A Two-Stage Fusion Framework for Few-Shot Species Identification from Melting Curve Data

Chengqian Zhang
Worcester Polytechnic Institute
Worcester, MA, USA
czhang12@wpi.edu

Kyumin Lee
Worcester Polytechnic Institute
Worcester, Massachusetts, USA
kmlee@wpi.edu

Zhongyi Tong
Worcester Polytechnic Institute
Worcester, MA, USA
ztong@wpi.edu

Abstract

Illegal wildlife trade is a major driver of biodiversity loss, and species-level identification from fragmented wildlife products remains a key bottleneck for conservation enforcement. High-Resolution Melt (HRM) analysis provides a portable and cost-effective molecular tool for identifying sharks and rays in trade, but existing melting-curve approaches often require large reference libraries and are less suitable for low-resource field settings. We propose a two-stage framework for few-shot species classification from HRM melting-curve time series. First, statistical curve features are converted into structured textual prompts to warm up a text encoder. Second, patch embeddings from the raw time series are fused with the text representation through a lightweight multi-head self-attention layer. On a dataset of 65 shark and ray species, our method achieves robust classification with only three training samples per species. Across 10 random splits, it outperforms seven state-of-the-art baselines and reaches 93.08% average accuracy, demonstrating the potential of lightweight HRM-based learning systems for scalable, field-deployable wildlife enforcement. Our code will be available at https://github.com/czhang657/Two-Stage-HRM-Melting-Curve-Classification.

CCS Concepts

• **Computing methodologies** → **Supervised learning by classification**; • **Applied computing** → **Bioinformatics**.

Keywords

Few-shot species identification, Melting-curve data, HRM assay, Two-stage fusion framework, Wildlife trade enforcement

ACM Reference Format:
Chengqian Zhang, Kyumin Lee, and Zhongyi Tong. 2026. A Two-Stage Fusion Framework for Few-Shot Species Identification from Melting Curve Data. In *17th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '26), June 30–July 03, 2026, Rende (CS), Italy*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3807503.3819372

1 Introduction

High-Resolution Melt (HRM) assay analysis provides a portable and cost-effective way to obtain DNA melting curves for species identification (Figure 1), and has recently been explored for detecting illegally traded shark and ray products. Compared with traditional DNA sequencing, HRM-based screening is faster, cheaper, and more suitable for preliminary enforcement decisions. Cardeñosa et al. [2] proposed converting HRM curves into images and classifying them with ResNet18. However, this image-based formulation is less suitable for low-resource settings with only a few reference samples per species.

HRM curves are inherently one-dimensional numerical sequences rather than images. Rendering a 3,478-point curve into a 224×224 image may compress fine-grained signal information and obscure quantitative properties such as intensity, variance, amplitude, and frequency-domain characteristics. Thus, image-based models mainly learn curve morphology rather than directly exploiting the full numerical signal.

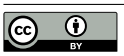
We therefore revisit HRM-based species identification as a lightweight sequence modeling problem and propose a two-stage fusion framework that combines patch-based HRM sequence encoding with structured textual prompts derived from statistical curve features. By fusing native time-series signals with high-level semantic summaries through compact self-attention, our method supports few-shot HRM species classification.

In this paper, we make the following contributions:

- We introduce a statistical-to-semantic transformation that converts global melting curve descriptors into discretized textual prompts, enabling models to exploit quantitative HRM signal characteristics beyond raw curve morphology.
- We propose a lightweight two-stage fusion framework that combines patch-based time-series encoding with structured statistical prompts through intermediate self-attention fusion.
- Experiments on 65 shark and ray species show that our method consistently outperforms seven baselines in the 3-shot setting, achieving 93.08% average accuracy across 10 random splits, with ablation results confirming the effectiveness of each component.

2 Related Work

HRM and Melting Curve-Based Classification. High-Resolution Melt (HRM) analysis identifies sequence variations by monitoring fluorescence changes during DNA denaturation [16]. The resulting melting curve reflects thermodynamic properties such as GC content, sequence length, and nucleotide composition, supporting SNP, mutation, and species-level discrimination [13]. Traditional HRM analysis relies on T_m, curve normalization, difference plotting, and clustering [6, 13], while recent learning-based approaches use CNN-LSTM models for temporal classification [12] or image-based ResNet models for curve morphology classification [7]. However,



BCB '26, Rende (CS), Italy

ACM ISBN 979-8-4007-2653-8/26/06
https://doi.org/10.1145/3807503.3819372

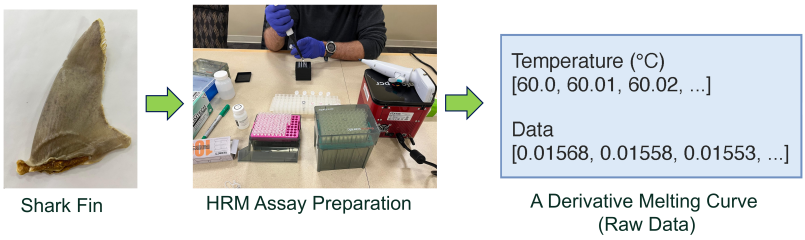


Figure 1: Overview of the HRM workflow. A shark-fin sample is processed through a high-resolution melt (HRM) assay. The resulting fluorescence data are processed to generate a derivative melting curve (–dF/dT), which is used for downstream modeling.

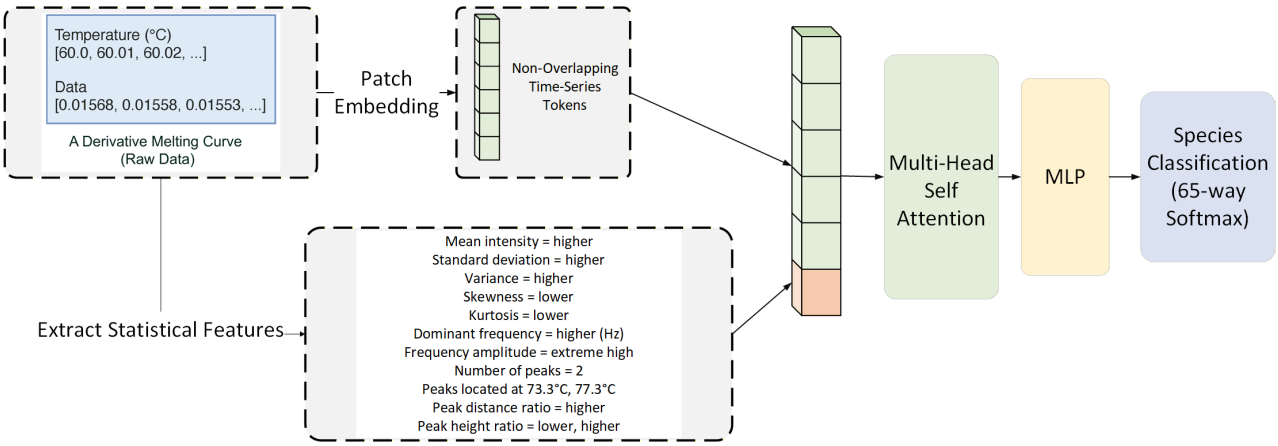


Figure 2: Overview of our framework. Raw melting curves obtained from High-Resolution Melt (HRM) assays are first processed to extract global statistical features, which are used to construct textual information that will be fed into a textual encoder. The assays are simultaneously fed into a patch embedding layer, which outputs non-overlapping time-series patch tokens. We concatenate the patch tokens and [CLS] token output by textual encoder and apply multi-head self attention layer and MLP layer to get final classification decision.

these methods mainly emphasize curve shape and often underutilize explicit global statistical descriptors.

Time-Series Modeling and Patching. Temporal convolutional models such as TCN and CDIL-CNN capture long-range or periodic temporal patterns through dilated convolutions [1, 3]. Transformer-based time-series models further improve temporal representation through self-attention, frequency-domain modeling, sparse attention, or decomposition mechanisms [15, 17, 19, 21, 22]. Patch-based tokenization, popularized by ViT [5] and PatchTST [11], reduces sequence length while preserving local temporal patterns. This makes patching well suited for smooth HRM melting curves, where local variations and global dependencies are both informative.

Multi-Modal Fusion between Time-Series and Textual Data. Multi-modal fusion captures interactions between heterogeneous inputs beyond simple concatenation. Prior work has explored late fusion [14], intermediate interaction such as cross-attention [20], and time-series–text integration through models such as TimeCAP and Time-LLM [8, 9]. Motivated by these studies, we fuse HRM patch embeddings with structured statistical-prompt representations using a lightweight single self-attention layer.

3 Our Proposed Approach

In this section, we describe our proposed framework. Under the few-shot setting, we introduce a two-stage training framework for species identification: (i) structured text prompt generation and first-stage classification; and (ii) second-stage self-attention–based time-series representation learning.

3.1 Overview of the Framework

As shown in Figure 2, our framework consists of statistical feature extraction, time-series patch embedding, and self-attention-based fusion.

Melting curve measurements are long one-dimensional sequences (length 3,478), representing the negative first derivative of fluorescence with respect to temperature, collected from 60°C to 94.77°C in 0.01°C increments. Given a derivative melting curve of length 3,478, the time-series branch divides the signal into fixed-length patches and projects them into pseudo-token embeddings to capture local curve patterns. In parallel, the statistical branch extracts global descriptors, including distributional moments, peak-related features, and frequency statistics, and converts them into a structured textual

prompt encoded by BERT-base [4]. The resulting prompt representation is concatenated with the patch embeddings and processed by a lightweight multi-head self-attention layer. Finally, the fused tokens are aggregated and passed to an MLP classification head for species prediction.

3.2 First Training Stage

Stage 1 encodes global statistical information from each HRM melting curve. We extract statistical descriptors from the raw curve, convert them into a structured textual prompt, and train a BERT-based text encoder with cross-entropy loss for species classification.

Unlike the ResNet18 baseline [2], which mainly learns morphology from rasterized curve images, this stage explicitly preserves numerical properties such as intensity magnitude, distributional statistics, and global signal trends. It therefore provides semantic support for the subsequent fusion stage.

Statistical Feature Extraction. For each melting curve, we extract descriptors capturing global statistics, peak structure, and frequency-domain behavior. Global features include mean intensity, standard deviation, variance, skewness, and kurtosis. Peak features are obtained using SciPy `find_peaks` with a prominence threshold of 0.001 and a minimum distance of 10 sampling points, from which we record the number of peaks, peak temperature locations, normalized inter-peak distance ratios, and peak height ratios. Frequency features are computed with Fast Fourier Transform (FFT) by retaining positive frequency components and recording the dominant frequency and its normalized amplitude. These descriptors provide compact quantitative summaries of HRM curve dynamics beyond raw morphology.

Prompt Construction and Encoding. Since transformer-based language models are generally insensitive to precise numerical magnitudes, directly feeding continuous statistics may not yield stable representations. Therefore, we discretize scalar-valued statistics—except for the number of peaks and peak locations—into four quartile-based categories according to the empirical distribution computed from the training set. The 25%, 50%, and 75% thresholds are fixed after preprocessing training data and reused during validation and testing to strictly prevent information leakage. Each statistic is then converted into a structured textual token indicating its relative quartile membership (i.e., *extreme low*, *lower*, *higher*, *extreme high*).

As illustrated in Figure 3, the resulting textual prompt is first embedded and combined with positional encoding before being processed by a BERT-base encoder consisting of 12 transformer layers with multi-head self-attention and feed-forward blocks. The final `[CLS]` token representation is extracted as a global summary of the statistical prompt.

Stage-1 Supervised Classification. In Stage 1, we train the BERT encoder to make statistical prompts discriminative for species identification. The `[CLS]` representation is passed through a linear head and softmax layer to predict 65 species labels using cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = -\sum_i y_i \log \hat{y}_i, \tag{1}$$

where y_i and $\hat{y}_i$ denote the ground-truth label and predicted probability, respectively. The trained encoder is then reused in Stage 2 to provide statistical representations for fusion with raw time-series signals.

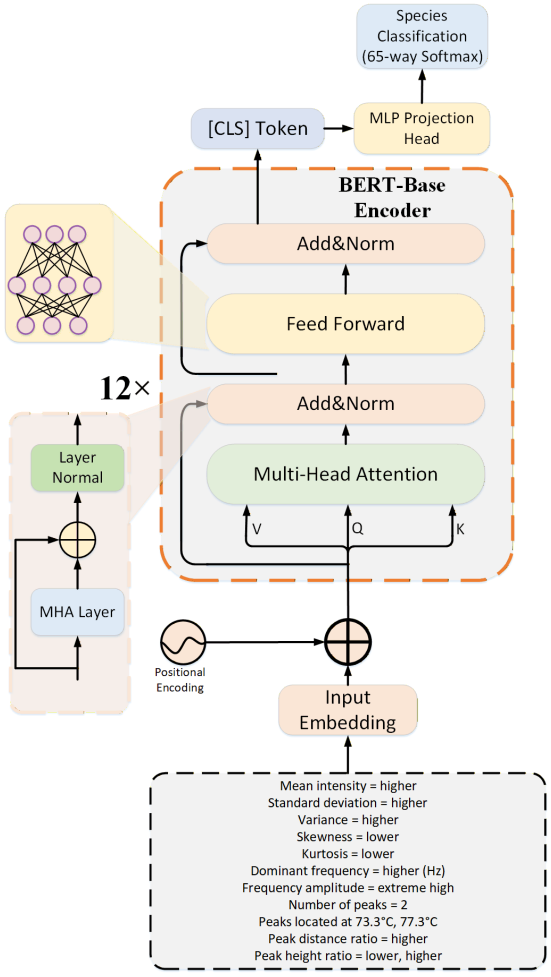


Figure 3: Overview of Stage 1. Statistical features extracted from each melting curve are discretized into quartile-based categories. The resulting structured prompt is fed into a BERT-base encoder. The final `[CLS]` representation is projected through an MLP head for 65-way species classification.

3.3 Second Training Stage with Intermediate Fusion

As shown in Figure 4, Stage 2 fuses local temporal patterns from raw melting curves with the global statistical representation learned in Stage 1.

Patching and Token Construction. Following TimeCAP [9] and PatchTST-style patching [11], each melting curve is segmented into non-overlapping patches of length 16. Each patch is linearly projected into a 128-dimensional token to capture local temporal patterns from the native one-dimensional signal.

Reusing the Stage-1 Text Encoder. After Stage 1, we discard the classification head and retain the trained BERT encoder. The `[CLS]`

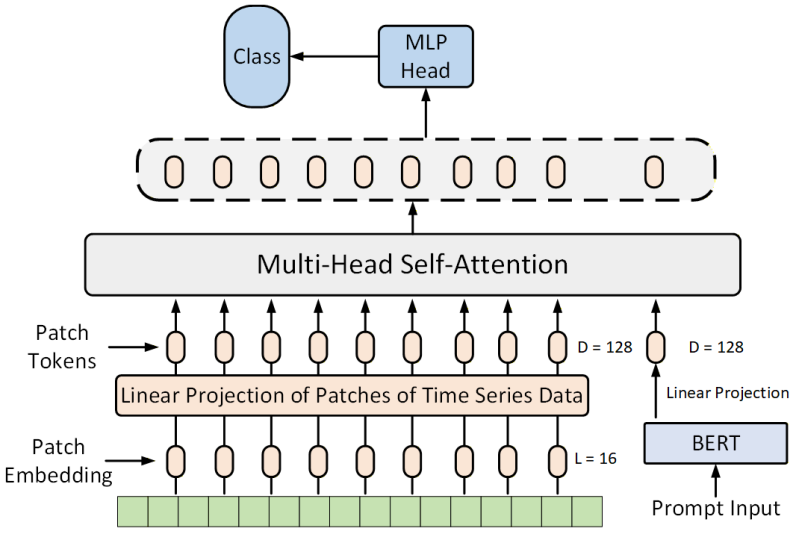


Figure 4: Overview of the second training stage. The raw melting curve is segmented into non-overlapping patches and projected into a 128-dimensional latent space. The Stage-1 statistical prompt is encoded by the fine-tuned BERT encoder, whose `[CLS]` token is projected as a global semantic token. The patch tokens and statistical token are concatenated and processed by multi-head self-attention to fuse local temporal patterns with global statistical descriptors.

representation of the structured statistical prompt is projected from 768 to 128 dimensions to match the patch-token space. During Stage 2, the BERT encoder is unfrozen and jointly optimized with the time-series branch.

Intermediate Fusion via Self-Attention. The patch tokens and statistical token are concatenated and processed by a multi-head self-attention layer, enabling interaction between local temporal structures and global statistical summaries. The fused tokens are aggregated and passed to an MLP classifier trained with the cross-entropy loss in Eq. 1.

3.4 Inference

At inference time, given a new melting curve, we first extract the same set of statistical descriptors described in Section 3.2. Scalar-valued statistics are discretized using the fixed quartile thresholds computed from the training set to ensure consistency and prevent data leakage. The resulting structured prompt is processed by the trained BERT encoder to obtain the `[CLS]` representation. In parallel, the raw melting-curve data is segmented into non-overlapping patches and projected into the 128-dimensional latent space. The projected patch tokens and the projected `[CLS]` token are concatenated and jointly processed by the multi-head self-attention module. The output tokens are aggregated and passed through the final MLP classification head to produce the predicted species label.

4 Experiments

4.1 Dataset and Data Preprocessing

We evaluate our method in a strictly low-resource setting using the shark and ray HRM dataset from Cardeñosa et al. [2]. Since melting curve representations achieved the best performance in their study, we use melting curves as the input modality.

The original dataset contains 1,154 samples across 87 species. To build a balanced few-shot benchmark, we remove species with fewer than five samples and randomly subsample five curves for species with more than five samples. The final dataset includes 65 species and 325 curves. Each curve is a one-dimensional sequence of length 3,478, representing the negative first derivative of fluorescence with respect to temperature from 60°C to 94.77°C at 0.01°C intervals.

For each species, we use a 3/1/1 train–validation–test split, yielding a 3-shot classification setting. We repeat this procedure over 10 random seeds (0–9) and report average performance across splits.

4.2 Baseline Models

We compare our method with seven baselines covering HRM-specific and general time-series models. **CNN-LSTM** [12] combines 1D convolutional feature extraction with LSTM-based temporal modeling. **ResNet18** [2] converts melting curves into 224×224 images and uses a frozen pre-trained ResNet18 backbone with a trainable classification head. **CDIL-CNN** [3] uses stacked dilated convolutions for multi-scale temporal patterns. **TimesNet** [17] transforms 1D time series into 2D representations using dominant periods identified by the Fast Fourier Transform (FFT). **PatchTST** [11] segments long sequences into patches and applies Transformer encoding. **iTransformer** [10] is adapted to single-channel HRM curves by segmenting each sequence into equal-length tokens. **Autoformer** [19] uses decomposition-based Transformer encoding with global pooling and a linear classifier. Further details of the baseline design and its description are presented in the Appendix A.1.

4.3 Hyperparameter Search

To ensure a fair comparison, we conduct architecture-aware hyperparameter tuning for all baselines based on validation performance. For CNN-LSTM and CDIL-CNN, we grid search learning rate

Model	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	Mean
MLP (Numeric Features)	1.54	0.00	1.54	1.54	0.00	0.00	0.00	0.00	1.54	1.54	0.77
BERT (Numerical Prompt)	75.38	76.92	63.08	64.62	69.23	76.92	80.00	66.15	58.46	49.23	68.00
BERT (Proposed Stage 1)	61.54	70.77	70.77	70.77	69.23	78.46	72.31	72.31	72.31	75.38	**71.39**

Table 1: Stage 1 classification accuracy (%) of our model and variants across 10 independent random splits.

Model	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	Mean
CNN-LSTM	60.00	53.85	76.92	76.92	80.00	81.54	76.92	69.23	76.92	80.00	73.23
ResNet18	83.07	92.31	90.77	89.23	89.23	93.85	87.69	90.77	93.85	93.85	90.46
CDIL-CNN	40.00	52.31	52.31	50.77	43.08	52.32	50.77	36.92	43.08	49.23	47.08
TimesNet	80.00	86.15	80.00	80.00	84.62	84.62	78.46	72.31	78.46	81.54	80.62
PatchTST	84.62	93.85	87.69	87.69	89.23	90.77	87.69	87.69	89.23	87.69	88.62
iTransformer	92.31	93.85	93.85	89.23	90.77	90.77	89.23	90.77	90.77	87.69	90.92
Autoformer	1.54	1.54	3.08	3.08	1.54	1.54	1.54	0.00	1.54	1.54	1.69
Our Model	90.77	95.38	90.77	92.31	92.31	93.85	96.92	92.31	95.38	90.77	**93.08**

Table 2: Classification accuracy (%) of our model and the baselines across 10 independent random splits.

and weight decay within architecture-specific ranges: $[10^{-4}, 10^{-2}]$ and $[10^{-6}, 10^{-3}]$ for CNN-LSTM, and $[5 \times 10^{-4}, 3 \times 10^{-3}]$ and $[0, 3 \times 10^{-3}]$ for CDIL-CNN. For ResNet18, we use a broader search space due to its optimization sensitivity, with learning rate and weight decay both searched from 10^{-6} to 10^{-1}.

For TSLib-based models, including PatchTST, Autoformer, iTransformer, and TimesNet, we use the official TSLib implementations [18]. Each model is trained for up to 3,000 epochs with early stopping patience of 100. Learning rate and weight decay are searched over $\{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$, resulting in 16 combinations per model per seed. All models use AdamW with cosine learning rate decay and 10-epoch linear warmup, and the best configuration is selected by validation loss.

For our model, we additionally tune patch size and stride over $\{4, 8, 16\}$, yielding nine combinations. The non-overlapping setting with patch size 16 and stride 16 achieves the best validation performance and is used in all reported experiments.

4.4 Effectiveness of Structured Prompts and Stage 1 Training

Before evaluating the full model, we assess the statistical prompt representation alone. For each of the 10 random splits, quartile thresholds are computed only from the training set and applied to validation and test samples to prevent data leakage.

Fine-tuning BERT-base on the generated prompts with a classifier over the final `[CLS]` representation achieves 71.39% average accuracy (Table 1), showing that statistical descriptors alone are discriminative. In contrast, an MLP trained directly on raw numerical features under the same protocol collapses to near-random performance, achieving only 0.77% accuracy despite having over $100K$ trainable parameters. This suggests that structured textual encoding is more effective than direct numerical learning in the 3-shot setting.

Prompts with exact numerical values but without quartile discretization also outperform the MLP baseline, but underperform the proposed structured prompts, indicating that discretization reduces numerical noise and improves semantic alignment. Separability analysis further shows that among all $\binom{65}{2} = 2080$ inter-species pairs, only 0.1% share identical discretized profiles, while 69.4% differ in at least 6 of 8 features, confirming strong species-level separability before fusion. The experimental results show effectiveness of the proposed structured prompt and stage 1 training.

4.5 Effectiveness of the Proposed Method

As shown in Table 2, our full two-stage model achieves the best average accuracy of **93.08%** across 10 random splits, outperforming all seven baselines. Compared with ResNet18, the strongest prior species classifier, our method improves accuracy from 90.46% to 93.08%. CNN-LSTM reaches 73.23%, CDIL-CNN obtains 47.08%, and TimesNet achieves 80.62%, suggesting that conventional temporal convolutional or frequency-aware architectures are less effective for few-shot HRM species classification.

Our method also outperforms transformer-based baselines, including PatchTST at 88.62% and iTransformer at 90.92%. Although these models use deeper self-attention stacks, our framework achieves higher accuracy with only a single self-attention layer, indicating that the Stage-1 statistical prompt provides complementary global guidance beyond purely numerical patch tokens. Autoformer performs poorly at 1.69%, likely because its forecasting-oriented decomposition mechanism is not well aligned with species classification.

Paired t-tests across the same 10 splits show statistically significant improvements over ResNet18 ($p = 0.0488$) and iTransformer ($p = 0.0499$). Species-level error analysis further shows that 40 of 65 species achieve perfect accuracy across all splits, with at most 3 errors out of 10 for any species. These results indicate that our gains are stable across random splits and are not driven by a small

Model	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	Mean
AB1: w/o the BERT Component	87.69	95.38	86.15	87.69	93.85	89.23	86.15	86.15	95.38	89.23	89.69
AB2: Prompt with Numerical Values	86.15	87.69	84.61	86.15	92.30	92.30	92.30	87.69	95.38	93.85	89.84
AB3: Single-Stage Training w/o Stage-1 Pretraining	87.69	95.38	92.31	92.31	93.85	93.85	93.85	89.23	90.77	90.77	92.00
Our Model	90.77	95.38	90.77	92.31	92.31	93.85	96.92	92.31	95.38	90.77	**93.08**

Table 3: Ablation study and fusion-variant analysis across 10 independent random splits evaluating key design choices of our framework.

subset of species. This experiment result confirms that combining structured statistical prompts with lightweight patch-based self-attention yields superior performance.

4.6 Ablation Study

We conduct ablation studies under the same protocol to assess each component. As shown in Table 3, removing the BERT branch reduces accuracy to 89.69%, confirming that statistical prompts provide complementary global information beyond patch tokens. Replacing quartile-discretized prompts with exact numerical prompts achieves 89.84%, indicating that discretization improves robustness and semantic alignment. Training the full model without Stage-1 pretraining decreases accuracy to 92.00%, showing that supervised prompt pretraining helps establish a stable statistical representation before fusion.

5 Conclusion

In this paper, we presented a two-stage fusion framework for few-shot species identification from HRM melting curves. By combining structured statistical prompts with patch-based time-series representations through lightweight self-attention fusion, our method outperforms seven baselines across 10 independent 3-shot splits on 65 shark and ray species. These results show that discretized statistical prompts complement raw curve morphology and improve robustness under limited data, suggesting a promising direction for low-resource scientific time-series classification.

Acknowledgments

This work was supported by the National Science Foundation under Grant IOS-2430277 and the Paul G. Allen Family Foundation.

References

[1] Shaojie Bai, J. Kolter, and Vladlen Koltun. 2018. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. (03 2018).

[2] Diego Cardeñosa, Zhuang Luo, Kyumin Lee, Emma Aitken, Maria A. Herrera, DeEtta Mills, John Carlson, and Gavin Naylor. 2025. Integrating portable qPCR and image recognition to combat illegal trade in sharks and rays. *Scientific Reports* 15 (2025), 38629.

[3] Lei Cheng, Ruslan Khalitov, Tong Yu, and Zhirong Yang. 2023. Classification of long sequential data using circular dilated convolutional neural networks. *Neurocomputing* 518 (2023), 50–59. https://www.sciencedirect.com/science/article/pii/S0925231222013364

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *ACL*.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*. https://openreview.net/forum?id=YicbFdNTTy

[6] Maria Erali, Karl V. Voelkerding, and Carl T. Wittwer. 2008. High resolution melting applications for clinical laboratory medicine. *Experimental and Molecular Pathology* 85, 1 (2008), 50–58. https://www.sciencedirect.com/science/article/pii/S0014480008000397 Special Issue: Molecular Pathology and Molecular Diagnostics.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR (CVPR '16)*. IEEE, 770–778. http://ieeexplore.ieee.org/document/7780459

[8] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. 2024. Time-LLM: Time Series Forecasting by Reprogramming Large Language Models. In *ICLR*. https://openreview.net/forum?id=Unb5CVPtae

[9] Geon Lee, Wenchao Yu, Kijung Shin, Wei Cheng, and Haifeng Chen. 2025. Timecap: Learning to contextualize, augment, and predict time series events with large language model agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[10] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. 2023. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. *ArXiv* abs/2310.06625 (2023). https://api.semanticscholar.org/CorpusID:263830644

[11] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *ICLR*. https://openreview.net/forum?id=Jbdc0vTOcol

[12] Fatma Ozge Ozkok and Mete Celik. 2022. A hybrid CNN-LSTM model for high resolution melting curve classification. *Biomedical Signal Processing and Control* 71 (2022), 103168. https://www.sciencedirect.com/science/article/pii/S1746809421007655

[13] Gudrun H Reed, Jana O Kent, and Carl T Wittwer. 2007. High-Resolution DNA Melting Analysis for Simple and Efficient Molecular Diagnostics. *Pharmacogenomics* 8, 6 (2007), 597–608. https://doi.org/10.2217/14622416.8.6.597 PMID: 17559349.

[14] Yao-Hung Tsai, Shaojie Bai, Paul Liang, J. Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Multimodal Transformer for Unaligned Multimodal Language Sequences. *Proceedings of the conference. Association for Computational Linguistics. Meeting* 2019, 6558–6569.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

[16] Carl T Wittwer, Gudrun H Reed, Cameron N Gundry, Joshua G Vandersteen, and Robert J Pryor. 2003. High-Resolution Genotyping by Amplicon Melting Analysis Using LCGreen. *Clinical Chemistry* 49, 6 (06 2003), 853–860. arXiv:https://academic.oup.com/clinchem/article-pdf/49/6/853/32732208/clinchem0853.pdf

[17] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. 2023. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *ICLR*.

[18] Haixu Wu, Yuxuan Wang, and Yong Liu. 2026. Time Series Library (TSLib): A Library for Advanced Deep Time Series Models for General Time Series Analysis. https://github.com/thuml/Time-Series-Library

[19] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. 2021. Autoformer: decomposition transformers with auto-correlation for long-term series forecasting. In *NeurIPS (NIPS '21)*. Article 1717, 12 pages.

[20] Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018. Memory fusion network for multi-view sequential learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[21] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *Proceedings of the AAAI conference on artificial intelligence*.

[22] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. 2022. FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting. In *ICML (Proceedings of Machine Learning Research, Vol. 162)*. 27268–27286.