SUPPLEMENTAL MATERIALS

This supplementary material provides additional details supporting the explanation workflows and case studies presented in the main paper. Section A describes the system implementation. Section B presents the prompt templates used by the *Neighborhood Explainer* and *Path Explainer*; Section C provides the custom prompts used in the case studies; and Section D compares the proposed human–AI collaborative workflow with current reaction-atlas exploration practices.

A IMPLEMENTATION DETAILS

The TMAP layout and associated metadata are precomputed using a standalone Python script. The resulting data are exported as a JavaScript file using the Faerun visualization library [4] and loaded into the *Explainable Reaction Atlas* interface at runtime. Once the file is loaded, users can interactively explore the visualization and perform LLM-assisted analyses. The system is implemented as a web application with a Python (Flask) backend and a JavaScript frontend. LLM-generated explanations are produced using OpenAI's GPT-5 mini through the backend, which is model-agnostic and can be readily configured to use alternative LLMs.

Precomputing the reaction atlas separates computationally intensive layout construction from interactive exploration. At runtime, the interface manages user selections, retrieves the corresponding reaction records and metadata, and forwards the selected instances to the explanation backend. This separation maintains interface responsiveness while enabling a consistent explanation pipeline across interaction modes.

Each explanation request is constructed solely from the user's current selection rather than the entire dataset. Neighborhood selections are represented as unordered sets of reaction records, whereas path selections preserve the sequential order of reactions along the selected TMAP path. In label-aware mode, reaction class labels are provided alongside the reaction SMILES; in label-free mode, the labels are omitted, requiring the model to infer recurring patterns directly from the reaction strings. The language model is not fine-tuned on the Schneider 50K dataset. Instead, its responses are conditioned entirely on the prompt template, selected reaction records, optional class labels, and requested analysis task.

This design keeps the explanation pipeline model-agnostic and facilitates its application to new reaction datasets and alternative LLMs without task-specific retraining. Although fine-tuning can improve domain-specific performance and reduce hallucinations in some settings, it may specialize the model to a particular data distribution, limiting flexibility across datasets and requiring additional fine-tuning for each new dataset [3, 7]. It also requires task-specific training data and high-quality explanation annotations, increasing development effort and working against the system's goal of supporting rapid exploratory analysis [7].

Prompt design. The prompt templates are designed to support exploratory visual analysis by generating concise, region-level descriptions of reaction subsets rather than detailed analyses of individual instances. Adapted from the prompting strategies of Explainable Mapper [8] and tailored to chemical reactions, the templates guide the model toward distinguishing characteristics of selected latent-space regions, including recurring reaction types, reagents, bond changes, and functional-group transformations.

The templates are intentionally conservative, requesting recurring patterns and representative transformations while avoiding unsupported mechanistic details. For comparison tasks, prompts emphasize contrastive evidence, encouraging the model to explain what distinguishes two selections rather than summarize them independently. This distinction is important in visual analytics, where users may seek to understand why nearby regions are separated or distant regions are connected. To support efficient exploration, the prompts emphasize semantic abstraction over individual reactions and constrain responses to concise summaries and descriptors. Structured JSON output enables consistent downstream processing and presentation. By default, responses consist of a concise summary and a small set of keywords to facilitate rapid exploration while minimizing cognitive load. These outputs serve as starting points rather than substitutes for detailed inspection; users can refine the default prompt or provide custom prompts for more targeted or detailed analyses.

B PROMPT TEMPLATES

For the *Neighborhood Explainer*, Figure 1 and Figure 2 present the prompt templates used for neighborhood summarization and neighborhood comparison, respectively. Likewise, Figure 3 and Figure 4 present the corresponding prompt templates for path summarization and path comparison in the *Path Explainer*.

Instruction: You are a computational chemist analyzing chemical reactions given as reaction SMILES.
Identify:
- Common reaction types or mechanisms
- Shared functional group transformations
- Recurring bond changes or reagents
- The chemical rationale for grouping these reactions together
Focus on patterns common across the set, not individual reactions. Include brief examples of recurring transformations where helpful.

Summarize in under 50 words and list 3 descriptors. Provide your response strictly in the following JSON format:

```
{
  "keywords": string[],
  "summary": string
}
```

Input:
[*Instances in the selected neighborhood*]

Figure 1: Prompt template for summarizing an individual TMAP neighborhood.

Instruction:
You are a computational chemist analyzing two neighborhoods of reaction SMILES representing related transformations.
Analyze the transition between neighborhoods by:
(1) Identifying shared reactants, products, catalysts, or functional group transformations (if any).
(2) Highlighting differences in mechanisms or structures.
(3) Summarizing how one neighborhood shifts into the other (if there is overlap).

Summarize in under 50 words and list 3 descriptors. Provide your response strictly in the following JSON format:

```
{
  "keywords": string[],
  "summary": string
}
```

Input:
Overlapping Instances: [*Instances in both neighborhoods*]
Unique Instances 1: [*First neighborhood*]
Unique Instances 2: [*Second neighborhood*]

Figure 2: Prompt template for comparing two TMAP neighborhoods.

Instruction:
You are a computational chemist analyzing how reactions evolve across a sequence of reactions, represented by reaction SMILES. Identify specific recurring reactants, products, catalysts, reagents, or functional group changes that are shared within and across reactions. Clearly reference representative reactants, catalysts, or products

Summarize in under 50 words and list 3 descriptors. Provide your response strictly in the following JSON format:

```
{
  "keywords": string[],
  "summary": string
}
```

Input:
Path: [*Instances in the path in order*]

Figure 3: Prompt template for summarizing a TMAP path.

Instruction:
You are a computational chemist analyzing how reaction patterns evolve across two reaction pathways represented as sequences of reaction SMILES.
Your task:
- Identify recurring motifs within each pathway
- Compare patterns across the two pathways
- Highlight whether transformations are analogous or mechanistically distinct between pathways
Explicitly cite representative reaction SMILES or fragments from each pathway. Point to specific molecules or substructures that illustrate the pattern.

Summarize in under 50 words and list 3 descriptors. Provide your response strictly in the following JSON format:

```
{
  "keywords": string[],
  "summary": string
}
```

Input:
Path: [*Ordered path instances*]

Figure 4: Prompt template for comparing two TMAP paths.

C Case Study Custom Prompts

Figure 5 and Figure 6 present the custom prompt templates used in the Label-Aware and Label-Free Explanation Modes case study. Figure 7 presents the custom prompt used to analyze the neighborhood of a locally inconsistent reaction.

Instruction:
You are a computational chemist analyzing a path through a reaction latent-space atlas. The input is an ordered sequence of reactions from a TMAP path. Each item contains a path index, a reaction SMILES, and optionally a reaction class label.
This path was selected because it appears to transition between two reaction classes: 3.1.5 and 1.7.9. Identify the reaction(s) or adjacent path segment(s) that define the chemical boundary between the classes.
Analyze the path by: (1) segmenting the ordered reactions into chemically coherent regions; (2) identifying the most likely boundary position(s) using path indices; (3) explaining the chemical evidence before and after the boundary using reaction SMILES; and (4) reporting whether the transition appears gradual, abrupt, or ambiguous. If class labels are provided, use them only as auxiliary context and ground the explanation in the reaction SMILES.
Return strictly valid JSON with fields for boundary indices, boundary reactions, pre-boundary pattern, post-boundary pattern, transition type, confidence, summary, and keywords.
Input:
Path: [*Ordered path instances*]

Figure 5: Condensed custom prompt for identifying the reactions that define a transition boundary along a TMAP path. The version used in the interface additionally instructs the model to return a concise summary of its analysis together with the corresponding keywords.

Instruction:
You are a computational chemist analyzing a set of reactions from different reaction classes, represented by reaction SMILES. Explain the key chemical similarities that connect these reactions along a path transitioning between reaction classes 3.1.5 and 1.7.9.
Identify recurring reactants, products, catalysts, reagents, functional groups, reaction motifs, or transformation patterns that are shared across reactions from both classes. Focus on common chemical features that could explain their similarity, and clearly reference representative reactants, catalysts, products, or functional group changes. Avoid highlighting features that are unique to only one reaction class unless they directly support a broader shared pattern. If class labels are provided, use them only as auxiliary context and ground the explanation in the reaction SMILES.

Summarize in under 50 words and list 3 descriptors. Provide your response strictly in the following JSON format:

```
{
  "keywords": string[],
  "summary": string
}
```

Input:
Path: [*Ordered path instances*]

Figure 6: Custom prompt for identifying the chemical similarities that connect reactions from different classes along a TMAP path.

Instruction:
You are analyzing a selected neighborhood from a chemical reaction atlas.
Do not assume in advance whether the reactions in this neighborhood are chemically consistent or inconsistent.
Based only on the provided reaction SMILES and, if available, the class labels, evaluate whether this neighborhood appears chemically consistent.
Explain your reasoning clearly. In particular:
- What chemical similarities, if any, make the reactions seem consistent?
- What differences, if any, make the reactions seem inconsistent?
- If the reactions belong to different classes, do those classes still share enough chemistry to justify their proximity?
- Could the observed neighborhood plausibly reflect a meaningful chemical relationship, or does it seem more likely to be a limitation of the graph/MST representation?
Conclude with one of the following:
- consistent
- partially consistent
- inconsistent
- insufficient evidence

Return concise structured JSON:

```
{
  "chemical_similarities": ["...", "..."],
  "chemical_differences": ["...", "..."],
  "class_relationship": "...",
  "representation_assessment": "...",
  "conclusion": "consistent | partially consistent |
                 inconsistent | insufficient evidence"
}
```

Input:
[*Instances in the selected neighborhood*]

Figure 7: Custom prompt for explaining locally inconsistent reactions.

D Comparison with current practice

Existing reaction atlas exploration workflows use TMAP to analyze the global organization of reaction space and inspect nearest neighbors of selected reactions to identify local relationships [1, 2, 5, 6]. Analysts typically examine multiple metadata color maps and manually inspect reaction SMILES to identify patterns within neighborhoods or along paths. This process is time-consuming, becomes increasingly difficult as the number of selected reactions and metadata attributes grows, and is inherently limited to the available metadata. In contrast, *Explainable Reaction Atlas* automatically summarizes selected regions, identifies recurring reaction motifs and distinguishing characteristics directly from the reaction representations, and generates natural-language explanations within the visualization. This enables users to rapidly explore both known patterns encoded by existing metadata and previously unannotated chemical relationships that may inform the creation of additional metadata for further analyses.

References

[1] V. S. Boorla and C. D. Maranas. Catpred: a comprehensive framework for deep learning in vitro enzyme kinetic parameters. *Nature Communications*, 16(1):2072, 2025. doi: 10.1038/s41467-025-57215-9

[2] A. Capecchi, D. Probst, and J.-L. Reymond. One molecular fingerprint to rule them all: drugs, biomolecules, and the metabolome. *Journal of Cheminformatics*, 12(1):43, 2020. doi: 10.1186/s13321-020-00445-4

[3] G. Niess, H. Razouk, S. Mandic, and R. Kern. Addressing hallucination in causal Q&A: The efficacy of fine-tuning over prompting in LLMs. In *Proceedings of the Joint Workshop of the 9th Financial Technology and Natural Language Processing (FinNLP), the 6th Financial Narrative Processing (FNP), and the 1st Workshop on Large Language Models for Finance and Legal (LLMFinLegal)*, pp. 253–258. Association for Computational Linguistics, Abu Dhabi, UAE, 2025.

[4] D. Probst and J.-L. Reymond. FUn: a framework for interactive visualizations of large, high-dimensional datasets on the web. *Bioinformatics*, 34(8):1433–1435, 2018. doi: 10.1093/bioinformatics/btx760

[5] D. Probst and J.-L. Reymond. Visualization of very large high-dimensional data sets as minimum spanning trees. *Journal of Cheminformatics*, 12(1):12, 2020. doi: 10.1186/s13321-020-0416-x

[6] P. Schwaller, D. Probst, A. C. Vaucher, V. H. Nair, D. Kreutter, T. Laino, and J.-L. Reymond. Mapping the space of chemical reactions using attention-based neural networks. *Nature Machine Intelligence*, 3(2):144–152, 2021. doi: 10.1038/s42256-020-00284-w

[7] M. Szep, D. Rueckert, R. von Eisenhart-Rothe, and F. Hinterwimmer. Fine-tuning large language models with limited data: A survey and practical guide. *Transactions of the Association for Computational Linguistics*, 14:341–377, 2026. doi: 10.1162/tacl.a.627

[8] X. Yan, R. Sevastjanova, S. van der Ben, M. El-Assady, and B. Wang. Explainable Mapper: Charting LLM embedding spaces using perturbation-based explanation and verification agents. *IEEE Transactions on Visualization and Computer Graphics*, 2026. To appear.