

GraphATC: advancing multilevel and multi-label anatomical therapeutic chemical classification via atom-level graph learning

Wengyu Zhang^{1,2,†}, Qi Tian^{1,†}, Yi Cao¹, Wenqi Fan², Dongmei Jiang³, Yaowei Wang³, Qing Li², Xiao-Yong Wei^{1,2,*}

¹Department of Computer Science, Sichuan University, Chengdu 610065, China

²Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong

³Peng Cheng Laboratory, Shenzhen 518000, China

*Corresponding author. Department of Computer Science, Sichuan University, Chengdu 610065, China, and Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong. E-mail: x1wei@polyu.edu.hk

†Wengyu Zhang and Qi Tian have contributed equally to this work and share first authorship.

Abstract

The accurate categorization of compounds within the anatomical therapeutic chemical (ATC) system is fundamental for drug development and fundamental research. Although this area has garnered significant research focus for over a decade, the majority of prior studies have concentrated solely on the Level 1 labels defined by the World Health Organization (WHO), neglecting the labels of the remaining four levels. This narrow focus fails to address the true nature of the task as a multilevel, multi-label classification challenge. Moreover, existing benchmarks like Chen-2012 and ATC-SMILES have become outdated, lacking the incorporation of new drugs or updated properties of existing ones that have emerged in recent years and have been integrated into the WHO ATC system. To tackle these shortcomings, we present a comprehensive approach in this paper. Firstly, we systematically cleanse and enhance the drug dataset, expanding it to encompass all five levels through a rigorous cross-resource validation process involving KEGG, PubChem, ChEMBL, ChemSpider, and ChemicalBook. This effort culminates in the creation of a novel benchmark termed ATC-GRAPH. Secondly, we extend the classification task to encompass Level 2 and introduce graph-based learning techniques to provide more accurate representations of drug molecular structures. This approach not only facilitates the modeling of Polymers, Macromolecules, and Multi-Component drugs more precisely but also enhances the overall fidelity of the classification process. The efficacy of our proposed framework is validated through extensive experiments, establishing a new state-of-the-art methodology. To facilitate the replication of this study, we have made the benchmark dataset, source code, and web server openly accessible.

Keywords: anatomical therapeutic chemical; graph learning; molecular structures; polymers; macromolecules; multi-component drugs

Introduction

Identification of compounds into the anatomical therapeutic chemical (ATC) system is crucial for drug development and basic research. It has been researched for over a decade since its initial proposal by Dunkel et al. [1] in 2008. For a given compound, the task is to identify its ATC codes that are associated with its therapeutic, pharmacological, and chemical properties. One of the most significant advantages of ATC identification is that the properties of a new drug can be pre-assessed before actual development, which can save resources that would otherwise be spent on drugs without the desired properties. The ATC system developed by the World Health Organization (WHO) is commonly adopted for this purpose (https://www.whocc.no/atc/structure_and_principles/). It is a hierarchical system consisting of five levels: L1 for main anatomical/pharmacological groups, L2 for pharmacological or therapeutic groups, L3 and L4 for chemical, pharmacological or therapeutic subgroups, and L5 for chemical substances. At every level, medications exhibit numerous labels, transforming this into a multilevel, multi-label task that poses

a substantial challenge to conventional supervised learning algorithms.

The complexity emerges from the hierarchical nature of the classification system, which expands into a greater number of categories at the L2, L3, L4, and L5 levels, while the count of drugs diminishes accordingly. This results in a scarcity of samples available for supervised learning. Therefore, most current research endeavors opt for a simplified scheme, focusing solely on the first level (i.e. L1) of ATC codes [1–20]. Certain studies have explored a compromised approach that converts the challenge of predicting ATC codes into predicting drug-code relationships [21–29]. Within these methodologies, a drug is paired with a chosen ATC code and inputted into the model to predict the presence of a connection between the drug and a designated label [23–25, 27–29]. In technical terms, this compromised approach works by enhancing the training dataset with “negative” pairs, comprising a drug paired with any code that it does not possess. Nevertheless, models trained on such imbalanced datasets are inclined to prioritize predicting “non-relationships” rather than accurately pinpointing

Received: December 03, 2024. Revised: March 06, 2025. Accepted: April 07, 2025

© The Author(s) 2025. Published by Oxford University Press.

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<https://creativecommons.org/licenses/by-nc/4.0/>), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact journals.permissions@oup.com

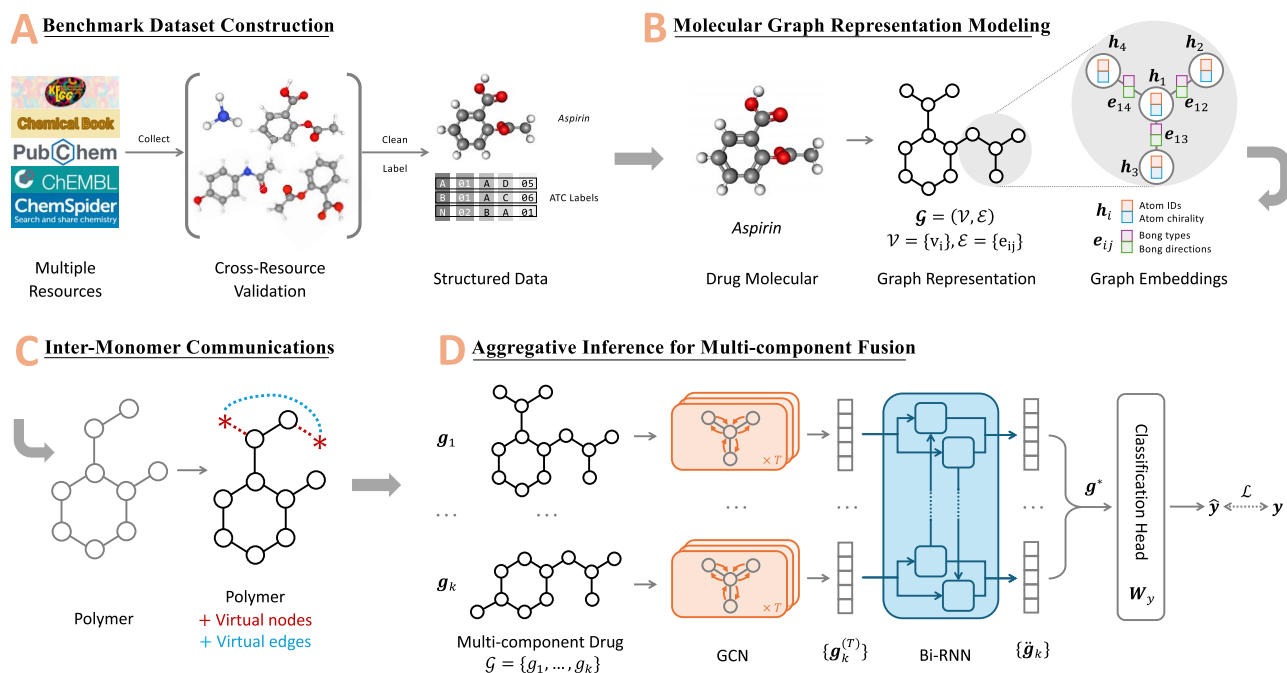


Figure 1. Overall framework of the proposed GraphATC.

the ATC properties associated with a specific drug. Consequently, this method falls short of tackling the multilevel, multi-label challenge effectively.

The effectiveness of ATC classification is typically determined by the model employed and the drug representations utilized. In recent years, ATC models have transitioned from traditional machine learning algorithms such as ML-GKR [5, 6], LIFT [7, 8], SVM [13, 14], logistic regression [25], naive Bayes [26], and random forests [26] toward more sophisticated deep learning techniques like DNN [12], LSTM [11, 19], CNN [8, 11, 16, 27], and Text-CNN [20]. The prevailing consensus suggests that deep models are notably more effective. Regarding representations, substantial effort has been dedicated to enhancing them by incorporating extra physicochemical properties alongside molecular fingerprints. These additions encompass chemical-chemical interactions [2, 5–8, 10–13, 16, 19], compound descriptions from Wikipedia [16], structural similarities [2, 5–8, 10–13, 16, 19, 20], and chemical ontology [6, 11, 12, 19]. However, incorporating these elements necessitates additional resources like STITCH [30] and tools such as RDKit [31], SIMCOMP [32], and SUBCOMP [32]. Moreover, the accessibility of these supplementary properties relies on clinical or laboratory experiments, making it less feasible for newly developed drugs.

To address this challenge, a recent study by Wei et al. [20] demonstrated that the state-of-the-art (SOTA) performance levels can be achieved solely by utilizing compound structure information as input, effectively reducing the dependency on additional resources. Nevertheless, this underscores the necessity for more elaborate representations of compound structures beyond basic fingerprints [1] and Simplified Molecular Input Line Entry System (SMILES)-based [33] sequential embeddings [34]. Graph-based techniques like graph convolutional network (GCN) naturally align with this requirement since molecular structures inherently form graphs, a dimension that has been underexplored in ATC endeavors (Previous works [16, 17, 35, 36] have utilized graph neural networks, but they constructed graphs at the drug level for modeling the inter-drug relationship rather than at the atom

level for modeling the molecular structure of drugs.) In this study, we propose the GraphATC framework as an initial effort to bridge this gap by customizing atom-level graph construction and message passing. We illustrate that these enhanced representations can be applied to L2 ATC tasks, offering advantages in handling polymers, macromolecules, and multicomponent drugs that have not been extensively investigated before. The framework of the proposed approach is shown in Fig. 1. Our contributions include the following:

- **We have constructed the most extensive ATC dataset to date.** We have expanded the preexisting ATC datasets from an initial scale of 3883 [2] to 5311 entries. All compounds have undergone cleaning of their mol files [37] by cross-validating with multiple resources such as KEGG [38], PubChem [39], and ChEMBL [40]. This results in a dataset that encompasses greater diversity, including a broader range of polymers, macromolecules, and multicomponent drugs that have not been extensively explored before (see statistics in Table 1 and Fig. 2).
- **We implement the multilevel, multi-label study by extending the task to Level-2 (i.e. L2).** Prior research has predominantly concentrated on the 14 primary groups (classes) of L1 within the WHO ATC system. Expanding the focus beyond these L1 classes to L2 would escalate the scale of the challenge from tens to potentially hundreds or even thousands. The subdivision of classes into finer categories results in limited data availability for certain minor classes, intensifying the learning difficulty. For instance, widely utilized benchmarks like Chen-2012 [2] and ATC-SMILES [20] encompass thousands of samples, but transitioning to L2, which comprises 94 classes, reduces the number of training samples to only a few dozen (Fig. 2c). In this study, we introduce a molecular graph-based approach designed to enhance representation learning and tackle the few-shot learning issue, marking an initial step toward extending the ATC task to L2.

Table 1. Comparison of ATC Benchmark Datasets: Chen-2012, ATC-SMILES, and ATC-GRAPH (Ours)

	Dataset	Chen-2012	ATC-SMILES	ATC-GRAPH
Group by Polymer	Year	2012	2022	2024
	Non-Poly	3852	4545	5259
	Polymer	23	0	52
Mass	Small	3715	4353	4822
	Macro	160	192	489
	Single	2275	2685	2931
#Comp	Multiple	1600	1860	2380
	Total	3883	4545	5311
	Coverage	67.84%	79.40%	92.78%

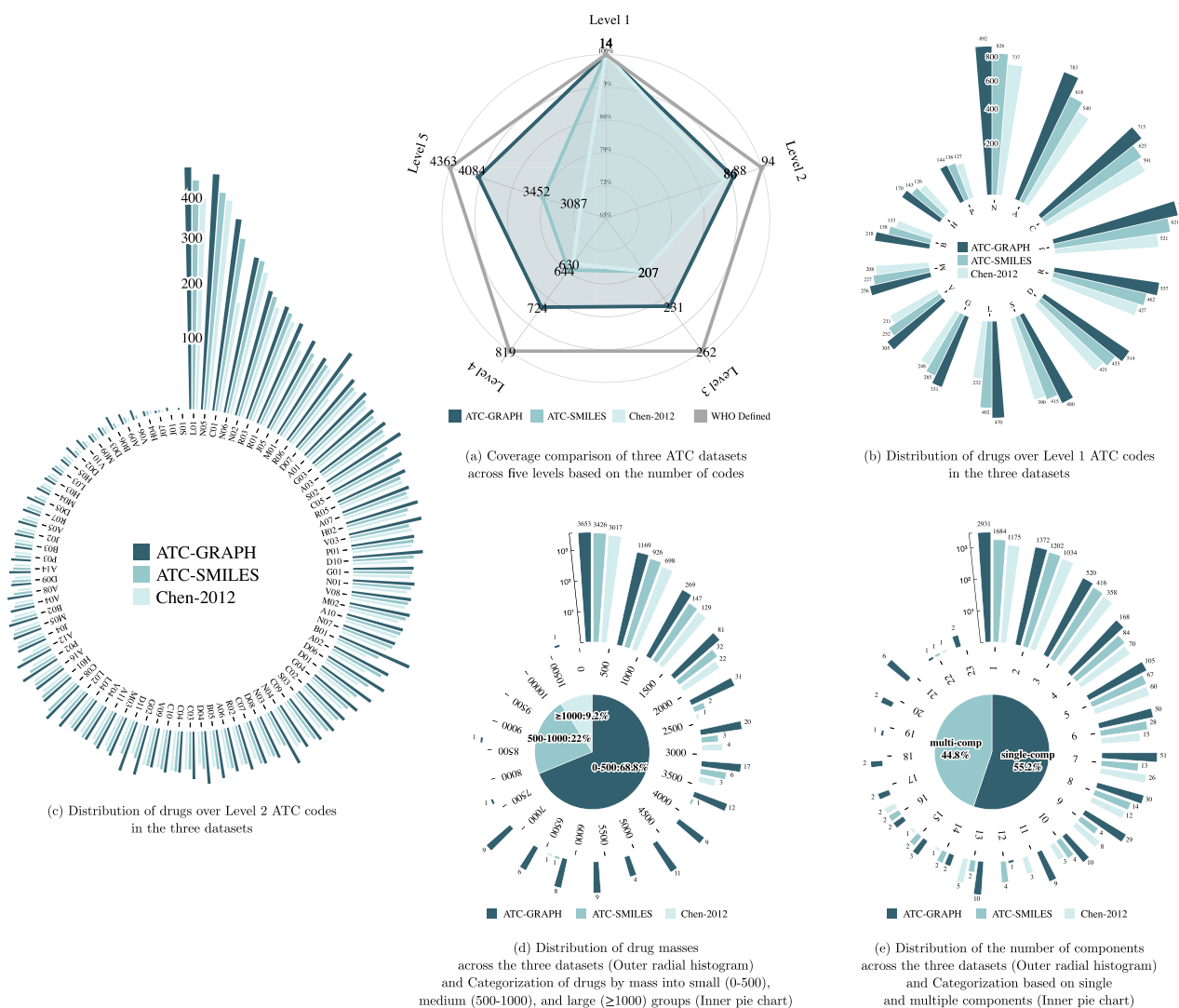


Figure 2. Comparative statistics of ATC-GRAPH versus Chen-2012 and ATC-SMILES, where ATC-GRAPH exhibits the most extensive coverage across levels, mass, and component quantities.

- **We build more accurate representations for polymers.** Previous studies have often neglected polymers, represented them as zero vectors, or treated them as their monomer forms [14, 20], due to the lack of SMILES data. The use of graphs as representations in this study is more intuitive and informative for non-Euclidean geometries like molecular structures (compared with the commonly used sequential SMILES), and enables ATC classification for all types of compounds. In addition, we have introduced virtual atoms

and bonds between the connecting points of the member monomers to stimulate the inter-monomer communications (Fig. 1C).

- **We optimize the representation learning for macromolecular drugs.** Previous study [20] based on sequence models involved truncation during the processing of input sequences, necessitating a balance between small and macromolecules. By representing macromolecules using graphs, we have eliminated the need for truncation and thus preserve structural

information. Additionally, we found the propagation distance of node information could be enhanced by increasing the number of layers in the message passing mechanism, thereby improving the representational quality of macromolecular drugs.

- **We build a more effective framework for aggregating component representations of multicomponent drugs.** Based on our data analysis, multicomponent medications represent 44.8% of the compounds in ATC benchmarks, underscoring their significance in this context. Each component of a drug plays a distinct role in shaping the drug's properties, adding layers of complexity to representation learning for multicomponent drugs. However, prior research has often oversimplified this by assuming equal contributions from all components. In sequential models utilizing SMILES, component sub-sequences are segmented with a [dot] notation [33]. In GCN-based models, graph representation is achieved through flat pooling, averaging node features rather than component features. Both approaches treat components as equal contributors, which may not accurately reflect their individual impacts. To tackle this challenge, we introduce an aggregative inference framework that integrates component representations regarding their interactions. As shown in Fig. 1D, we employ a bidirectional recurrent neural network (Bi-RNN) to blend component representations successively, dynamically assessing each component's contribution based on the evolving "context" established by earlier fused components. This method aligns more closely with our understanding of chemical interactions among components.

The organization of this paper follows a widely adopted five-step guideline in ATC studies [2, 4, 5, 7, 12, 20], as outlined in [41]. The guideline consists of five steps: (1) selecting a benchmark dataset, (2) formulating the samples, (3) designing the operation algorithm, (4) anticipating accuracy, and (5) creating a web-server.

Materials and methods

Benchmark dataset construction

To kick off the study, we have established ATC-GRAPH as the most extensive ATC benchmark dataset to date. The construction process commenced with a review of two existing benchmarks: Chen-2012 [2], widely adopted in prior studies, and ATC-SMILES [20], the most comprehensive and current benchmark before this research. A detailed comparison of the three datasets is presented in Table 1 and Fig. 2. A key characteristic of ATC-GRAPH is that all drugs in the benchmarks are linked to their Mol files instead of the SMILES sequences utilized in earlier benchmarks. This shift allows for more precise and detailed modeling and learning. In terms of scale, ATC-GRAPH surpasses Chen-2012 and ATC-SMILES by 36.78% and 16.85%, respectively. Significantly, ATC-GRAPH was curated through a cross-validation process involving multiple resources such as KEGG, PubChem, ChEMBL, ChemSpider, and ChemicalBook. This results in ATC-GRAPH being distinguished by its timeliness and comprehensive coverage across all five levels and drug genres.

Improvement on timeliness: over a decade of development, the Chen-2012 dataset no longer accurately reflects the current drug landscape, yet it continues to be used as an important performance evaluation dataset to align with previous research. After verifying drug IDs, we discovered that some drug codes had been updated or some drugs were no longer in use, with no records in various databases. Examples include D02859, D06425, D06488,

D06526, D06527, D06535, D06536, and D06537. Additionally, ATC labels of some drugs are missing, which we obtained by consulting pharmacological experts or by searching historical pages via the Internet Archive's Wayback Machine [42]. For example, the drug D07536, named Boldenone undecylenate, is currently used primarily to enhance the physical condition and performance of horses. In human medicine, due to significant side effects, its use has been discontinued in current medical practice, thus it lacks an ATC code. Pharmacological experts suggest that it likely falls under "Anabolic agents for systemic use," specifically within the category of androgenic drugs, based on its pharmacological effects and its biological targets in the body (androgen receptors), typically starting with the code "A14A." Another example is the drug D00728, named Bismuth subsalicylate, whose therapeutic effects include Antacid, Antidiarrheal, Anti-ulcerative, and which has had its ATC code removed. Using the Wayback Machine, we found that in 2012 this drug's code was "D01AE12" and "S01BC08."

Better coverage over five levels: Chen-2012 and ATC-SMILES datasets exclusively utilized Level 1 labels, while ATC-GRAPH has compiled all ATC codes spanning the first through the fifth level, enabling support for multilevel, multi-label studies. The comparison of coverage across these five levels is depicted in Fig. 2(a). It is worth mentioning that the coverages of Chen-2012 and ATC-SMILES across Level 2 to Level 5 are indeed zeros in the original datasets. To enhance visualization, we have supplemented the missing level labels in these two datasets with our labels. The detailed distributions of drugs at Level 1 and Level 2 for the three datasets are presented in Fig. 2(b) and Fig. 2(c), respectively. It is evident from the figures that ATC-GRAPH stands out for its superior comprehensiveness compared with the other benchmarks.

Better coverage over drug genres: compared with the Chen-2012 and ATC-SMILES datasets, ATC-GRAPH explicitly includes polymeric, macro, and multicomponent drugs.

According to the drug mass distribution in Fig. 2(d), it is evident that the relative mass of the drugs exhibits a long-tail distribution. Most drugs have a relative mass within the 0–1000 range, belonging to small molecule drugs. Drugs with a relative mass of 1000 or greater are considered macromolecular drugs, accounting for ~9.2% of the dataset. Drugs within the 0–499 range account for 68.8%, and those in the 500–999 range make up 22.0%. In Fig. 2(e), the drug distribution across the number of components also shows a significant long-tail distribution. The vast majority of drugs contain only one component, with these single-component drugs accounting for 55.2%. However, drugs containing two or more components, the multicomponent drugs, make up a significant 44.8%, a substantial proportion that cannot be overlooked.

Graphic representations: in ATC-GRAPH, the drug molecules are graphically represented with atoms as nodes and chemical bonds as edges. The atomic numbers start with hydrogen (symbol H), coded as 0, up to the largest element on the periodic table, oganesson (symbol Uuo), coded as 117. To facilitate network processing of aggregate drugs and drugs with other functional groups or pharmacophores, the symbol "R" is added as the functional groups or pharmacophores. Node attributes consider atom type and chirality, while edge attributes consider the type of chemical bond and the direction of the bond, which are formally defined in the subsequent sections.

Problem formulation

We formulate the graph-based ATC as a learning problem for a function $f: \mathcal{G} \rightarrow \{0, 1\}^c$, which takes features encapsulated on a molecular graph $\mathcal{G}$ of a compound x and predict its ATC

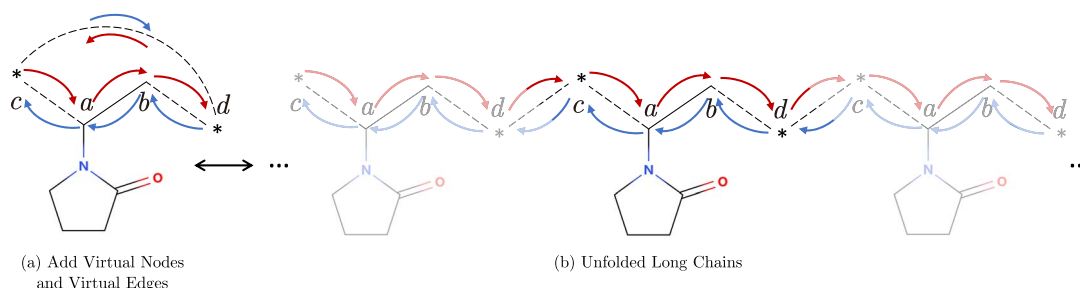


Figure 3. Illustration of the incorporation of virtual atoms and bonds to stimulate inter-monomer communication in elongated chain structures.

label as

$$\hat{\mathbf{y}} = f(\mathcal{G}; \theta), \quad (1)$$

where $\hat{\mathbf{y}} \in \{0, 1\}^c$ is a multi-hot binary vector with the i th element being 1/0 indicating the membership of x to the i th ATC class, the c is the number of classes, and θ is a set of parameters to learn. The molecular graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a set $\mathcal{V} = \{v_i\}$ of vertices and a set $\mathcal{E} = \{e_{ij}\}$ of edges, which represent the atoms and bonds of x , respectively.

The learning goal is to find an optimal set of parameters θ that minimizes the loss $\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y})$ between the prediction $\hat{\mathbf{y}}$ and the ground-truth $\mathbf{y}$, which indicates the real ATC class labels of x as

$$\begin{aligned} \hat{\theta} &= \operatorname{argmin}_{\theta} \mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) \\ &= \operatorname{argmin}_{\theta} \mathcal{L}(f(\mathcal{G}; \theta), \mathbf{y}). \end{aligned} \quad (2)$$

In the following subsections, we will introduce our implementation of representation learning on $\mathcal{G}$, the prediction function f , and the loss function $\mathcal{L}$. To ease the elaboration, the framework of our implementation is shown in Fig. 1. The prediction process (e.g. f) can be further decomposed into three sub-processes of Molecular Graph Construction, Representation Learning, and Aggregative Inference. The advantage of the proposed framework over previous work mainly lies in its capacity to deal with polymeric, macromolecular, and multicomponent drugs. We will give more details.

Construction and featurization of molecular graph $\mathcal{G}$

While converting from a structural formula to its molecular graph is straightforward, several strategies have been introduced in our conversion scheme. Firstly, as shown in Fig. 3, a virtual self-loop bond has been included for each vertex (atom) to enhance the model's ability to capture local structural information and to ensure that the intrinsic features of each vertex are not overlooked during the feature update process. This has been validated in many papers [43–45]. The symbol “*” is added to the molecular graph to represent virtual atoms. Secondly, for each polymer, we include an additional virtual bond by connecting the covalent bonds of the corresponding monomer to encourage the interaction, as shown in Fig. 3. In previous ATC studies, the prediction of polymers has simply been skipped [16, 19, 20]. In other tasks, a popular way to simplify the problem is to ignore the repeating structure and keep the corresponding monomer [46–48]. It will make no difference between the polymers and monomers. The interactions (message passing) between monomers are neglected

in the model. The virtual bond we included is thus designed to encourage the interactions, which is more faithful to the chemical model. As the example shown in Fig. 1B, the carbon atom in the middle is able to receive messages from the three adjacent atoms through the virtual bond. To featurize the graph, we reserve four types of embedding matrices including those for atom IDs, chirality tags, bond types, and bond directions. All embeddings share the same dimensionality of m , which is the same as that of the hidden vectors. More specifically, the matrices are

$$\begin{aligned} \text{Atom IDs: } \mathbf{A} &\in \mathbb{R}^{a \times m}, a \in \mathbb{N} \cap [0, 118] \\ \text{Chirality Tags: } \mathbf{C} &\in \mathbb{R}^{c \times m}, c \in \mathbb{N} \cap [0, 2] \\ \text{Bond Types: } \mathbf{B} &\in \mathbb{R}^{b \times m}, b \in \mathbb{N} \cap [0, 4] \\ \text{Bond Directions: } \mathbf{D} &\in \mathbb{R}^{d \times m}, d \in \mathbb{N} \cap [0, 4], \end{aligned} \quad (3)$$

where rows in $\mathbf{A}$ are ordered with the 0th row for non-registered atoms and the rest of 118 for known atoms, $\mathbf{C}$ are ordered with chirality tags of Unspecified, Tetrahedral CW, and Tetrahedral CCW, $\mathbf{B}$ are ordered with bond types of Single, Double, Triple, Aromatic, and Self-Loop, and $\mathbf{D}$ are ordered with bond directions of Linear, End_Up_Right, and End_Down_Right. With the embedding matrices, we can assign for each atom vertex a hidden vector by fusing its ID and chirality embeddings as

$$\mathbf{h}_i = \frac{1}{2} (\mathbf{A}_{I(\mathbf{A}, i)} + \mathbf{C}_{I(\mathbf{C}, i)}), \quad (4)$$

where $I(\cdot)$ is an indexing function that returns the index of a given atom (or bond) in a specified embedding matrix. Similarly, a hidden vector for each bond edge can be calculated with its type and direction embeddings as

$$\mathbf{e}_{ij} = \frac{1}{2} (\mathbf{B}_{I(\mathbf{B}, ij)} + \mathbf{D}_{I(\mathbf{D}, ij)}). \quad (5)$$

Representation learning on the molecular graph $\mathcal{G}$

We adopt GCNs from DeeperGCN [49] to implement the representation learning on $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. It can be considered as an iterative process to simulate the interactions among atoms through bonds. More specifically, at a step (or layer) t , each atom vertex v_i holds a d -dimensional hidden state vector $\mathbf{h}_i^{(t)} \in \mathbb{R}^m$ and accumulates the messages passed from all its adjacent vertices v_j 's through the corresponding bond edges e_{ij} 's on the basis of which the feature $\mathbf{h}_i^{(t)}$ will be updated to its state at step (or layer) $t + 1$ as $\mathbf{h}_i^{(t+1)}$. The

process can be formulated as

$$\begin{aligned}
 \text{Message: } \mathbf{m}_{ij}^{(t)} &= \mathbf{h}_j^{(t)} + \mathbf{e}_{ij}^{(t)}, \\
 \text{Softmax: } \alpha_{ij} &= \frac{\exp(\beta^{(t)} \mathbf{m}_{ij}^{(t)})}{\sum_{k \in N(i)} \exp(\beta^{(t)} \mathbf{m}_{ik}^{(t)})}, \\
 \text{Aggregation: } \tilde{\mathbf{m}}_{ij}^{(t)} &= \sum_{j \in N(i)} \alpha_{ij} \mathbf{m}_{ij}^{(t)}, \\
 \text{Updating: } \mathbf{h}_i^{(t+1)} &= \text{MLP} \left(\mathbf{h}_i^{(t)} + \lambda^{(t)} \frac{\|\mathbf{h}_i^{(t)}\|}{\|\tilde{\mathbf{m}}_{ij}^{(t)}\|} \tilde{\mathbf{m}}_{ij}^{(t)} \middle| \boldsymbol{\theta}_w^{(t)} \right), \quad (6)
 \end{aligned}$$

where $\mathbf{e}_{ij}^{(t)} \in \mathbb{R}^m$ is the strength vector on the edge e_{ij} , $N(i)$ is a set of v_i 's adjacent vertices, and $\text{MLP}(\cdot | \boldsymbol{\theta}_w^{(t)})$ is a multilayer perceptron based on the parameter $\boldsymbol{\theta}_w^{(t)} \in \mathbb{R}^m$. The $\beta^{(t)} \in \mathbb{R}$ and $\lambda^{(t)} \in \mathbb{R}$ are the temperature and scalar, respectively. $\beta^{(t)}$, $\lambda^{(t)}$, and $\boldsymbol{\theta}_w^{(t)}$ are all layer-dependent and learnable weights at step (layer) t . The process will be repeated T steps, resulting in the feature $\mathbf{h}_i^{(T)}$ on each atom v_i . An average pooling can be used to get the final representation of the molecular graph $\mathcal{G}$ as

$$\mathbf{g}^{(T)} = \frac{1}{\|\mathcal{V}\|} \sum_{i \in \mathcal{V}} \mathbf{h}_i^{(T)}. \quad (7)$$

Aggregative inference

We made a single graph assumption in the preceding subsections. However, for multicomponent drugs, there are multiple subgraphs (denoted as $\mathcal{G}_k$'s hereafter), each requiring graph construction and representation learning. This generates a set of molecular graph representations $\{\mathbf{g}_k^{(T)}\}$. We further arrange the representations in descending order by subgraph size and convert the set into a sequence. To streamline the discussion, we will hide the layer indicator (T). The sequence is then written

$$\langle \mathbf{g}_k \mid k \in \mathbb{N}_{\geq 0}, \|\mathcal{G}_k\| \geq \|\mathcal{G}_l\| \text{ if } k > l \rangle. \quad (8)$$

To implement the aggregative inference, the goal is to fuse this sequence into a single representation $\mathbf{g}^*$ on the basis of which the inference is conducted.

We propose to adapt RNN for this purpose, as shown in Fig. 1D. The process starts by accumulating the representations using average pooling as

$$\bar{\mathbf{g}} = \frac{1}{\|\langle \mathbf{g}_k \rangle\|} \sum_{k=1} \|\mathbf{g}_k\| \mathbf{g}_k. \quad (9)$$

Although the average pooling is popularly adopted for multiple graph aggregation, it fuses representations in a coarse level, in the sense that the inter-dependency among subgraphs is crudely modeled during the pooling. However, the result $\bar{\mathbf{g}}$ is still a fair base for the aggregation.

To learn the inter-dependency, RNN is a more sophisticated model. With the ordered representation sequence $\langle \mathbf{g}_k \rangle$ as the input, our RNN fuses the representations in an iterative way from those of large subgraphs to small ones. The design is based on the intuition that large compounds often play the primary role during chemical interactions. In addition, RNNs are known to be capable of modeling the inter-item dependency of a sequence by using early inputs to learn a "context" for further aggregation, which

thus serves as a more sophisticated way of aggregating subgraph representations. Our RNN-based aggregation is formulated as

$$\mathbf{s}_k = \tanh(\mathbf{g}_k \mathbf{W}_g^\top + \mathbf{s}_{k-1} \mathbf{W}_s^\top + \mathbf{b}_s) \quad (10)$$

$$\tilde{\mathbf{g}}_k = \text{softmax}(\mathbf{s}_k \mathbf{W}_g^\top + \mathbf{b}_g), \quad (11)$$

where $\mathbf{s}_k$ denotes the hidden state of the RNN at the k th iteration (i.e. the k th subgraph representation $\mathbf{g}_k$ has been fused). $\mathbf{W}_g$, $\mathbf{W}_s$, and $\mathbf{b}_s$ are the learnable weights and bias for the state, respectively. $\tilde{\mathbf{g}}_k$ is the intermediate output at the k th iteration (i.e. an RNN-fused representation of the first k subgraph representations). $\mathbf{W}_g$ and $\mathbf{b}_g$ are learnable weights and bias for the output, respectively. Eventually, the intermediate outputs $\tilde{\mathbf{g}}_k$'s are fused to refine the coarse-level result $\bar{\mathbf{g}}$, which results in $\mathbf{g}^*$ for prediction as

$$\mathbf{g}^* = \bar{\mathbf{g}} + \sum_{k=1}^{\|\langle \mathbf{g}_k \rangle\|} \tilde{\mathbf{g}}_k. \quad (12)$$

The prediction can then be made with FC layer. We can simply use a linear layer for this purpose as

$$\hat{\mathbf{y}} = \mathbf{g}^* \mathbf{W}_y^\top + \mathbf{b}_y, \quad (13)$$

where $\mathbf{W}_y$ and $\mathbf{b}_y$ are learnable weights and bias of the layer.

Ground truth $\mathbf{y}$ and loss function $\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y})$

We evaluate the prediction $\hat{\mathbf{y}}$ by comparing it with the corresponding ground truth label $\mathbf{y}$. As our goal is to extend the ATC task to level-2, the number of potential labels has increased to $C = 102$, resulting from the iteration of 14 level-1 labels up to their respective children (e.g. A04, B05...). A ground truth label $\mathbf{y}$ is then a multi-hot vector by setting the bits of its classes to 1's and leaving others to 0's.

We implement the loss function using a multi-label one-versus-all loss based on max-entropy as

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{C} \left(\mathbf{y}^\top \log \left(\frac{1}{1 + e^{-\hat{\mathbf{y}}}} \right) + (1 - \mathbf{y})^\top \log \left(1 - \frac{e^{-\hat{\mathbf{y}}}}{1 + e^{-\hat{\mathbf{y}}}} \right) \right). \quad (14)$$

Results and discussion

Metrics

We evaluate the results using the five metrics that have been established in [50] and widely adopted in literature as follows:

$$\text{Aiming} = \frac{1}{N} \sum_{i=1}^N \left(\frac{\|L_i \cap \hat{L}_i\|}{\|\hat{L}_i\|} \right) \quad (15)$$

$$\text{Coverage} = \frac{1}{N} \sum_{i=1}^N \left(\frac{\|L_i \cap \hat{L}_i\|}{\|L_i\|} \right) \quad (16)$$

$$\text{Accuracy} = \frac{1}{N} \sum_{i=1}^N \left(\frac{\|L_i \cap \hat{L}_i\|}{\|L_i \cup \hat{L}_i\|} \right) \quad (17)$$

$$\text{Absolute True} = \frac{1}{N} \sum_{i=1}^N \left(\Delta(L_i, \hat{L}_i) \right) \quad (18)$$

$$\Delta(L_i, \hat{L}_i) = \begin{cases} 1, & \text{if } L_i = \hat{L}_i. \\ 0, & \text{otherwise.} \end{cases} \quad (19)$$

Table 2. Performance comparison with SOTA methods on ATC Level 1. The best results are in bold font

Method	Year	Dataset	#Drugs	Rep.	Model	Aiming ↑	Coverage ↑	Accuracy ↑	Abs. True ↑	Abs. False ↓
Chen et al. [2]	2012	Chen-2012	3883	I S	Similarity Search	50.76%	75.79%	49.38%	13.83%	8.83%
iATC-mISF [5]	2017	Chen-2012	3883	I S F	ML-GKR	67.83%	67.10%	66.41%	60.98%	5.85%
iATC-mHyb [6]	2017	Chen-2012	3883	I S F O	ML-GKR	71.91%	71.46%	71.32%	66.75%	2.43%
EnsLIFT [7]	2017	Chen-2012	3883	I S F	LIFT	78.18%	75.77%	71.21%	63.30%	2.85%
EnsANet_LR [8]	2018	Chen-2012	3883	I S F	CNN,LIFT,RR	75.40%	82.49%	75.12%	66.68%	2.62%
EnsANet_LR@DO [8]	2018	Chen-2012	3883	I S F O	CNN,LIFT,RR	79.57%	83.35%	77.78%	70.90%	2.40%
ATC-NLSP [10]	2019	Chen-2012	3883	I S F	NLSP	81.35%	79.50%	78.28%	74.97%	3.43%
iATC-NRAKEL [13]	2020	Chen-2012	3883	I S	RAKEL,SVM	78.88%	79.36%	77.86%	75.93%	3.63%
iATC-FRAKEL [14]	2020	Chen-2012	3883	F	RAKEL,SVM	78.51%	78.40%	77.21%	75.11%	3.70%
FUS3 [11]	2020	Chen-2012	3883	I S F	CNN,LSTM,LIFT,RR	87.55%	69.73%	73.46%	68.71%	2.38%
FUS3@DO [11]	2020	Chen-2012	3883	I S F O	CNN,LSTM,LIFT,RR	79.79%	84.22%	79.64%	73.04%	2.09%
iATC_Deep-mISF [12]	2020	Chen-2012	3883	I S F O	DNN	74.70%	73.91%	71.57%	67.01%	0.00%
CGATCPred [16]	2021	Chen-2012	3883	I S E A	CNN,GCN	81.94%	82.88%	80.81%	76.58%	2.75%
EnsATC [19]	2022	Chen-2012	3883	I S F	hMuLab,LSTM	91.39%	84.32%	83.38%	80.09%	1.31%
ATC-CNN [20]	2022	Chen-2012	3883	S	CNN	93.01%	90.72%	90.53%	87.77%	1.53%
ATC-CNN [20]	2022	ATC-SMILES	4545	S	CNN	95.83%	94.14%	93.99%	91.77%	0.94%
ATC-CNN [20]	2022	ATC-GRAPH	5311	S	CNN	77.34%	76.42%	75.63%	73.11%	3.55%
GraphATC (Ours)	2024	Chen-2012	3883	S	GCN,BiRNN	95.73%	95.64%	94.68%	92.56%	0.83%
GraphATC (Ours)	2024	ATC-SMILES	4545	S	GCN,BiRNN	96.08%	96.09%	95.42%	93.97%	0.68%
GraphATC (Ours)	2024	ATC-GRAPH	5311	S	GCN,BiRNN	96.94%	96.88%	96.14%	94.56%	0.57%

Representation (Rep.) abbreviations: I—chemical interactions, S—chemical structural features, F—molecular fingerprint features, O—drug ontology information, E—pretrained word embedding, and A—ATC codes association information.

$$\text{Absolute False} = \frac{1}{N} \sum_{i=1}^N \left(\frac{\|L_i \cup \hat{L}_i\| - \|L_i \cap \hat{L}_i\|}{M} \right), \quad (20)$$

where M is the number of labels and N is the total number of all samples. y_i and $\hat{y}_i$ are the ground truth and predicted labels of the i th drug, respectively. $\cup$ and $\cap$ denote the union and intersection operations, and $\|\cdot\|$ is an operator to count the number of elements in a set. In the remainder of this section, we will use the symbol $\uparrow$ to indicate positive indices (i.e. Aiming, Coverage, Accuracy, and Absolute True) and $\downarrow$ for negative indices (e.g. Absolute False).

Cross validation

Cross-validation has been conducted using the Jackknife test, which has been considered as a standard and adopted in nearly all previous ATC studies [2, 5–8, 10, 14, 16, 20]. Methodologically, the Jackknife test is a “leave-one-out” test or a special case of k -fold cross-validation where k equals the total number of data samples. By employing the Jackknife test, the consistency in data splits and alignment in experiment results across different studies is thus ensured.

Comparison with SOTA methods

In this study, we evaluate the performance of the proposed GraphATC method against 15 SOTA methods that use various representations and models. We have included the results on Chen-2012 benchmark to be consistent with previous studies. In addition, we have conducted experiments on ATC-SMILES and ATC-GRAPH. This is the most comprehensive comparison found in the literature. The results on ATC Level 1, shown in Table 2, reveal that GraphATC outperforms the SOTA methods in all five metrics, with improvements of 2.72%, 4.92%, 4.15%, 4.79%, and 0.7% in Aiming, Coverage, Accuracy, Absolute True, and Absolute False, respectively. In terms of the ATC-CNN dataset, GraphATC outperforms the SOTA methods with gains of 0.25%, 1.95%, 1.43%, 2.2%, and 0.26% across the five metrics. Regarding the proposed

ATC-GRAPH dataset, GraphATC outperforms the SOTA methods with gains of 19.60%, 20.46%, 20.51%, 21.45%, and 2.98% across the five metrics.

As discussed earlier, the ATC experiments on Level 2 in a multi-label classification setting are under-explored in the literature. We have addressed this issue in this paper. Note that, due to the availability of source code on Level 2 experiments, we can only compare with the SOTA method ATC-CNN, which introduces the ATC benchmark with Level 2 drugs and has the best performance reported in the literature. The comparison of L2 drugs has been conducted extensively on three datasets, including Chen-2012, ATC-SMILES, and ATC-GRAPH. The results are shown in Table 3. Our method outperforms ATC-CNN over all three datasets by 24.79%, 26.43%, 26.16%, 27.7%, and 1.32% in Aiming, Coverage, Accuracy, Absolute True, and Absolute False, respectively. In comparison with Level 1, there was a slight degradation in the performance of all models at Level 2. This deterioration can be attributed to the reduced training samples available for each class at Level 2, particularly as the class definitions become more finely grained. The performance gap observed serves as an indicator of potential areas for enhancement in future research endeavors.

Ablation study

In order to ascertain the superiority of the proposed method, we conduct an ablation study to assess the significance of each component. However, given the extensive set of experiments to be carried out, we substituted the costly jackknife validation with 100-fold cross-validation. For the same reason, our attention will be directed toward Level 1. The outcomes are detailed in Fig. 4 and Fig. 6.

The importance of virtual atoms and bonds

As shown in Fig. 4, the incorporation of virtual atoms and bonds has clearly enhanced performance. Particularly on Polymers, this enhancement has led to 8.33%, 14.74%, 12.18%, 11.54%, and 2.06% gain in Aiming, Coverage, Accuracy, Absolute True, and Absolute

Table 3. Performance comparison with SOTA methods on ATC Level 2. The best results are in bold font

Method	Dataset	Aiming ↑	Coverage ↑	Accuracy ↑	Abs. True ↑	Abs. False ↓
ATC-CNN	Chen-2012	79.39	78.42	77.20	73.78	0.53
ATC-CNN	ATC-SMILES	82.95	82.29	81.13	78.32	0.63
ATC-CNN	ATC-GRAPH	67.93	66.40	65.73	62.62	1.50
GraphATC (Ours)	Chen-2012	86.97	86.65	85.52	83.05	0.29
GraphATC (Ours)	ATC-SMILES	91.75	92.22	90.74	88.43	0.22
GraphATC (Ours)	ATC-GRAPH	92.72	92.83	91.89	90.32	0.19

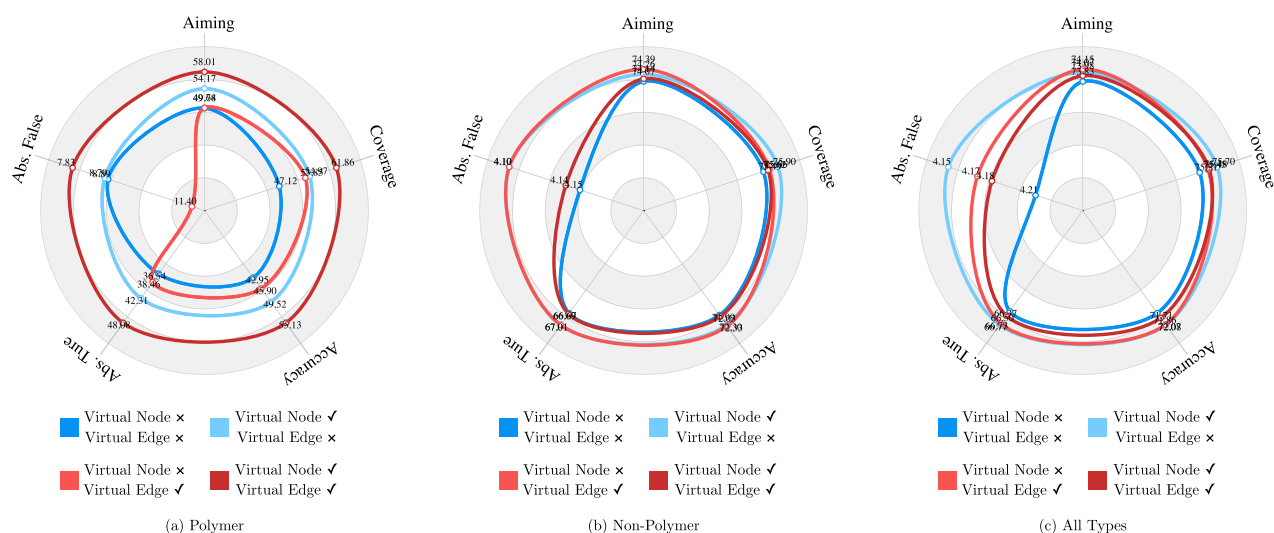


Figure 4. Performance comparison of different polymer modeling methods on (a) Polymer, (b) Non-Polymer, and (c) All Types.

False metrics, respectively, compared with the baseline run without virtual entities.

This outcome is unsurprising as, according to the design, the inclusion of virtual entities promotes interactions among monomer atoms within a Polymer, aligning the message passing in the GCN more effectively with real-world scenarios. As shown in Fig. 5, this effect becomes apparent when visualizing the neural network attention through class activation mapping (CAM). In Fig. 5(d), the reinforcement of attention on the connecting atoms and bonds within Polymers is evident following the addition of the virtual atoms and bonds. By contrast, without the help of the virtual atoms and bonds, Fig. 5(a) shows less focus on connections when a polymer has been treated as its monomer form.

The importance of aggregative inference

As shown in Fig. 6, by enabling the aggregative inference using subgraph fusion, the performance has been boosted on all types. The largest gain over the run without aggregative inference has been observed on the multicomponent compound with 13.43%, 14.46%, 12.71%, 10.59%, and 0.41% in Aiming, Coverage, Accuracy, Absolute True, and Absolute False, respectively.

In Fig. 7, the CAM attentions without and with the aggregative inference have been visualized. It is evident in Fig. 7(a) that the attention has been dominated by the subgraphs with large scales and the contribution of small subgraphs has been ignored. By contrast, in Fig. 7(b), the attentions are fairly paid on those small subgraphs when aggregative inference has been integrated.

Comparison of GraphATC using different backbone models

We compare the GraphATC with different graph-based models as its backbone on the ATC-GRAPH dataset on Level 1 ATC labels, including GCN [43], GAT [51], GIN [52] implemented in DGL [53], and DeeperGCN [49] implemented by following the corresponding paper. The results are presented in Table 4. Our method surpasses the baseline models, achieving maximum performance improvements of 10.16%, 10.52%, 9.76%, 8.59%, and 0.38% across five metrics. The result primarily demonstrates the effectiveness of the proposed GraphATC framework across various graph backbone models, highlighting its extensibility.

Comparison of GraphATC with various graph-based models

We compare our method with the latest graph-based approaches specifically designed for this task, including SAN [54], GraphGPS [55], Expformer [56], and Graph-Mamba [57]. The results of these comparisons are shown in Table 5. Our method demonstrates superior performance over these approaches across nearly all five metrics, with maximum gains of 37.15%, 38.83%, 37.84%, 36.76%, and 3.38% across five metrics.

Web server

Alongside releasing the source code of GraphATC on Github.com, we have created a web server accessible at <https://github.com/lookwei/GraphATC> to enhance the accessibility of both the method and dataset. This web server accepts a drug/compound ID as input and provides predictions for labels and the top five

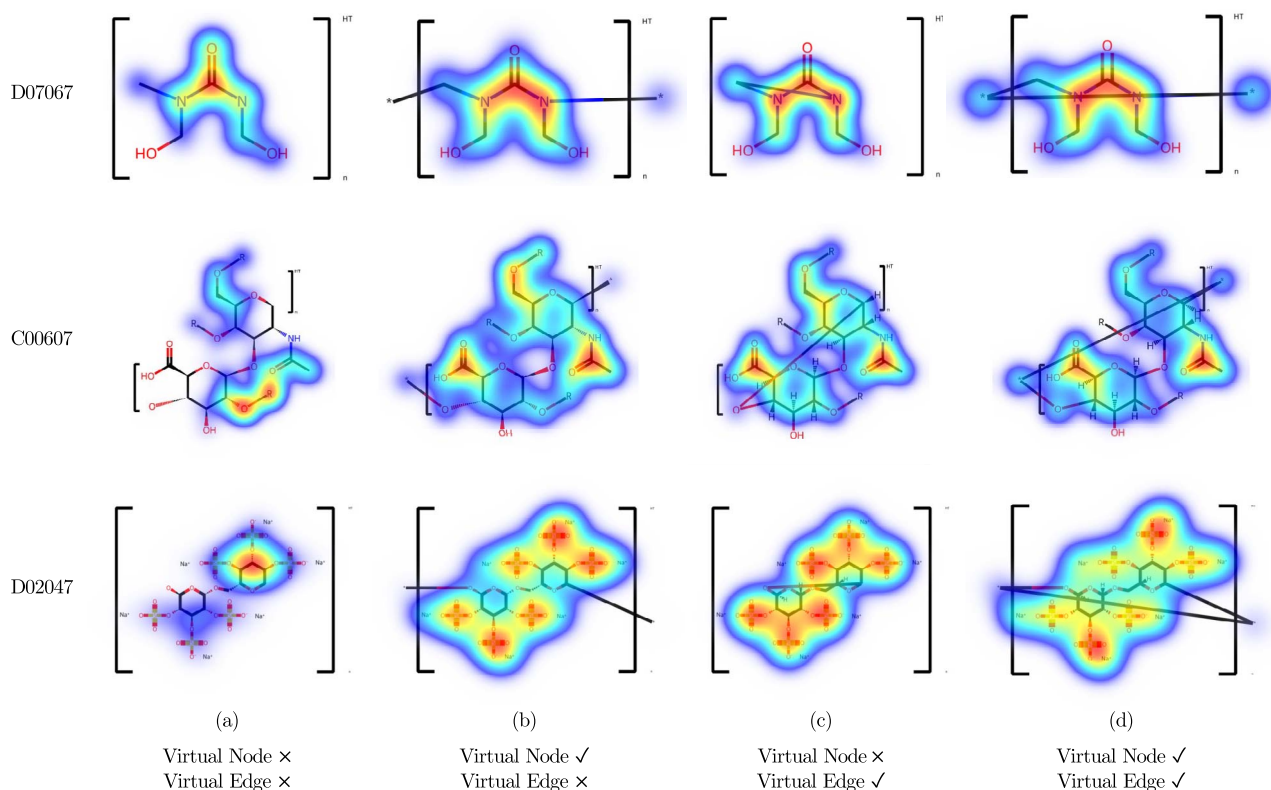


Figure 5. Attention maps exemplifying drugs with a varied range of scales and shapes. (a) When polymer drugs have been treated as their monomer forms, the attention is concentrated around the central parts (e.g. the carbonyl (C=O) group and nitrogen atoms (N) in the central ring of D07067), ignoring the contributions of the connecting parts; (b) by adding the virtual nodes, the attention expands and helps the model capture end-group interactions (e.g. two terminal hydroxyl (OH) groups and two nitrogen atoms (N) are emphasized in D07067); (c) by adding virtual edges, attention extends along bonds, especially toward the connected atoms (e.g. the N-N bond within the central ring of D07067). However, without virtual nodes, attention remains limited to bond pathways. (d) By adding both virtual nodes and edges, more uniform attention distributions are observed across the entire molecules, with a strong focus not only on the central parts and terminal groups (e.g. the C=O, N-N bonds and the OH of D07067) but also on virtual atoms particularly. The refined attention maps reflect inter-monomer interactions within polymers in a better way.

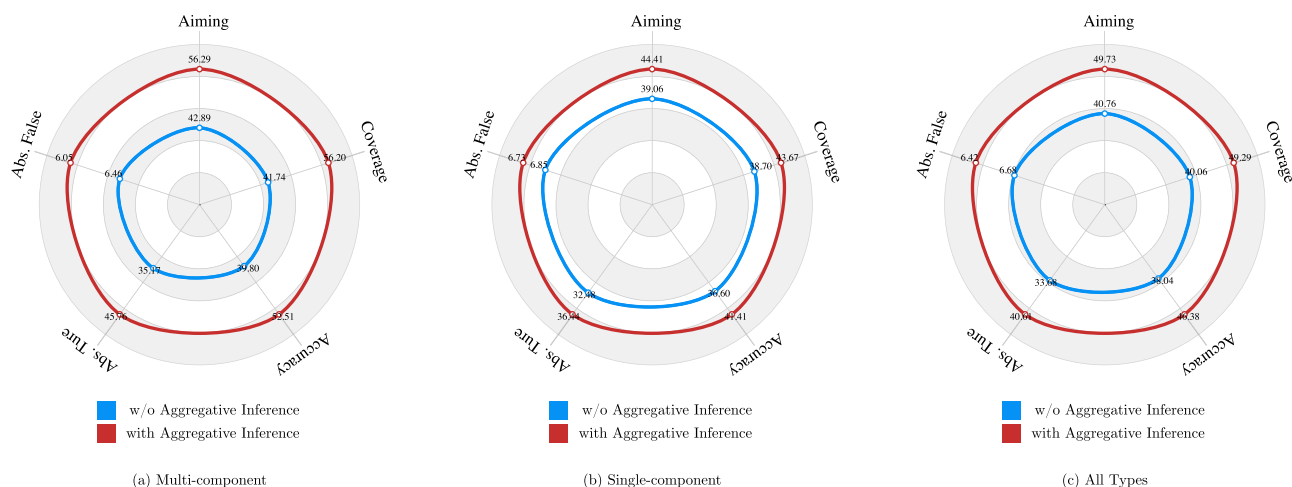


Figure 6. Performance comparison of different subgraph feature fusion networks in handling (a) multicomponent drugs, (b) single-component drugs, and (c) all types of drugs.

related drugs/compounds. It is important to note that the ID does not have to be from ATC-GRAPH; the server can predict labels for any drugs or compounds with valid IDs or sequences.

Conclusion

This research has adopted a holistic strategy to propel the multilevel and multi-label ATC classification through graph

learning techniques. By methodically enriching the drug dataset to encompass all five levels and employing a cross-resource validation process involving key databases like KEGG, PubChem, ChEMBL, ChemSpider, and ChemicalBook, we have introduced a fresh benchmark known as ATC-GRAPH. Moreover, we have expanded the classification task to Level 2, introducing graph-based learning strategies to create more precise representations of drug molecular structures and improving the modeling of

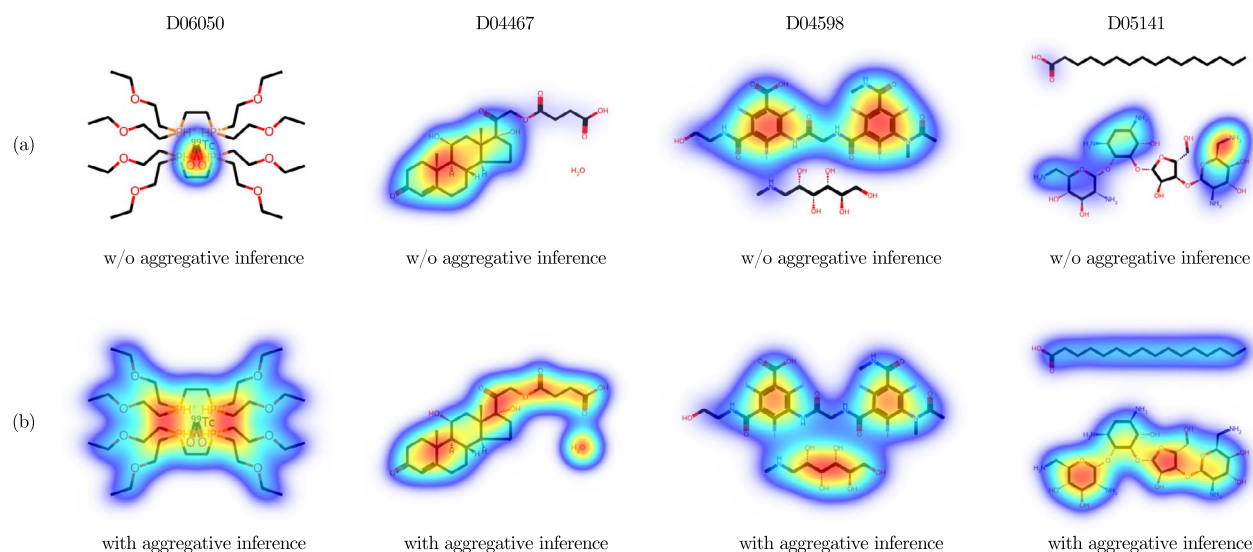


Figure 7. Attention maps exemplifying drugs consisting of components with a varied range of scales and shapes. (a) Without using the aggregative inference, most of the attention has been dominated by the subgraphs with large scales (e.g. long chain in D04467, two large rings of D04598, NH_2 in D05141), while the contribution of small subgraphs has been ignored (e.g. long chains in D06050, H_2O in D04467, short chain in D04598, long chain in D05141); (b) when integrating aggregative inference, attentions are fairly paid to those small subgraphs. The refined attention maps indicate subgraph interactions within multicomponent molecular in a better way.

Table 4. Performance (%) comparison of GraphATC using different graph-based backbone models on the ATC-GGRAPH dataset (Level 1). 100-fold cross-validation was applied, and the best results are highlighted in bold

Backbone	Method	Aiming ↑	Coverage ↑	Accuracy ↑	Abs. True ↑	Abs. False ↓
GCN [43]	Base	40.31	39.58	37.43	32.85	6.81
	Ours	50.47	50.10	47.19	41.44	6.43
GAT [51]	Base	40.62	39.59	37.77	33.51	6.71
	Ours	49.89	49.49	46.61	40.86	6.40
GIN [52]	Base	40.55	39.85	37.76	33.39	6.74
	Ours	50.20	49.58	46.83	41.10	6.42
DeeperGCN [49]	Base	70.72	73.96	68.06	59.59	4.67
	Ours	77.98	78.90	75.79	70.63	3.52

Table 5. Performance (%) comparison of GraphATC with various *ad hoc* graph-based models on the ATC-GGRAPH dataset (Level 1). 100-fold cross-validation was performed, and the best results are highlighted in bold

Method	Aiming ↑	Coverage ↑	Accuracy ↑	Abs. True ↑	Abs. False ↓
SAN [54]	40.83	40.07	37.95	33.87	6.90
GraphGPS [55]	65.83	65.91	62.91	57.14	4.61
Exphormer [56]	64.69	64.14	61.86	56.87	4.43
Graph-Mamba [57]	75.17	75.85	73.17	68.67	3.36
GraphATC (Ours)	77.98	78.90	75.79	70.63	3.52

intricate drug categories. Following rigorous experimentation, our proposed framework has emerged as a SOTA methodology in the field. In a bid to enhance study reproducibility, we have made the benchmark dataset, source code, and web server openly accessible, emphasizing our dedication to advancing the realm of multilevel and multi-label ATC classification through innovative graph-based learning approaches.

Key Points

- We have constructed the most extensive ATC dataset to date.

- We implement the multilevel, multi-label study by extending the task to Level-2 (i.e. L2).
- We build more accurate representations for polymers.
- We optimize the representation learning for macro-molecular drugs.
- We build a more effective framework for aggregating component representations of multicomponent drugs.

Supplementary data

Supplementary data is available at *Briefings in Bioinformatics* online.

Code and data availability

The dataset, source code, and web server are open to public at <https://github.com/lookwei/GraphATC> for easier production of this study.

Funding

This research was supported by the National Natural Science Foundation of China (Grant No.: 62372314).

References

- Dunkel M, Günther S, Ahmed J. et al. Superpred: drug classification and target prediction. *Nucleic Acids Res* 2008;**36**:W55–9.
- Chen L, Zeng W-M, Cai Y-D. et al. Predicting anatomical therapeutic chemical (ATC) classification of drugs by integrating chemical–chemical interactions and similarities. *PLoS One* 2012;**7**:e35254. <https://doi.org/10.1371/journal.pone.0035254>
- Nickel J, Gohlke B-O, Erehman J. et al. Superpred: update on drug classification and target prediction. *Nucleic Acids Res* 2014;**42**:W26–31. <https://doi.org/10.1093/nar/gku477>
- Chen L, Jing L, Zhang N. et al. A hybrid method for prediction and repositioning of drug anatomical therapeutic chemical classes. *Mol Biosyst* 2014;**10**:868–77. <https://doi.org/10.1039/c3mb70490d>
- Cheng X, Zhao S-G, Xiao X. et al. iATC-mISF: a multi-label classifier for predicting the classes of anatomical therapeutic chemicals. *Bioinformatics* 2017;**33**:341–6. <https://doi.org/10.1093/bioinformatics/btw644>
- Cheng X, Zhao S-G, Xiao X. et al. iATC-mHyb: a hybrid multi-label classifier for predicting the classification of anatomical therapeutic chemicals. *Oncotarget* 2017;**8**:58494–503. <https://doi.org/10.18632/oncotarget.17028>
- Nanni L, Brahnam S. Multi-label classifier based on histogram of gradients for predicting the anatomical therapeutic chemical class/classes of a given compound. *Bioinformatics* 2017;**33**:2837–41. <https://doi.org/10.1093/bioinformatics/btx278>
- Lumini A, Nanni L. Convolutional neural networks for ATC classification. *Curr Pharm Des* 2018;**24**:4007–12.
- Chen L, Liu T, Zhao X. Inferring anatomical therapeutic chemical (ATC) class of drugs using shortest path and random walk with restart algorithms. *Biochim Biophys Acta, Mol Basis Dis* 2018;**1864**:2228–40. <https://doi.org/10.1016/j.bbadis.2017.12.019>
- Wang X, Wang Y, Zhenyu X. et al. ATC-NLSP: prediction of the classes of anatomical therapeutic chemicals using a network-based label space partition method. *Front Pharmacol* 2019;**10**:971. <https://doi.org/10.3389/fphar.2019.00971>
- Nanni L, Brahnam S, Lumini A. Ensemble of deep learning approaches for ATC classification. In: *Smart Intelligent Computing and Applications: Proceedings of the Third International Conference on Smart Computing and Informatics*, Vol. 1, pp. 117–25. Singapore: Springer Singapore, 2019.
- Zhe L, Chou K-C. iATC_Deep-mISF: a multi-label classifier for predicting the classes of anatomical therapeutic chemicals by deep learning. *Advances in Bioscience and Biotechnology* 2020;**11**:153–9.
- Zhou J-P, Chen L, Guo Z-H. iATC-NRAKEL: an efficient multi-label classifier for recognizing anatomical therapeutic chemical classes of drugs. *Bioinformatics* 2020;**36**:1391–6. <https://doi.org/10.1093/bioinformatics/btz757>
- Zhou J-P, Chen L, Wang T. et al. iATC-FRAKEL: a simple multi-label web server for recognizing anatomical therapeutic chemical classes of drugs with their fingerprints only. *Bioinformatics* 2020;**36**:3568–9. <https://doi.org/10.1093/bioinformatics/btaa166>
- Liang H, Bin H, Chen L. et al. Recognizing novel chemicals/drugs for anatomical therapeutic chemical classes with a heat diffusion algorithm. *Biochim Biophys Acta, Mol Basis Dis* 2020;**1866**:165910. <https://doi.org/10.1016/j.bbadis.2020.165910>
- Zhao H, Li Y, Wang J. A convolutional neural network and graph convolutional network-based method for predicting the classification of anatomical therapeutic chemicals. *Bioinformatics* 2021;**37**:2841–7. <https://doi.org/10.1093/bioinformatics/btab204>
- Wang X, Liu M, Zhang Y. et al. Deep fusion learning facilitates anatomical therapeutic chemical recognition in drug repurposing and discovery. *Brief Bioinform* 2021;**22**:bbab289.
- Yan C, Suo Z, Wang J. et al. DACPGTN: drug ATC code prediction method based on graph transformer network for drug discovery. *Front Pharmacol* 2022;**13**:907676.
- Nanni L, Lumini A, Brahnam S. Neural networks for anatomical therapeutic chemical (ATC) classification. *Appl Comput Inf (ahead-of-print)* 2022.
- Cao Y, Zhen-Qun Yang X-L, Zhang WF. et al. Identifying the kind behind smiles–anatomical therapeutic chemical classification using structure-only representations. *Brief Bioinform* 2022;**23**:bbac346.
- Gurulingappa H, Kolářik C, Hofmann-Apitius M. et al. Concept-based semi-automatic classification of drugs. *J Chem Inf Model* 2009;**49**:1986–92. <https://doi.org/10.1021/ci9000844>
- Leihong W, Ai N, Liu Y. et al. Relating anatomical therapeutic indications by the ensemble similarity of drug sets. *J Chem Inf Model* 2013;**53**:2154–60.
- Wang Y-C, Chen S-L, Deng N-Y. et al. Network predicting drug's anatomical therapeutic chemical code. *Bioinformatics* 2013;**29**:1317–24. <https://doi.org/10.1093/bioinformatics/btt158>
- Chen F-S, Jiang Z-R. Prediction of drug's anatomical therapeutic chemical (ATC) code by integrating drug–domain network. *J Biomed Inform* 2015;**58**:80–8. <https://doi.org/10.1016/j.jbi.2015.09.016>
- Liu Z, Guo F, Jiangyong G. et al. Similarity-based prediction for anatomical therapeutic chemical classification of drugs by integrating multiple data sources. *Bioinformatics* 2015;**31**:1788–95. <https://doi.org/10.1093/bioinformatics/btv055>
- Olson T, Singh R. Predicting anatomic therapeutic chemical classification codes using tiered learning. *BMC Bioinformatics* 2017;**18**:1–13.
- Zhao H, Duan G, Ni P. et al. RNPredATC: a deep residual learning-based model with applications to the prediction of drug-ATC code association. *IEEE/ACM Trans Comput Biol Bioinform* 2021;**20**:2712–23.
- Peng Y, Wang M, Yixiang X. et al. Drug repositioning by prediction of drug's anatomical therapeutic chemical code via network-based inference approaches. *Brief Bioinform* 2021;**22**:2058–72. <https://doi.org/10.1093/bib/bbaa027>
- Chen L, Jing X, Zhou Y. PDATC-NCPMKL: predicting drug's anatomical therapeutic chemical (ATC) codes based on network consistency projection and multiple kernel learning. *Comput Biol Med* 2024;**169**:107862. <https://doi.org/10.1016/j.combiomed.2023.107862>
- Szklarczyk D, Santos A, Von Mering C. et al. STITCH 5: augmenting protein–chemical interaction networks with tissue and affinity data. *Nucleic Acids Res* 2016;**44**:D380–4. <https://doi.org/10.1093/nar/gkv1277>
- European Bioinformatics Institute. RDKit: Open-Source Cheminformatics. <https://www.rdkit.org/>, 2024.

32. Hattori M, Tanaka N, Kanehisa M. et al. SIMCOMP/SUBCOMP: chemical structure search servers for network analyses. *Nucleic Acids Res* 2010;**38**:W652–6.
33. Wikipedia contributors. *Simplified Molecular-Input Line-Entry System* 2024. (1 June 2004, date last accessed)..
34. Weininger D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J Chem Inf Comput Sci* 1988;**28**:31–6. <https://doi.org/10.1021/ci00057a005>
35. Duvenaud DK, Maclaurin D, Iparraguirre J. et al. Convolutional networks on graphs for learning molecular fingerprints. *Advances in neural information processing systems* 2015;**28**.
36. Kearnes S, McCloskey K, Berndl M. et al. Molecular graph convolutions: moving beyond fingerprints. *J Comput Aided Mol Des* 2016;**30**:595–608. <https://doi.org/10.1007/s10822-016-9938-8>
37. Wikipedia contributors. *Chemical Table File* 2024. (1 June 2004, date last accessed)..
38. Ogata H, Goto S, Sato K. et al. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res* 1999;**27**:29–34. <https://doi.org/10.1093/nar/27.1.29>
39. Kim S, Chen J, Cheng T. et al. Pubchem 2023 update. *Nucleic Acids Res* 2023;**51**:D1373–80. <https://doi.org/10.1093/nar/gkac956>
40. Gaulton A, Bellis LJ, Patricia Bento A. et al. ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Res* 2012;**40**:D1100–7. <https://doi.org/10.1093/nar/gkr777>
41. Chou K-C. Some remarks on protein attribute prediction and pseudo amino acid composition. *J Theor Biol* 2011;**273**:236–47. <https://doi.org/10.1016/j.jtbi.2010.12.024>
42. Internet Archive. <https://archive.org/web/>, 2024. (2 June 2004, date last accessed).
43. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks arXiv preprint arXiv:1609.02907. 2016.
44. Hamilton W, L, Ying R, Leskovec J. Inductive representation learning on large graphs. In: *Advances in Neural Information Processing Systems, NIPS'17*, pp. 1025–35. Red Hook, NY, USA: Curran Associates Inc., 2017.
45. Velickovic P, Cucurull G, Casanova A. et al. Graph attention networks. In: *International Conference on Learning Representations*, 2018.
46. Park J, Shim Y, Lee F. et al. Prediction and interpretation of polymer properties using the graph convolutional network. *ACS Polymers Au* 2022;**2**:213–22. <https://doi.org/10.1021/acspolymersau.1c00050>
47. Queen O, McCarver GA, Thatigotla S. et al. Polymer graph neural networks for multitask property learning. *npj Comput Mater* 2023;**9**:90.
48. Zhang X, Sheng Y, Liu X. et al. Polymer-unit graph: advancing interpretability in graph neural network machine learning for organic polymer semiconductor materials. *J Chem Theory Comput* 2024;**20**:2908–20. <https://doi.org/10.1021/acs.jctc.3c01385>
49. Li G, Xiong C, Thabet A. et al. DeeperGCN: all you need to train deeper GCNs arXiv preprint arXiv:2006.07739 2020.
50. Chou K-C. Some remarks on predicting multi-label attributes in molecular biosystems. *Mol Biosyst* 2013;**9**:1092–100. <https://doi.org/10.1039/c3mb25555g>
51. Velickovic P, Cucurull G, Casanova A. et al. Graph attention networks stat 2017;**1050**:10–48550.
52. Xu K, Hu W, Leskovec J. et al. How powerful are graph neural networks? arXiv preprint arXiv:1810.00826, 2018.
53. Wang M, Da Zheng ZY, Gan Q. et al. Deep graph library: a graph-centric, highly performant package for graph neural networks arXiv preprint arXiv:1909.01315. 2019.
54. Kreuzer D, Beaini D, Hamilton W. et al. Rethinking graph transformers with spectral attention. *Advances in Neural Information Processing Systems* 2021;**34**:21618–29.
55. Rampásek L, Galkin M, Dwivedi VP. et al. Recipe for a general, powerful, scalable graph transformer. *Advances in Neural Information Processing Systems* 2022;**35**:14501–15.
56. Shirzad H, Vellingker A, Venkatachalam B. et al. Exphormer: sparse transformers for graphs. In: *International Conference on Machine Learning*, pp. 31613–32. PMLR, 2023.
57. Wang C, Tsepa O, Ma J. et al. Graph-Mamba: towards long-range graph sequence modeling with selective state spaces arXiv preprint arXiv:2402.00789 2024.