A Blueprint for Open Science: How Transatlantic Teams Built and Deployed Knowledge Graphs to Enable Biological (AI) Models

Mina P. Peyton [1], **Cheng-Han Chung** [2], **Samarpan Mohanty** [3], **Van Q. Truong** [4], **Andrew Scouten** [5], **Sangeeta Shukla** [6], **Chantera Lazard** [7], **Daniall Masood** [8], **John D. Murphy** [9], **Anne Ketter** [10], **Taha Mohseni Ahooyi** [6], **Viren R. Amin** [11], **Arshi Arora** [12], **J. Allen Baron** [13], **Fateema Bazzi** [14], **Arun Bondali**[15], **Nishat Anjum Bristy** [16], **E. Kathleen Carter** [2], **Yibei Chen** [17], **Jean Paul Courneya** [18], **Camille Daniels** [19], **Marcin J. Domagalski** [20], **Victor Felix** [13], **Vivien W. Ho** [21], **Michelle Holko** [10], **Toshiaki Katayama** [22], **Yaphet Kebede** [2], **Mariia Kim** [8], **Raghavendra Kini**[15], **Carmen Lidia Diaz Soria** [21], **Anurag Limdi** [21], **Shakuntala Mitra** [23], **Evan Molinelli** [24], **Evan Morris** [2], **Aymen Maqsood Mulbagal** [25], **Bijan Paul** [26], **Seeta Ramaraju Pericherla** [27], **Kara Quaid** [28], **Radu Robotin** [29], **Polina Rusina** [21], **Irene Lopez Santiago** [21], **Ben Stear** [6], **Karthick Subramanian** [27], **Shilpa Sundar** [30], **Likhitha Surapaneni** [27], **Deanne M. Taylor** [31], **Simone Weyand** [27], **Benjamin Wingfield** [27], **Christine Withers** [21], **David Yu Yuan** [27], **Emily Richardson**[32], **Beryl Rabindran** [32], **and Ben Busby** [33]

1 Bioinformatics and Computational Biosciences Branch, OCICB, NIAID, NIH, Bethesda, MD, USA **2** Renaissance Computing Institute,University of North Carolina-Chapel Hill, Chapel Hill, NC, USA **3** University of Nebraska-Lincoln, Lincoln, NE, USA **4** Institute for Biomedical Informatics, University of Pennsylvania, Philadelphia, PA, USA **5** Texas State University, San Marcos, TX, USA **6** Department of Biomedical and Health Informatics, Children's Hospital of Philadelphia, Philadelphia, PA, USA **7** Northeastern University Boston, Boston, MA, USA **8** The George Washington University, Washington, DC, USA **9** EPAM Systems, Newtown, PA, USA **10** Computercraft Corporation, Washington DC, USA **11** BioSkryb Genomics, Durham, NC, USA **12** Volastra Therapeutics, New York, NY, USA **13** Institute for Genome Sciences, University of Maryland School of Medicine, Baltimore, MD, USA **14** Independent Researcher, Ann Arbor, MI, USA **15** Astrazeneca, Gaithersburg, MD, USA **16** Carnegie Mellon University, Pittsburgh, PA, USA **17** McGovern Institute for Brain Research, Massachusetts Institute of Technology, Cambridge, MA, USA **18** Center for Vaccine Development and Global Health, University of Maryland School of Medicine, Baltimore, MD, USA **19** Medical Device Innovation Consortium, Arlington, VA, USA **20** ICF International, Reston, VA, USA **21** Open Targets, European Molecular Biology Laboratory, EBI, Hinxton, UK **22** Database Center for Life Science, Japan **23** Johns Hopkins University, Baltimore, MD, USA **24** Chan Zuckerberg Initiative, Redwood City, CA, USA **25** Northeastern University, Boston, MA, USA **26** School of Computing, University of Nebraska-Lincoln, Lincoln, NE, USA **27** European Molecular Biology Laboratory, EBI, Wellcome Genome Campus, Hinxton, CB10 1SD, UK **28** Children's Tumor Foundation, New York, NY, USA **29** CloudR Solutions, North Potomac, MD, USA **30** Carolina Health Informatics Program, University of North Carolina at Chapel Hill, NC, USA **31** Department of Biomedical and Health Informatics, Children's Hospital of Philadelphia, University of Pennsylvania Perelman Medical School, Philadelphia, PA, USA **32** Amazon Web Services, Arlington, VA, USA **33** NVIDIA, CA, USA

Authors are listed according to their contributions, followed by alphabetical order.

BioHackathon series:
NVIDIA - AWS Open Data Knowledge Graph Hackathon
Arlington, VA, USA, 2025
NVIDIA - AWS Open Data Knowledge Graph Hackathon Group

Submitted: 25 Nov 2025

Introduction

Knowledge graphs (KGs) are structured representations of information that model real-world entities and their relationships in a network format. Unlike traditional databases that store data in tables, KGs organize information as nodes (entities) connected by edges (relationships) that may carry labels or attributes providing semantic meaning to those connections. As directed, labeled graphs, KGs associate meaning with nodes and edges; information can be incorporated through manual curation, semi-automated extraction, or fully automated data integration methods (Chaudhri et al., 2022; Hogan et al., 2021; Peng et al., 2023). Once established, these graphs can be efficiently explored through graph navigation where search and query operations can be performed, enabling complex reasoning over large, heterogeneous datasets (Chaudhri et al., 2022). Although semantic networks predate KGs, modern KGs extend these earlier constructs by incorporating ontologies or schema layers, achieving massive scalability (e.g., applying graph algorithms to billions of entities), and integrating multimodal and cross-domain data sources resulting in more comprehensive structured knowledge (Chaudhri et al., 2022).

In parallel, biomedical investigators are increasingly adopting large language models (LLMs) to accelerate discovery. However, LLMs often lack transparency into the evidence underpinning their outputs (Joseph et al., 2025; Liao & Vaughan, 2024; Palikhe et al., 2025). This opacity is problematic for biomedical research, where verifiable, evidence-based reasoning is essential. Furthermore, generative artificial intelligence (GenAI) systems are prone to hallucinations, generating plausible-sounding but false statements (Huang et al., 2025; Z. Xu et al., 2025). Thus, for safe application of GenAI in biomedical research, its outputs must be valid, traceable, and contextually grounded. To address these challenges, the graph-based retrieval-augmented generation (GraphRAG) framework integrates knowledge graph structure into the retrieval and reasoning process of LLMs (Potts, 2024; Shi et al., 2025). GraphRAG reduces hallucinations observed in free-form GenAI systems by constraining generation to graph-anchored evidence, and scales to heterogeneous biomedical datasets without discarding graph topology. This approach enhances both interpretability and trust in LLM-driven biomedical discovery.

The following sections describe the seven transatlantic projects undertaken during the NVIDIA - AWS Open Data Knowledge Graph Hackathon focused on building or integrating KGs and deploying the GraghRAG framework. The collaborative event brought together interdisciplinary teams of scientists, bioinformaticians, computational biologists, data scientists, and software engineers to prototype and deploy pipelines that connected heterogeneous biomedical datasets. The hackathon leveraged AWS Open Data resources (Registry of Open Data on AWS), AWS Neptune (a graph database service, Build Graph Applications - Amazon Neptune), and open-source tools to construct and/or integrate biomedical KGs. Collectively, these projects and pipelines showcase methods for constructing KGs from existing biomedical datasets and exemplify the practical deployment of GraphRAG in real-world biomedical context, thereby advancing the biomedical sciences through the creation of new KGs and the promotion of evidence-grounded, graph-aware GenAI methodologies.

GeNETwork

Cancer remains a leading cause of mortality worldwide, accounting for nearly one in six deaths globally (*WHO Cancer*, 2025). Treatment selection is increasingly guided by molecular profiling of patient tumors, leveraging genomic data to identify targetable alterations. The explosion of genomic sequencing data (Katz et al., 2021), coupled with expanding databases of drug-disease associations, pathway annotations, and clinical variant interpretations, has created unprecedented opportunities for precision oncology. However, these data sources remain largely siloed across specialized databases, each using distinct identifier systems and data models (Triplet & Butler, 2011). Researchers seeking to connect molecular variants observed in patient samples to potential therapeutic interventions must manually navigate multiple disconnected resources, a time-consuming process that limits systematic exploration of treatment options.

Knowledge graphs offer a promising approach to integrate heterogeneous biological and pharmacological data into unified, queryable frameworks. Recent implementations, such as Petagraph, developed by some members of our hackathon group, demonstrate the feasibility of large-scale multi-omics integration with over 32 million nodes (Stear et al., 2024). However, reproducibility remains a critical challenge in knowledge graph development. A recent analysis found only 0.4% of published KGs provide sufficient code and data for reproduction (Babalou et al., 2023). Furthermore, most knowledge graphs present final products without transparently documenting integration challenges, limiting practical guidance for future development efforts.

To address these gaps, we developed GeNETwork during the hackathon focusing on cancer data integration. GeNETwork is a multi-scale knowledge graph connecting variants, genes, pathways, diseases, and drugs data from multiple sources. The graph architecture supports pathway-centric queries, enabling systematic exploration from biological mechanisms to therapeutic opportunities. Addressing the documented reproducibility crisis where only 0.4% of knowledge graphs provide adequate reproduction materials, we make all data files, loading scripts, and documentation publicly available, with transparent documentation of integration challenges to guide future development efforts.

ECoGraph

Colorectal cancer remains a significant public health burden, ranking as the third most common cancer by both incidence (9.6% of new cancer cases per year) and mortality (9.3% of cancer deaths) globally (Bray et al., 2024). Recent epidemiological evidence indicates that colorectal cancer incidence has been rising among younger populations around the world, highlighting an urgent need for improved understanding of disease mechanisms and identification of actionable therapeutic targets (Lui et al., 2019; Vuik et al., 2019). This shifting demographic pattern underscores the importance of age-stratified analyses in colorectal cancer research to identify potential age-specific biomarkers and treatment strategies.

The heterogeneity of colorectal cancer presents both challenges and opportunities for precision oncology approaches. Multiple subtypes of colorectal cancer have been characterized based on histological features and anatomical subsites, each carrying distinct survival patterns and clinical outcomes (Lech et al., 2016; Q. Li et al., 2024). Furthermore, colorectal cancer incidence and mortality rates exhibit substantial variation across racial and ethnic groups and other demographic factors in the United States, suggesting that both genetic and environmental determinants contribute to disease progression and patient outcomes (Carethers, 2021; Díaz-Gay et al., 2025). Although molecular profiling of colorectal cancer has advanced considerably, critical knowledge gaps remain regarding which specific genes are associated with favorable or unfavorable survival outcomes, particularly when stratified by demographic and clinical characteristics.

To address this, we developed an integrative analytical framework that combines genomic and proteomic data to identify potential biomarkers and therapeutic targets for colorectal adenocarcinoma (COAD). We selected colorectal adenocarcinoma for this study because of its status as a disease of public health concern and because it is well-represented in both The Cancer Genome Atlas (TCGA) and the Clinical Proteomic Tumor Analysis Consortium (CPTAC), enabling cross-platform validation of our findings (Ellis et al., 2013; Tomczak et al., 2015). We present our findings within a knowledge graph framework that captures the complex relationships between genes, proteins, mutations, clinical outcomes, and patient characteristics. This integrated approach provides a systematic method for identifying and validating potential therapeutic targets while accounting for the demographic and molecular heterogeneity of colorectal cancer.

ClassiGraph: A Colon Adenocarcinoma GNN Based Classifier

Large-scale initiatives such as the Clinical Proteomic Tumor Analysis Consortium (CPTAC) have generated comprehensive multi-omics datasets across several cancer types, such as

breast, colorectal, and ovarian cancers (Ellis et al., 2013). These datasets provide a unique opportunity to study the interplay between genomic alterations and proteomic expression at scale. However, the integration of such high-dimensional, heterogeneous data remains a significant challenge. Traditional machine learning methods struggle to exploit complex relationships across modalities, often leading to suboptimal predictive performance.

Graph neural networks (GNNs) offer a promising solution to this challenge (Şimşek et al., 2025; Zohari & Chehreghani, 2025). By modeling biological entities (e.g., genes, proteins) as nodes and their interactions as edges, GNNs naturally capture the relational structure of molecular systems. This graph-based representation enables the integration of heterogeneous omics data into a unified framework, where both local molecular features and global network context contribute to classification. Applying GNNs to multi-omics data thus has the potential to improve cancer subtype identification, reveal biologically meaningful interactions, and guide personalized treatment strategies.

EasyGiraffe - Validator of multisite polygenicity extraction on sequence graph

Sizeable cancer genome repositories such as TCGA offer a wealth of genomic data with the potential to transform cancer diagnostics, therapeutics, and precision medicine (Tomczak et al., 2015). However, extracting meaningful polygenic insights from this complex and heterogeneous dataset remains a major challenge. Traditional methods for variant calling rely heavily on alignment against a single linear reference genome, typically GRCh38 (DePristo et al., 2011; H. Li & Durbin, 2009). While effective in many settings, this approach often fails to account for the extensive genomic variation across human populations, leading to biases in variant detection and interpretation.

Recent advancements in pangenomics have introduced graph-based representations of the genome, which encode known population-level variations directly into the reference itself (Paten et al., 2017). Tools such as the Variation Graph (VG) toolkit enable the construction of sequence graphs from short reads, allowing for the generation of pangenome structures that better reflect the genetic diversity of the sample cohort (Garrison et al., 2018). These graphs not only improve alignment accuracy but also enhance the detection of structural and polygenic variants that may be missed by linear references. The GIRAFFE mapper, a component of the VG toolkit, further accelerates this process by providing fast and accurate read alignment against these variation graphs.

Yet, despite these methodological improvements, there remains a critical need for comprehensive validation frameworks. The existing pipelines tackle production steps —such as alignment, variant calling and indexing—and do not offer an integrated method to evaluate the accuracy and reproducibility of polygenic variant extraction, particularly in the context of cancer genomes where multicentric and population-specific variants can play pivotal roles.

To address this gap, we introduce a simulator-based validation framework tailored for multicentric polygenic variant extraction. This simulator generates synthetic sequencing data (FASTQ files) embedded with known variants across multiple loci and samples. When processed through the variant calling pipeline, these outputs enable robust benchmarking by comparing detected variants against the known ground truth.

Model Integration and Data Assembly System (MIDAS)

By integrating diverse datasets into a connected graph, KGs enable discovery of hidden associations and support advanced querying. However, the full potential of KGs is often constrained by heterogeneity across data sources, including differences in formats, identifiers, and semantic standards. Data in the biomedical field often follow different standards or ontologies based on the use case for that particular data. This presents a challenge in connecting KGs together and making them interoperable.

To address this, modular KGs can be developed and integrated into existing frameworks by adopting shared identifier spaces and community-driven data models. This project focuses on investigating data from cBioPortal (Cerami et al., 2012), CIViC (Griffith et al., 2017), and 1000Genome (Fairley et al., 2020). Modular KGs from these data sources will help in understanding what data needs to be collected and normalized. NCATS Translator Babel Node Normalizer helps normalize identifiers and the Biolink Model provides a standardized schema that allows for data to be integrated seamlessly with existing knowledge graphs (Fecho et al., 2025; Unni et al., 2022). We have created the Model Integration and Data Assembly System (MIDAS), an established pipeline that takes these existing tools and converts and normalizes data to allow for these modular KGs to be created and connected to existing KGs. It allows for interoperability of current resources but also creates scalable pathways for integrating future datasets, ultimately enhancing the utility of KGs in biomedical research.

KG Model Garbage Collection

Biomedical KGs are powerful tools for linking genes, diseases, and phenotypes — but when AI models generate new edges, they often hallucinate or introduce errors. Our project focuses on pruning these errors. KG Model Garbage Collection is a proof-of-concept for how a combination of human review, grounded AI, and graph learning can work together to keep biomedical knowledge graphs accurate and trustworthy (Putman et al., 2024).

KG Model Garbage Collection uses a subset of a trusted graph (Monarch) (Putman et al., 2024) and randomly removes some edges. Then, three strategies are used to fill the missing edges: random guessing, a general LLM, and an LLM using RAG. Participants (SMEs) validate (some of) these edges through a simple interface to evaluate how close each method comes to the truth. This data is used to train a graph neural network to see if it can automatically spot questionable edges and flag them for review and removal. The resulting knowledge graph is tested against the original, trusted knowledge graph (Payong, 2025; Yu et al., 2024).

BioGraphRAG

The limitations of generative AI (GenAI) systems are well documented, notably their propensity to produce hallucinated information and their lack of mechanisms for tracing outputs to verifiable sources of evidence (Huang et al., 2025; Joseph et al., 2025; Liao & Vaughan, 2024; Palikhe et al., 2025; Z. Xu et al., 2025). In biomedical research, reliability and reproducibility is paramount and such limitations are unacceptable. Therefore, a GenAI system that generates responses grounded explicitly in verified biomedical knowledge and peer-reviewed publications is essential. BioGraphRAG addresses this need by producing natural language responses that are anchored in biomedical data and publication-based KGs. Specifically, BioGraphRAG integrates the Precision Medicine Knowledge Graph (PrimeKG) (Chandak et al., 2023)–which encodes disease–gene–drug relationships with the PubMed Knowledge Graph (PubKG) (J. Xu et al., 2025)–representing literature and citation networks– into a unified property graph. The integration leverages the GraphRAG framework and is implemented using the G-retriever architecture (Lewis et al., 2021; Shi et al., 2025). Through this design, BioGraphRAG ensures that every generated response is grounded in validated relationships extracted from the underlying graphs, thereby enhancing transparency, trustworthiness, and auditability.

Briefly, to tackle biomedical question answering, we combined graph-native retrieval with neural reasoning. For each question Qi, the system retrieves a targeted subgraph $Gi' \subseteq G = (V,E)$ that maximally captures relevant entities and relations, and passes this evidence to a graph-aware neural reader. The retrieval step is formalized as a Prize-Collecting Steiner Tree (PCST) optimization problem, which identifies the minimal, high-value subgraph connecting relevant biomedical concepts. The GNN + LLM hybrid reader then conditions its reasoning on this subgraph to generate a response set $Ai \subseteq V$.

This architecture mitigates hallucinations by constraining generation to graph-grounded evidence, while enabling scalable reasoning across dense, multimodal biomedical domains. By

preserving graph topology and integrating structured retrieval with neural generation, BioGraphRAG aligns with the GraphRAG paradigm—delivering precise, evidence-backed, and interpretable answers suitable for biomedical discovery.

Methods/Implementation

The hackathon was conducted at the AWS Skills Center in Arlington, VA and European Bioinformatics Institute, Cambridge, UK. Both locations provided a collaborative environment equipped with high-performance computing resources and cloud-based services. Participants utilized AWS Open Data platform, Neptune graph database service, and a suite of open-source technologies to design, build, and integrate KGs. Additionally, team collaboration was coordinated through Slack for communication and GitHub for project documentation, version tracking, and code management. The following sections describe the methods and implementation of each project.

GeNETwork

We developed GeNETwork, a multi-scale knowledge graph integrating cancer genomics and pharmacological data to answer questions that would aid in making better personalized treatment decisions. Our data was pulled from multiple sources including OpenTargets Platform version 25.09 (drug-disease associations and drug-target data), TCGA (variant-cohort relationships), OncoDB (gene-disease associations), and multiple pathway databases (Gene Ontology, MSigDB, Reactome, WikiPathways). The graph was constructed in Neo4j using Cypher loading scripts. Data was standardized to triple format and loaded with unique constraints on identifiers. Our data files are hosted on Open Science Framework (OSF) and our scripts and some additional information can be found on our Github repository.

ECoGraph

This research developed a knowledge graph-based framework to identify molecular drivers and therapeutic targets for colorectal cancer, with a specific focus on younger patient populations. The approach integrated multiple data types, including genetic markers, proteomic data, clinical outcomes, and patient demographics, into a comprehensive graph structure to capture complex biological relationships. Using data from TCGA, we stratified patients by age (≤ 50 vs > 50 years) and performed Cox proportional hazards regression analyses to identify genes whose mutations showed different survival associations between age groups, prioritizing those with substantial differences in hazard ratios as potential age-specific therapeutic targets.

ClassiGraph

A multi-omics graph was constructed from CPTAC COAD data by integrating proteomic, phosphoproteomic, RNA, CNV, and mutation profiles annotated by CMS subtypes. Data were cleaned, harmonized across gene identifiers, imputed, and z-normalized using tumor–normal baselines while preserving feature availability masks. Protein–protein edges were derived from feature-space k-nearest-neighbor similarity (or, additionally, high-confidence STRING-DB interactions), gene–protein links were based on shared identifiers, and patient–protein mutation edges represented individual alterations. The resulting heterogeneous graph was processed with a two-layer GraphSAGE-based HeteroConv model that learned embeddings across node types and predicted CMS subtypes from pooled patient representations. Stratified data splits, class-weighted loss, and weighted sampling were used to address CMS label imbalance, with model selection guided by validation macro-F1 and final testing on a held-out test cohort.

EasyGiraffe

We used the JaSaPaGe pangenome graph developed by Kulmatov et al. as the backbone for variant extraction (Kulmanov et al., 2025). This graph, constructed from whole-genome sequences of 10 Japanese and 9 Saudi Arabian individuals, encoded common population-level variants and served as the reference structure. To evaluate the accuracy of variant detection, we developed a simulator that generated synthetic sequencing reads (FASTQ files) embedded with known variants. These reads were designed to reflect realistic multicentric polygenic configurations and were produced in short-read or long-read formats. The synthetic reads were mapped to the JaSaPaGe backbone using the GIRAFFE mapper from the VG toolkit, which performed graph-based alignment and outputted giraffe.gam alignment files. Variant calling was then conducted using the VG pipeline, producing VCF files containing the detected variants. The expected output was the accurate identification of the genetic variants implanted by the simulator in the synthetic FASTQ samples. The simulation will generate SNP, MNP, InDel, inversions and translocations. The simulation was repeated across multiple replicates and background graphs to evaluate reproducibility and performance across population-specific contexts.

MIDAS

We built our knowledge graph by leveraging existing open tools including Node Normalizer for identifier harmonization, the Biolink Model for semantic alignment, and ORION for graph construction. To demonstrate the feasibility of combining heterogeneous datasets into a modular knowledge graph, we collected data from CIViC, cBioPortal, and the 1000 Genomes Project, focusing on variants located on chromosome 6. From CIViC, we ingested curated variant records linked to allele registry IDs in addition to diseases with which they are associated. CIVic also contains information about therapies applied to treat diseases. From this, we can construct various edge types such as variant-genetically_associated_with-disease and drug_applied_to_treat-disease. In addition to the edges provided by CIVic, we also produced genetically_associated_with condition edges from cBioPortal. Information from 1000 Genomes enriches our graph by adding population frequency data to the variant nodes and is further annotated with predicted molecular consequences using Ensembl Variant Effect Predictor (VEP).

The transformation pipeline standardized all the entities collected from the primary data sources using node normalizer, ensuring that all nodes resolve to consistent Biolink-compliant identifiers. The node normalizer ensures that the name space IDs obtained from each resource are harmonized to align under one single representation. This allows for data to be connected and uncover relationships between knowledge graphs and data that may have been hidden. By using the semantic framework developed by the Biolink Model, we are representing relationships in a consistent way. Once the data is normalized and harmonized, we can merge the primary data sources into an interoperable knowledge graph using ORION.

We have also developed a pipeline to convert the output to a format ready for upload to Neptune. Uploading the knowledge graph to Amazon Neptune provides a scalable, high-performance graph database where all nodes, edges, and metadata can be stored in a query-optimized format. Once loaded, the graph can be accessed using openCypher or Gremlin queries, enabling fast exploration of complex biological relationships across diseases, drugs, genes, and proteins. By connecting Neptune to a Model Context Protocol (MCP) agent, the graph becomes directly usable within AI workflows: the MCP can expose endpoints for status checks, schema inspection, and query execution, allowing AI systems to dynamically pull structured biomedical knowledge into reasoning pipelines, enrich responses with curated evidence, and support interactive discovery at scale.

KG Model Garbage Collection

We developed the KG Model Garbage Collection Tool as a proof-of-concept framework to evaluate and correct AI-generated edges in biomedical knowledge graphs. The workflow begins with a trusted subset of the Monarch Knowledge Graph (Putman et al., 2024) related to Alzheimer's disease, derived via Cypher query against a Neo4j API provided by ROBOKOP (Bizon et al., 2019), from which 50% of predicates were randomly removed to create incomplete subgraphs for reconstruction experiments. These masked graphs simulate real-world data incompleteness, allowing systematic testing of edge generation and validation strategies. Missing edges were then synthetically reconstructed using three approaches: (1) Random Guessing, which randomly pairs entities and relations as a baseline; (2) a General LLM hosted on AWS Bedrock, which predicts relations through structured batch prompts; and (3) an LLM augmented with retrieval (RAG) from NCBI E-utilities and PubTator3 to ground predictions in biomedical literature. Each approach produces a set of candidate edges representing potential hallucinations or errors for downstream evaluation.

To benchmark and refine these reconstructions, we extracted 29 representative "backbone" edges and expanded each into local subgraphs within a three-hop radius, resulting in a contextual dataset of 222 verified ground-truth edges linking genes, diseases, phenotypes, and biological processes. These focused graphs were used to prototype a graph neural network (GNN) model designed to discriminate between valid and spurious edges. The model architecture, inspired by Relational Graph Convolutional Networks (R-GCNs), integrates local graph structure, edge type frequency, and AI-assigned prediction scores to generate an edge trustworthiness score. Planned extensions include a Human-in-the-Loop (HITL) validation layer where subject-matter experts review model-flagged edges through a lightweight web interface, with their feedback aggregated into confidence-weighted labels that guide iterative retraining.

This modular pipeline—implemented in PyTorch and DGL—was designed for scalability and transparency. Each stage of the process, from graph preprocessing and edge simulation to model training and evaluation, is saved in structured CSV or LMDB formats for reproducibility. The initial minimal dataset supports rapid experimentation and retraining cycles ($<$5 minutes), while the architecture is designed to scale seamlessly to larger biomedical graphs such as the full 1,754-edge Alzheimer's subgraph. Together, this workflow demonstrates how combining grounded AI, human validation, and graph learning can support trustworthy curation and maintenance of biomedical knowledge graphs at scale.

BioGraphRAG

BioGraphRAG is a system that utilizes KGs and GraphRAG for biomedical question and answer. We fused PrimeKG (Chandak et al., 2023) with PubKG (J. Xu et al., 2025) into one property graph for precise, auditable reasoning. In parallel, we embedded textual surfaces (paper titles/abstracts, trial summaries, entity names) into dense vectors and indexed them in OpenSearch kNN (HNSW), enabling semantic retrieval that handles synonyms, abbreviations, and paraphrases. At query time, the question is embedded and kNN retrieves the most semantically relevant seeds across PrimeKG and PubKG. From these seeds, we ran controlled graph expansions (bounded hops/degree) in the unified graph to gather mechanistic and contextual evidence: mentions, citations, disease–gene–drug links, and trial associations. Lightweight graph features (e.g., node2vec) can rerank evidence. The final answer is composed from this subgraph with explicit citations and provenance. Operationally, data lands in S3, loads into the graph in Neptune, and vectors live in OpenSearch.

Operation

Seven prototype projects were completed during the NVIDIA - AWS Open Data Knowledge Graph Hackathon. These projects present innovative pipelines for building KGs or integrating

KGs with GraphRAG. The following section provides an overview of the workflow and outlines the minimal system requirements needed to run the software for each project.

GeNETwork

GitHub, Usage Guide

Workflow Overview By harmonizing multiple distinct data sources, GeNETwork provides unified access to previously siloed cancer biology and pharmacology datasets. Users can initiate queries starting with a specific variant, gene, pathway, disease, or drug, and traverse the graph to discover related entities and relationships. We demonstrated that the knowledge graph supports pathway-centric queries, enabling identification of genes in specific biological processes and their associated mutation profiles across cancer types. Users can generate induced subgraphs, extracting focused networks for detailed analysis or downstream machine learning applications. Network statistics including degree distribution, betweenness centrality, and clustering coefficients can be computed on these subgraphs to identify regulatory nodes and network properties.

System Requirements GeNETwork data files are hosted on the Open Science Framework, while Cypher loading scripts are available on GitHub. Users can browse source data and documentation with no installation beyond git cloning and dataset access. For users who wish to reconstruct the knowledge graph locally, Neo4j Desktop (version 5.x or later) is required, with a minimum 16 GB RAM, and 10 GB disk space. R (version 4.x) or Python 3.x are optional for further data preprocessing.

ECoGraph

GitHub, Usage Guide

Workflow Overview ECoGraph's analysis proceeded through seven sequential stages: (1) acquisition and harmonization of TCGA and CPTAC colorectal cancer datasets; (2) quality control and preprocessing of genomic, proteomic, and clinical data; (3) age-stratified survival analysis using Cox proportional hazards regression; (4) construction and population of the knowledge graph with statistical results; (5) validation of TCGA-derived associations in the independent CPTAC cohort; (6) development of enhanced prognostic models incorporating proteomic data; and (7) generation of survival curves and knowledge graph visualizations.

System Requirements A Unix-like operating system with the Conda package manager installed. A minimum of 10-15 GB of available disk space is necessary to accommodate project data.

All software dependencies are specified in Conda environment files, available in the repository:

- *scripts/conda_environment_full.yml*: Provides a complete environment with all dependencies pinned to exact versions for maximum reproducibility.
- *scripts/conda_environment.yml*: Provides a minimal environment with flexible versioning.

Core dependencies managed by these environments include bedtools (~2.26.0), bcftools (~1.20), htslib (~1.20), and R (4.3.x).

ClassiGraph

GitHub, Usage Guide

Workflow Overview ClassiGraph integrates multi-omics and ontology-based data into a unified knowledge graph where nodes and edges capture biological relationships. From these relationships, users can train graph neural network models for tasks like cancer sub-type classification. The resulting embeddings and predictions can then be evaluated, visualized, and exported for downstream biological analysis.

System Requirements ClassiGraph is implemented in Python 3.12 and has been utilized in high-performance environments such as AWS SageMaker. It requires a Unix-based system with Python, Git, and pip installed. All dependencies are listed in the pyproject.toml file and can be installed using standard Python environment managers (e.g., pip install -e .). Refer to the repository's README for detailed setup instructions and environment configuration steps.

EasyGiraffe

GitHub, Usage Guide

Workflow Overview EasyGiraffe is a simulator-based benchmarking framework designed to evaluate the accuracy of polygenic variant extraction using graph-based genome references and the VG toolkit. To get started, first ensure you meet the system requirements. Clone the repository and navigate to the bootstrap-scripts directory, then run the setup scripts in sequence: first download-pangenome-data.sh to fetch the JaSaPaGe pangenome reference graph, then install_vg.sh and install-tools.sh to install the VG toolkit and ART simulator, and finally disease_to_variant_resolver.sh with a disease name (e.g., "Sickle Cell Anemia") to generate simulated FASTQ reads with implanted variants. Once setup is complete, you'll have a validated VG environment ready for read mapping against the pangenome graph using VG GIRAFFE, variant calling to produce VCF files, and evaluation of detected variants against known ground truth data to assess precision, recall, and F1-scores—making it particularly useful for validating variant calling pipelines across diverse population structures.

System Requirements A Unix-based system with bash, wget, curl, git, Python 3.7+, conda, and a C++ compiler along with at least 100 GB of free disk space.

MIDAS

GitHub, Usage Guide

Workflow Overview The Model Integration and Data Assembly System (MIDAS) allows for users to take heterogeneous datasets and allow for them to be made into a modular knowledge graph which can then be connected to an external knowledge graph. Because entities are harmonized with Node Normalizer and aligned with the Biolink Model, users can combine and query data from multiple sources without dealing with inconsistent identifiers. For example, curated variant–disease–therapy associations from CIViC, disease links from cBioPortal, and population frequency data from the 1000Genomes Projects are all represented in a unified, semantically consistent structure.

By deploying the graph to Amazon Neptune, researchers can execute openCypher or Gremlin queries to retrieve disease-specific variants, identify therapies linked to particular genetic alterations, or examine population-level variant frequencies. The Biolink-compliance of the pipeline also means that the graph can be connected to external resources such as the ROBOKOP Knowledge Graph (Bizon et al., 2019; Morton et al., 2019), allowing users to expand their analyses beyond our integrated sources.

Finally, the system can be accessed through an agentic large language model with graph-based retrieval-augmented generation (graph-RAG). Users can pose natural language questions (e.g., "Which therapies target EGFR variants on chromosome 6?"), and the MCP-enabled agent will issue structured queries to Neptune and return evidence-grounded responses. This approach makes the knowledge graph directly usable in AI-driven discovery workflows, enabling clinicians, bioinformaticians, and data scientists to interactively explore complex biological relationships at scale.

System Requirements To run MIDAS locally, you need Python 3.12+ and uv for environment and dependency management. For optional graph database deployment, you'll also need an AWS setup with AWS CLI, awscurl, and jq, along with access to an Amazon Neptune cluster (with OpenCypher) or Neo4j for local development.

KG Model Garbage Collection

GitHub, Usage Guide

Workflow Overview The system proceeds in four main stages: (1) Graph preparation, where a subset of the Monarch Knowledge Graph is downloaded and partially masked to simulate missing or noisy edges; (2) Synthetic edge generation, using Random, LLM, and LLM + RAG strategies to reconstruct missing links; (3) Edge evaluation and model training, where reconstructed graphs are compared against ground truth and used to train a GNN classifier that predicts edge trustworthiness; and (4) Human-in-the-loop validation, where domain experts review model-flagged edges through a lightweight web interface, providing labels that feed back into iterative model retraining. This workflow allows researchers to experiment with scalable curation strategies for improving the reliability of AI-augmented biomedical knowledge graphs.

User Interaction and Front-End Our prototype includes a conceptual front-end designed to bridge human and AI-assisted curation workflows. Built with React and Flask, the interface offers an intuitive environment where users can explore, validate, and refine graph-derived edges generated by the GNN models. Developers can currently interact with the system in developer mode by running the components in order — data preparation, model training, and front-end launch — to visualize flagged edges, review predictions, and log curation outcomes. As the interface matures, the goal is to create a fully interactive UX that integrates directly with the backend pipeline, allowing researchers to approve, reject, or comment on edges in real time.

System Requirements All components of the KG Model Garbage Collection Tool were developed and tested in a Linux-based environment (Ubuntu 22.04) using Python 3.11. Core dependencies include PyTorch 2.1, Deep Graph Library (DGL) 1.1, pandas, networkx, and Flask for the web-based validation interface. The pipeline can be executed on a standard workstation with 16 GB RAM, 4 CPU cores, and at least 10 GB of free disk space. GPU acceleration (CUDA 11+) is optional but recommended for faster GNN training. Full dependency specifications and setup instructions are provided in the project's requirements.txt file and README.

BioGraphRAG

GitHub, Usage Guide

Workflow Overview The system is designed to transform a free-text biomedical question into a grounded, citation-supported answer. First, the user provides a natural language query, which is embedded into a dense vector representation. This embedding is used to query OpenSearch indices constructed from both PrimeKG and PubKG, yielding the top-k semantically relevant seed nodes and document fragments. Next, the seed nodes are expanded within Amazon Neptune using openCypher queries, restricted t o one or two hops with label filters and degree caps to control graph explosion. The resulting expanded subgraph is then pruned via a Prize-Collecting Steiner Tree (PCST)-like optimization to yield a compact, evidence-rich subgraph that balances semantic relevance with structural connectivity.

This subgraph is subsequently encoded by a GNN to enrich node representations, which are combined with textual attributes and evidence snippets to form a serialized graph context. The graph context, along with the original question, is passed to an instruction-tuned LLM. The LLM produces an answer that is explicitly grounded in the evidence, including PubMed IDs (PMIDs) and experiment identifiers, thereby reducing hallucinations and increasing reproducibility. Optionally, the output can include an evidence table summarizing the supporting snippets. This pipeline thus integrates graph-based retrieval, neural pruning, and language generation to provide accurate, evidence-linked biomedical question answering.

System Requirements The software requires Python version 3.11 or higher and a minimum of 16 GB of system memory to ensure smooth execution of graph expansion and LLM inference. A processor with at least four cores is recommended, and while the system can run on CPU-only

environments, an NVIDIA CUDA-enabled GPU with at least 12 GB of VRAM is strongly recommended to accelerate graph neural network (GNN) training and inference. Approximately 20 GB of available disk space is necessary to store the biomedical knowledge graph, embeddings, and intermediate artifacts. The core dependencies include PyTorch (with CUDA support if available), PyTorch Geometric, OpenSearch 2.x with the k-NN plugin for vector retrieval, Amazon Neptune (with openCypher enabled) or Neo4j for local development, and FastAPI for service orchestration. An active LLM provider key (e.g., OpenAI, Anthropic, or a local deployment) is required to generate answers.

Discussion

Across all seven prototype systems, the use of open biomedical KGs and graph-based retrieval methods presents both significant opportunities and inherent limitations. Open KGs such as PrimeKG, PubKG, and other public datasets often reflect underlying data and curation biases—favoring well-studied genes, diseases, and pathways while underrepresenting rare or emerging areas (Karki et al., 2025; Norori et al., 2021). Static graph snapshots and heterogeneous schema designs may introduce temporal and structural inconsistencies, and text-derived graphs can inherit linguistic and publication biases from their source literature (Koukaras, 2025; Zhang et al., 2025). Although the integration of GraphRAG and related graph-aware approaches constrains generative outputs to verifiable, evidence-grounded knowledge, it cannot fully eliminate hallucinations or inference drift (M. Li et al., 2025). Retrieval gaps, incomplete entity coverage, and overreliance on densely connected graph regions may still influence reasoning outcomes (Ju et al., 2024; Tian et al., 2022). Addressing these limitations will require continual data updates, ontology harmonization, and transparent provenance tracking to strengthen the reliability and interpretability of graph-driven biomedical AI systems.

Future Vision

The NVIDIA–AWS Open Data Knowledge Graph Hackathon convened a diverse transatlantic cohort of researchers, computational biologists, data scientists, and software engineers to collaboratively advance methods for integrating heterogeneous biomedical datasets into KGs and promoting the adoption of GraphRAG frameworks. Over three intensive days, participants completed seven prototype projects, each demonstrating innovative strategies for KG construction or deploying graph-based, evidence-grounded generative AI. The following section outlines the prospective directions and development priorities identified by each team to extend these prototypes into reproducible, scalable, and open-source solutions for the biomedical community.

GeNETwork

Critical technical improvements include variant identifier normalization (TCGA variants currently in chromosome-position format require conversion to HGVSg using Ensembl VEP), HGNC gene nomenclature standardization, and disease ontology mapping to unify EFO/MONDO terms across data layers, enabling seamless cross-layer queries.

Several prepared datasets await integration, including CIViC clinical variant interpretations for therapeutic actionability annotations; Molecular Targets Project and FDA Pediatric Molecular Target List data to fulfill the original pediatric oncology focus; StringDB protein-protein interactions, which connect genes through functional relationships beyond pathway membership, reducing network fragmentation for genes lacking formal pathway annotations; and GTEx gene co-expression data for functional relationship context. These additions would substantially enhance GeNETwork's analytical capabilities and pediatric oncology applications.

Scalable infrastructure deployment via cloud-based graph databases would support integration of all datasets and enable development of accessible, user-friendly interfaces for clinical researchers without bioinformatics expertise, broadening GeNETwork's impact. Integration of clinical

outcome data would enable validation of computational predictions and support development of predictive models for treatment response, advancing precision oncology applications.

ECoGraph

To further enrich the knowledge graph and enhance the predictive power of our models, our immediate next steps will focus on incorporating multi-omics data. Specifically, we plan to integrate epigenomic and transcriptomic data sourced from TCGA to capture a more comprehensive view of regulatory mechanisms influencing the disease state. Following this, we will validate our findings by testing the resulting knowledge graph in independent cohorts, such as those available through the CPTAC, and incorporate proteomic data to bridge the gap between genomic alterations and their functional consequences at the protein level. Finally, to facilitate more dynamic and sophisticated interpretation of the complex network of molecular and clinical relationships, we intend to explore integrating the knowledge graph with generative AI services, such as Amazon Bedrock, to derive novel, mechanistically relevant hypotheses and insights from the integrated multi-omic data.

ClassiGraph

Potential next steps could include collaborating with related teams, such as ECoGraph, to enhance interoperability and integration across CPTAC and TCGA datasets. The project could also expand to additional CPTAC cohorts beyond COAD to improve model generalization and robustness across diverse cancer types. Building on this foundation, further work might optimize the GNN architecture by incorporating recent state-of-the-art designs as starting points. To enhance interpretability, explainable GNN techniques such as GNNExplainer and integrated gradients could be explored. Lastly, the GNN's performance could be systematically compared with baseline models, including random forests and multi-layer perceptrons (MLPs), to more clearly contextualize its predictive effectiveness.

EasyGiraffe

We successfully developed a comprehensive pipeline to extract pathogenic variants from user-queried phenotypic features or diseases, generate synthetic sequence reads with embedded variants, and output the assigned variants from the simulator. The simulator was designed with adaptable parameters, enabling precise control over minor allele frequencies in the simulated sequences to reflect diverse population genetic scenarios. Future development could further expand the capabilities of the simulator to generate long reads up to 1 megabase in length and to accommodate more complex variant types, including large insertions and deletions (InDels), inversions, and translocations, thereby providing a robust tool for comprehensive genomic variant simulation and analysis.

MIDAS

We developed a scalable data integration pipeline leveraging the Node Normalizer, the Biolink Model, and ORION to harmonize and integrate heterogeneous biomedical datasets into an interoperable knowledge graph. This framework enables consistent semantic representation of biomedical entities and relationships, facilitating cross-dataset interoperability and deployment at scale within Amazon Neptune. The resulting graph architecture can seamlessly link to external biomedical KGs, including ROBOKOP and GWAS, thereby enhancing data connectivity and discovery potential. Future work focuses on expanding the pipeline to incorporate additional biomedical resources and advancing GraphRAG applications using Neptune and agentic LLMs. These extensions aim to enable context-aware reasoning, automated hypothesis generation, and interactive exploration across federated knowledge graphs to accelerate biomedical discovery and translational research.

KG Model Garbage Collection

Building on the current work, our future directions aim to significantly enhance the system's performance, scalability, and usability. We plan to immediately expand human validation by upgrading the existing web interface to support expert curation, edge review, and direct feedback loops. This will tightly link human insight to iterative GNN retraining cycles. Concurrently, we will rigorously benchmark our edge reconstruction pipelines, evaluating Random, LLM, and LLM + RAG approaches using key metrics like precision, recall, and AUC to assess performance and scalability. For broader utility, the workflow will be scaled and extended to cover larger modules of the Monarch KG, integrating GraphRAG capabilities to allow for interactive, LLM-driven exploration. We also intend to test the method on other relevant biomedical KGs, specifically those produced by collaborating groups described in this paper.

BioGraphRAG

BioGraphRAG integrated PrimeKG and PubKG into a unified property graph to enable precise and auditable reasoning. We utilized the full PrimeKG dataset; however, due to the large size of PubKG (217.92 GB) and time constraints during the hackathon, only a 1 GB subset containing biomedical entities and paper linkage data was extracted and ingested. Unlike PrimeKG—which provides readily formatted node and edge files—PubKG required additional preprocessing before integration, resulting in longer data preparation times.

Future development of BioGraphRAG will focus on improving data coverage, scalability, and interpretability to address current biases and technical constraints. Expanding ingestion capacity for large-scale resources such as PubKG and integrating multimodal data—including clinical, imaging, and molecular profiles—will enable richer, context-aware reasoning. Dynamic graph updating pipelines will be implemented to ensure timely incorporation of new biomedical literature and database releases, thereby reducing temporal drift. Enhanced ontology alignment and cross-source entity resolution will improve semantic consistency across integrated datasets. To further minimize hallucination and strengthen evidence attribution, future iterations will incorporate human-in-the-loop validation and confidence scoring mechanisms that quantify the provenance and reliability of retrieved graph evidence. Finally, a graphical user interface will be developed to enhance accessibility for researchers and clinicians. Collectively, these efforts aim to advance BioGraphRAG toward a robust, continually learning framework for transparent and trustworthy generative AI in biomedicine.

Data and Software Availability

All codes and scripts for all seven projects are under an open MIT license.

GeNETwork

All processed data files and knowledge graph triples are available on the Open Science Framework. Cypher loading scripts and documentation are available on GitHub. The integrated knowledge graph uses data from TCGA, OncoDB, OpenTargets Platform, and pathway databases (Gene Ontology, MSigDB, Reactome, WikiPathways). Additional curated datasets including CIViC, Molecular Targets Project (MTP), and FDA Pediatric Molecular Target List (PMTL) are available on OSF for future integration.

ECoGraph

The knowledge graphs and scripts for producing this work are available in an open-source repository on GitHub. Data used to populate the graphs were sourced from TCGA.

ClassiGraph

All data is available through a frozen S3 bucket of CPTAC through Open Data on AWS, with data loading methods implemented in the code. There are also other versions of the data available through the Registry of Open Data on AWS. Specifically, our implementation uses AWS's data freeze of CPTAC v1.2.

EasyGiraffe

All reference pangenomes are available on JASAPAGE GitHib repo. The software tools used in this study are publicly available through open-source repositories. The variation graph toolkit used for genomic analysis is available at vg. The Phenome-Mapper tool for phenotypic mapping and analysis is available at Phenome-Mapper. Both repositories include documentation, installation instructions, and example usage. All custom scripts and analysis pipelines developed for this study, including the knowledge graph construction and query methods, are available from the corresponding authors upon reasonable request.

MIDAS

All source code, scripts and documentation are available on the MIDAS GitHub Repository. Data used to build modular knowledge graphs is available at CIViC, 1000Genomes, and TCGA. The generated modular graphs were connected to ROBOKOP.

KG Model Garbage Collection

All source code, scripts, and documentation for the KG Model Garbage Collection Tool are openly available on the KG Model Garbage Collection GitHub Repository. Processed graph datasets and synthetic edge reconstruction outputs are derived from the publicly available Monarch Knowledge Graph (Monarch Initiative, v2024), which integrates data from resources such as OMIM, GO, HGNC, NCBI Gene, and UPheno. Retrieval-augmented generation (RAG) components additionally query open biomedical text sources via NCBI E-utilities API and PubTator3. A permanent archived version of the code and datasets may be deposited on Zenodo and/or Amazon ODP for long-term access.

BioGraphRAG

Data files for the PrimeKG and PubKG can be found here: PrimeKG - Harvard Dataverse, PubMed knowledge graph 2.0

All documentation, code, and scripts for BioGraphRAG are available on the BiographRAG GitHub Repository

Competing Interests

BR and ER are full time employees of AWS. BB is a full time employee of NVIDIA. JM is a full time employee of EPAM Systems, Inc. VH is a full time employee of EMBL-EBI. RR is a full time employee of CloudR Solutions.

Grant Information

This project has been funded in part with Federal funds from the National Institute of Allergy and Infectious Diseases (NIAID), National Institutes of Health, Department of Health and Human Services under BCBB Support Services Contract HHSN316201300006W/75N93022F00001 to Guidehouse Digital. VQT is supported by the Microsoft Research PhD Fellowship, ACM SIGHPC Computational and Data Science Fellowship. JAB is supported by the National Institutes of Health–National Human Genome Research Institute (NHGRI) 5U24HG012557-04. YC is supported by NIH U24MH136628 and 5U24MH130918-04. DT is supported by

NIH 1U24OD038422. DM is supported by NIH U24 1U24OD038423-01. SRP is funded by the PARADIGM initiative, Wellcome Trust WT226083/Z/22/Z. LS is funded by the Wellcome Trust WT222155/Z/20/Z. ROBOKOP U24: Supporting Biomedical Discovery with the ROBOKOP Graph Knowledgebase: NIH 5-U24-ES035214-02. BioData Catalyst: NIH 1-OT3-HL147154. DOGSLED for Data, Ontologies, and Graphs Supporting Learning and Enhanced Discovery: NIH/NCATS 1-OT2-TR005712-01. MATRIX: ML/AI-Aided Therapeutic Repurposing in Extended Uses: ARPA-H 140D042490001-2024-1001-SOW002.

Acknowledgements

We thank the following individuals for their contributions to code generation and technical support across multiple teams, which substantially accelerated project development during the hackathon: Taylor Teske - AWS, Archana Sharma - AWS, Emmanuela Derival - AWS, Jessie Johnson - AWS, Ashley Chen - AWS, Kayla Taylor - AWS, Cristian Chicas - AWS, Gargi Singh Chhatwal - AWS, Nate Haynes - AWS, Gregg Grieff - AWS, and Rishi Puri - NVIDIA.

Supplemental information

A list of resources introducing knowledge graphs for biomedical data was put together in preparation for the hackathon. This list includes relevant literature, github repos, technical blogs, videos, tools, and workspace resources. We are making this available to the community so that anyone looking to learn about KGs for biomedical datasets can use this list to get a head start. The supplemental resources and materials are available here: ODP Hackathon Supplemental Materials.

Github Repos

Team 1: https://github.com/collaborativebioinformatics/GeNETwork.git GeNETwork is a reproducible multi-source knowledge graph integrating cancer genomics and pharmacological data for precision oncology applications, with ongoing development toward pediatric cancer therapeutic decision support.

Team 2: https://github.com/collaborativebioinformatics/Cancer_target_KG ECoGraph is a knowledge graph that integrates genomic and clinical data from colorectal cancer patients to identify biomarkers and, thus, potential therapeutic targets, t hat are associated with early-onset disease.

Team 3: https://github.com/collaborativebioinformatics/Proteomic_Genomic_Cancer_KG ClassiGraph: a Colon Adenocarcinoma GNN Based Classifier is a system of multi-omnics graph-based models including GNNMutation and a custom implementation that leverages a heterogeneous graph-based framework for cancer detection and subtype classification.

Team 4: https://github.com/collaborativebioinformatics/GiraffeAgent2 EasyGiraffe: Sequence simulator and variant validator of genetic variations generated by vg

Team 5: https://github.com/collaborativebioinformatics/Adding_Datasets_to_KG Creating modular knowledge graphs from primary datasets which adopt shared identifier spaces and defined semantic models to create interoperable knowledge graphs by connecting to existing knowledge graphs.

Team 6: https://github.com/collaborativebioinformatics/Model_Garbage_Collection The KG-LLM Garbage Collection Tool uses a combination of human review, grounded AI, and graph learning to identify and prune erroneous edges in biomedical knowledge graphs, improving their accuracy and trustworthiness in a more automated human-in-the-loop fashion.

Team 7: https://github.com/collaborativebioinformatics/BioGraphRAG BioGraphRAG bridges biomedical data and publication knowledge graphs with GraphRAG, based upon the G-Retriever

architecture, to enhance and ensure that the natural language responses are generated solely from trusted knowledge sources.

References

Babalou, S., Samuel, S., & König-Ries, B. (2023). *Reproducible Domain-Specific Knowledge Graphs in the Life Sciences: A Systematic Literature Review*. https://doi.org/10.48550/arXiv.2309.08754

Bizon, C., Cox, S., Balhoff, J., Kebede, Y., Wang, P., Morton, K., Fecho, K., & Tropsha, A. (2019). ROBOKOP KG and KGB: Integrated Knowledge Graphs from Federated Sources. *Journal of Chemical Information and Modeling*, *59*(12), 4968–4973. https://doi.org/10.1021/acs.jcim.9b00683

Bray, F., Laversanne, M., Sung, H., Ferlay, J., Siegel, R. L., Soerjomataram, I., & Jemal, A. (2024). Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, *74*(3), 229–263. https://doi.org/10.3322/caac.21834

Carethers, J. M. (2021). Racial and ethnic disparities in colorectal cancer incidence and mortality. *Advances in Cancer Research*, *151*, 197–229. https://doi.org/10.1016/bs.acr.2021.02.007

Cerami, E., Gao, J., Dogrusoz, U., Gross, B. E., Sumer, S. O., Aksoy, B. A., Jacobsen, A., Byrne, C. J., Heuer, M. L., Larsson, E., Antipin, Y., Reva, B., Goldberg, A. P., Sander, C., & Schultz, N. (2012). The cBio Cancer Genomics Portal: An Open Platform for Exploring Multidimensional Cancer Genomics Data. *Cancer Discovery*, *2*(5), 401–404. https://doi.org/10.1158/2159-8290.CD-12-0095

Chandak, P., Huang, K., & Zitnik, M. (2023). Building a knowledge graph to enable precision medicine. *Scientific Data*, *10*(1), 67. https://doi.org/10.1038/s41597-023-01960-3

Chaudhri, V. K., Baru, C., Chittar, N., Dong, X. L., Genesereth, M., Hendler, J., Kalyanpur, A., Lenat, D. B., Sequeda, J., Vrandečić, D., & Wang, K. (2022). Knowledge graphs: Introduction, history, and perspectives. *AI Magazine*, *43*(1), 17–29. https://doi.org/10.1002/aaai.12033

DePristo, M. A., Banks, E., Poplin, R., Garimella, K. V., Maguire, J. R., Hartl, C., Philippakis, A. A., Angel, G. del, Rivas, M. A., Hanna, M., McKenna, A., Fennell, T. J., Kernytsky, A. M., Sivachenko, A. Y., Cibulskis, K., Gabriel, S. B., Altshuler, D., & Daly, M. J. (2011). A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, *43*(5), 491–498. https://doi.org/10.1038/ng.806

Díaz-Gay, M., Dos Santos, W., Moody, S., Kazachkova, M., Abbasi, A., Steele, C. D., Vangara, R., Senkin, S., Wang, J., Fitzgerald, S., Bergstrom, E. N., Khandekar, A., Otlu, B., Abedi-Ardekani, B., Carvalho, A. C. de, Cattiaux, T., Penha, R. C. C., Gaborieau, V., Chopard, P., … Alexandrov, L. B. (2025). Geographic and age variations in mutational processes in colorectal cancer. *Nature*, *643*(8070), 230–240. https://doi.org/10.1038/s41586-025-09025-8

Ellis, M. J., Gillette, M., Carr, S. A., Paulovich, A. G., Smith, R. D., Rodland, K. K., Townsend, R. R., Kinsinger, C., Mesri, M., Rodriguez, H., Liebler, D. C., & on behalf of the Clinical Proteomic Tumor Analysis Consortium (CPTAC). (2013). Connecting Genomic Alterations to Cancer Biology with Proteomics: The NCI Clinical Proteomic Tumor Analysis Consortium. *Cancer Discovery*, *3*(10), 1108–1112. https://doi.org/10.1158/2159-8290.CD-13-0219

Fairley, S., Lowy-Gallego, E., Perry, E., & Flicek, P. (2020). The International Genome Sample Resource (IGSR) collection of open human genomic variation resources. *Nucleic Acids Research*, *48*(D1), D941–D947. https://doi.org/10.1093/nar/gkz836

Fecho, K., Glusman, G., Baranzini, S. E., Bizon, C., Brush, M., Byrd, W., Chung, L., Crouse, A., Deutsch, E., Dumontier, M., Foksinska, A., Hadlock, J., He, K., Huang, S., Hubal, R., Hyde, G. M., Israni, S., Kenmogne, K., Koslicki, D., … Yakaboski, C. (2025). Announcing

the Biomedical Data Translator: Initial Public Release. *Clinical and Translational Science*, *18*(7), e70284. https://doi.org/10.1111/cts.70284

Garrison, E., Sirén, J., Novak, A. M., Hickey, G., Eizenga, J. M., Dawson, E. T., Jones, W., Garg, S., Markello, C., Lin, M. F., Paten, B., & Durbin, R. (2018). Variation graph toolkit improves read mapping by representing genetic variation in the reference. *Nature Biotechnology*, *36*(9), 875–879. https://doi.org/10.1038/nbt.4227

Griffith, M., Spies, N. C., Krysiak, K., McMichael, J. F., Coffman, A. C., Danos, A. M., Ainscough, B. J., Ramirez, C. A., Rieke, D. T., Kujan, L., Barnell, E. K., Wagner, A. H., Skidmore, Z. L., Wollam, A., Liu, C. J., Jones, M. R., Bilski, R. L., Lesurf, R., Feng, Y.-Y., … Griffith, O. L. (2017). CIViC is a community knowledgebase for expert crowdsourcing the clinical interpretation of variants in cancer. *Nature Genetics*, *49*(2), 170–174. https://doi.org/10.1038/ng.3774

Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G. D., Gutierrez, C., Kirrane, S., Gayo, J. E. L., Navigli, R., Neumaier, S., Ngomo, A.-C. N., Polleres, A., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., & Zimmermann, A. (2021). Knowledge Graphs. *ACM Comput. Surv.*, *54*(4), 71:1–71:37. https://doi.org/10.1145/3447772

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., & Liu, T. (2025). A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems*, *43*(2), 1–55. https://doi.org/10.1145/3703155

Joseph, J., Jose, B., & Jose, J. (2025). The generative illusion: How ChatGPT-like AI tools could reinforce misinformation and mistrust in public health communication. *Frontiers in Public Health*, *13*, 1683498. https://doi.org/10.3389/fpubh.2025.1683498

Ju, W., Yi, S., Wang, Y., Xiao, Z., Mao, Z., Li, H., Gu, Y., Qin, Y., Yin, N., Wang, S., Liu, X., Luo, X., Yu, P. S., & Zhang, M. (2024). *A Survey of Graph Neural Networks in Real world: Imbalance, Noise, Privacy and OOD Challenges*. https://doi.org/10.48550/arXiv.2403.04468

Karki, R., Gadiya, Y., Zaliani, A., Pokharel, B., Babaiha, N. S., Ostaszewski, M., Hofmann-Apitius, M., & Gribbon, P. (2025). KGG: A fully automated workflow for creating disease-specific knowledge graphs. *Bioinformatics*, *41*(7), btaf383. https://doi.org/10.1093/bioinformatics/btaf383

Katz, K., Shutov, O., Lapoint, R., Kimelman, M., Brister, J. R., & O'Sullivan, C. (2021). The Sequence Read Archive: A decade more of explosive growth. *Nucleic Acids Research*, *50*(D1), D387–D390. https://doi.org/10.1093/nar/gkab1053

Koukaras, P. (2025). Data Integration and Storage Strategies in Heterogeneous Analytical Systems: Architectures, Methods, and Interoperability Challenges. *Information*, *16*(11), 932. https://doi.org/10.3390/info16110932

Kulmanov, M., Ashouri, S., Liu, Y., Abdelhakim, M., Alsolme, E., Nagasaki, M., Ohkawa, Y., Suzuki, Y., Tawfiq, R., Tokunaga, K., Katayama, T., Abedalthagafi, M. S., Hoehndorf, R., & Kawai, Y. (2025). Phased genome assemblies and pangenome graphs of human populations of Japan and Saudi Arabia. *Scientific Data*, *12*(1), 1316. https://doi.org/10.1038/s41597-025-05652-y

Lech, G., Słotwiński, R., Słodkowski, M., & Krasnodębski, I. W. (2016). Colorectal cancer tumour markers and biomarkers: Recent therapeutic advances. *World Journal of Gastroenterology*, *22*(5), 1745–1755. https://doi.org/10.3748/wjg.v22.i5.1745

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2021). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. https://doi.org/10.48550/arXiv.2005.11401

Li, H., & Durbin, R. (2009). Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics*, *25*(14), 1754–1760. https://doi.org/10.1093/bioinformatics/btp324

Li, M., Miao, S., & Li, P. (2025). *Simple Is Effective: The Roles of Graphs and Large Language Models in Knowledge-Graph-Based Retrieval-Augmented Generation*. https://doi.org/10.48550/arXiv.2410.20724

Li, Q., Geng, S., Luo, H., Wang, W., Mo, Y.-Q., Luo, Q., Wang, L., Song, G.-B., Sheng, J.-P., & Xu, B. (2024). Signaling pathways involved in colorectal cancer: Pathogenesis and targeted therapy. *Signal Transduction and Targeted Therapy*, *9*(1), 266. https://doi.org/10.1038/s41392-024-01953-7

Liao, Q. V., & Vaughan, J. W. (2024). AI Transparency in the Age of LLMs: A Human-Centered Research Roadmap. *Harvard Data Science Review*, *Special Issue 5*. https://doi.org/10.1162/99608f92.8036d03b

Lui, R. N., Tsoi, K. K. F., Ho, J. M. W., Lo, C. M., Chan, F. C. H., Kyaw, M. H., & Sung, J. J. Y. (2019). Global Increasing Incidence of Young-Onset Colorectal Cancer Across 5 Continents: A Joinpoint Regression Analysis of 1,922,167 Cases. *Cancer Epidemiology, Biomarkers & Prevention: A Publication of the American Association for Cancer Research, Cosponsored by the American Society of Preventive Oncology*, *28*(8), 1275–1282. https://doi.org/10.1158/1055-9965.EPI-18-1111

Morton, K., Wang, P., Bizon, C., Cox, S., Balhoff, J., Kebede, Y., Fecho, K., & Tropsha, A. (2019). ROBOKOP: An abstraction layer and user interface for knowledge graphs to support question answering. *Bioinformatics (Oxford, England)*, *35*(24), 5382–5384. https://doi.org/10.1093/bioinformatics/btz604

Norori, N., Hu, Q., Aellen, F. M., Faraci, F. D., & Tzovara, A. (2021). Addressing bias in big data and AI for health care: A call for open science. *Patterns*, *2*(10), 100347. https://doi.org/10.1016/j.patter.2021.100347

Palikhe, A., Yu, Z., Wang, Z., & Zhang, W. (2025). *Towards Transparent AI: A Survey on Explainable Large Language Models*. https://arxiv.org/html/2506.21812v1. https://arxiv.org/html/2506.21812v1

Paten, B., Novak, A. M., Eizenga, J. M., & Garrison, E. (2017). Genome graphs and the evolution of genome inference. *Genome Research*, *27*(5), 665–676. https://doi.org/10.1101/gr.214155.116

Payong, A. (2025). *An Overview of AI Hallucinations with RAG and Knowledge Graphs | DigitalOcean*. https://www.digitalocean.com/community/conceptual-articles/ai-hallucinations-with-rag-and-knowledge-graphs. https://www.digitalocean.com/community/conceptual-articles/ai-hallucinations-with-rag-and-knowledge-graphs

Peng, C., Xia, F., Naseriparsa, M., & Osborne, F. (2023). Knowledge Graphs: Opportunities and Challenges. *Artificial Intelligence Review*, *56*(11), 13071–13102. https://doi.org/10.1007/s10462-023-10465-9

Potts, B. (2024). *GraphRAG: A new approach for discovery using complex information*. https://www.microsoft.com/en-us/research/blog/graphrag-unlocking-llm-discovery-on-narrative-private-data/. https://www.microsoft.com/en-us/research/blog/graphrag-unlocking-llm-discovery-on-narrative-private-data/

Putman, T. E., Schaper, K., Matentzoglu, N., Rubinetti, V. P., Alquaddoomi, F. S., Cox, C., Caufield, J. H., Elsarboukh, G., Gehrke, S., Hegde, H., Reese, J. T., Braun, I., Bruskiewich, R. M., Cappelletti, L., Carbon, S., Caron, A. R., Chan, L. E., Chute, C. G., Cortes, K. G., … Munoz-Torres, M. C. (2024). The Monarch Initiative in 2024: An analytic platform integrating phenotypes, genes and diseases across species. *Nucleic Acids Research*, *52*(D1), D938–D949. https://doi.org/10.1093/nar/gkad1082

Shi, B., Clemedtson, A., Blumenfeld, Z., & Puri, R. (2025). *Boosting Q&A Accuracy with GraphRAG Using PyG and Graph Databases*. https://developer.nvidia.com/blog/boosting-qa-accuracy-with-graphrag-using-pyg-and-graph-databases/. https://developer.nvidia.com/blog/boosting-qa-accuracy-with-graphrag-using-pyg-and-graph-databases/

Şimşek, N. Ö. Özcan, Özgür, A., & Gürgen, F. (2025). GNNMutation: A heterogeneous graph-based framework for cancer detection. *BMC Bioinformatics*, *26*(1), 153. https://doi.org/10.1186/s12859-025-06133-0

Stear, B. J., Mohseni Ahooyi, T., Simmons, J. A., Kollar, C., Hartman, L., Beigel, K., Lahiri, A., Vasisht, S., Callahan, T. J., Nemarich, C. M., Silverstein, J. C., & Taylor, D. M. (2024). Petagraph: A large-scale unifying knowledge graph framework for integrating biomolecular and biomedical data. *Scientific Data*, *11*(1), 1338. https://doi.org/10.1038/s41597-024-04070-w

Tian, L., Zhou, X., Wu, Y.-P., Zhou, W.-T., Zhang, J.-H., & Zhang, T.-S. (2022). Knowledge graph and knowledge reasoning: A systematic review. *Journal of Electronic Science and Technology*, *20*(2), 100159. https://doi.org/10.1016/j.jnlest.2022.100159

Tomczak, K., Czerwińska, P., & Wiznerowicz, M. (2015). The Cancer Genome Atlas (TCGA): An immeasurable source of knowledge. *Contemporary Oncology*, *19*(1A), A68–A77. https://doi.org/10.5114/wo.2014.47136

Triplet, T., & Butler, G. (2011). *Systems Biology Warehousing: Challenges and Strategies toward Effective Data Integration*. ISBN: 978-1-61208-115-1

Unni, D. R., Moxon, S. A. T., Bada, M., Brush, M., Bruskiewich, R., Caufield, J. H., Clemons, P. A., Dancik, V., Dumontier, M., Fecho, K., Glusman, G., Hadlock, J. J., Harris, N. L., Joshi, A., Putman, T., Qin, G., Ramsey, S. A., Shefchek, K. A., Solbrig, H., … Consortium, T. B. D. T. (2022). Biolink Model: A universal schema for knowledge graphs in clinical, biomedical, and translational science. *Clinical and Translational Science*, *15*(8), 1848–1855. https://doi.org/10.1111/cts.13302

Vuik, F. E., Nieuwenburg, S. A., Bardou, M., Lansdorp-Vogelaar, I., Dinis-Ribeiro, M., Bento, M. J., Zadnik, V., Pellisé, M., Esteban, L., Kaminski, M. F., Suchanek, S., Ngo, O., Májek, O., Leja, M., Kuipers, E. J., & Spaander, M. C. (2019). Increasing incidence of colorectal cancer in young adults in Europe over the last 25 years. *Gut*, *68*(10), 1820–1826. https://doi.org/10.1136/gutjnl-2018-317592

WHO Cancer. (2025). https://www.who.int/news-room/fact-sheets/detail/cancer. https://www.who.int/news-room/fact-sheets/detail/cancer

Xu, J., Yu, C., Xu, J., Torvik, V. I., Kang, J., Sung, M., Song, M., Bu, Y., & Ding, Y. (2025). PubMed knowledge graph 2.0: Connecting papers, patents, and clinical trials in biomedical science. *Scientific Data*, *12*(1), 1018. https://doi.org/10.1038/s41597-025-05343-8

Xu, Z., Jain, S., & Kankanhalli, M. (2025). *Hallucination is Inevitable: An Innate Limitation of Large Language Models*. https://doi.org/10.48550/arXiv.2401.11817

Yu, G., Ye, Q., & Ruan, T. (2024). Enhancing Error Detection on Medical Knowledge Graphs via Intrinsic Label. *Bioengineering*, *11*(3), 225. https://doi.org/10.3390/bioengineering11030225

Zhang, B., He, Y., Pintscher, L., Peñuela, A. M., & Simperl, E. (2025). *Schema Generation for Large Knowledge Graphs Using Large Language Models*. https://doi.org/10.48550/arXiv.2506.04512

Zohari, P., & Chehreghani, M. H. (2025). *Graph Neural Networks in Multi-Omics Cancer Research: A Structured Survey*. https://doi.org/10.48550/arXiv.2506.17234

Author contributions

Mina P. Peyton: Writing Cheng-Han Chung: Writing Samarpan Mohanty: Writing Van Q. Truong: Writing Andrew Scouten: Writing Sangeeta Shukla: Writing Chantera Lazard: Writing Daniall Masood: Writing John D. Murphy: Writing Anne Ketter: Writing Beryl Rabindran: Conceptualization, Writing Ben Busby: Conceptualization, Writing