Fcg-Former: Identification of Functional Groups in FTIR Spectra Using Enhanced Transformer-Based Model

Vu Hoang Minh Doan,[#] Cao Duong Ly,[#] Sudip Mondal,[#] Thi Thuy Truong, Tan Dung Nguyen, Jaeyeop Choi, Byeongil Lee,* and Junghwan Oh*
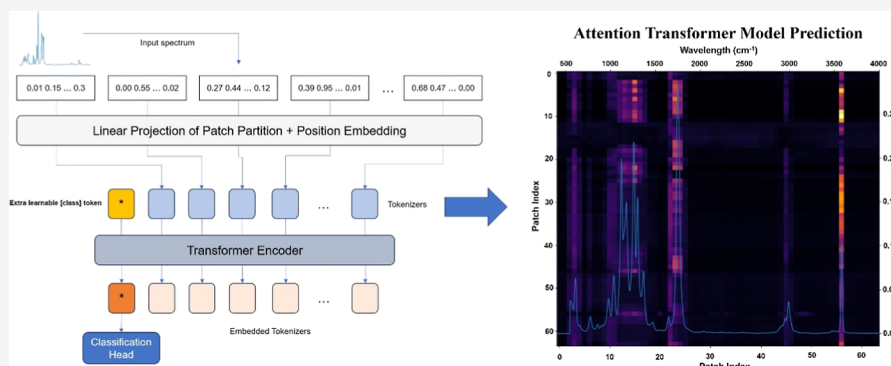
Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Deep learning (DL) is becoming more popular as a useful tool in various scientific domains, especially in chemistry applications. In the infrared spectroscopy field, where identifying functional groups in unknown compounds poses a significant challenge, there is a growing need for innovative approaches to streamline and enhance analysis processes. This study introduces a transformative approach leveraging a DL methodology based on transformer attention models. With a data set containing approximately 8677 spectra, our model utilizes self-attention mechanisms to capture complex spectral features and precisely predict 17 functional groups, outperforming conventional architectures in both functional group prediction accuracy and compound-level precision. The success of our approach underscores the potential of transformer-based methodologies in enhancing spectral analysis techniques.

INTRODUCTION

Functional groups serve to identify the physical properties including boiling point,[1] melting point,[2] solubility,[3] and viscosity[4] of chemical compounds. By recognizing functional groups, researchers can classify and categorize compounds, aiding in their characterization and identification.[5] In the field of medicinal chemistry, functional groups play a critical role in determining a compound's biological activity and pharmacological properties.[6] Specific functional groups might impart desirable therapeutic effects or influence the compound's interaction with biological targets.[7] Understanding these relationships is essential for drug design and optimization. In polymer chemistry, by controlling the types and distribution of functional groups, researchers can tailor their mechanical, thermal, and chemical properties for specific applications, such as in materials science, engineering, and biomedicine.[8−10] Knowledge of functional groups is thus essential for understanding and predicting the behavior of substances in various environments.

Functional groups are specific arrangements of atoms within a molecule that give the compound its unique chemical characteristics.[11] Infrared (IR) spectroscopy is a widely used qualitative and quantitative analytical method utilized for the identification and characterization of chemical compounds, relying on their molecular vibrations.[12] When a sample is exposed to infrared radiation, certain wavelengths are selectively absorbed by its chemical bonds, causing transitions between quantized vibrational energy levels.[13] Even at the lowest energy state, known as the zero point energy, molecular bonds possess intrinsic vibrational energy.[14] The absorption of infrared radiation provides the exact amount of energy required to elevate the molecule from a lower vibrational state to a higher one.[15] Each type of chemical bond has a characteristic vibrational frequency associated with it, corresponding to the energy difference between these quantized
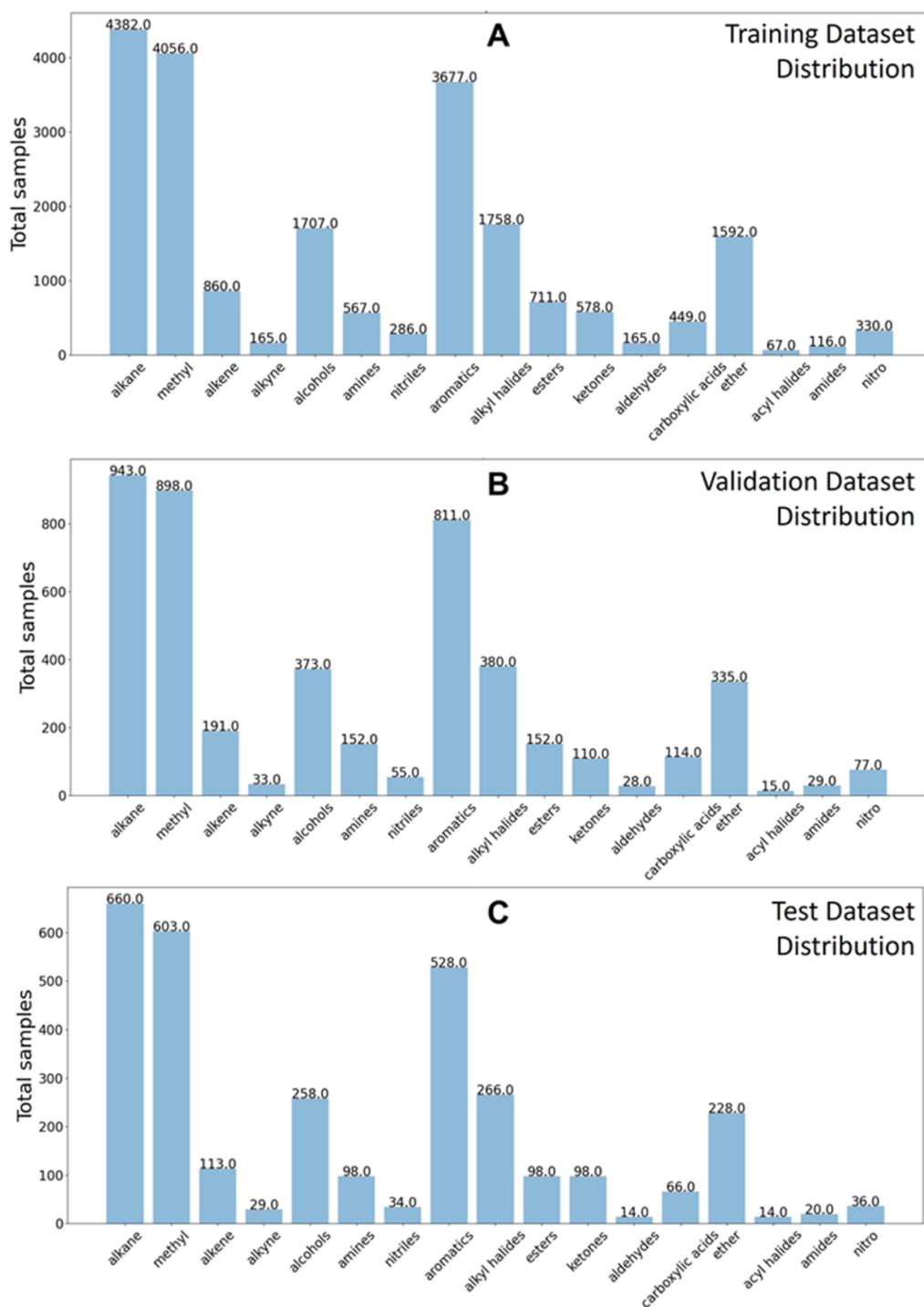
Figure 1. Distribution of (A) training data set, (B) validation data set, and (C) test data set.

levels.[16] This characteristic absorption allows IR spectroscopy to detect and analyze the structural features of organic compounds effectively.[15]

The IR spectra produced from IR spectroscopy are typically investigated by manually marking all relevant peaks corresponding to specific functional groups. Even with the expertise of researchers and the assistance of available documents, this manual procedure is time-consuming, especially in the case of complex compounds. Moreover, various physical and chemical factors that affect the sample's constituents may cause changes in the structural environment of specific functional groups,

leading to notable deviations in those groups' typical peak frequencies from their representative ranges.[17] There might be an overlap phenomenon in the fingerprint region (400–1500 cm^{-1}) of most IR spectra. Because each material's distinctive properties are contained in this frequency band, it becomes much more difficult for researchers to identify specific peaks and corresponding functional groups. To reduce the inefficiency of manual interpretation, the computer and AI-aided approach such as innovative deep learning (DL) is suggested when analyzing the IR spectra.
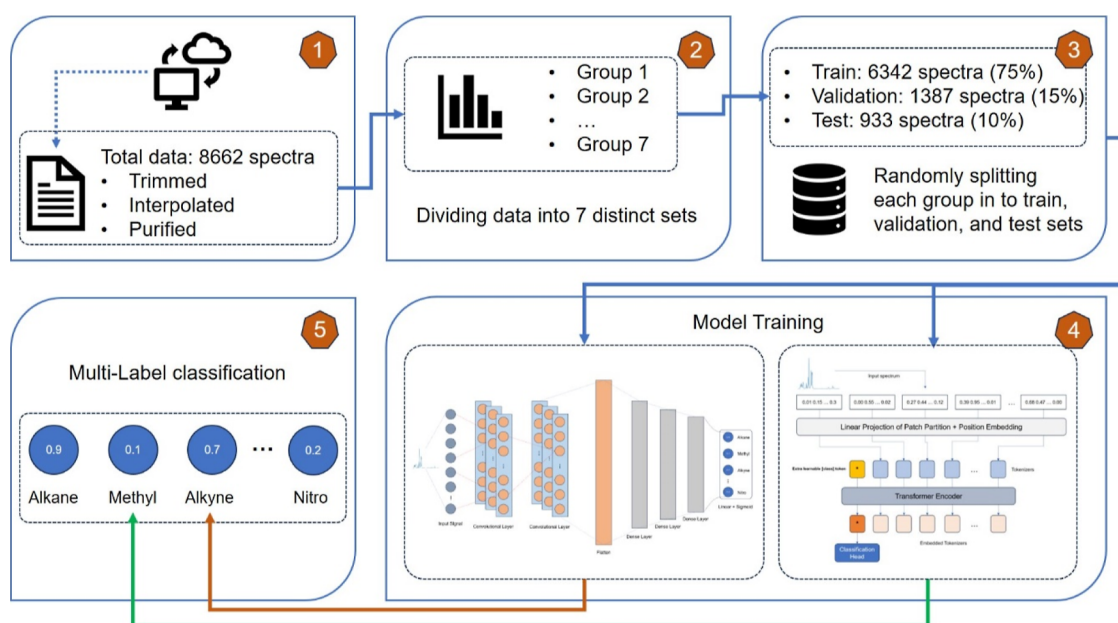
Figure 2. Process for classifying FTIR spectra using two models, one based on convolutional neural network (IRCNN) and the other on transformer architecture.

DL is a subset of machine learning that harnesses neural networks to process data like human brains. These computer models are trained to extract features from raw data and make predictions and classifications. DL models can be trained in chemistry to predict the presence or absence of specific functional groups in chemical compounds based on IR spectra.[18] This is particularly useful when dealing with complex spectra with multiple overlapping peaks.[19] Recently, several studies were reported using DL models to investigate the IR spectra. The most successful and efficient model presented in this field is the convolutional neuron network (CNN) or CNN-based model,[18−27] recognized for its remarkable efficacy in predicting functional groups. However, evaluating these models solely based on functional group prediction precision may lead to an incomplete assessment. It is imperative to also consider the accuracy of the model in predicting entire molecules. This consideration arises from data imbalance issues, where certain functional groups are more prevalent than others within the data set. To illustrate, if a molecule comprises five functional groups and a model accurately predicts four out of the five, conventional accuracy metrics may suggest an 80% success rate. However, from the perspective of predicting the entire molecule, the accuracy would be 0%. This underscores the need for a more comprehensive evaluation framework encompassing functional group prediction precision and compound-level accuracy.

In recent years, transformer models have emerged as the cornerstone of various machine learning applications, revolutionizing the field with their remarkable capabilities in handling diverse data types such as signals, images, speech, and text.[28] The transformer architecture, initially introduced for natural language processing tasks,[29] has showcased exceptional performance across various domains, including translation,[30,31] time series forecasting,[32−34] and signal classification.[35−39] Overview of the existing landscape of machine learning models for functional group characterization, traditional models, such as 1D-CNNs, recurrent neural networks (RNNs),[40] and long short-term memory,[41] have historically dominated functional

group analysis tasks.[19,27,36,42−47] Despite these achievements, there remains a noticeable gap in the literature regarding the application of transformer models to chemical spectra signals, particularly in the functional group characterization data sets.

To explore the potential benefits and challenges associated with adopting transformer approaches in this domain, an attention-based transformer model was utilized for predicting the function groups within IR spectra. The architecture of the transformer model, encompassing 17 multilabel functional groups as inputs, is depicted in Figure 1. The model's performance was evaluated by assessing both the accuracy of functional group predictions and the precision of compound-level predictions.

■ MATERIALS AND METHODS

Data Collection and Functional Groups Assignment. We obtained the FTIR absorbance spectra for all compounds from the National Institute of Standards and Technology (NIST) Chemistry WebBook.[48] These spectra were initially downloaded in the JDX format and subsequently converted to XY files. Finally, all converted spectra were consolidated and stored in a single CSV file, as per the specifications outlined in a species file. We match the ID of each compound to the IUPAC InChi strings by using the PubChem API.[49] Substructure matching was afterward carried out by RDKit on each string to determine whether a predetermined compound topology was present.[50] Each SMARTS string was tested independently and if a match was found, the functional group was classified as belonging to the corresponding compound. Another group of spectra—an external data set—including 17 spectra was downloaded from the Emission Measurement Center Spectral Database. All spectra were processed the same way as those from the NIST Chemistry WebBook.

Framework. Figure 2 depicts the comprehensive FITR functional groups classification flowchart, encompassing the preparation of input spectra, data segmentation into functional groups within a single molecule, training-validation-testing data
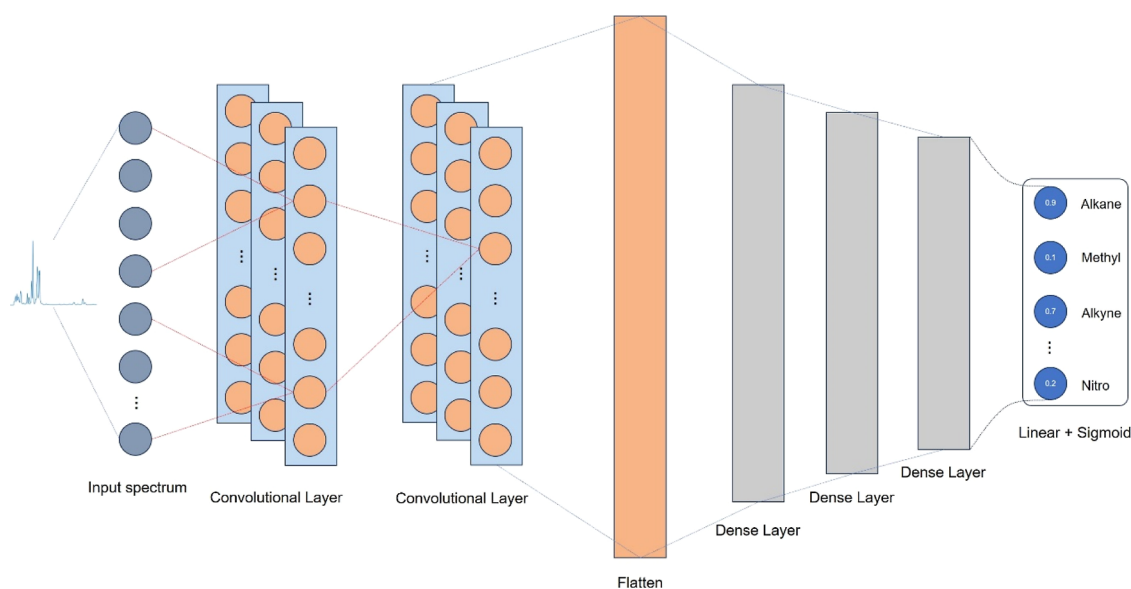
Figure 3. IRCNN architecture.

splicing, utilization of a DL network, optimization of hyperparameters, model comparison, postprocessing, and the deployment of the model. First, the spectra, initially presented in XY format within a CSV file, underwent conversion into a unified 1D-array data set in NumPy format. Each data was then paired with a corresponding text annotation file. To ensure uniformity, all spectra were standardized to a consistent set of feature points (3600 points spanning the range from 400 to 4000 cm^{-1}) by employing the linear interpolation method, as defined in eq 1. The spectra data concerning the number of functional groups that existed in each molecule were also investigated. The maximum number of functional groups in a single molecule is seven, and the distribution of each group reveals an imbalance, as illustrated in Figure S1A−C. Spectra were then segmented into seven groups to address this imbalance. This method ensures that models are trained and evaluated on data sets accurately representing the distribution of functional groups, thereby promoting a more balanced data set regarding the number of functional groups per molecule.[51] Subsequently, all groups were randomly partitioned into training, validation, and test sets with allocation ratios of 75, 15, and 10%, respectively. During the training process, input spectra values were scaled to a range of 0 to 1 using the min−max normalization method, ensuring consistent data scales.

We conducted a multilabel classification with 17 classes of functional groups, assessing the performance of both the CNN-based model[27] (specifically IRCNN, a published model) and our proposed transformer-based model based on the test set. The external data set was used to further evaluate our model's reliability. Two classical machine learning classifiers: decision trees (DTs) and K-nearest neighbors (KNN) were also used to compare with our proposed model.

$$y = y_1 + \frac{(x - x_1)(y_1 - y_2)}{x_2 - x_1} \qquad (1)$$

where x_1, y_1 represent the coordinates of the left position, x_2, y_2 represent the coordinates of the right position, x is the interpolation point and y denotes the interpolated value.

DL Model. *Irchracterization Convolutional Neural Networks.* The irchracterization convolutional neural networks

(IRCNN)[27] is a DL method proposed for identifying functional groups within organic molecules. Unlike other approaches that utilize artificial neural networks, IRCNN employs sliding convolutional filters with a shared-weight architecture across input features, resulting in translational equivariant responses referred to as feature maps. In this research, we chose to reimplement the IRCNN model, which offers a novel spectral interpretation approach. The IRCNN architecture is composed of two convolution blocks, one flattened layer, three dense layers, and one activation layer. We utilized PyTorch to carry out the reimplementation while maintaining all parameters consistent with the original paper, as detailed in both the paper itself and a provided GitHub link at https://github.com/gj475/irchracterizationcnn. Each convolution block consists of a dense convolution layer, batch normalization, ReLU activation, and a max-pooling layer. In the activation layer, unlike typical classification tasks, the spectra signal corresponds to multiple class labels. Consequently, the authors opted for sigmoid activation instead of softmax activation to accommodate these multilabel class labels. The IRCNN architecture is shown in Figure 3.

Transformer Architecture. We propose an approach that directly adapts the full transformer architecture,[29] initially designed as a neural machine translation model. This model is particularly effective for handling sequential data through the aiding of attention mechanisms.[52] The transformer network utilizes fundamental concepts of an encoder-decoder architecture, with each block incorporating simple word embeddings, attention mechanisms, and softmax. It avoids the structural complexities present in RNNs or CNNs. The encoder extracts features from an input sequence, while the decoder utilizes these features to generate an output sequence. In this study, our primary objective is spectra classification. Therefore, we opted to implement only the encoder part, which is designed to learn embeddings suitable for the efficient classification task. The encoder component in transformer architecture plays a vital role in comprehending and extracting relevant information from the input sequence. Over the years, numerous encoder-only architectures have been utilized, drawing inspiration from the encoder module of the original
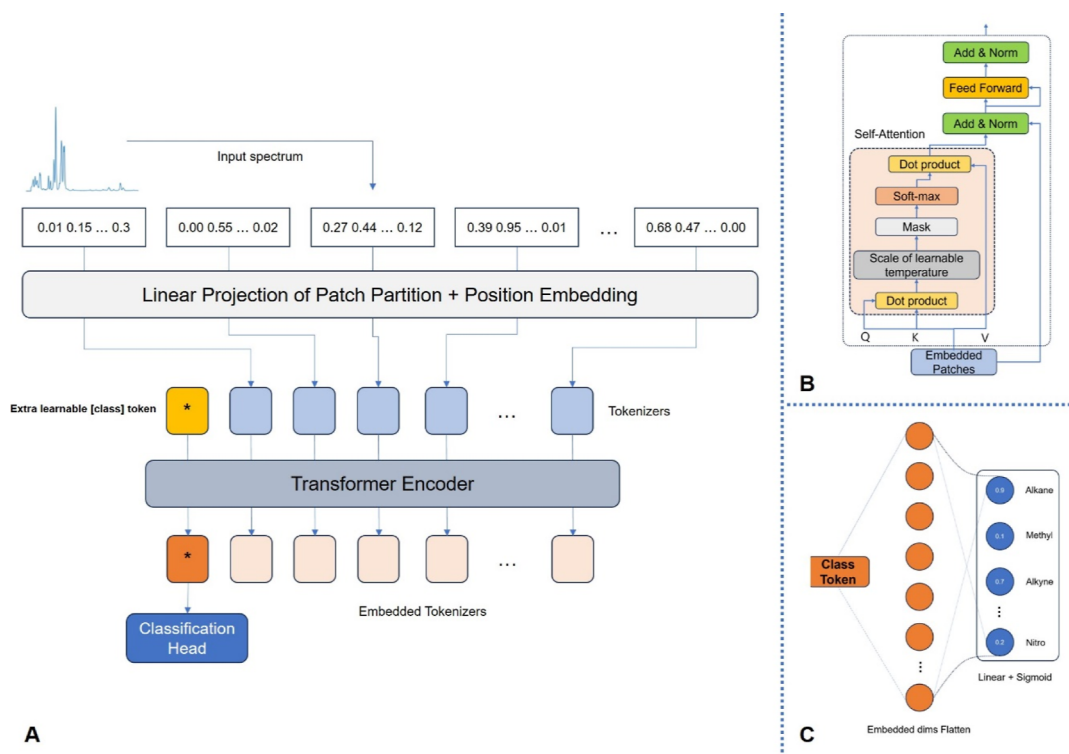
Figure 4. Transformer-based model: (A) Fcg-former architecture; (B) transformer encoder block with the self-attention mechanism; (C) classification head.

transformer model. Examples include BERT (bidirectional encoder representations from transformers),[53] RoBERTa (a Robustly optimized BERT pretraining approach),[54] and ViT (vision transformer).[55] These architectures have been instrumental in advancing various classification domains. Despite the transformer architecture's emergence as state-of-the-art (SOTA) for natural language processing and vision tasks, its application to chemical signals has remained limited. In this study, we introduced a transformer model (Fcg-former) (Figure 4A) inspired by the transformer encoder architecture and the ViT model. By adapting this architecture to process sequences of signal patches, our approach has shown outstanding performance on spectra classification tasks. Leveraging attention mechanisms, Fcg-former achieves remarkable results compared to SOTA convolutional networks trained on the same resources. The overall architecture of Fcg-former is shown in Figure 4.

Self-attention (Figure 4B) is a crucial mechanism utilized in transformer encoders, enabling the model to focus on different parts of the input sequence when processing each element-(token). The self-attention mechanism generates three versions of the input embeddings: queries, keys, and values. These are linear projections of the original embeddings and are used to calculate attention scores. Attention scores, representing the importance or relevance of each patch sequence to the current patch, are computed by taking the dot product of a query with the keys. The softmax function is then applied to the attention scores to convert them into probabilities, ensuring that the attention weights sum up to 1 and indicating the relative importance of each element signal. Following the softmax operation, the model calculates a weighted average of the value vectors associated with all patch sequences. These weights are determined by the softmax probabilities obtained earlier, ensuring that patches deemed more pertinent to the

current signal contribute more significantly to the final output. The resulting vector represents a signal-aware representation of the current patch, considering its relationship with other patch signals in the sequence. By applying self-attention to the spectra signal, the transformer model can capture dependencies between different patches in the input sequence and learn to focus on the most relevant patches for each position, which aids in understanding the signal and improves classification accuracy.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (2)$$

To effectively process the spectral signal data while ensuring consistent input size and facilitating tokenization for subsequent processing, each signal is first resized to a fixed signal length of 1024. Subsequently, the signal is divided into a sequence of fixed-size non-overlapping patches. These patches are then linearly embedded into tokens, which serve as the input to the Fcg-former model. Like BERT and ViT architectures, an additional learnable token known as the [class] token is introduced to act as the representation of the entire input signal. This token is utilized to capture global information and understanding of the signal, which proves beneficial for various tasks such as classification. It typically serves as the input to the classification head located at the output of the transformer encoder block. Each token, including an additional special token [class], is assigned learnable position embeddings. These position embeddings play an essential role in transformer-based architecture, which encodes the positional information on each token within the sequence by a unique sinusoidal extrapolability, allowing the model to understand the relative positions of tokens. However, static (not trained) value does not always perform well, due to the

lack of learnability and flexibility,[56] most pretrained language models[57] utilize learnable (trainable parameters) vector embedding. Subsequently, the sequence of vectors, comprising both the token embeddings and their corresponding position embeddings, is input into a transformer encoder. This encoder processes the input sequence, leveraging self-attention mechanisms to capture dependencies between patches and generate representations for each token in the sequence. A classification head is responsible for producing the final classification output, the class token (*) plays a crucial role in this process (Figure 4C). This additional learnable token is appended to the input sequence and does not correspond to any specific patch of spectrum in the input. Each layer of the transformer encoder processes the tokens, updating their representations. The class token is initialized as a fixed-size and learnable vector, which matches the embedding dimensions of the model. After passing through the final layer of the transformer encoder, the class token holds a comprehensive representation of the entire input sequence. This representation is then used by the classification head to produce the final output. Hyperparameter tuning was conducted using Neural Network Intelligence,[58] aiming to optimize parameters within a transformer architecture, specifically focusing on patch size, layer count, and attention head count. This process goal was to identify the optimal configurations for these parameters that would lead to improved performance or efficiency in the given task or model architecture. The top-performing model, characterized by a signal size of 1024, patch size of 16, 2 layers, an embedded dimension of 768, and 4 attention heads, was chosen for evaluation on an independent test data set.

Training Methods. Both IRCNN and Fcg-former utilized the same training parameters during the evaluation of their performance. This consistency ensures a fair comparison between the two models, as they are trained under similar conditions, allowing for a more accurate assessment of their relative effectiveness in handling the given task or data set. The learning rate was set to 0.002, employing the Adam optimizer algorithm[59] and a cosine annealing warm restarts[60] learning rate scheduler, with the number of iterations set to 600 epochs. To mitigate overfitting, the training code was configured for early stopping if the model's loss on the validation set did not improve for the subsequent 10 patience epochs. Furthermore, the best weights of the model built at each iteration were retained if they achieved the minimum validation loss. Various activation functions are employed in neural networks, with the selection often influenced by the network's architecture and its predictive accuracy. In the context of multilabel classification, we opted for the Sigmoid function. This function is capable of transforming values into a range of 0 to 1 for each class, which could be defined as $\sigma(z) = \frac{1}{1 + e^{-z}}$, which aligns well with the multilabel classification task. For training the model, we utilized the weighted binary cross-entropy loss function (L) (eq 3). This choice has shown superior performance in handling the imbalanced data set in infrared spectra signal classification when using CNN networks.[27]

$$L = -\frac{1}{N}\sum_{i=1}^{N} W_i y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i) \tag{3}$$

where N is the number of classes, W_i, y_i, and $\hat{y}_i$ correspond to weight, the ground truth value, and the predicted value for class i.

To address potential overfitting caused by the limitations of spectral signals, various data augmentation techniques were exclusively applied to the training data set. These techniques encompassed the addition of random noise within a signal-to-noise ratio (SNR) range of 2 to 20 dB (dB), random vertical shifts with a 0.3 probability, and random masking of signal portions with zeros, also with a 0.3 probability. It is crucial to highlight that none of these augmentation processes were extended to the validation and testing data sets, ensuring an unbiased evaluation of the model's performance on unseen data, thereby preserving its generalization capability.

In implementing DL approaches, PyTorch was the framework of choice. The hardware platform employed in this study consisted of a high-performance computer equipped with eight Intel Core i7-12700F processors running at 4.0 GHz, along with a high-speed graphics computing unit NVIDIA GeForce RTX 2060 with 12 GB of graphic memory. The networks were configured using Python 3.9 within an Anaconda environment, with PyTorch 2.0 serving as the backend for model development and training. This setup provided the necessary computational resources and software environment to conduct the experiments effectively.

Evaluation Metrics. Various metrics have been employed to assess the performance of the proposed DL models for functional group prediction. Accuracy serves as a comprehensive measure of the model's correctness, providing an overview of its success rate in identifying functional groups. Precision, particularly crucial in scenarios where false positives are costly, ensures the accuracy and trustworthiness of identified functional groups. In functional group prediction, recall reflects the model's effectiveness in capturing all occurrences of each functional group, thereby ensuring comprehensive coverage and preventing the oversight of critical information. The $F1$-score is a commonly used metric in classification tasks and it considers both precision and recall, providing a balanced measure of a model's performance. In data sets where certain functional groups are more prevalent than others, class imbalance can affect the interpretation of traditional accuracy metrics. The $F1$-score, being based on both precision and recall, is less sensitive to class imbalance and provides a more robust evaluation of model performance in such scenarios. Moreover, the exact match ratio (EMR) evaluates the model's precision in identifying all functional groups within a molecule, offering a strict criterion for performance assessment. EMR is particularly vital in applications necessitating precise identification of functional groups, such as drug discovery or material science. These metrics collectively contribute to the thorough evaluation of the model's efficacy in functional group prediction. The formulas of these metrics (accuracy, precision, recall, $F1$-score, and EMR) are as follows eqs 4−8.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

$$F1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{7}$$
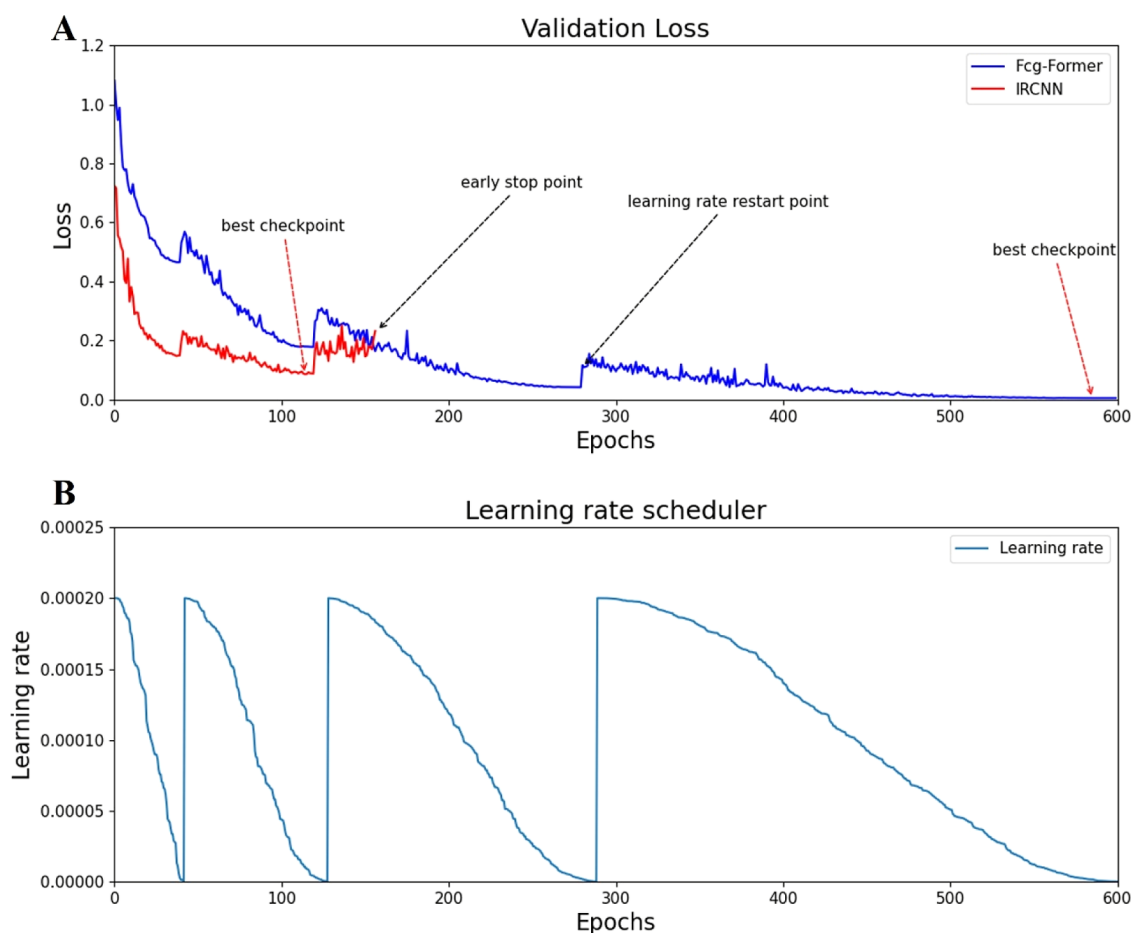
Figure 5. (A)Validation loss during training of IRCNN and Fcg-former; (B) learning rate scheduler.

where TP, TN, FP, FN represent the number of true positive, true negative, false positive, and false negative samples, respectively.

$$\text{EMR} = \frac{1}{n}\sum_{i=1}^{n} I(Y_i = \widehat{Y}_i) \tag{8}$$

where n is the number of testing signals, Y_i and $\widehat{Y}_i$ are true labels and predicted labels for spectral i.

■ RESULTS AND DISCUSSIONS

Training Results. The data set was randomly divided into three subsets: training (75%), validation (15%), and testing (10%), facilitating rigorous investigation into the training and evaluation of the proposed DL models. Following training and hyperparameter tuning, the model underwent validation over 600 epochs. A conventional IRCNN model was trained and validated in parallel with the proposed Fcg-former, enabling direct comparison within the confines of the same data set. As depicted in Figure 5, the loss function of the Fcg-former model exhibits a reliable reduction indicative of optimal convergence, whereas the IRCNN model stops training early under predefined stopping criteria. The optimal epochs for Fcg-former and IRCNN are identified at 585 and 115, respectively.

Prediction of Functional Groups. The ROC curve, PR curve, and the overall functional group confusion matrix are presented in Figure 6. The predictive outcomes of both IRCNN, Fcg-former, DTs, and KNN models on the testing subdata set were calculated based on the confusion matrices

and shown in Table 2. In terms of accuracy, both models demonstrate high performance, with Fcg-former slightly outperforming IRCNN by achieving an accuracy of 0.9715 compared to 0.9613. Both models also demonstrate strong precision values, indicating high accuracy in positive predictions (0.9355 versus 0.9396). However, the Fcg-former model exhibits better recall (0.9227), capturing a higher proportion of actual positive instances in the data set compared to IRCNN (0.8754). The Fcg-former model's improved recall suggests its effectiveness in capturing a broader range of functional groups within IR spectra, potentially due to its attention architecture. The F1-score, a harmonic mean of precision and recall, further confirms the overall superior performance of Fcg-former, with a score of 0.929 compared to 0.9063 for IRCNN. Additionally, Fcg-former demonstrates a higher EMR of 0.702 compared to 0.6249 for IRCNN, indicating its capability to accurately predict all functional groups within a given molecule (Figure 7).

Furthermore, regarding resource management, despite Fcg-former having significantly fewer trainable parameters (6,210,065) compared to IRCNN (61,540,416), it still achieves comparable performance. Also, Fcg-former requires substantially less GPU RAM, with an estimate of 142 MB compared to 1409 MB for IRCNN, making it more memory-efficient. Notably, while IRCNN employs optimal threshold tuning for individual functional groups, resulting in enhanced accuracy evaluation, our study adopts a uniform threshold (0.5) for all functional groups. This approach highlights the
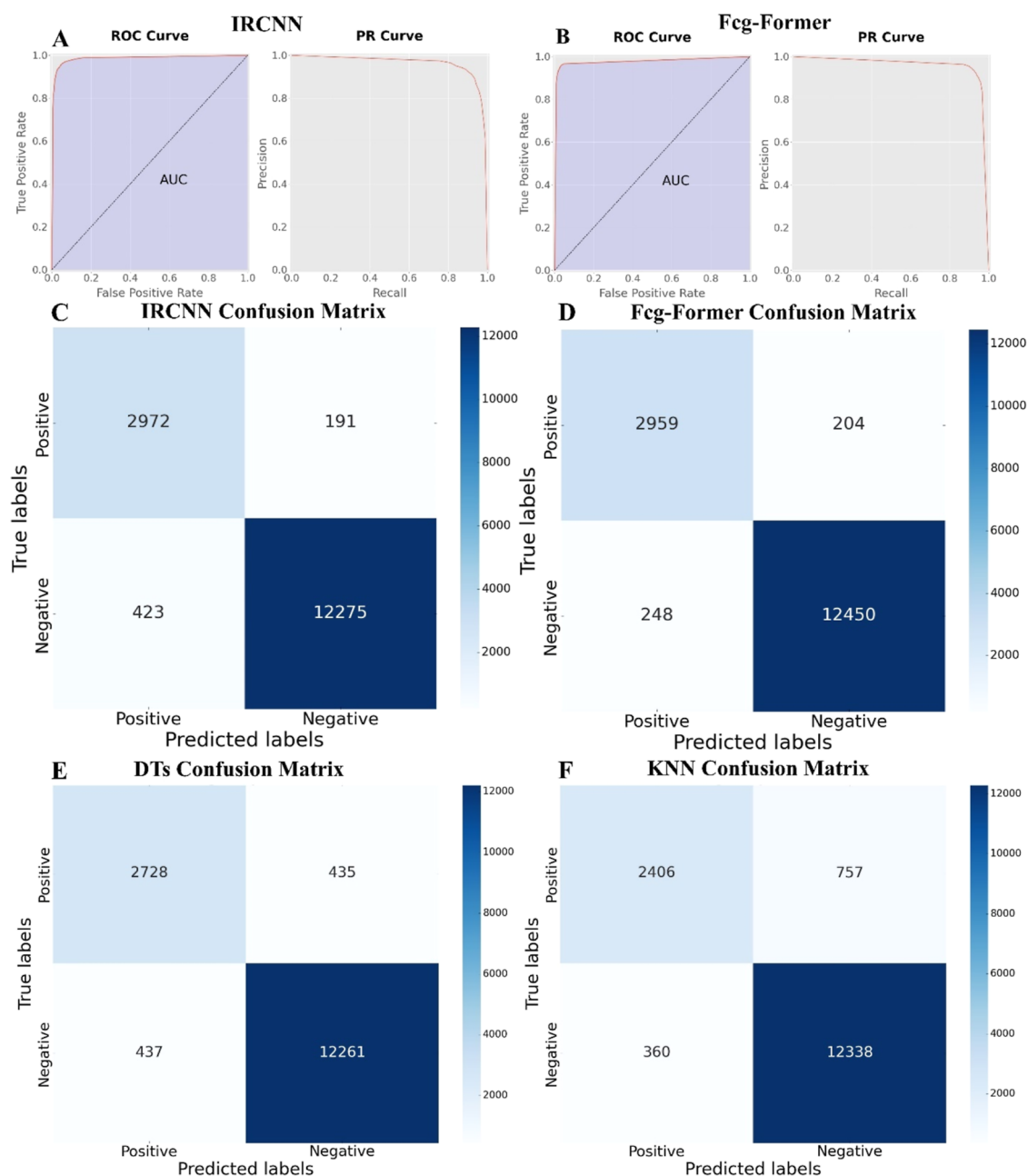
Figure 6. ROC and PR curve of (A) the IRCNN model and (B) the Fcg-former; the functional group confusion matrix of (C) the IRCNN model, (D) the Fcg-former model, (E) the DTs model, and (F) the KNN model performed on the test data set.

Table 1. Self-Attention Map Calculations

self-attention map calculation
input: input spectrum
output: self-attention map
for each patch in sequence:
for each attention head:
calculate query, key, and value for the current patch
calculate attention scores between the current patch and all other patches
apply softmax to obtain attention weights
store attention weights for each patch
calculate mean attention scores across all attention heads
represent mean attention scores as a color map

reliability of our proposed Fcg-former attention model in functional group prediction. Figures S2 and S3 show the confusion matrix of individual functional group prediction results of both models, revealing similarities to their overall performance, and confirming the enhanced performance of Fcg-former over IRCNN in predicting functional groups within IR spectra analysis.

The conventional machine learning techniques DTs and KNN also exhibit performance characteristics in the context of functional group prediction. The functional group confusion matrices for DTs and KNN are depicted in Figures 6E,F, S4, and S5, while their compound-level confusion matrices are illustrated in Figures 7C and 7D. DTs achieve an accuracy of 0.945, a recall of 0.8619, and a precision of 0.8625, resulting in an F1-score of 0.8622. While DTs offer a balance between

Table 2. IRCNN, Fcg-Former, DTs, and KNN Performance Summary on Test Data

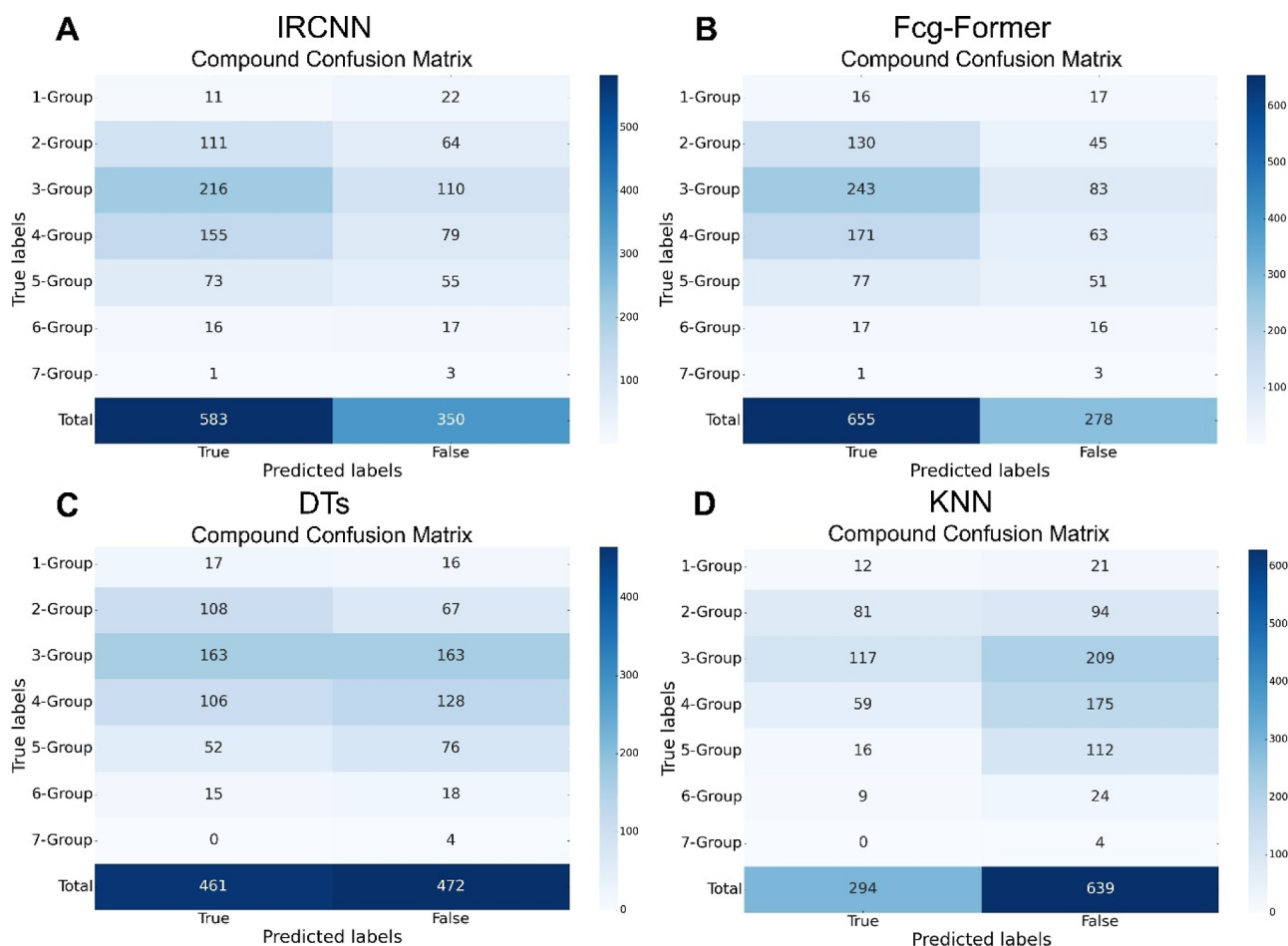|	accuracy	recall	precision	$F1$-score	EMR	parameters	GPU ram
IRCNN	0.9613	0.8754	0.9396	0.9063	0.6249	61,540,416	1409 MiB
Fcg-former	0.9715	0.9227	0.9355	0.929	0.702	6,210,065	142 MiB
DTs	0.945	0.8619	0.8625	0.8622	0.4941		
KNN	0.9296	0.8698	0.7607	0.8116	0.3151		

Figure 7. Compound-level functional group confusion matrix of (A) the IRCNN model, (B) the Fcg-former model, (C) the DTs model, and (D) the KNN model performed on the test data set.

precision and recall, their EMR is relatively lower at 0.4941, indicating moderate effectiveness in predicting all functional groups within a molecule correctly. On the other hand, KNN shows an accuracy of 0.9296, with a recall of 0.8698 and a lower precision of 0.7607, leading to an $F1$-score of 0.8116. The KNN model has an even lower EMR of 0.3151, suggesting it struggles more with accurately identifying the full set of functional groups. Overall, the DL models, IRCNN and Fcg-former, significantly outperformed the classical machine learning approaches, DTs, and KNN, particularly regarding the EMR. This highlights the superior ability of DL techniques to handle compound-level predictions, demonstrating a clear advantage over traditional machine learning methods in capturing the complexity of functional group identification.

The robustness of our models is evident in their performance on both the initial test data set and the external data set. As shown in Figures S6, S7, and Table S1, while there is a slight decrease in performance metrics when evaluated on the external data set, the Fcg-former model consistently shows better performance, indicating its stability and reduced

likelihood of overfitting compared to the classical machine learning models. On the initial test data set, the Fcg-former achieved the highest accuracy (0.9715) and EMR (0.702), outperforming IRCNN, DTs, and KNN. On the external data set, the Fcg-former maintained its superior performance with an accuracy of 0.9585 and an EMR of 0.6471, demonstrating its ability to generalize well to new data. The results indicate that the DL models, particularly Fcg-former, exhibit robust generalization capabilities without significant overfitting, especially in dealing with complex compound-level predictions. We believe these measures address the concern regarding data set independence and provide a comprehensive assessment of our models' performance.

Self-Attention Map in Functional Group Prediction. Figure 8 depicts an example of the attention map generated during the processing of IR spectra and its corresponding outputs. In the calculation process, attention scores are computed for each patch in a sequence by comparing it to all other patches. This is achieved by calculating the dot product between the Query of the current patch and the Key
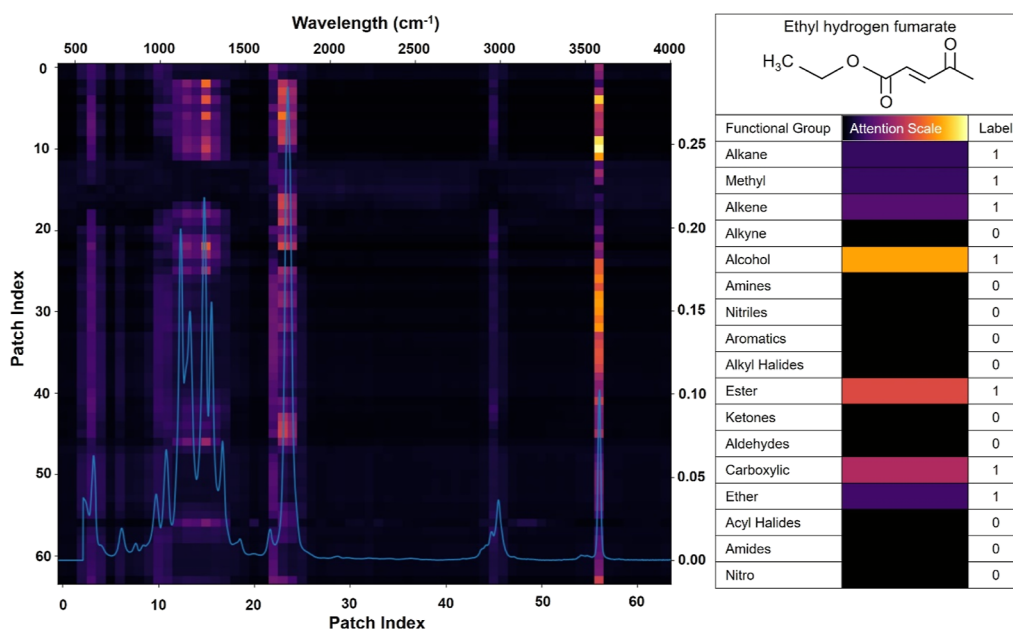
Figure 8. Self-Attention mechanism works on ethyl hydrogen fumarate compound. Each cell (patch index) in the figure reflects how attention heads distribute their attention across different parts of the input. This visualization helps understand which patches receive more focus from specific attention heads during the model's processing.
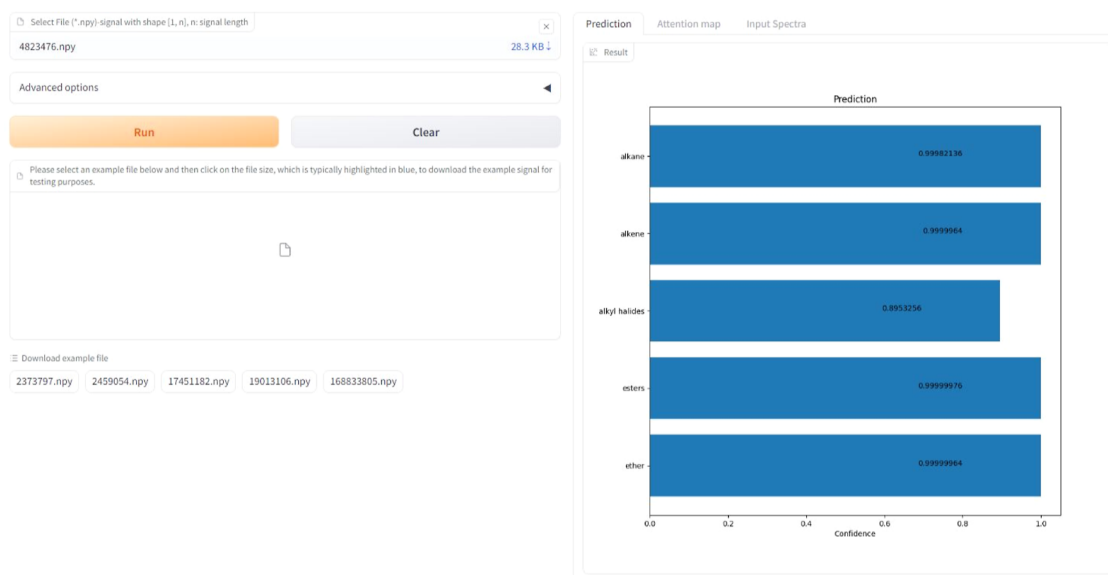

Figure 9. Functional group prediction result performed on the web-based application.

of every other patch, followed by a softmax activation (as depicted in eq 2). These attention scores, calculated for each head, provide insight into the significance of different patches with one another (Table 1). Upon examination, the attention model demonstrates a notable focus on relevant peaks within the spectra. The attention model accurately identifies the strong absorption bands associated with the alcohol functional group, particularly at 3600 cm^{-1}. However, for groups such as alkane and methyl, whose absorption bands overlap within the 2800−3000 cm^{-1} range, the attention transformer detects these features less prominently, reflecting their weaker signals. Moreover, the prominent peak observed at 1700−1750 cm^{-1}, revealing ester stretching vibrations and potentially carboxylic acid groups, receives significant attention from the model.

During the training phase, the attention transformer algorithm learns and defines the bonding interactions among various functional groups. This acquired knowledge allows the model to predict the potential presence of specific functional groups within the unknown molecules.

Deployment of Fcg-Former. Fcg-former is an open-source library dedicated to making strides in chemical signal research accessible to the wider machine-learning community. It offers meticulously designed FcgFormer architectures through a unified API. Fcg-former emphasizes extensibility for researchers, simplicity for practitioners, and efficiency and reliability for tasks like fine-tuning and deployment.

Additionally, users can access the library and its associated Hugging Face application, powered by Gradio, at https://

huggingface.co/spaces/lycaoduong/FcgFormerApp. The HuggingFace App prediction result is shown in Figure 9. Other examples of our model deployment were demonstrated in Figures S8–S11.

CONCLUSION

In conclusion, this study presents a novel approach utilizing a transformer attention model for the prediction of functional groups in FTIR spectra. Our findings underscore the importance of exploring cutting-edge DL techniques in spectroscopy, paving the way for future research avenues aimed at enhancing spectral analysis and interpretation. As the field continues to evolve, integrating transformer-based models into analytical workflows could lead to significant advancements in compound characterization and identification. Our model demonstrates better performance compared to conventional CNN architectures, both in terms of functional group prediction accuracy (0.9715 over 0.9613) and compound-level accuracy (0.702 over 0.6249). The success of our transformer attention model highlights the efficacy of self-attention mechanisms in capturing intricate spectral patterns and relationships, thus enabling more accurate predictions. Overall, this work contributes to the ongoing convergence of artificial intelligence and spectroscopic analysis, offering a robust framework for accurate and efficient functional group prediction in FTIR spectra.

ASSOCIATED CONTENT

Data Availability Statement

The data sets used in this study are available from the NIST Chemistry WebBook https://webbook.nist.gov and Emission Measurement Center Spectral Database https://www3.epa.gov/ttn/emc/ftir/refnam.html. The Python code implementation of Fcg-former, model checkpoints, and all data sets used in this study are available on GitHub at https://github.com/lycaoduong/FcgFormer.

ⓢ Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.analchem.4c01622.

> The data set distribution regarding the number of functional groups within each molecule. The IRCNN, Fcg-former, DTs, and KNN performance summary on the external data set. The confusion matrix of each functional group from IRCNN, Fcg-former, DTs, and KNN models performed on the test data set. The functional group and compound-level confusion matrix of IRCNN, Fcg-former, DTs, and KNN models performed on the external data set. Examples of model deployment. Please refer to Supporting Information for additional details (PDF)

AUTHOR INFORMATION

Corresponding Authors

Byeongil Lee − *Digital Healthcare Research Center and Industry 4.0 Convergence Bionics Engineering, Department of Biomedical Engineering, Pukyong National University, Busan 48513, Republic of Korea*; Email: bilee@pknu.ac.kr

Junghwan Oh − *Smart Gym-Based Translational Research Center for Active Senior's Healthcare, Pukyong National University, Busan 48513, Republic of Korea; Digital Healthcare Research Center and Industry 4.0 Convergence Bionics Engineering, Department of Biomedical Engineering,*

Pukyong National University, Busan 48513, Republic of Korea; Ohlabs Corp., Busan 48513, Republic of Korea; ⓞ orcid.org/0000-0002-5837-0958; Phone: +82-51-629-5771; Email: jungoh@pknu.ac.kr; Fax: +82-51-629-5779

Authors

Vu Hoang Minh Doan − *Smart Gym-Based Translational Research Center for Active Senior's Healthcare, Pukyong National University, Busan 48513, Republic of Korea*

Cao Duong Ly − *Research and Development Department, Senior AI Research Engineer, Vision-in Inc., Seoul 08505, Republic of Korea*

Sudip Mondal − *Digital Healthcare Research Center, Pukyong National University, Busan 48513, Republic of Korea;* ⓞ orcid.org/0000-0002-0638-9657

Thi Thuy Truong − *Industry 4.0 Convergence Bionics Engineering, Department of Biomedical Engineering, Pukyong National University, Busan 48513, Republic of Korea*

Tan Dung Nguyen − *Industry 4.0 Convergence Bionics Engineering, Department of Biomedical Engineering, Pukyong National University, Busan 48513, Republic of Korea*

Jaeyeop Choi − *Smart Gym-Based Translational Research Center for Active Senior's Healthcare, Pukyong National University, Busan 48513, Republic of Korea*

Complete contact information is available at:
https://pubs.acs.org/10.1021/acs.analchem.4c01622

Author Contributions

#V.H.M.D., C.D.L. and S.M. contributed equally to this work.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

REFERENCES

(1) Struyf, J. J. *J. Chem. Educ.* **2011**, *88* (7), 937−943.

(2) Hong, S.; Shen, X.-J.; Xue, Z.; Sun, Z.; Yuan, T. Q. *Green Chem.* **2020**, *22* (21), 7219−7232.

(3) Naqvi, S. T. R.; Rasheed, T.; Hussain, D.; ul Haq, M. N.; Majeed, S.; Ahmed, N.; Nawaz, R. J. *J. Mol. Liq.* **2020**, *297*, 111919.

(4) Rothfuss, N. E.; Petters, M. D. *Environ. Sci. Technol.* **2017**, *51* (1), 271−279.

(5) Pellenz, L.; de Oliveira, C. R. S.; da Silva Júnior, A. H.; da Silva, L. J. S.; da Silva, L.; de Souza, A. A. U.; de Souza, S. M. d. A. G. U.; Borba, F. H.; da Silva, A. *Sep. Purif. Technol.* **2023**, *305*, 122435.

(6) Ertl, P.; Altmann, E.; McKenna, J. M. *J. Med. Chem.* **2020**, *63* (15), 8408−8418.

(7) Wang, Y.; Wu, C. *Biomacromolecules* **2018**, *19* (6), 1804−1825.

(8) Perovic, M.; Qin, Q.; Oschatz, M. *Adv. Funct. Mater.* **2020**, *30* (41), 1908371.

(9) George, A.; Sanjay, M.; Srisuk, R.; Parameswaranpillai, J.; Siengchin, S. *Int. J. Biol. Macromol.* **2020**, *154*, 329−338.

(10) Bhong, M.; Khan, T. K.; Devade, K.; Krishna, B. V.; Sura, S.; Eftikhaar, H.; Thethi, H. P.; Gupta, N. *Mater. Today: Proc.* **2023**.

(11) Mäder, P.; Kattner, L. J. *J. Med. Chem.* **2020**, *63* (23), 14243−14275.

(12) Baker, M. J.; Trevisan, J.; Bassan, P.; Bhargava, R.; Butler, H. J.; Dorling, K. M.; Fielden, P. R.; Fogarty, S. W.; Fullwood, N. J.; Heys, K. A.; et al. *Nat. Protoc.* **2014**, *9* (8), 1771−1791.

(13) Dong, Y.; Zhang, X.; Chen, L.; Meng, W.; Wang, C.; Cheng, Z.; Liang, H.; Wang, F. *Renew. Sustain. Energy Rev.* **2023**, *188*, 113801.

(14) Rhodes, C. J.; Macrae, R. M. *Sci. Prog.* **2015**, *98* (1), 12−33.

(15) Smith, B. C. *Infrared Spectral Interpretation: A Systematic Approach*; CRC Press, 2018.

(16) Cremer, D.; Kraka, E. *Curr. Org. Chem.* **2010**, *14* (15), 1524−1560.

(17) Bacsik, Z.; Mink, J.; Keresztury, G. *Appl. Spectrosc. Rev.* **2004**, *39* (3), 295−363.

(18) Fine, J. A.; Rajasekar, A. A.; Jethava, K. P.; Chopra, G. *Chem. Sci.* **2020**, *11* (18), 4618−4630.

(19) Wang, T.; Tan, Y.; Chen, Y. Z.; Tan, C. *J. Chem. Inf. Model.* **2023**, *63* (15), 4615−4622.

(20) Acquarelli, J.; van Laarhoven, T.; Gerretzen, J.; Tran, T. N.; Buydens, L. M.; Marchiori, E. *Anal. Chim. Acta* **2017**, *954*, 22−31.

(21) Yuanyuan, C.; Zhibin, W. *Chemom. Intell. Lab. Syst.* **2018**, *181*, 1−10.

(22) Ng, W.; Minasny, B.; Montazerolghaem, M.; Padarian, J.; Ferguson, R.; Bailey, S.; McBratney, A. B. *Geoderma* **2019**, *352*, 251−267.

(23) Chen, Y. Y.; Wang, Z. B. *J. Chemom.* **2019**, *33* (5), No. e3122.

(24) McCarthy, M.; Lee, K. L. K. *J. Phys. Chem. A* **2020**, *124* (15), 3002−3017.

(25) Enders, A. A.; North, N. M.; Fensore, C. M.; Velez-Alvarez, J.; Allen, H. C. *Anal. Chem.* **2021**, *93* (28), 9711−9718.

(26) Awotunde, O.; Roseboom, N.; Cai, J.; Hayes, K.; Rajane, R.; Chen, R.; Yusuf, A.; Lieberman, M. *Anal. Chem.* **2022**, *94* (37), 12586−12594.

(27) Jung, G.; Jung, S. G.; Cole, J. M. *Chem. Sci.* **2023**, *14* (13), 3600−3609.

(28) Lin, T.; Wang, Y.; Liu, X.; Qiu, X. *AI Open* **2022**, *3*, 111−132.

(29) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. *arXiv* **2017**, arXiv:1706.03762.

(30) Ott, M.; Edunov, S.; Baevski, A.; Fan, A.; Gross, S.; Ng, N.; Grangier, D.; Auli, M. *arXiv* **2019**, arXiv:1904.01038.

(31) Wang, Q.; Li, B.; Xiao, T.; Zhu, J.; Li, C.; Wong, D. F.; Chao, L. S. *arXiv* **2019**, arXiv:1906.01787.

(32) Li, S.; Jin, X.; Xuan, Y.; Zhou, X.; Chen, W.; Wang, Y.-X.; Yan, X. *arXiv* **2019**, arXiv:1907.00235.

(33) Wu, N.; Green, B.; Ben, X.; O'Banion, S. *arXiv* **2020**, arXiv:2001.08317.

(34) Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, 2021, pp 11106−11115..

(35) Jin, C.-c.; Chen, X. *Expert Syst. Appl.* **2021**, *171*, 114570.

(36) Che, C.; Zhang, P.; Zhu, M.; Qu, Y.; Jin, B. *BMC Med. Inf. Decis. Making* **2021**, *21* (1), 184.

(37) Sun, J.; Xie, J.; Zhou, H. EEG classification with transformer-based models. In *2021 IEEE 3rd Global Conference On Life Sciences And Technologies (Lifetech)*; IEEE, 2021, pp 92−93.

(38) Cai, J.; Gan, F.; Cao, X.; Liu, W. *IEEE Trans. Cogn. Commun. Netw.* **2022**, *8* (3), 1348−1357.

(39) Xue, R.; Bai, X.; Cao, X.; Zhou, F. *IEEE Trans. Geosci. Rem. Sens.* **2022**, *60*, 5111411.

(40) Grossberg, S. *Scholarpedia* **2013**, *8* (2), 1888.

(41) Graves, A. Long short-term memory. In *Studies in Computational Intelligence*; Springer, 2012; pp 37−45..

(42) Lin, S.; Runger, G. C. *IEEE Transact. Neural Networks Learn. Syst.* **2018**, *29* (10), 4709−4718.

(43) Cueva, C. J.; Wei, X.-X. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. In *International Conference on Learning Representations (ICLR)*, 2018.

(44) Agar, J. C.; Naul, B.; Pandya, S.; van Der Walt, S.; Maher, J.; Ren, Y.; Chen, L.-Q.; Kalinin, S. V.; Vasudevan, R. K.; Cao, Y.; et al. *Nat. Commun.* **2019**, *10* (1), 4809.

(45) Tang, W.; Long, G.; Liu, L.; Zhou, T.; Jiang, J.; Blumenstein, M. Rethinking 1d-cnn for time series classification: A stronger baseline. In *The Tenth International Conference on Learning Representations (ICLR 2022)*, 2020, pp 1−7.

(46) Yu, G.; Ma, B.; Chen, J.; Li, X.; Li, Y.; Li, C. *J. Food Process. Eng.* **2021**, *44* (1), No. e13602.

(47) Yoo, S.-H.; Huang, G.; Hong, K.-S. *Bioengineering* **2023**, *10* (6), 685.

(48) Stein, S.; Linstrom, P.; Mallard, W.; Gaithersburg, T. *NIST Chemistry WebBook; NIST Standard Reference Database Number 69*, 2005, p 20899..

(49) Kim, S.; Thiessen, P. A.; Bolton, E. E.; Chen, J.; Fu, G.; Gindulyte, A.; Han, L.; He, J.; He, S.; Shoemaker, B. A.; et al. *Nucleic Acids Res.* **2016**, *44* (D1), D1202−D1213.

(50) Landrum, G. *RDKit: Open-Source Cheminformatics*, 2006.

(51) Pala, A.; Oleynik, A.; Utseth, I.; Handegard, N. O. *ICES J. Mar. Sci.* **2023**, *80* (10), 2530−2544.

(52) Niu, Z.; Zhong, G.; Yu, H. *Neurocomputing* **2021**, *452*, 48−62.

(53) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. *arXiv* **2018**, arXiv:1810.04805.

(54) Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. *arXiv* **2019**, arXiv:1907.11692.

(55) Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. *arXiv* **2020**, arXiv:2010.11929.

(56) Wang, G.; Lu, Y.; Cui, L.; Lv, T.; Florencio, D.; Zhang, C. A simple yet effective learnable positional encoding method for improving document transformer model *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, 2022, pp 453−463.

(57) Gehring, J.; Auli, M.; Grangier, D.; Yarats, D.; Dauphin, Y. N. Convolutional sequence to sequence learning. In *International conference on machine learning*; PMLR, 2017, pp 1243−1252.

(58) Liu, X.; Wang, Y.; Ji, J.; Cheng, H.; Zhu, X.; Awa, E.; He, P.; Chen, W.; Poon, H.; Cao, G. *arXiv* **2020**, arXiv:2002.07972.

(59) Kingma, D. P.; Ba, J. *arXiv* **2014**, arXiv:1412.6980.

(60) Loshchilov, I.; Hutter, F. *arXiv* **2016**, arXiv:1608.03983.