PSO-Optimized DistilBERT for Financial Sentiment Analysis: An Empirical Study on Hyperparameter Optimization

Wenxuan Zhu
Department of Biostatistics
University of Michigan
Ann Arbor, MI, USA

***Abstract*—This paper investigates the application of Particle Swarm Optimization (PSO) for hyperparameter tuning of DistilBERT in financial sentiment classification. Using the Financial PhraseBank dataset, we systematically compare baseline DistilBERT performance against PSO-optimized configurations across two experimental settings: the high-agreement subset (2,264 sentences) and the complete dataset (4,846 sentences). On the high-agreement subset, our PSO-optimized model achieves 96.03% accuracy compared to the 95.14% baseline. However, on the complete dataset with noisier labels, the baseline (83.81%) outperforms PSO-optimized configurations (81.65%). These contrasting results reveal important insights about the relationship between dataset characteristics and optimization effectiveness, demonstrating that well-established default hyperparameters often represent near-optimal configurations for mature NLP architectures.**

***Index Terms*—sentiment analysis, financial NLP, DistilBERT, particle swarm optimization, hyperparameter tuning, transfer learning**

I. Introduction

Financial sentiment analysis has emerged as a critical application of natural language processing (NLP), enabling automated extraction of market sentiment from textual data such as news articles, earnings reports, and social media [1]. Accurate sentiment classification supports applications ranging from algorithmic trading to risk assessment, making methodological improvements in this domain practically significant.

The advent of transformer-based pre-trained language models has substantially advanced NLP capabilities. BERT (Bidirectional Encoder Representations from Transformers) [2] introduced deep bidirectional pre-training, enabling state-of-the-art performance across diverse tasks. DistilBERT [3] subsequently demonstrated that knowledge distillation could reduce BERT's size by 40% while retaining 97% of its language understanding capabilities and achieving 60% faster inference. These efficiency gains make DistilBERT particularly attractive for resource-constrained deployment scenarios.

Despite the success of pre-trained models, hyperparameter selection remains crucial for optimal fine-tuning performance. While default hyperparameters are often effective, task-specific optimization may yield improvements. Particle Swarm Optimization (PSO), introduced by Kennedy and Eberhart [4], offers a population-based metaheuristic approach that explores continuous parameter spaces through simulated swarm behavior. PSO has been successfully applied to neural network optimization and represents a compelling alternative to grid search or random search methods.

This paper makes the following contributions: (1) systematic evaluation of PSO-based hyperparameter optimization for DistilBERT fine-tuning on financial sentiment classification; (2) comparative analysis across dataset subsets with varying label agreement levels; and (3) empirical insights into the conditions under which metaheuristic optimization provides benefits over default configurations.

II. Related Work

A. Financial Sentiment Analysis

The Financial PhraseBank dataset, introduced by Malo et al. [5], established a benchmark for financial sentiment classification. The dataset contains 4,845 sentences from financial news, annotated by 16 domain experts. Sentences are labeled as positive, neutral, or negative based on their potential impact on stock prices. The dataset's stratification by annotator agreement levels (50%, 66%, 75%, 100%) enables analysis of model robustness to label noise.

FinBERT [1] demonstrated that domain-specific pre-training on financial corpora improves sentiment classification, achieving state-of-the-art results on Financial PhraseBank. Subsequent work has explored various transformer architectures and training strategies for financial NLP tasks.

B. Knowledge Distillation and Efficient Models

Knowledge distillation, formalized by Hinton et al. [6], enables training compact models that approximate larger teacher networks. Sanh et al. [3] applied this approach to BERT, creating DistilBERT through a triple loss combining language modeling, distillation, and cosine-distance losses. DistilBERT's balance of performance and efficiency makes it suitable for applications requiring fast inference.

C. Particle Swarm Optimization

PSO [4] simulates the social behavior of bird flocks searching for food sources. Each particle in the swarm represents a candidate solution, with movement influenced by personal best positions and global best positions discovered by the swarm. The standard PSO update equations are:

$$v_i^{t+1} = w \cdot v_i^t + c_1 r_1 (p_i - x_i^t) + c_2 r_2 (g - x_i^t) \quad (1)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (2)$$

where v_i is particle velocity, x_i is particle position, p_i is personal best, g is global best, w is inertia weight, c_1 and c_2 are cognitive and social coefficients, and r_1, r_2 are random values in $[0, 1]$.

PSO has been applied to neural network hyperparameter optimization with promising results, offering advantages over grid search through its ability to explore continuous spaces efficiently [7].

III. Method

A. Problem Formulation

Given a financial sentence s, the task is to predict sentiment label $y \in \{negative, neutral, positive\}$. We frame this as a sequence classification problem where DistilBERT encodes the input sentence and a classification head produces class probabilities.

B. Dataset Description

We utilize the Financial PhraseBank dataset in two configurations:

AllAgree Subset: 2,264 sentences with 100% annotator agreement. Distribution: neutral (61.4%), positive (25.2%), negative (13.4%). This high-agreement subset provides cleaner labels but smaller sample size.

Complete Dataset: 4,846 sentences including all agreement levels (50%-100%). Distribution: neutral (59.4%), positive (28.1%), negative (12.5%). This larger dataset introduces label noise from lower-agreement sentences.

Both configurations use stratified 70%/10%/20% train/validation/test splits.

C. Model Architecture

We employ DistilBERT-base-uncased [3], comprising 6 transformer layers with 768 hidden dimensions and 12 attention heads (66M parameters). A classification head maps the [CLS] token representation to three sentiment classes. Input sequences are tokenized using WordPiece with maximum length 128.

D. Baseline Configuration

The baseline model uses standard fine-tuning hyperparameters established in prior work: learning rate 2×10^{-5}, dropout rate 0.3, batch size 16, AdamW optimizer [9] with linear warmup scheduling, and early stopping with patience 3.

E. PSO-Based Hyperparameter Optimization

We optimize three hyperparameters using PSO:

- Learning rate: $[10^{-5}, 10^{-3}]$ (log scale)
- Dropout rate: $[0.2, 0.5]$
- Classification head hidden size: $[64, 512]$ (integer)

PSO configuration: inertia weight $w = 0.9$, cognitive coefficient $c_1 = 2.0$, social coefficient $c_2 = 2.0$. The fitness function is validation accuracy after training for 5 epochs with early stopping.

For the AllAgree subset, we use 5 particles over 3 iterations (15 total evaluations) as a computational constraint. For the complete dataset, we use 10 particles over 10 iterations (100 evaluations), representing approximately 30 hours of computation on CPU. The optimization converged after approximately 55 evaluations, at which point we stopped to avoid computational waste.

F. Training Procedure

Models are trained using cross-entropy loss with gradient clipping (max norm 1.0). The learning rate follows a linear schedule with warmup. Early stopping monitors validation accuracy with patience of 3 epochs. All experiments use random seed 42 for reproducibility and are implemented using PyTorch [10] and the Hugging Face Transformers library [8].

IV. Results

A. Experiment 1: AllAgree Subset

Table I presents results on the high-agreement subset.

TABLE I
Performance on AllAgree Subset (2,264 sentences)

Model	**Test Accuracy**	**Test Loss**
Baseline DistilBERT	95.14%	0.209
PSO-Optimized	96.03%	0.242
Improvement	**+0.89 pp**	–

PSO identified optimal hyperparameters: learning rate 1.07×10^{-4}, dropout rate 0.312, hidden size 417. The optimized model achieves modest improvement (+0.89 percentage points) while using a learning rate approximately 5× higher than the baseline.

Table II shows per-class performance metrics.

TABLE II
Per-Class F1 Scores on AllAgree Subset

Class	**Baseline**	**PSO**	Δ
Negative	0.936	0.908	-0.0279
Neutral	0.973	0.986	+0.0130
Positive	0.911	0.928	+0.0163
Macro Avg	0.940	0.940	+0.0004

Despite the overall accuracy improvement, the macro-averaged F1-score shows minimal change (+0.04%), suggesting sensitivity-specificity trade-offs.

B. Experiment 2: Complete Dataset

Table III presents results on the complete dataset including lower-agreement sentences.

TABLE III
PERFORMANCE ON COMPLETE DATASET (4,846 SENTENCES)

Model	Test Accuracy	Test Loss
Baseline DistilBERT	83.81%	0.810
PSO-Optimized	81.65%	0.911
PSO + Regularization	82.37%	0.872
PSO vs Baseline:	**-2.16 pp**	–
PSO+Reg vs Baseline:	**-1.44 pp**	–

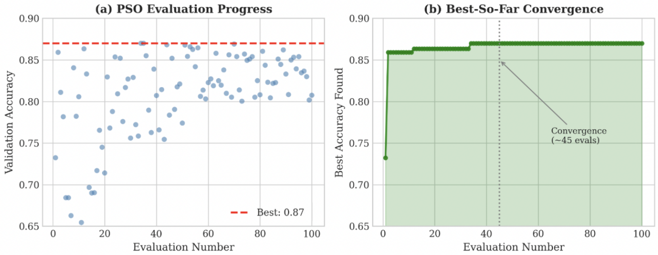

Fig. 1. PSO optimization progress showing validation accuracy across approximately 55 evaluations (left) and best-so-far convergence curve (right). The algorithm converges after approximately 40 evaluations, with the best configuration achieving 87.42% validation accuracy.

Contrary to Experiment 1, PSO optimization on the complete dataset yields *worse* performance than the baseline. The PSO-discovered hyperparameters (learning rate 1.02×10^{-4}, dropout 0.211, hidden size 386) lead to overfitting, as evidenced by the higher test loss. Adding stronger regularization (dropout 0.4, weight decay 0.01) partially recovers performance but still underperforms the baseline.

TABLE IV
PER-CLASS F1 SCORES ON COMPLETE DATASET

Class	Baseline	PSO	Δ
Negative	0.817	0.768	-0.049
Neutral	0.867	0.858	-0.009
Positive	0.791	0.753	-0.038
Macro Avg	0.825	0.793	-0.032

Table IV reveals consistent degradation across all classes, with the negative class most affected.

C. Analysis of Optimal Hyperparameters

Table V compares discovered hyperparameters across experiments.

Both PSO runs converge to similar learning rates ($\approx 10^{-4}$), suggesting this represents a local optimum in the search space. However, the lower dropout discovered for the complete dataset (0.211 vs. 0.312) may contribute to overfitting on noisier data.

TABLE V
OPTIMAL HYPERPARAMETERS DISCOVERED BY PSO

Parameter	Baseline	AllAgree	Complete
Learning Rate	2×10^{-5}	1.07×10^{-4}	1.02×10^{-4}
Dropout Rate	0.30	0.312	0.211
Hidden Size	768	417	386

V. DISCUSSION

A. Dataset Characteristics and Optimization

The contrasting results between experiments reveal a critical insight: PSO optimization is more effective on clean, high-agreement data. The AllAgree subset provides consistent supervision signals, allowing PSO to identify configurations that genuinely improve learning dynamics. In contrast, the complete dataset's label noise creates an adversarial optimization landscape where PSO may overfit to validation set idiosyncrasies.

B. Robustness of Default Hyperparameters

The baseline learning rate of 2×10^{-5} represents years of community-driven optimization for BERT-family fine-tuning. Our results suggest this default is particularly robust to label noise, providing implicit regularization that aggressive optimization may undermine. This aligns with the broader observation that well-established defaults often encode substantial empirical wisdom.

C. Computational Considerations

PSO optimization required 15 evaluations (approximately 3 hours on CPU) for the AllAgree subset and 55 evaluations (approximately 20 hours) for the complete dataset. The marginal gains on clean data and negative transfer on noisy data suggest that for financial sentiment analysis with DistilBERT, default hyperparameters may be preferable unless clean, high-agreement training data is available.

D. Limitations

This study has several limitations. First, computational constraints limited PSO iterations, potentially missing better configurations. Second, we optimize only three hyperparameters; batch size, warmup steps, and architecture modifications remain unexplored. Third, our experiments focus on a single dataset; generalization to other financial NLP tasks requires validation.

VI. CONCLUSION

This paper presented an empirical study of PSO-based hyperparameter optimization for DistilBERT fine-tuning on financial sentiment analysis. Our experiments reveal a nuanced picture: PSO achieves modest improvements (95.14% → 96.03%) on high-agreement data but underperforms baseline configurations on noisier complete datasets (83.81% vs. 81.65%). These results underscore that metaheuristic optimization is not universally beneficial and that dataset characteristics significantly influence optimization outcomes.

Our findings suggest that practitioners should carefully consider data quality before investing computational resources in hyperparameter optimization. For mature architectures like DistilBERT with well-established defaults, the marginal gains from optimization may not justify the computational cost, particularly when training data contains label noise. Future work could explore adaptive regularization strategies that adjust hyperparameters based on estimated label noise, potentially combining the exploration benefits of PSO with robustness to annotation inconsistencies.

References

[1] D. Araci, "FinBERT: Financial sentiment analysis with pre-trained language models," *arXiv preprint arXiv:1908.10063*, 2019.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, pp. 4171–4186, 2019.

[3] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," in *NeurIPS EMC^2 Workshop*, 2019.

[4] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[5] P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala, "Good debt or bad debt: Detecting semantic orientations in economic texts," *Journal of the Association for Information Science and Technology*, vol. 65, no. 4, pp. 782–796, 2014.

[6] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *NIPS Deep Learning Workshop*, 2015.

[7] F. Marini and B. Walczak, "Particle swarm optimization (PSO): A tutorial," *Chemometrics and Intelligent Laboratory Systems*, vol. 149, pp. 153–165, 2015.

[8] T. Wolf et al., "Transformers: State-of-the-art natural language processing," in *Proc. EMNLP: System Demonstrations*, pp. 38–45, 2020.

[9] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. ICLR*, 2019.

[10] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.