2	19	5.2%	11.6
RCV1 (F1)	59.3	63.2	+3.9	39	7.0%	10.0
GoEmotions (F1)	27.9	30.2	+2.3	34	6.1%	6.8
TREC-6 (Acc)	84.2	89.7	+5.5	4	4.5%	137.5
Amazon Review (Acc)	51.7	53.2	+1.5	59	5.5%	2.5
Yelp Review (Acc)	50.4	52.7	+2.3	67	5.8%	3.4
Average	59.8	62.8	+3.0	37.0	5.7%	28.6

Table 5: Cost-benefit analysis of AccentMatch vs. SoftMatch across all datasets, using results from the 20 labels/class setting.

To account for the offline teacher pseudo-labeling phase, Table 6 reports the average input and output token lengths per unlabeled sample across all datasets. Because classification targets require only short label generations, output lengths remain small (30 to 120 tokens), keeping decoding costs low. Input token counts vary with task text length, reaching approximately 4K tokens for long document benchmarks such as RCV1 and Yelp Review. Because this inference step is executed only once offline before SSL training begins, it avoids repeated forward passes through LLM during optimization.

Dataset	**Classes**	**Average Input Tokens**	**Average Output Tokens**
IMDB	2	3,805	120
RCV1	103	4,067	120
GoEmotions	27	3,848	30
TREC-6	6	753	30
Amazon Review	5	3,280	60
Yelp Review	5	4,061	75

Table 6: Token profiling for the offline LLM pseudo-labeling stage across benchmarks.

C GENERALIZABILITY TO HARD-THRESHOLDING METHODS

While our primary analysis focuses on integrating AccentMatch with SoftMatch, the core framework is versatile and can be readily adapted to hard-thresholding SSL methods like FixMatch and FlexMatch. The application is straightforward: we replace the original one-hot pseudo-label with the fused pseudo-label generated by our teacher annealing framework, while the native confidence thresholding mechanism of the underlying SSL method remains unchanged.

The results, presented in Table 7, demonstrate that our AccentMatch framework provides significant and consistent performance gains when applied to these hard-thresholding strategies. For example, on the IMDB dataset with only 20 labels per class, applying our framework with fixed thresholds boosts accuracy by a remarkable 3.0%. Similarly, utilizing our framework in combination with flexible thresholds on RCV1 improves F1 by 2.1% under the same challenging low-data setting. This pattern of improvement holds across the majority of datasets and label counts, confirming that the benefits of using LLM-informed pseudo-labels are not unique to SoftMatch, but represent a general and highly effective enhancement for consistency-based SSL.

Dataset (Metric)	IMDB (Accuracy)		RCV1 (F1)		GOEMOTIONS (F1)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
FlexMatch	$84.1_{\pm 3.96}$	$86.3_{\pm 3.52}$	$56.3_{\pm 4.56}$	$58.9_{\pm 4.21}$	$26.4_{\pm 4.61}$	$26.2_{\pm 4.51}$
AccentMatch (flexible thresholds)	$\mathbf{85.6_{\pm 2.64}}$	$\mathbf{87.5_{\pm 3.35}}$	$\mathbf{58.4_{\pm 1.54}}$	$\mathbf{59.5_{\pm 3.17}}$	$\mathbf{27.1_{\pm 3.46}}$	$\mathbf{26.9_{\pm 3.10}}$
FixMatch	$84.2_{\pm 2.61}$	$84.2_{\pm 2.32}$	$55.2_{\pm 3.51}$	$56.3_{\pm 3.25}$	$24.1_{\pm 4.63}$	$24.9_{\pm 4.21}$
AccentMatch (fixed thresholds)	$\mathbf{87.2_{\pm 1.74}}$	$\mathbf{86.3_{\pm 2.78}}$	$\mathbf{57.1_{\pm 2.54}}$	$\mathbf{57.5_{\pm 2.33}}$	$\mathbf{24.9_{\pm 2.41}}$	$\mathbf{25.3_{\pm 2.64}}$
Dataset (Metric)	TREC-6 (Accuracy)		AMAZON REVIEW (Accuracy)		YELP REVIEW (Accuracy)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
FlexMatch	$83.9_{\pm 4.22}$	$88.4_{\pm 3.58}$	$51.8_{\pm 3.42}$	$51.3_{\pm 2.45}$	$50.1_{\pm 1.45}$	$48.4_{\pm 1.53}$
AccentMatch (flexible thresholds)	$\mathbf{85.3_{\pm 1.63}}$	$\mathbf{89.3_{\pm 1.73}}$	$\mathbf{53.1_{\pm 2.98}}$	$\mathbf{52.3_{\pm 1.20}}$	$\mathbf{52.6_{\pm 2.31}}$	$\mathbf{49.3_{\pm 1.51}}$
FixMatch	$80.1_{\pm 1.98}$	$89.1_{\pm 1.91}$	$50.6_{\pm 2.35}$	$51.1_{\pm 2.22}$	$49.7_{\pm 1.26}$	$48.1_{\pm 1.48}$
AccentMatch (fixed thresholds)	$79.7_{\pm 1.32}$	$\mathbf{89.6_{\pm 1.22}}$	$\mathbf{52.1_{\pm 1.39}}$	$\mathbf{51.7_{\pm 1.74}}$	$\mathbf{51.9_{\pm 2.68}}$	$\mathbf{49.2_{\pm 1.19}}$

Table 7: Performance of the AccentMatch framework when utilizing flexible and fixed confidence thresholding mechanisms (derived from FlexMatch and FixMatch, respectively).

D SOFT LABELS

D.1 LLM SOFT LABELS

In this section, we investigate the impact of utilizing soft labels derived from LLM teacher, rather than the standard hard (one-hot) pseudo-labels employed in the main AccentMatch framework. To obtain these soft labels, we compute the probability distribution across the valid categorical class tokens generated by the LLM during inference. We then use this normalized token probability distribution as the target for the LLM teacher term in our dynamic fusion mechanism.

As detailed in Table 8, replacing hard labels with soft labels does not yield performance improvements. On average, the AccentMatch formulation utilizing soft LLM labels performs approximately 0.3% worse than the standard approach with hard labels across all evaluated benchmarks. We hypothesize that providing a soft probability distribution introduces unnecessary entropy into the target signal. Because the primary role of the LLM teacher in our framework is to provide a strong, confident prior during the volatile early stages of student training, the ambiguity introduced by token-level soft distributions slightly degrades the stabilizing effect of the teacher scaffolding.

D.2 SSL SOFT LABELS

In the standard formulation of AccentMatch, the student's prediction is sharpened into a hard, one-hot pseudo-label ($\hat{c} = \arg\max(p_\theta(y|\pi(\hat{x})))$) before being dynamically fused with the LLM's prior. In this section, we explore an alternative variant, AccentMatch (Soft SSL), where we bypass this sharpening step. Instead, we directly fuse the student's continuous probability distribution over all classes with the LLM teacher's label.

Dataset (Metric)	IMDB (Accuracy)		RCV1 (F1)		GoEmotions (F1)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
BASE	$69.1_{\pm 4.25}$	$78.4_{\pm 3.55}$	$22.1_{\pm 4.55}$	$24.19_{\pm 4.36}$	$09.2_{\pm 6.34}$	$20.4_{\pm 4.51}$
FewShot	85.8	85.8	58.6	58.6	26.2	26.2
AccentMatch	$\mathbf{87.7_{\pm 2.60}}$	$\mathbf{89.8_{\pm 2.93}}$	$\mathbf{63.2_{\pm 2.17}}$	$\mathbf{61.5_{\pm 2.54}}$	$\mathbf{30.2_{\pm 3.43}}$	$\mathbf{30.6_{\pm 2.27}}$
AccentMatch (Soft LLM)	$87.3_{\pm 2.61}$	$89.6_{\pm 2.85}$	$63.1_{\pm 2.14}$	$61.1_{\pm 2.49}$	$29.9_{\pm 3.35}$	$30.4_{\pm 2.18}$
Dataset (Metric)	TREC-6 (Accuracy)		Amazon Review (Accuracy)		Yelp Review (Accuracy)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
BASE	$80.4_{\pm 2.51}$	$85.1_{\pm 2.31}$	$47.6_{\pm 3.15}$	$48.4_{\pm 3.01}$	$38.5_{\pm 1.45}$	$46.2_{\pm 1.52}$
FewShot	83.8	83.8	50.4	50.4	49.7	49.7
AccentMatch	$\mathbf{89.7_{\pm 2.22}}$	$\mathbf{91.2_{\pm 2.23}}$	$\mathbf{53.2_{\pm 1.72}}$	$\mathbf{54.6_{\pm 1.87}}$	$\mathbf{52.7_{\pm 2.33}}$	$\mathbf{56.9_{\pm 2.34}}$
AccentMatch (Soft LLM)	$89.2_{\pm 2.18}$	$90.9_{\pm 2.28}$	$53.0_{\pm 1.75}$	$54.1_{\pm 1.92}$	$52.3_{\pm 2.28}$	$56.8_{\pm 2.41}$

Table 8: Comparison of model performance utilizing hard vs. soft LLM pseudo-labels. The soft label variant consistently underperforms the standard hard label formulation by a small margin across all datasets and configurations.

The results, presented in Table 9, show that utilizing soft labels for the student's predictions leads to a consistent degradation in performance. On average, the AccentMatch (Soft SSL) variant underperforms the standard hard-label formulation by 0.5% across the evaluated benchmarks. This performance drop aligns with standard findings in semi-supervised learning regarding entropy minimization (Grandvalet & Bengio, 2004). Soft pseudo-labels from the student inherently contain uncertainty and high entropy, particularly in the low-resource regimes we evaluate. By enforcing a hard argmax, the standard AccentMatch framework implicitly performs entropy minimization, forcing the student to make confident class boundaries. Removing this sharpening step causes the model to reinforce its own uncertainty, weakening the self-training signal and resulting in slightly lower overall accuracy and F1 scores.

Dataset (Metric)	IMDB (Accuracy)		RCV1 (F1)		GoEmotions (F1)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
BASE	$69.1_{\pm 4.25}$	$78.4_{\pm 3.55}$	$22.1_{\pm 4.55}$	$24.2_{\pm 4.36}$	$9.2_{\pm 6.34}$	$20.4_{\pm 4.51}$
FewShot	85.8	85.8	58.6	58.6	26.2	26.2
AccentMatch	$\mathbf{87.7_{\pm 2.60}}$	$\mathbf{89.8_{\pm 2.93}}$	$\mathbf{63.2_{\pm 2.17}}$	$\mathbf{61.5_{\pm 2.54}}$	$\mathbf{30.2_{\pm 3.43}}$	$\mathbf{30.6_{\pm 2.27}}$
AccentMatch (Soft SSL)	$87.1_{\pm 2.58}$	$89.4_{\pm 2.91}$	$62.9_{\pm 2.11}$	$60.9_{\pm 2.53}$	$29.7_{\pm 3.41}$	$30.2_{\pm 2.25}$
Dataset (Metric)	TREC-6 (Accuracy)		Amazon Review (Accuracy)		Yelp Review (Accuracy)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
BASE	$80.4_{\pm 2.51}$	$85.1_{\pm 2.31}$	$47.6_{\pm 3.15}$	$48.4_{\pm 3.01}$	$38.5_{\pm 1.45}$	$46.2_{\pm 1.52}$
FewShot	83.8	83.8	50.4	50.4	49.7	49.7
AccentMatch	$\mathbf{89.7_{\pm 2.22}}$	$\mathbf{91.2_{\pm 2.23}}$	$\mathbf{53.2_{\pm 1.72}}$	$\mathbf{54.6_{\pm 1.87}}$	$\mathbf{52.7_{\pm 2.33}}$	$\mathbf{56.9_{\pm 2.34}}$
AccentMatch (Soft SSL)	$89.0_{\pm 2.15}$	$90.8_{\pm 2.33}$	$52.8_{\pm 1.79}$	$54.0_{\pm 1.86}$	$52.2_{\pm 2.31}$	$56.6_{\pm 2.45}$

Table 9: Comparison of model performance utilizing hard vs. soft student pseudo-labels. Removing the *argmax* sharpening step results in a degradation of performance by an average of 0.5%, confirming the importance of implicit entropy minimization in the student's self-training signal.

E EMA Influence on Performance

In our main AccentMatch framework, we compute the Average Pseudo-Margin (APM) using an Exponential Moving Average (EMA) rather than a simple cumulative average. The EMA relies on a smoothing parameter β (defined in Eq. 3) to control the memory length of the student's learning status. To understand the influence of this parameter, we evaluate the performance of AccentMatch on the RCV1 dataset (using 50 labels per class) across various values of β, as well as against a baseline using a simple moving average.

The results, presented in Table 10, demonstrate the necessity of balancing historical stability with adaptability to recent training dynamics. Using a simple average yields a sub-optimal F1 score

of 59.1%, as it weights all historical iterations equally. Consequently, the highly unstable and erroneous pseudo-margins from the earliest epochs continuously drag down the student's stability score, preventing the model from properly annealing the teacher's guidance. Conversely, setting $\beta = 0.0$ forces the model to rely exclusively on the current iteration's pseudo-margin. This completely discards historical stability, creates extreme volatility in the stability score, and yields the lowest performance at 58.5%. As β increases, performance steadily improves, indicating that a long memory is highly beneficial for filtering out stochastic noise between epochs. The performance peaks at $\beta = 0.997$ with an F1 score of 61.5%. However, if β is increased further to 0.999, the model adapts too slowly to recent improvements in the student's predictions, causing a slight dip in performance to 61.2%. Based on these findings, we utilize $\beta = 0.997$ as the optimal default parameter across all our main experiments.

Averaging Strategy / Parameter	**RCV1 (50 lb/cl) F1**
Simple Average (Avg)	59.1
EMA ($\beta = 0.0$)	58.5
EMA ($\beta = 0.5$)	58.9
EMA ($\beta = 0.75$)	59.0
EMA ($\beta = 0.95$)	59.9
EMA ($\beta = 0.97$)	60.7
EMA ($\beta = 0.99$)	61.3
EMA ($\beta = 0.997$)	**61.5**
EMA ($\beta = 0.999$)	61.2

Table 10: Ablation of the EMA smoothing parameter β on the RCV1 dataset (50 labels per class). The optimal balance between stability and adaptability is achieved at $\beta = 0.997$.

F LLM ABSTENTION

In our standard framework, the LLM teacher provides a pseudo-label for every unlabeled example, regardless of its certainty. To investigate whether forcing the teacher to only provide highly confident guidance improves performance, we experiment with an LLM abstention strategy. In this setup, we explicitly prompt the LLM to output a confidence percentage alongside its class prediction. If this self-assessed generation confidence falls below a stringent 95% threshold, the teacher abstains. For these specific examples, AccentMatch falls back entirely to the student's self-training signal, effectively disabling the teacher term.

We evaluate this strategy on the RCV1 and GoEmotions datasets, with the results presented in Table 11. On the RCV1 dataset, the abstention strategy consistently degrades performance; most notably, in the 20 labels per class setting, the F1 score drops from 63.2% to 62.5%. This indicates that even when the LLM reports low confidence, its predictions still provide a valuable directional signal that prevents the student from reinforcing its own early errors. Conversely, on the GoEmotions dataset, which features 28 fine-grained and highly nuanced emotion classes, abstention provides a marginal benefit in the extreme low-data regimes (30.5% vs 30.2% for 20 labels, and 30.9% vs 30.6% for 50 labels). In such complex tasks, an incorrect guess by the LLM can occasionally mislead the student when labeled data is too scarce to correct it. Overall, these findings suggest that a hard abstention threshold based on LLM self-assessment is generally sub-optimal, as our dynamic stability weighting mechanism is already well-equipped to naturally modulate the influence of the teacher without explicitly discarding its predictions.

G TWO-STAGE LLM RELABELING

Our standard AccentMatch framework relies on calling the LLM once at the beginning of training to generate a static prior. However, to investigate whether the LLM's guidance can be further refined by the student's domain adaptation, we explore a two-stage relabeling strategy.

In this setup, we pause training once at the midpoint (i.e., Epoch 100). We extract the student model's highest-APM predictions on the unlabeled set and dynamically construct new few-shot

Dataset / Method	20 lb/cl	50 lb/cl
RCV1 (F1)		
AccentMatch	**63.2**	**61.5**
AccentMatch (Abstain)	62.5	60.9
GoEmotions (F1)		
AccentMatch	30.2	30.6
AccentMatch (Abstain)	**30.5**	**30.9**

Table 11: Performance comparison of the standard AccentMatch versus an LLM abstention variant. In the abstention setup, the LLM is prompted to generate its confidence, and abstains from providing a pseudo-label if this self-assessed score is $< 95\%$.

Chain-of-Thought (CoT) prompts using these examples. We then re-prompt the LLM a single time to generate a fresh set of pseudo-labels for the remaining uncertain examples before resuming the student's training. This approach creates a reciprocal teacher-student feedback loop: the LLM provides the initial generalist scaffolding, the student specializes to the domain-specific nuances, and these nuances are then used to ground the LLM for a more accurate secondary prior.

The results on the RCV1 and GoEmotions datasets, presented in Table 12, show that the two-stage relabeling strategy yields a slight average decrease of 0.4% compared to the standard single-stage approach. This mild degradation suggests that replacing the LLM's initial independent prior with a student-biased prior often risks introducing confirmation bias. The student's high-APM predictions, while confident, may encode domain-specific shortcuts that the LLM then echoes back, effectively collapsing the diversity of the teacher's external knowledge.

We observe that across the 20 and 50 label configurations, the two-stage relabeling strategy generally underperforms or only marginally matches the single-stage approach. This suggests that in extreme low-resource settings, the student lacks sufficient labeled data to form a robust representation by the midpoint, making interactive feedback loops less effective than the standard AccentMatch framework, which remains more robust and computationally efficient for general low-resource learning.

Dataset (Metric)	RCV1 (F1)		GoEmotions (F1)	
Num Labels	20 lb/cl	50 lb/cl	20 lb/cl	50 lb/cl
AccentMatch	$\mathbf{63.2_{\pm 2.17}}$	$\mathbf{61.5_{\pm 2.54}}$	$\mathbf{30.2_{\pm 3.43}}$	$\mathbf{30.6_{\pm 2.27}}$
Relabeling	$62.7_{\pm 2.22}$	$61.0_{\pm 2.49}$	$29.8_{\pm 3.38}$	$30.4_{\pm 2.31}$

Table 12: Comparison of the standard AccentMatch versus the Two-Stage Relabeling variant on RCV1 and GoEmotions. While the interactive feedback loop improves performance in specific configurations, it generally leads to a slight overall degradation due to confirmation bias in the low resource regime.

H Statistical Significance and 10-Run Robustness Analysis

To ensure the performance gains of AccentMatch are statistically robust across random initializations, we expanded our evaluation across all six benchmarks to 10 independent runs. Table 13 reports the mean and standard deviation over 10 runs for SoftMatch, VerifyMatch, and AccentMatch, along with paired t-test p-values computed against the stronger of the two baselines for each setting.

AccentMatch achieves statistically significant improvements ($p < 0.05$) in 10 out of the 12 evaluated configurations. The gains are especially pronounced in the extreme low-data regime of 20 labels per class, where $p < 0.01$ across nearly all datasets (e.g., $p = 0.002$ on RCV1, $p = 0.001$ on GoEmotions, and $p < 0.001$ on TREC-6). The only two configurations where the advantage is not statistically significant are RCV1 with 50 labels ($p = 0.412$ vs. VerifyMatch) and Amazon Review with 50 labels ($p = 0.184$ vs. SoftMatch).

Dataset (Metric)	Labels/Cl	SoftMatch (10 runs)	VerifyMatch (10 runs)	AccentMatch (10 runs)	p-value	Significant? ($p < 0.05$)
IMDB (Acc.)	20	$85.3_{\pm 2.24}$	$86.0_{\pm 2.15}$	$\mathbf{87.8}_{\pm \mathbf{2.42}}$	$p = 0.012$	Yes
	50	$87.9_{\pm 2.08}$	$88.1_{\pm 2.10}$	$\mathbf{89.9}_{\pm \mathbf{2.31}}$	$p = 0.014$	Yes
RCV1 (F1)	20	$59.1_{\pm 3.10}$	$60.5_{\pm 2.28}$	$\mathbf{63.4}_{\pm \mathbf{2.12}}$	$p = 0.002$	Yes
	50	$60.1_{\pm 3.18}$	$62.2_{\pm 1.91}$	$\mathbf{62.3}_{\pm \mathbf{1.85}}$	$p = 0.412$	No
GoEmotions (F1)	20	$28.0_{\pm 2.32}$	$27.1_{\pm 2.10}$	$\mathbf{30.3}_{\pm \mathbf{2.18}}$	$p = 0.001$	Yes
	50	$27.9_{\pm 1.84}$	$26.5_{\pm 1.95}$	$\mathbf{30.8}_{\pm \mathbf{1.76}}$	$p = 0.001$	Yes
TREC-6 (Acc.)	20	$84.0_{\pm 2.15}$	$87.9_{\pm 2.18}$	$\mathbf{89.6}_{\pm \mathbf{2.05}}$	$p < 0.001$	Yes
	50	$89.9_{\pm 1.95}$	$90.1_{\pm 1.82}$	$\mathbf{91.4}_{\pm \mathbf{1.98}}$	$p = 0.043$	Yes
Amazon (Acc.)	20	$51.9_{\pm 1.55}$	$51.4_{\pm 1.70}$	$\mathbf{53.3}_{\pm \mathbf{1.61}}$	$p = 0.008$	Yes
	50	$53.8_{\pm 1.68}$	$52.3_{\pm 1.65}$	$\mathbf{54.5}_{\pm \mathbf{1.72}}$	$p = 0.184$	No
Yelp (Acc.)	20	$50.3_{\pm 1.98}$	$51.3_{\pm 2.02}$	$\mathbf{52.8}_{\pm \mathbf{2.10}}$	$p = 0.003$	Yes
	50	$55.8_{\pm 2.08}$	$54.9_{\pm 1.55}$	$\mathbf{57.0}_{\pm \mathbf{2.15}}$	$p = 0.041$	Yes

Table 13: Model performance averaged over 10 independent runs across all benchmarks. Paired t-test p-values are calculated between AccentMatch and the highest-performing baseline for each setting.

I Cross-Domain Transfer and Distribution Shift

To test the robustness of AccentMatch under distribution shift, we evaluate cross-domain transfer between the Amazon Review and Yelp Review benchmarks. Both datasets share a 5-class sentiment classification target, but feature distinct review vocabularies, stylistic conventions, and topic distributions. We take the student model trained with AccentMatch on the source domain and evaluate it directly on the target domain test set without fine-tuning or retraining.

Table 14 displays the out-of-domain transfer performance alongside the in-domain reference scores from the main paper. Even under severe label scarcity, the student models retain substantial sentiment categorization capability across domains. For instance, an Amazon-trained student achieves 41.2% accuracy on Yelp at 50 labels per class, while a Yelp-trained student reaches 40.4% accuracy on Amazon. This indicates that competence-guided teacher annealing regularizes the student toward generalizable semantic features rather than domain-specific artifacts.

Source Domain	Target Domain	20 Labels/Class	50 Labels/Class
Amazon Review	Amazon Review (In-Domain)	53.2	54.6
Amazon Review	Yelp Review (Cross-Domain)	35.4	41.2
Yelp Review	Yelp Review (In-Domain)	52.7	56.9
Yelp Review	Amazon Review (Cross-Domain)	36.4	40.4

Table 14: Cross-domain transfer performance between Amazon Review and Yelp Review compared against in-domain reference results.

J Prompt Sensitivity and Few-Shot Depth

Because AccentMatch uses an LLM teacher to provide early pseudo-labels, we analyze the sensitivity of the teacher's supervision to the number of few-shot demonstrations provided in the prompt. Table 15 compares the teacher's accuracy and F1 score across zero-shot, 1-shot, 5-shot (our default setup in the main paper), and 10-shot demonstration depths across all benchmarks in the 20 labels per class regime.

Performance gains are most visible when transitioning from zero-shot to 1-shot prompting (e.g., $+5.0\%$ on TREC-6 and $+1.7\%$ on IMDB). Beyond 1-shot, performance remains stable: the gap between 5-shot and 10-shot prompting is negligible across all benchmarks (e.g., 85.8 vs. 86.0 on IMDB and 58.6 vs. 58.4 on RCV1). This confirms that the teacher guidance is not overly sensitive to prompt engineering or demonstration depth, and that standard few-shot prompting is sufficient to provide a reliable prior.

Few-Shot Depth	IMDB (Acc)	RCV1 (F1)	GoEmotions (F1)	TREC-6 (Acc)	Amazon (Acc)	Yelp (Acc)
Zero-Shot	82.9	57.8	24.6	78.1	48.1	47.2
1-Shot	84.6	58.1	25.7	83.1	49.6	49.1
5-Shot (Default)	85.8	58.6	26.2	83.8	50.4	49.7
10-Shot	86.0	58.4	26.1	83.5	50.8	49.5

Table 15: LLM teacher performance across demonstration depths with 20 labels per class.

K Analysis of Prediction Error vs. Average Pseudo-Margin (APM)

In Section 5 and Figure 2b, we evaluated model performance across bins of the normalized stability score $S = \sigma(\text{APM})$. To inspect the underlying training dynamics directly on the unbounded logit margin scale, we evaluate the pseudo-label error rates of both the student (e_S) and the LLM teacher (e_T) across bins of the raw Average Pseudo-Margin (APM).

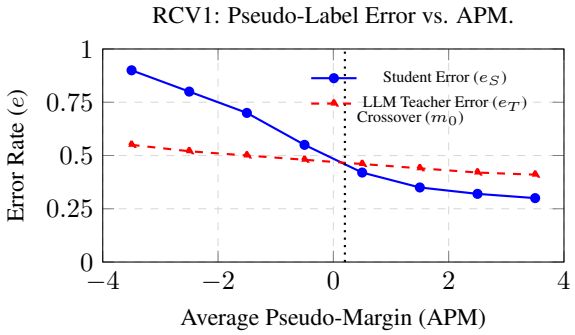


Figure 3: Pseudo-label error rates of the student (e_S) and LLM teacher (e_T) binned by Average Pseudo-Margin on RCV1 (20 labels/class). Student error decreases monotonically with higher APM, crossing below the teacher's error rate once predictions stabilize.

As shown in Figure 3, the student's error rate decreases monotonically as APM increases. In the low-margin and negative-margin setting (APM < 0), the student is unstable and exhibits a high error rate ($e_S > 0.6$), whereas the static LLM teacher maintains a substantially lower error rate. In this phase, assigning higher supervisory authority to the teacher prevents the student from compounding early errors.

As the student stabilizes over training iterations and its APM increases into positive values, its error rate drops below that of the static LLM teacher. This crossover demonstrates that the student successfully adapts to domain-specific features, eventually outperforming the generalist teacher on stable instances and justifying the smooth transfer of authority governed by $S = \sigma(\text{APM})$.

L Theoretical Motivation

Here we show the proof of Proposition 3.1.

Proof. For any fixed x, the conditional risk can be rewritten as

$$R_\alpha(x) = e_T(x) + \alpha(x)\big(e_S(x) - e_T(x)\big).$$

Thus, $R_\alpha(x)$ is linear in $\alpha(x)$. If $e_S(x) \le e_T(x)$, the minimum is attained at $\alpha(x) = 1$, corresponding to using the student. If $e_T(x) \le e_S(x)$, the minimum is attained at $\alpha(x) = 0$, corresponding to using the teacher. By the crossover assumption, these two regimes are separated by whether $m(x) \ge m_0$. Therefore, the optimal gate is $\alpha^\star(x) = \mathbf{1}\{m(x) \ge m_0\}$, and the resulting risk is $\min\{e_S(x), e_T(x)\}$. Taking expectation over x gives the desired result. □

M PROMPT DESIGN

To ensure reproducible pseudo-label generation across diverse task formats while restricting teacher token output, we use a structured, few-shot prompt template. The candidate class list is fixed upfront, and the model is instructed to output a short rationale followed by a deterministic terminal line.

```
System: Classify the input text into exactly one of the
candidate classes.
Candidate Classes: [CLASS_LIST]
Constraint: Provide a short rationale, then output the
final label on a new line formatted strictly as "Label:
<class>".

--
Example 1
Input: "[EXAMPLE_TEXT_1]"
Reasoning: [EXAMPLE_REASONING_1]
Label: [EXAMPLE_LABEL_1]
--

Input: "[UNLABELED_TEXT]"
Reasoning:
```