Early Stopping Based on Unlabeled Samples in Text Classification

Paper Authors: HongSeok Choi, Dongha Choi, Hyunju Lee
Presenters: Swapnali Kadam, Yash Varpe
Paper Link

Source: AI Generated

Problem Statement

Key challenge: **When to stop training** the model. If training continues too long, the model **overfits**; if stopped too early, model **underfits** and performance drops.

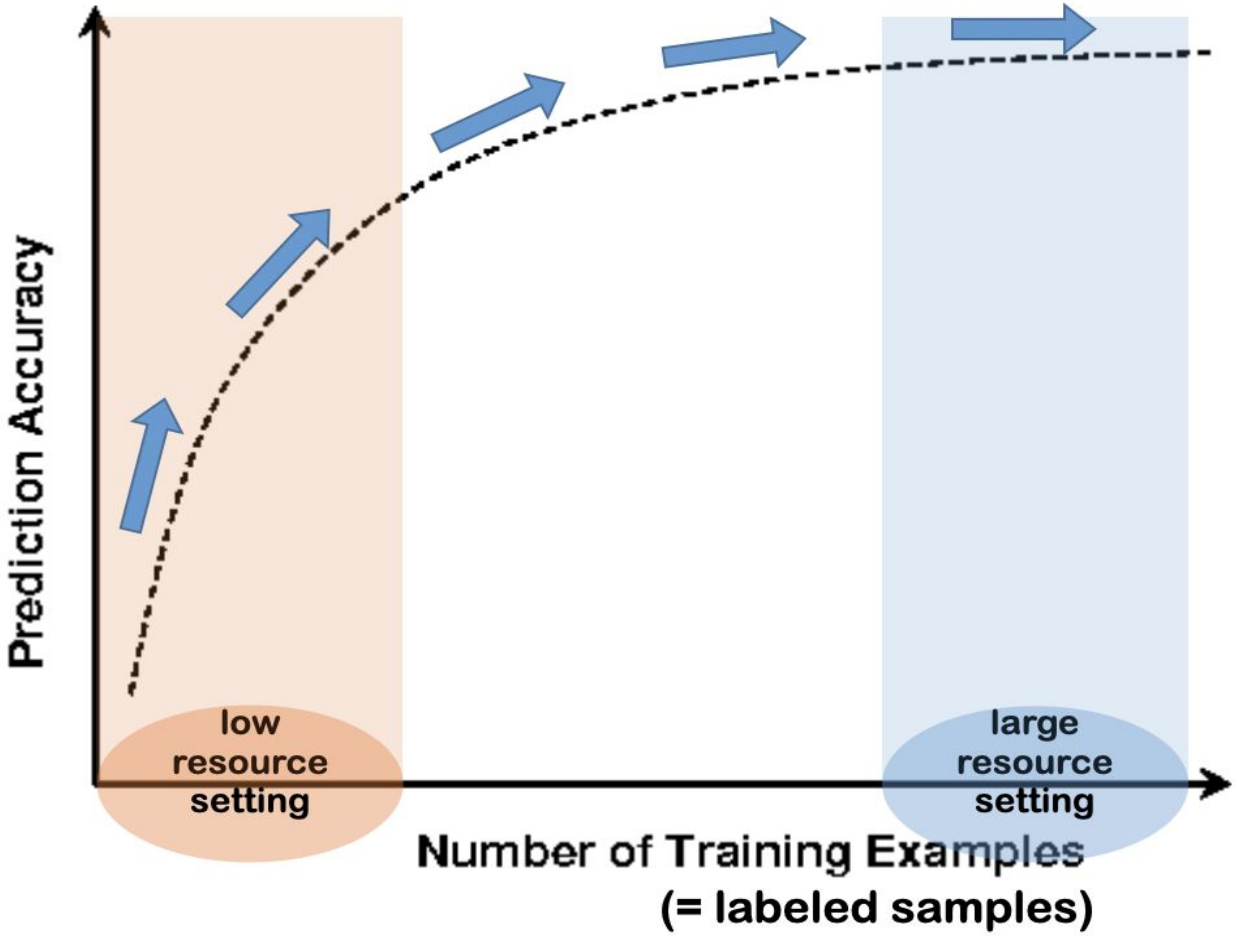
Traditional Early Stopping (Validation Based) Challenge

The Low-Resource Dilemma

Real-World Context: Labeling data is expensive and time-consuming

The core problem addressed is: **Can we perform early stopping *without* a labelled validation set by using unlabelled samples instead?**

Source- https://github.com/DMCB-GIST/BUS-stop/blob/main/ppts/bus_stop_ppt.pptx

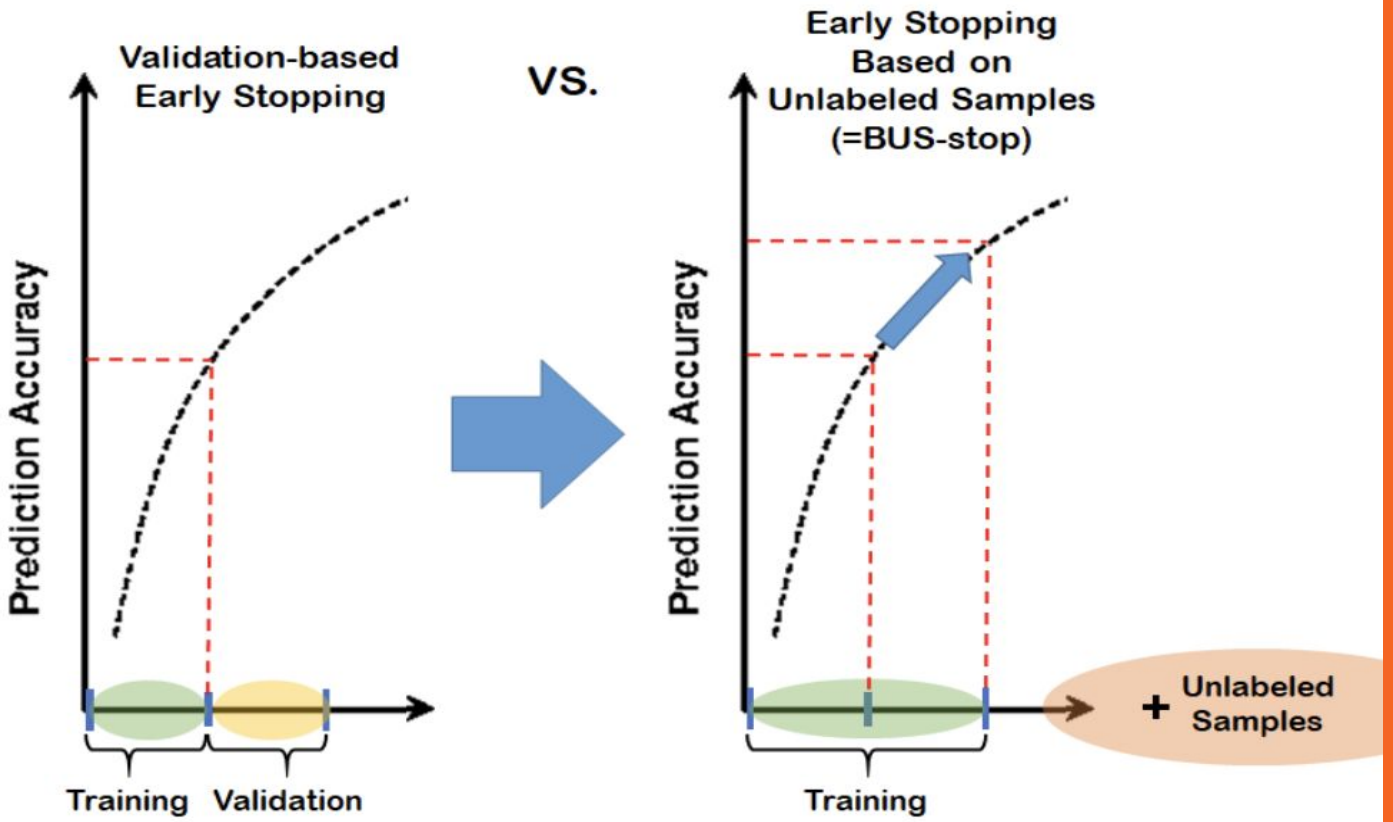


The Ice Cream Taster

Labeled scoops->
labeled dataset
Mystery scoops->
unlabeled dataset
Overconfident tasting->
overfitting

Source: AI Generated

Source:https://github.com/DMCB-GIST/BUS-stop/blob/main/ppts/bus_stop_ppt.pptx



Gap in Literature

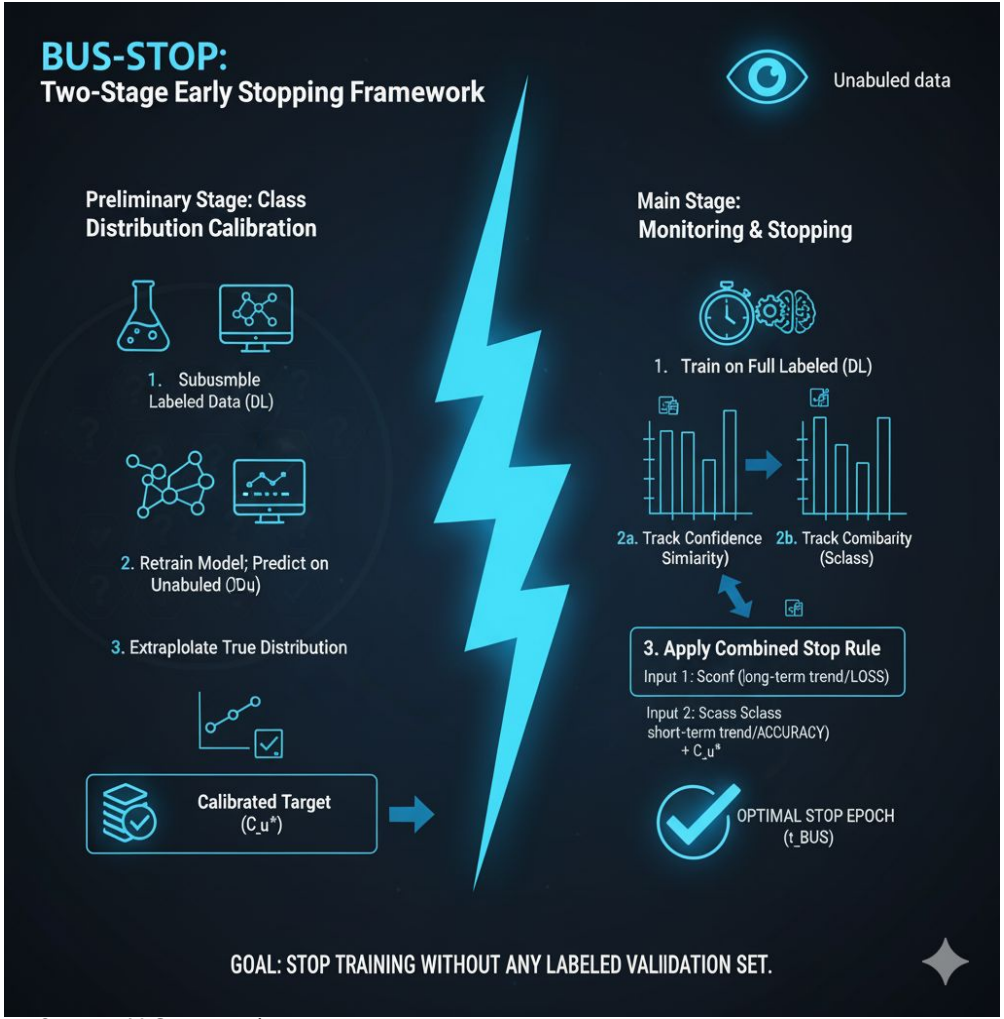
Early stopping is a common technique in machine learning to prevent **overfitting**, where the model memorizes training data instead of learning general patterns.

- **Static heuristics:** Not adaptive, ignores task dynamics.
- **Gradient-based stopping methods:** Indirect, noisy, and dataset/model dependent.
- **Intrinsic measures (like LID-Local Intrinsic Dimensionality)**

There is a need for a method that:

- Uses **all labeled data** for training
- **Detects overfitting** reliably
- Works even when l**abeled data is scarce**

Source: AI Generated



Source: AI Generated

Proposed System

BUS-Stop has **two major stages**:

1. **Preliminary Stage → Class Distribution Calibration**
   (Estimate the *true* class distribution from unlabeled data)

2. **Main Stage → Confidence + Class-Similarity Monitoring**
   (Track signals on unlabeled data to decide when to stop training)

The goal: **Stop training without any labeled validation set.**

I.Preliminary Stage: Class Distribution Calibration

This stage exists because:

- **Labeled data is tiny, often biased, and not representative.**
- **Unlabeled data is huge and mirrors real class proportions, but we don't know the labels.**

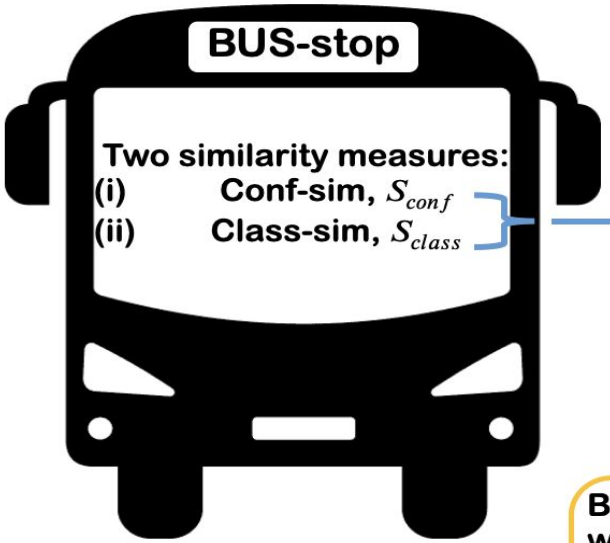
BUS-Stop wants to estimate:

The "true" class distribution of the unlabeled population → $\vec{C}_u^*$

This becomes the target during early stopping.

Source:https://github.com/DMCB-GIST/BUS-stop/blob/main/ppts/bus_stop_ppt.pptx

Overview



$$S_{conf} = sim_1(\vec{P}_l, \ddot{P}_u^e)$$
$$S_{class} = sim_2(\vec{C}_{cali}, \hat{C}_u^e)$$

Main stage

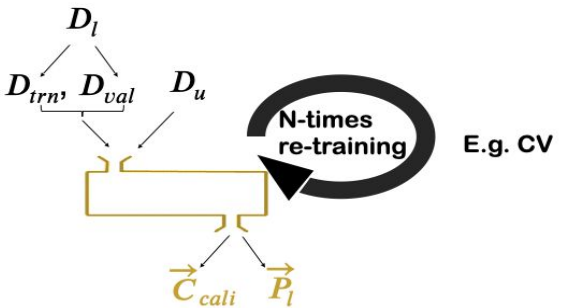
For epoch e in $\{1,2,3,\ldots\}$:
Train the model M one epoch on D_l.
Feed D_u into M and obtains
the confidences, $\ddot{P}_u^e$, and
the output class distribution, $\hat{C}_u^e$.

$$S_{conf} = sim_1\left(\vec{P}_l, \ddot{P}_u^e\right)$$
$$S_{class} = sim_2\left(\vec{C}_{cali}, \hat{C}_u^e\right)$$

BUS-stop(S_{conf}, S_{class}) → stop?
if stop==yes → $M_{best} = M$
else stop==no → continue training

Preliminary stage

Before main stage,
we need the following two:
$\vec{P}_l =$ pre-calculated confidences on D_l
$\vec{C}_{cali} =$ pre-estimated class distribution for D_u



Steps

- ➔ Subsampling the Labeled Set & Multiple Mini-Trainings
- ➔ Predict on the Entire Unlabeled Dataset
- ➔ Discover & Model the Bias Relationship
- ➔ Extrapolation to Estimate the “Ideal” Target Distribution

—

Step 1: Subsampling the Labeled Set & Multiple Mini-Trainings

Goal: understand how biased labeled sets distort predictions.
Actions:

1. Split labeled data Dl into **K random subsets**, each with a different (possibly skewed) class distribution e.g., K = 20
2. For each subset:
   - Fine-tune the pre-trained model (e.g., BERT)
   - Only for a **small number of epochs** (e.g., 3–5)
   - Repeat over **T iterations** to stabilize noise.

   This gives you many "experimental runs" showing how initial class imbalance affects predictions.

Step 2: Predict on the Entire Unlabeled Dataset

After each mini-training:

- Use the model to classify **all** unlabeled samples in Du
- Compute predicted class proportions: Cu,k
- This gives the *output* distributions associated with each *input* distribution : Cl,k

—

Step 3: Discover & Model the Bias Relationship

There is a linear relationship between the skew in the labeled subset and the predicted skew on unlabeled data.

Why linear?

Because small labeled sets consistently distort predictions in a predictable, linear way.

This is a **major contribution** of the paper.

Step 4: Extrapolation to Estimate the “Ideal” Target Distribution

What would the predicted distribution look like if the labeled data were perfectly balanced?

Cu* = The estimated true class distribution of unlabeled data

This becomes the anchor for the class-similarity score later.

II.Main Stage — Monitoring Unlabeled Patterns During Training

Now, train normally on all labeled data: Dl
But at every epoch, use the unlabeled dataset to collect two stability signals:

Signal 1 — Confidence Similarity: Sconf
Signal 2 — Class Distribution Similarity: Sclass

Why Two Signals?
Because:

- Sconf = stable, slow, smooth (like validation loss)
- Sclass = sensitive, local, high-frequency (like accuracy)
  Combining both gives the benefits of loss + accuracy without labels.

—

Signal 1 — Confidence Similarity Sconf

Tracks long-term generalization.
When models start overfitting:

- They become **over-confident too quickly**
- Their confidence distributions collapse to one class

Calculation:
Distance between:

- Current unlabeled confidence vector Pu,t
- Reference initial confidence vector Pu,init

Using Euclidean distance: $S_{\text{conf}}(t) = \|P_{u,t} - P_{u,init}\|_2$
Interpretation:

- This curve generally decreases → lowest point = optimal generalization
- After that, overconfidence increases = overfitting

So:
tconf= epoch where Sconf hits its minimum .
This forms the **center** of the BUS window.

—

Signal 2 — Class Distribution Similarity Sclass

Tracks short-term accuracy trends.

Compares:

- Current prediction distribution Cu,t
- Target calibrated distribution Cu*

Using **cosine similarity**:

$$S_{\text{class}}(t) = \cos(C_{u,t}, C_u^*)$$

Interpretation:

- If model predictions resemble the “true” class distribution → good accuracy
- If predictions collapse (e.g., predict everything as positive) → similarity drops

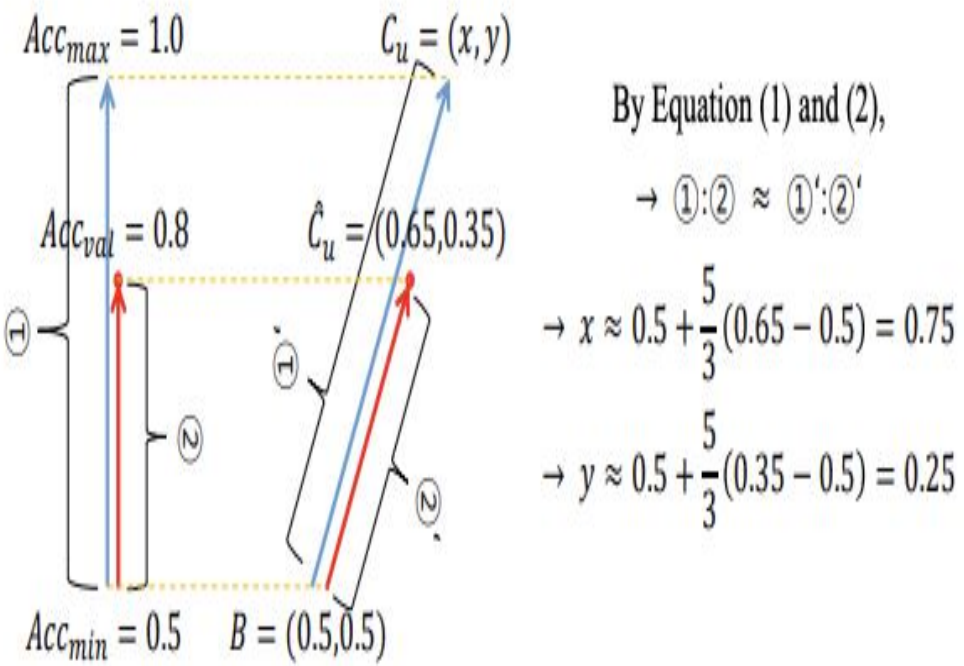


Fig. Confidence Similarity and Class Distribution Similarity Curves Over Training Epochs

The figure displays **two curves over training epochs**:

1. **Confidence similarity curve**
   - Starts high
   - Gradually decreases
   - Reaches a minimum → represents **the best generalization point**
   - After that, confidence sharply increases again ▫ **overfitting begins**
2. **Class distribution similarity curve**
   - Shows more local fluctuations
   - Peaks slightly after the S_conf minimum
   - That peak gives the **final BUS stopping epoch**

Combined BUS Stopping Rule

- **Find the confidence minimum**
  This gives: tconft

- **Define a queue window**
  Look at the next: n_{que} epochs
  This allows the peak accuracy to emerge slightly *after* the minimal loss point.

- **Choose the best class-similarity peak**
  Within the window:
  That is the final stopping epoch.

$$t_{\mathrm{BUS}} = \arg \max_{t \in [t_{\mathrm{conf}},\, t_{\mathrm{conf}} + n_{que}]} S_{\mathrm{class}}(t)$$

Save the Final Model

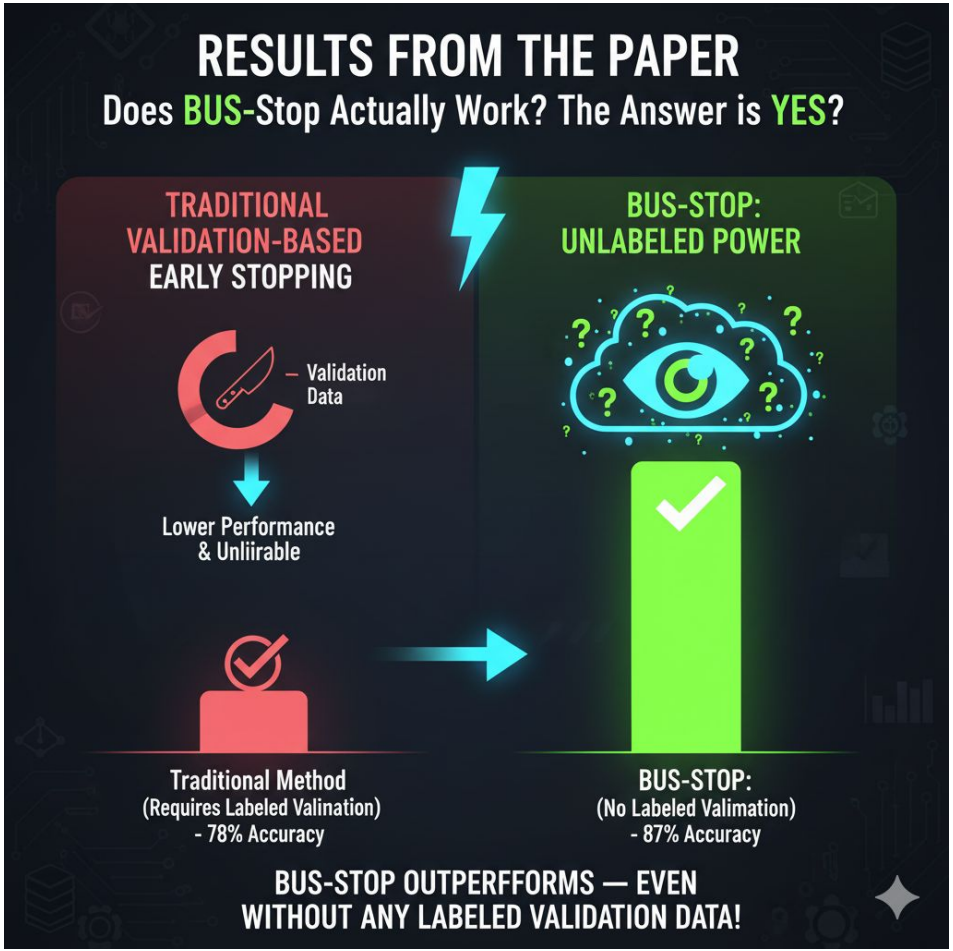
The model checkpoint at: tBUS is selected as the **BUS-Stop model**.

This ensures:

- Maximum labeled data used
- No dedicated validation set wasted
- Overfitting avoided using unlabeled stability signals
- Bias corrected using calibration stage

How Bus Stop Addresses The Gap

- ➔ **Eliminates the need for labeled validation data** in low-resource NLP
- ➔ **Introduces new stability-based stopping criteria** using hidden representations
- ➔ **Works for modern deep transformer models** and multiple learning paradigms
- ➔ **Ensures reliable stopping** even when training loss is misleading
- ➔ **Solves the fundamental trade-off** between validation size and training size



Results From the Paper

“Does BUS-Stop actually work?”

The paper evaluates BUS-Stop across **multiple models**, **datasets**, and **learning paradigms**.

The key message:
BUS-Stop outperforms or matches validation-based early stopping — even without using any labeled validation data.

1. Datasets Used for the experiment

The author conducted extensive experiments using five text classification datasets. SST-2 and IMDB include movie reviews, Elec includes reviews on Amazon electronics. AG-news and DBpedia are topic classification tasks for Wikipedia and news articles resp.

Data	Class	Train	Test	Len
SST-2	2	6.9K	1.8K	19
IMDB	2	25K	25K	231
Elec	2	25K	25K	107
AG-news	4	120K	7.6K	38
DBpedia	14	560K	70K	49

Table 1: Statistics for datasets. **Len** denotes the average number of words per sample.

2. Performance comparison of different stop-criteria in balanced classification

We used 50 labeled samples per class for all stop-criteria except for Val-stop*add*(25). *
Note that the Val-stop*add*(25) has an unfair advantage:for each class, it used 25 additional labeled samples for validation while using 50 labeled samples for training.
The best performances, except for the Val-stop*add*(25), are denoted in bold. '$^{\simeq}$' denotes that the performance is statistically similar to the BUS-stop

Dataset	SST-2		IMDB		Elec		AG-news		DBpedia		Average	
Method	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
Val-stop$_{split(25)}$	0.775	0.516	0.746	0.572	0.781	0.507	0.846	0.477	0.982	0.085	0.826	0.431
EB	0.826$^{\simeq}$	0.565	**0.833**$^{\simeq}$	0.551	0.843$^{\simeq}$	0.534	0.861	0.491	0.986$^{\simeq}$	0.103	0.869	0.449
LID	0.794	0.602	0.761	0.571	0.815	0.494	0.859	0.515	0.971	0.765	0.840	0.589
PE-stop-epoch	0.816	0.628	0.826$^{\simeq}$	0.585	0.837	0.524	0.859	0.487	0.985	0.079	0.865	0.460
Conf-sim (ours)	0.807	**0.442**$^{\simeq}$	0.793	0.484$^{\simeq}$	0.823	0.433$^{\simeq}$	0.863$^{\simeq}$	**0.421**	0.985$^{\simeq}$	0.077$^{\simeq}$	0.854	0.371
Class-sim (ours)	0.795	0.570	0.789	0.560	0.793	0.531	0.857	0.561	**0.986**$^{\simeq}$	0.078	0.844	0.460
BUS-stop (ours)	**0.831**	0.455	0.828	**0.456**	**0.848**	**0.417**	**0.865**	0.432	**0.986**	**0.074**	**0.872**	**0.367**
*Val-stop$_{add(25)}$	0.819	0.431	0.824$^{\simeq}$	0.447$^{\simeq}$	0.842$^{\simeq}$	0.407$^{\simeq}$	0.867	0.415	0.986$^{\simeq}$	0.075$^{\simeq}$	0.868	0.355

3. Performance comparison of different stop-criteria in imbalanced classification

Performance comparison in an imbalanced setting of binary classification tasks.

We used 50 labeled samples per class for training (i.e., K=50), and the class distributions of the test sets were adjusted to 2:8 (negative:positive). '$^{\simeq}$' denotes that the performance is statistically similar to the BUS-stop

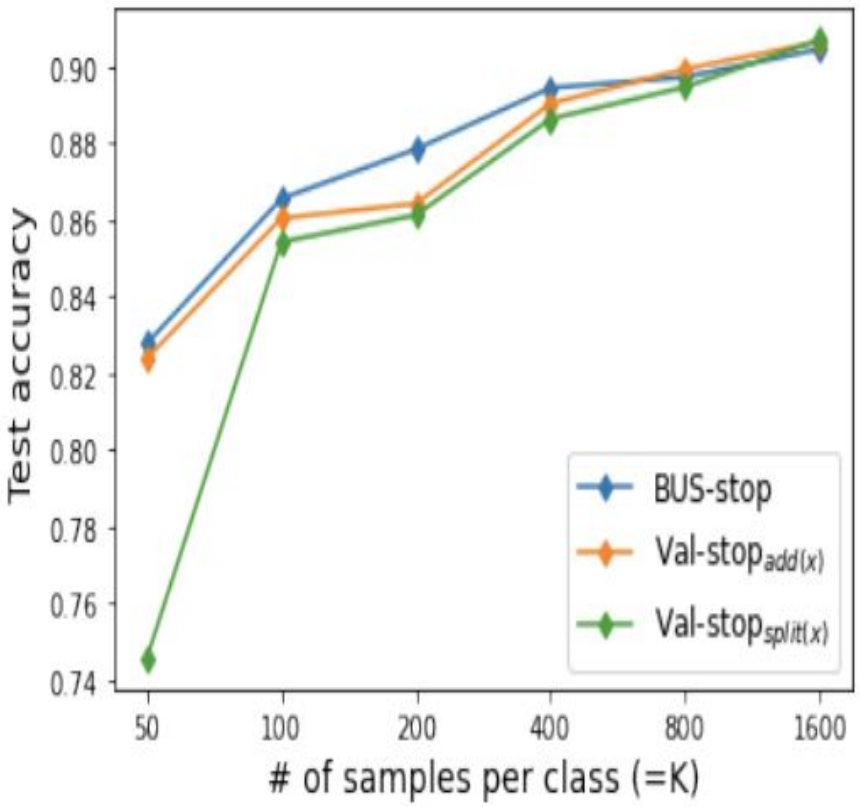
Dataset	SST-2			IMDB			Elec			Average		
Method	Acc	F1	Loss	Acc	F1	Loss	Acc	F1	Loss	Acc	F1	Loss
Val-stop$_{split(25)}$	0.788	0.719	0.499	0.732	0.674	0.589	0.783	0.724	0.507	0.768	0.706	0.532
EB	0.846$^{\simeq}$	0.786$^{\simeq}$	0.504	0.810	0.749	0.568	0.839	0.789	0.541	0.832	0.775	0.537
LID	0.750	0.698	0.632	0.712	0.668	0.678	0.780	0.728	0.574	0.747	0.698	0.628
PE-stop-epoch	0.843	0.779	0.527	0.821	0.763	0.589	0.843	0.789	0.521	0.836	0.777	0.545
Conf-sim (ours)	0.816	0.754	0.427	0.813	0.750	0.432$^{\simeq}$	0.835	0.775	0.398	0.821	0.760	0.419
Class-sim (ours)	**0.862**$^{\simeq}$	**0.797**$^{\simeq}$	0.489	0.844$^{\simeq}$	0.779$^{\simeq}$	0.510	0.873$^{\simeq}$	0.807$^{\simeq}$	0.409	0.860	0.794	0.469
BUS-stop (ours)	0.860	0.792	**0.379**	**0.849**	**0.787**	**0.406**	**0.876**	**0.815**	**0.343**	**0.861**	**0.798**	**0.376**
Val-stop$_{add(25)}$	0.823	0.767	0.412	0.820	0.767	0.457	0.837	0.784	0.407	0.827	0.773	0.426

4. Results - Accuracy comparison in various imbalance settings

In SST-2 dataset

Train	Test	2:8	4:6	6:4	8:2
2:8	EB	**0.845**	**0.732**	0.643	0.511
	BUS-stop (ours)	0.828	0.719	**0.669**	0.521
	Val-stop$_{add(25)}$	0.679	0.660	0.621	**0.634**
4:6	EB	0.860	0.820	0.790	0.728
	BUS-stop (ours)	**0.864**	**0.825**	**0.815**	**0.808**
	Val-stop$_{add(25)}$	0.820	0.808	0.801	0.794
6:4	EB	0.790	0.816	0.825	0.845
	BUS-stop (ours)	**0.845**	**0.826**	**0.833**	**0.864**
	Val-stop$_{add(25)}$	0.826	0.824	0.823	0.824
8:2	EB	0.611	0.696	0.774	**0.870**
	BUS-stop (ours)	**0.682**	**0.714**	**0.793**	0.865
	Val-stop$_{add(25)}$	0.667	0.707	0.733	0.782

Avg.: EB=0.760, BUS-stop=**0.779**, Val-stop$_{add(25)}$=0.750



What this graph proves:

- As the number of labeled samples increases, BUS-Stop consistently performs as well as or better than validation-based stopping.
- The improvement is **largest when labels are extremely limited** (50–200 samples per class).
- Validation-based stopping (green line, split(x)) is **unstable in low-resource settings** because the validation set is too tiny.
- BUS-Stop (blue) is **more stable** and **uses all labeled data**, giving higher accuracy.

6. BUS-Stop Equals Full-Validation Performance — But With Zero Validation Labels

The surprising result:

> **BUS-Stop achieves accuracy close to using a large validation set, even though BUS-Stop uses no labeled validation data.**

Example from SST-2 (numbers from the paper):

Method	Validation Size	Test Accuracy
Validation-based ES	500 labeled val samples	~89.8%
BUS-Stop	0 labeled val samples	**89–90%**

Interpretation:

Even with *zero* validation labels, BUS-Stop is as good as methods using hundreds of validation labels.

7. Each Individual Criterion Helps, But Their Combination Performs Best

The ablation study shows:

Criterion	Effect
S_class	Helps identify when representation become class-seperable
S_conf	Captures overconfidence/ underconfidence trends

Interpretation:
This shows that early stopping in low-resource NLP benefits from **multi-perspective stability**, not just accuracy curves.

Hypothesis

Since the paper proved that BUS-stop is better than validation-split, we propose a hypothesis that explores a new boundary or internal mechanism of the system.

- ➔ **Challenging the Domain Constraint: Is BUS-stop Universally Valid?**
- ➔ **Dissecting the Mechanism: Is the Dual-Criterion Stop Necessary?**

Challenging the Domain Constraint: Is BUS-stop Universally Valid?

Primary Hypothesis

We hypothesize that the core principle of BUS-stop—that **unlabeled prediction dynamics are a stable and superior proxy for generalization**—is **domain-agnostic**. The method can be effectively generalized to **low-resource vision tasks** (e.g., image classification) to achieve results superior to validation-split methods.

* **Claim:** BUS-stop → Domain Agnostic

Secondary Hypothesis

We further assert that the **Class-sim criterion** will prove essential for BUS-stop's success in vision tasks, demonstrating that **distribution stability is crucial** regardless of the data type (text or image)

* **Claim:** Sclass must remain essential in the new domain.

Experimental Design

Component	Setting	Justification
Task	Low-Resource Image Classification	Testing domain-agnostic nature.
Model	Pre-trained **ResNet-18**	Standard vision architecture for fine-tuning.
Dataset	CIFAR-10 Subset (3 classes)	Small, controlled dataset for low-resource focus.
Constraint	**K = 50 images per class** for D_l.	Maintains the paper's extreme scarcity condition.
Ablation Test	Compare: 1) Full BUS-stop vs. 2) $Val - stop_{split}$ vs. 3) S_{class}-only stop.	Isolates the essential criteria.

What are we trying to prove or expect from the experiments

1. **Proof of Domain-Agnostic Superiority**

Expectation: BUS-stop will yield a **significantly higher Top-1 Accuracy** (e.g., 5% gain) compared to the Val-stop_split baseline in the vision task.

Proof Point: This would prove the principle holds across domains: training with 100% Dl and monitoring with a large, stable Du is a **fundamental superiority in data-efficient learning**, not an NLP specialization

2. Proof of Robustness

Expectation: The **Coefficient of Variation (CV)** in the final accuracy of BUS-stop will be **lower** than the Val-stop_split method.

Proof Point: This shows that the unlabeled signal is a more **consistent and reliable** estimator of generalization than a small, noisy validation set, providing a more predictable result in resource-constrained environments.

3.Validation of the Sclass Signal

Expectation: The **Sclass only stop** (stopping at peak distribution similarity) will perform **statistically similarly** to the full BUS-stop method.

Proof Point: This demonstrates that the Sclass criterion is the dominant and necessary component, while Sconf primarily serves to narrow the search window, simplifying the underlying core mechanism.

Why will it work

- ➔ Stable Monitoring Signal
- ➔ Maximal Training Data
- ➔ Cross-Domain Robustness

Potential Failure Point

Chances that the linear relationship required for Class Distribution Calibration is specific to the representation space of BERT/text models (e.g., due to the nature of transformer attention) and does not hold true for the output layers of CNNs like ResNet. This would confine the method's utility primarily to the NLP domain.

Dissecting the Mechanism: Is the Dual-Criterion Stop Necessary?

Primary Hypothesis

We hypothesize that the **Confidence Similarity (S_{conf}) criterion is largely redundant** to the final stopping decision. The **Class Distribution Similarity (S_{class}) criterion alone** is the dominant and essential component, capable of identifying the peak performance epoch with comparable accuracy to the combined BUS-stop rule.

* **Claim:** Sclass → Domain Signal

Secondary Hypothesis

We further assert that stopping solely on the **global minimum of Sconf** (which tracks loss) will yield **significantly poorer performance** than stopping on the maximum of Sclass (which tracks accuracy), proving that minimum generalization loss does not equate to maximum classification performance.

* **Claim:** Loss Minimum != Accuracy Maximum

Experimental Design

Section	Content	Presentation Data
Headline	**Experimental Setup: An Ablation Study of BUS-stop Criteria**	**Visual:** A graphic showing a core model splitting into three distinct monitoring paths.
Model & Task	We will use **BERT-base** fine-tuned on the **SST-2** (binary) and **AG-News** (multi-class) datasets, maintaining the low-resource constraint.	* **Datasets:** SST-2 & AG-News
Low-Resource Setting	Labeled Samples (D_l): **K = 50 per class.** Unlabeled Samples (D_u) will be the monitoring base.	* **Constraint:** K=50 per class
Ablation Comparison Methods	We will compare four distinct stopping methods:	* **Method 1:** Full BUS-stop (Benchmark)
	1. **Full BUS-stop:** S_{conf} window $\rightarrow$ S_{class} peak (The original algorithm).	* **Method 2:** S_{class}-only Stop
	2. S_{class}**-only Stop:** Stop at the global maximum of S_{class} across all epochs.	* **Method 3:** S_{conf}-only Stop
	3. S_{conf}**-only Stop:** Stop at the global minimum of S_{conf} across all epochs.	* **Method 4:** $Val - stop_{split}$ (Baseline)
	4. $Val - stop_{split(K/2)}$**:** Traditional validation-split baseline (for context).	

What are we trying to prove or expect from the experiments

1. **Expected Verdict 1: Sclass is the True Performance Tracker**

Expectation: Accuracy(Sclass only) ~ Accuracy(Full BUS-stop)

Proof Point: Du provides enough stability for Sclass alone.

2. Expected Verdict 2: Loss vs. Accuracy Trade-off

Expectation: Loss (Sconf only) < Accuracy (Sclass only)

Proof Point: Accuracy signal is delayed relative to Loss signal.

Why will it work

- ➔ Accuracy-First Signal
- ➔ Loss-Accuracy Decoupling
- ➔ Simplicity Proof

Potential Failure Point

If the **Sclass-only stop** performs poorly compared to the Full BUS-stop, it would have suggested that the **Sconf criterion's role in setting the early stop window is indispensable.** The model might overfit significantly before Sclass peaked, proving that the combined approach is necessary to prevent premature convergence or catastrophic overfitting.

Conclusion

BUS-stop is more than just a stopping rule; it is a **paradigm shift** in how we leverage readily available data. It transforms the vast, often ignored unlabeled data into the **most reliable source of truth** for generalization, empowering practitioners to build high-performing, robust, and stable models even when labeled data is scarce.