

## REVIEW OPEN ACCESS

# Applicability Assessment of Technologies for Predictive and Prescriptive Analytics of Nephrology Big Data

Riste Stojanov<sup>1</sup>  | Milos Jovanovik<sup>1,2</sup> | Sasho Gramatikov<sup>1</sup> | Igor Mishkovski<sup>1</sup> | Eftim Zdravevski<sup>1</sup> | Darko Sasanski<sup>1</sup> | Zorica Karapancheva<sup>1</sup> | Goce Spasovski<sup>3</sup> | Ivona Vasileska<sup>4</sup>  | Tome Eftimov<sup>5</sup> | Wu Zhuojun<sup>6</sup> | Joachim Jankowski<sup>6,7</sup>  | Dimitar Trajanov<sup>1,8</sup>

<sup>1</sup>Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, Skopje, North Macedonia | <sup>2</sup>Institute of Logic and Computation, TU Wien, Vienna, Austria | <sup>3</sup>Department of Nephrology, Ss. Cyril and Methodius University in Skopje, Skopje, North Macedonia | <sup>4</sup>Faculty of Mechanical Engineering, University of Ljubljana, Ljubljana, Slovenia | <sup>5</sup>Jožef Stefan Institute, Ljubljana, Slovenia | <sup>6</sup>Institute for Molecular Cardiovascular Research IMCAR, University Hospital, Aachen, Germany | <sup>7</sup>School for Cardiovascular Diseases, Maastricht University, Maastricht, the Netherlands | <sup>8</sup>Department of Computer Science, Metropolitan College, Boston University, Boston, USA

**Correspondence:** Riste Stojanov ([riste.stojanov@finki.ukim.mk](mailto:riste.stojanov@finki.ukim.mk))

**Received:** 18 August 2024 | **Revised:** 18 April 2025 | **Accepted:** 25 April 2025

**Funding:** This work is supported by European Cooperation in Science and Technology, CA21165, HORIZON EUROPE Food, Bioeconomy, Natural Resources, Agriculture and Environment, 101060712. The Slovenian Research and Innovation Agency, GC-0001, P2-0098. Deutsche Forschungsgemeinschaft, 322900939, 445703531, 403224013. HORIZON EUROPE Widening participation and spreading excellence, 101159214. Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, KG-Enrich. H2020 Marie Skłodowska-Curie Actions, 722609 764474.

**Keywords:** big data analytics | data integration | data standardization | large language models | nephrology

## ABSTRACT

The integration of big data into nephrology research will open new avenues for analyzing and understanding complex biological datasets, driving advances in personalized management of kidney diseases. This paper describes the multifaceted challenges and opportunities by incorporating big data in nephrology, emphasizing the importance of data standardization, advanced storage solutions, and advanced analytical methods. We discuss the role of data science workflows, including data collection, preprocessing, integration, and analysis, in facilitating comprehensive insights into disease mechanisms and patient outcomes. Furthermore, we highlight predictive and prescriptive analytics, as well as the application of large language models (LLMs) in improving clinical decision-making and enhancing the accuracy of disease predictions. The use of high-performance computing (HPC) is also examined, showcasing its role in processing large-scale datasets and accelerating machine learning algorithms. Through this exploration, we aim to provide a comprehensive overview of the current state and future directions of big data analytics in nephrology, with a focus on enhancing patient care and advancing medical research.

## 1 | Introduction

Chronic kidney disease (CKD) affects more than 10% of the global population, contributing to significant mortality and financial burden [1, 2]. By 2040, CKD-related deaths can reach up to 4 million annually [3]. Big data can help mitigate these issues by

enabling more precise and data-driven interventions, reducing disparities in care, and improving overall disease management. Personalized management of kidney diseases tailors medical treatment to the unique characteristics of each patient, addressing the heterogeneity of these diseases and their complex genetic, environmental, and lifestyle interactions [4, 5]. Leveraging big

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial](https://creativecommons.org/licenses/by-nc/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

© 2025 The Author(s). *Proteomics* published by Wiley-VCH GmbH

data predictive and prescriptive analytics can enhance predictions of disease progression and treatment response, improving patient outcomes [6].

The advent of big data has revolutionized the field of nephrology research [6], providing unprecedented opportunities for the analysis and integration of complex biological data. The ability to collect and process vast amounts of data from various omics platforms—such as genomics, proteomics, and metabolomics—has enabled researchers to gain deeper insights into the molecular mechanisms underlying diseases [6–8]. Big data approaches are of critical importance in the context of chronic kidney disease (CKD), as they enable the integration and analysis of complex, multidimensional datasets—from genomics and proteomics to clinical parameters and longitudinal patient records. This holistic view facilitates the identification of novel biomarkers [21–23], prediction of disease progression [21, 24, 25], and discovery of patient-specific therapeutic targets [26]. By uncovering hidden patterns and interactions that are not discernible through traditional methods, big data empowers precision medicine and supports more effective, individualized care strategies for CKD patients [27].

Big data is characterized by four 'V's: volume, variety, velocity, and veracity [28]. Volume denotes the massive data generated, variety refers to the diverse data types (e.g., structured, unstructured), velocity captures the rapid data generation and processing, and veracity concerns the data's accuracy and reliability [8]. In nephrology, as well as in internal medicine, data is not only heterogeneous but also irregularly obtained from various sources, such as laboratory results, electronic health records (EHRs), imaging, and omics, complicating integration due to differing structures and timing. In the nephrology domain, an enormous amount of data is already generated, including genomic [15], proteomics [19, 20, 16], proteomic [19, 20], and clinical data [9, 11]. This wealth of information, exemplified by the datasets listed in Table 1, is invaluable for the advancement of personalized medicine [29]. These challenges require sophisticated tools to handle variety and velocity. Integrating big data into data science workflows allows for comprehensive analyses, generating new knowledge and actionable insights to address the complexities highlighted earlier.

The data science workflow in the context of medical research involves several key steps: data collection, data preprocessing, data integration, analysis, and interpretation. Initially, large datasets are collected from various sources, including EHRs, biobanks, and omics studies. These datasets undergo preprocessing to ensure quality and consistency. Subsequently, data integration combines multiple data types into a unified framework, enabling comprehensive analyses. Advanced machine learning algorithms are applied to identify patterns and predict disease progression. Finally, the results are interpreted to derive actionable insights to inform clinical decision-making [30, 31].

Despite the potential of big data in advancing medical research, several challenges persist. The large quantity of data presents issues for storage and computation. The variety of data types requires sophisticated integration techniques. The high velocity of data generation requires real-time processing capabilities. Additionally, ensuring the veracity of data is critical, as large

**TABLE 1** | Available datasets.

Name	Size
UK Biobank <sup>a</sup> [9]	379,432 participants (14,170 AKI) [10]
MIMIC-III <sup>b</sup> [11]	6.7 TB (8,770 AKI patients [12])
Nephroseq <sup>c</sup> [13]	36 datasets / 2K samples (nephrology-specific)
NephQTL <sup>d</sup> [14]	187 participants (nephrology-specific)
Gene Expression Omnibus (GEO) <sup>e</sup> [15]	4348 datasets with 7M+ samples
ProteomicsDB <sup>f</sup> [16]	Over 8 TB proteomic data across various human tissues, including the kidney (the quantity is not specified)
Metabolomics Workbench <sup>g</sup> [17]	169K samples (65 kidney studies with 3,939 samples)
Human Metabolome Database (HMDB) <sup>h</sup> [18]	221K small molecule metabolites associated with various organs (does not specify how many are kidney-related)
UniProt <sup>i</sup> [19]	Over 200 million protein sequences, many of which are associated with kidney function
ProteomeXchange <sup>j</sup> [20]	136 datasets annotated as kidney

<sup>a</sup><https://www.ukbiobank.ac.uk/>

<sup>b</sup><https://mimic.mit.edu/docs/iii/>

<sup>c</sup>[www.nephroseq.org](http://www.nephroseq.org)

<sup>d</sup><http://nephqtl.org/>

<sup>e</sup><https://www.ncbi.nlm.nih.gov/geo/>

<sup>f</sup><https://www.proteomicsdb.org/>

<sup>g</sup><https://www.metabolomicsworkbench.org/>

<sup>h</sup><https://hmdb.ca/>

<sup>i</sup><https://www.uniprot.org/>

<sup>j</sup><https://proteomecentral.proteomexchange.org/>

retrospective cohort datasets may suffer from biases and clinical trial data may not reflect real-world conditions, potentially leading to misleading conclusions [29]. Addressing these challenges is essential for harnessing the full potential of big data in nephrology [32, 33]. Resolving these issues requires advanced computational tools and methodologies tailored to handle big data in the biomedical domain [1, 6]. Leveraging big data and machine learning in CKD research holds immense promise for improving disease prediction and personalized management. By overcoming the challenges associated with omics data, we can pave the way for more effective and individualized treatment strategies, ultimately enhancing patient outcomes in nephrology.

## 2 | Nephrology Big Data Challenges

Despite the vast amounts of data available across various nephrology datasets (some examples are given in Table 1), these data are often stored in heterogeneous formats, each complying with different standards, which complicates integration and analysis.

For example, nucleotide sequences and their quality scores are typically stored in the text-based FASTQ format, which is widely used for raw sequencing reads from next-generation sequencing (NGS). Aligned sequence data, on the other hand, is often compressed into the Binary Alignment/Map (BAM) format, derived from the Sequence Alignment/Map (SAM) format. Genetic variants, including single-nucleotide polymorphisms (SNPs) and indels, are stored in the Variant Call Format (VCF). Mass spectrometry data, both raw and processed, is stored using the XML-based mzML format, while genomic features such as genes and transcripts are represented using formats like Gene Transfer Format (GTF) and General Feature Format (GFF). Clinical and genomic data are exchanged and integrated using standards like Health Level Seven (HL7) Fast Healthcare Interoperability Resources (FHIR)<sup>1</sup>, which supports interoperability across EHRs. Observational healthcare data is often standardized using the OHDSI Common Data Model (CDM)<sup>2</sup> for large-scale analytics. In representing biological pathways at the molecular and cellular levels, the Biological Pathway Exchange (BioPAX)<sup>3</sup> standard is used. Complex biological processes are stored and analyzed using the Systems Biology Markup Language (SBML)<sup>4</sup>. Finally, protein identification results are reported using the XML-based mzIdentML standard, developed by the Proteomics Standards Initiative [94].

Storing and organizing the data efficiently is an essential task, and selecting an appropriate format for storage is often dictated by the capabilities and limitations of the medical equipment used, resulting in different file storage formats. Moreover, users must go through various standards for representing common information, such as patients, diagnoses, and treatments, which can differ significantly across datasets.

Handling these large and diverse datasets generated in the medical domain presents several significant challenges. The primary challenge is the need for specific adapters for each dataset, which often involves manual searching and downloading parts of the data, adding to the complexity and time required for data integration. This lack of standardization complicates the process of harmonizing data from multiple sources. Effective big data management requires sophisticated tools and methodologies to ensure data is properly stored, processed, and analyzed, allowing for accurate and actionable insights in clinical settings.

Regulatory constraints further impact this process. The General Data Protection Regulation (GDPR)<sup>5</sup> for instance, imposes stringent restrictions on data migration, making it difficult to move and integrate data across different platforms and jurisdictions. While these regulations are crucial for safeguarding patient privacy, they magnify the challenges of efficiently using medical data for research and clinical applications.

### 3 | Big Data Management

Data management is a critical component of big data analytics, encompassing the collection, storage, standardization, and analysis processes, as shown in Figure 1. In the medical domain, the collection process involves acquiring data from various sources, such as EHRs, clinical trials, omics studies, and wearable devices. This data is usually heterogeneous, including structured data

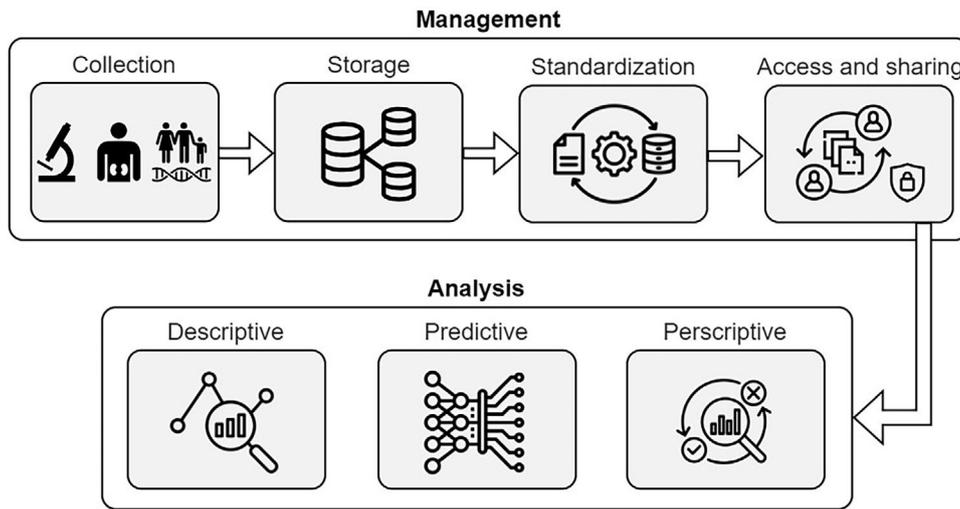
(e.g., lab results, demographics) and unstructured data (e.g., clinical notes, imaging).

#### 3.1 | Data Storage Solutions

Traditionally, data is stored in diverse storage systems. These source databases must be combined to create a unified view of all aspects of the data. One traditional approach is to create *analytical data warehouses* through the *extract, transform, load (ETL)* process and dimensional modeling. This process involves extracting data from multiple sources, transforming it into a consistent format, and loading it into a centralized repository for analysis. The questions being answered with the data warehouse must be defined in advance so that the data model can be adapted to answer them efficiently, limiting the flexibility to use the data for other purposes. However, when data sources need to be integrated gradually over time while considering data structure modifications and *change data capture (CDC)* in source systems, Data Vault 2.0 [34] modeling has emerged as a suitable methodology. Data Vault 2.0 makes it easier to adapt to changes in data sources and business needs. Despite being more complex and having a steeper learning curve than traditional data warehouses, it remains a strong alternative.

With the adoption of big data, new storage solutions such as data lakes and cloud platforms [35] have emerged beyond the warehouses. These platforms provide scalable and cost-effective storage options for large-scale omics datasets, enabling researchers to store and analyze vast amounts of data efficiently. In particular, data lakes have gained popularity due to their scalability and flexibility in handling diverse data types. Unlike warehouses, **data lakes store raw data** from various sources, making them ideal for medical research, since they can store the data in the format generated by the medical devices, which can be later transformed accordingly for the desired analysis. Examples of data lake enabling technologies include Hadoop Distributed File System (HDFS) [36], Amazon Simple Storage Service (S3)<sup>6</sup>, Google Cloud Storage (GCS)<sup>7</sup>, and Azure Data Lake Storage (ADLS)<sup>8</sup>. These platforms provide cost-effective storage options for large-scale omics datasets, enabling researchers to store and analyze vast amounts of data efficiently. The data lakes use the *extract, load, transform (ELT)* process, similar to ETL, but with the transformation step occurring after the data is loaded into the data lake [37]. This approach allows for more flexibility in data analysis, as the raw data is preserved in its original form, enabling researchers to apply different transformations and analyses as needed. However, data lakes also present challenges such as data silos, consistency, and retrieval efficiency. Data silos can arise when different teams or departments store data independently, leading to duplication and inconsistency. Ensuring data consistency across multiple sources can be complex, especially when dealing with real-time data updates [38].

An example of a data lake in the medical domain is the Helsinki University Hospital (HUS) Data Lake<sup>9</sup>, which provides researchers with access to a wide range of healthcare data, including EHRs, imaging data, and genomic data. By centralizing these datasets in a data lake, researchers can perform comprehensive analyses and gain insights into disease mechanisms, treatment outcomes, and patient outcomes [39, 40].



**FIGURE 1** | Data science process: Collect, store, standardize, and analyze.

Apart from the challenges of retrieving specific data from a data lake, metadata management is essential to ensure data is discoverable and accessible. Overcoming these limitations requires careful data governance, metadata management, and data quality assurance processes to ensure the integrity and usability of the stored data [41].

In the medical domain, metadata management is not just necessary but a crucial aspect of ensuring researchers can find and understand the data stored in data lakes. Metadata includes information about the data's source, format, quality, and lineage, enabling researchers to assess its relevance and reliability for analysis. By implementing robust metadata management practices, healthcare organizations significantly improve data discoverability, interoperability, and usability, ultimately enhancing the value of their data assets.

### 3.2 | Data Standardization

Standardization of data representation is essential for integrating diverse medical data sources and enabling interoperability. This process involves using shared formats and models to ensure that data can be exchanged, queried, and analyzed consistently across systems.

One widely used standard is the FHIR developed by HL7. This standard defines a set of resources for representing and exchanging EHRs and clinical data. It is designed to support interoperability between healthcare systems by providing a common data model (CDM) and API-based architecture. Another key standard is the CDM developed by the Observational Health Data Sciences and Informatics (OHDSI) initiative. OHDSI CDM<sup>10</sup> structures observational health data, such as EHRs and claims data, in a standardized way to enable large-scale analytics and collaborative research across institutions. An ongoing initiative aims to harmonize these two standards. OHDSI and HL7 organization are working together to align the OHDSI CDM with the HL7 FHIR standard for unified data representation<sup>11</sup>. This alignment seeks to enable seamless exchange and interoperability

between healthcare systems and research databases, combining the strengths of both frameworks.

Standardization also enables the use of powerful tools developed by the OHDSI community. For example, ATLAS [42] is a tool that provides a graphical interface for designing and analyzing observational studies using data represented with OHDSI CDM. Another tool, ACHILLES<sup>12</sup>, is an R package used for characterizing and visualizing data quality and content in CDM-formatted datasets.

A practical example of standardization is the mapping of the Medical Information Mart for Intensive Care (MIMIC) database [11] to both HL7 FHIR<sup>13</sup> and OHDSI CDM<sup>14</sup> representation standards. These mappings are publicly available and allow researchers to utilize MIMIC's rich critical care data within standardized frameworks, enhancing interoperability and usability.

It is a typical case that for each patient, the integration of multimodal data results in a comprehensive but complex graph of information, comprising EHR data, lab results, imaging studies, textual descriptions, prescriptions, diagnoses, time-series measurements (such as ECG), mass spectrometry results, and omic data. This data is often collected at irregular intervals, adding another layer of complexity to the analysis process. To manage this complexity, annotating the data using standard formats is essential. These standards facilitate system interoperability and allow for more efficient data querying and analysis. Developing robust data loaders that can extract and transform data based on these standards is critical for ensuring that the data is usable and can be analyzed effectively.

In addition to established data standards like FHIR and OHDSI CDM, ontologies offer a powerful approach for data standardization, particularly in the integration of heterogeneous medical datasets [43]. Ontologies are structured frameworks that define a set of concepts within a domain and the relationships between those concepts, providing a shared vocabulary and a formal representation of knowledge. In the context of healthcare, ontologies such as the Gene Ontology (GO) [44], the Human Disease

Ontology (DO) [45], and SNOMED CT [46] play a crucial role in standardizing the representation of biological and clinical concepts. By using ontologies, researchers can achieve a more granular and semantically rich integration of data from diverse sources, ensuring that different datasets are interoperable and can be meaningfully compared and analyzed [47, 48].

Ontology alignment, which involves mapping concepts from different ontologies to each other, further enhances this process [49, 50]. Techniques such as semantic matching and alignment algorithms enable data consolidation across ontologies, enabling more comprehensive and cohesive analyses. For example, integrating clinical data with genomic data through aligned ontologies can provide insights into disease mechanisms that are not apparent when examining these data types in isolation. Once data is standardized using ontologies, it can be consolidated into Knowledge Graphs [51], which represent data as interconnected entities and relationships. Knowledge Graphs provide flexibility and scalability by representing data as interconnected entities and relationships, allowing efficient traversal, integration, and querying across diverse and multidimensional datasets. They can encapsulate diverse data types, including clinical records, genomic data, and literature, into a unified structure, enabling researchers to derive actionable insights and enhance the understanding of disease processes and treatment outcomes.

### 3.3 | Data Access and Sharing

Managing and utilizing big data in healthcare is proper access and sharing of the data, especially in high-risk privacy environments. To ensure the protection of individual privacy, data owners must rigorously comply with regulatory requirements when publishing and sharing data with researchers. In this regard, two central frameworks for data privacy and security are the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA)<sup>15</sup>. GDPR, enforced by the European Union, mandates strict data protection and privacy measures for individuals within the EU and EEA, emphasizing transparency, consent, and the rights of individuals to control their personal data. It ensures that data handling practices are transparent and that individuals have significant control over their personal information. Similarly, HIPAA sets standards within the United States for protecting health information, focusing on the confidentiality, integrity, and security of electronically protected health information. HIPAA requires healthcare providers and organizations to implement physical, administrative, and technical safeguards to secure patient data and ensure privacy.

While essential for privacy, these regulations limit data sharing. To address these challenges, various techniques enhance privacy while enabling effective data analysis. Deidentification involves removing or masking personal identifiers to prevent identification of individuals. Common techniques include suppression, generalization [52], pseudonymization [53], and noise addition. Although deidentified data reduces reidentification risk, it may not fully prevent it, making this approach suitable when some data utility is necessary [54]. **Anonymization**, more stringent than deidentification, aims to make reidentification impossible by breaking the link between individuals and their data [52].

Techniques include aggregation, which combines data from multiple individuals to obscure individual identities; k-anonymity [55], ensuring that each record is indistinguishable from at least k-1 others based on certain identifying attributes; l-diversity [56], which enhances k-anonymity by requiring that sensitive attributes have at least l diverse values within each group; t-closeness [57], further refining l-diversity by ensuring the distribution of sensitive attributes in any group is close to their distribution in the overall dataset; and differential privacy [52], which adds controlled random noise to data analyses to protect individual privacy while allowing accurate aggregate insights. Anonymization provides stronger privacy guarantees but may reduce data utility [58]. **Homomorphic encryption** [59] enables secure computation on encrypted data, although it is computationally intensive.

Another approach for providing data privacy when the data is used for training models with machine learning is **federated learning** [60]. It enables multiple parties to train a model collaboratively without sharing their data. Each participant trains a model locally on their data and shares only the model parameters with a central server, which aggregates these values to create a global model. The global model is redistributed to participants for further training. **Split learning** [61] goes a step further and protects privacy by dividing the neural network layers into segments. Each client trains the initial layers of a neural network locally on their own data, generating intermediate activations. These activations are then sent to a central server, which continues training the remaining layers of the model. Both methods enhance data privacy and security, as raw data never leaves the local environment.

## 4 | Big Data Analysis Tools and Methods

Data analysis in medical research involves various methods and formats, each tailored to specific analytical requirements. For instance, regression and classification techniques require tabular data with fixed dimensions, while language models require text sequences to process natural language. On the other hand, time-series analysis demands temporal data that capture values over regular intervals, highlighting trends and patterns over time.

Figure 2 illustrates the diverse tools and frameworks employed in big data storage, standardization, and analytics. The tools are categorized by their ability to handle large datasets, their community-driven development, and their cloud-based or conventional nature. The elements highlighted in green are designed to efficiently manage and analyze big data and are often supported by open source communities. These tools are particularly useful for researchers working with vast amounts of data, offering flexibility and scalability. Blue elements represent cloud-based solutions managed by major providers, which are also capable of handling big data but with the added advantage of integrated cloud services, providing seamless scalability and maintenance, since they are operated by the provider's employees. Lastly, the light orange elements denote more conventional tools, which, while powerful in specific contexts, are not primarily designed for big data applications. These conventional solutions are typically used for smaller-scale or less complex data analysis tasks.

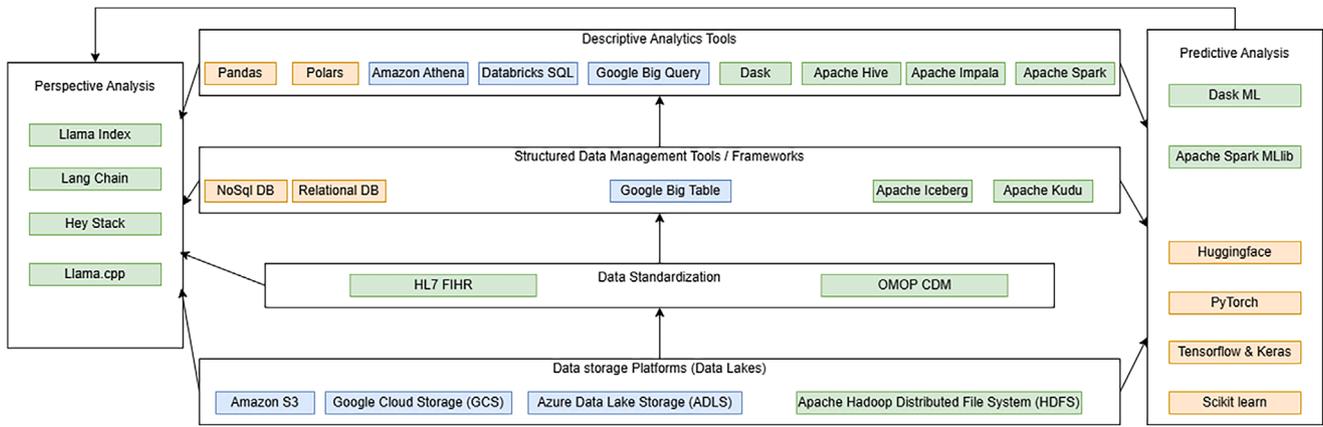


FIGURE 2 | Big data tools and frameworks.

In the field of nephrology and the broader medical domain, big data analytics can be categorized into descriptive, predictive, and prescriptive types, each serving distinct purposes. **Descriptive analytics** focuses on summarizing and interpreting historical data to understand past events and trends. Descriptive analytics might involve analyzing patient records to identify common patterns in disease progression or treatment outcomes. **Predictive analytics**, uses historical data and statistical models to forecast future events. This could involve predicting the likelihood of disease onset based on genetic, clinical, and lifestyle data, thus aiding in early diagnosis and personalized treatment plans. **Prescriptive analytics** takes a step further by recommending actions based on predictive insights. In nephrology, prescriptive analytics could suggest specific interventions or treatment modifications based on predicted disease trajectories and patient responses, aiming to optimize patient outcomes and healthcare efficiency. By employing these analytical approaches, researchers and clinicians can gain comprehensive insights, enhance decision-making, and improve patient care in nephrology and other medical fields.

#### 4.1 | Descriptive Big Data Analytics

Descriptive analytics involves summarizing and interpreting historical data to identify patterns or trends. However, one of the primary challenges in big data analysis is efficiently loading and exploring large datasets. Traditional tools like **Pandas**<sup>16</sup>, though powerful for small to medium-sized datasets, struggle with the volume of big data, leading to slow processing times and excessive memory usage. To overcome these limitations, platforms like **Polars**<sup>17</sup> try to provide lazy data loading, and thus more efficient memory allocation. However, greater data quantities require distributed solutions. **Dask**<sup>18</sup> extends Python's data analytics capabilities to larger-than-memory datasets by parallelizing operations across a cluster. It operates similarly to Pandas but distributes the computations over multiple nodes. Dask is capable of handling of complex workflows and provides real-time feedback, making it suitable for interactive and real-time data analysis, such as transcriptomics, where scalable solutions are necessary for handling the dimensionality of genomics datasets [62].

Even though all platforms have built-in toolboxes for descriptive analysis and data visualization, not all of them are capable of handling vast amounts of data efficiently. Pandas is designed to load the whole data-frame into memory, which limits the amount of data that can be analyzed. In contrast, Polars uses lazy evaluation to optimize the required memory. According to JetBrains research<sup>19</sup>, some of the most widely adopted platforms for distributed big data processing are **Dusk** [63], **Apache Hive**<sup>20</sup> [64], and **Apache Spark**<sup>21</sup> [65]. Table 2 explains the key differences among them.

The implementation of various big data technologies plays a crucial role in managing and analyzing large-scale biomedical datasets. Here, we discuss several key tools and their applications, focusing on their suitability for different types of data processing and analysis in the field of nephrology and bioinformatics.

**Apache Iceberg**<sup>22</sup> and **Apache Kudu**<sup>23</sup> are open source tools designed to efficiently manage large datasets. Apache Iceberg simplifies the management of vast analytic slowly changing datasets, while Apache Kudu is optimized for fast analytics on rapidly changing data. **Google BigQuery**<sup>24</sup>, on the other hand, is a fully managed, scalable and high-performance serverless data warehouse that allows for fast structured query execution on large datasets. BigQuery stands out for its scalability, high performance, and ease of use. It supports real-time data analysis and offers built-in machine learning capabilities through BigQuery ML.

Structured query execution technologies like **Apache Hive** and **Apache Impala**<sup>25</sup> are increasingly applied in biomedical data analysis due to their ability to efficiently query large-scale datasets. For instance, Apache Hive has been utilized in the SRIIdent pipeline to identify species from high-throughput sequencing reads in metagenomics and clinical diagnostic assays [66]. Similarly, Apache Kudu and Impala have been employed in the Variant-Kudu toolkit to analyze massive genetic variation datasets, enabling fast access and efficient querying of VCF files [67].

**Apache Spark** is a highly scalable distributed computing platform that processes large datasets using in-memory computing.

TABLE 2 | Comparison of Big Data Analysis Platforms.

Tool	Category	Strengths	Weaknesses
Polars	Data manipulation library	High performance execution; Efficient handling of large datasets; Lazy evaluation for optimization; Memory optimizations	Less mature ecosystem compared to others; Limited community support
Dask	Data manipulation library	Scales Python code for parallel computing; Integrates well with existing Python ecosystem; Dynamic task scheduling; Ideal for parallel computing and complex workflows	Overhead for task scheduling can be significant; Performance tuning can be complex
Apache Iceberg	Storage (Data lake)	High-performance format for huge analytic tables; Supports schema evolution and data partitioning; Fast read and write operations	Relatively new technology; Smaller community compared to more established tools
Apache Kudu	Storage (Columnar)	Low-latency random access; Efficient analytic access patterns; Integration with Apache Impala	Limited community and support; Performance can be impacted by write-heavy work-loads
Hive	Structured Query Engine	SQL-like querying interface; Integrates well with Hadoop; Suitable for data warehousing	High latency due to reliance on MapReduce; Not ideal for real-time queries
Apache Impala	Structured Query Engine	High-performance and low-latency SQL queries; MPP SQL query engine; Integration with various storage systems like HDFS, HBase, and S3	Requires significant resources for optimal performance; Can be complex to manage in large clusters
ApacheSpark	Distributed Computing	Highly scalable distributed computing; Rich ecosystem with many libraries (ML-lib, GraphX); Strong community support; Industry-standard for big data processing	High memory usage; Can be complex to configure and manage

It utilizes Resilient Distributed Datasets (RDDs) to manage and process data across a cluster efficiently. Spark's performance benefits make it a powerful tool for bioinformatics applications, such as NGS and other biological domains, where it ensures high fault tolerance and high scalability [68, 69].

Managed data storage and analysis services provided by cloud platforms offer robust, scalable, and user-friendly solutions that differ significantly from traditional big data analytics platforms such as Pandas, Dask, Hive, and Apache Spark. These managed services handle much of the infrastructure management, enabling users to focus more on data analysis and insights rather than operational complexities. **Amazon Athena**<sup>26</sup>, **Azure Synapse Analytics**<sup>27</sup>, and **Databricks SQL**<sup>28</sup> are cloud-based tools that enable users to analyze large datasets using familiar query languages. Amazon Athena allows direct querying of data stored in Amazon S3 without the need for setting up servers, making it ideal for quick, ad-hoc data analysis. Azure Synapse Analytics combines data warehousing and big data analytics, providing a unified platform for querying and managing data across various sources. Databricks SQL offers a high-performance environment for running SQL queries on large datasets, integrating seamlessly with business intelligence tools for interactive data exploration.

Cloud platforms provide robust and scalable solutions for handling large-scale medical data analysis, exemplified by using MIMIC-IV [70] on cloud services. MIMIC-IV, a comprehensive

clinical database containing deidentified health-related data for over forty thousand critical care patients, can be efficiently accessed and analyzed using cloud infrastructure. It is available on Google Cloud Platform, where users can leverage BigQuery for high-performance SQL queries on MIMIC-IV data stored in Google Cloud Storage. The dataset is also available through Amazon Athena, which provides an interactive query service for analyzing MIMIC-IV data stored in Amazon S3. With Athena, users can execute standard SQL queries directly on S3 data without loading it into a database, simplifying the analysis process. Additionally, AWS Glue can catalog and transform data, enhancing data management and preprocessing workflows. The integration with other AWS services like Amazon SageMaker allows for advanced machine learning applications on the processed MIMIC-IV data<sup>29</sup>.

With the rapid development of NGS technology, Apache Spark has emerged as a highly scalable and robust computational system, outperforming Hadoop by up to 100 times in memory access and ten times in disk access, making it ideal for large-scale genomic data processing and various bioinformatics applications, such as epigenetics, phylogeny, and drug discovery [69]. Furthermore, the AWS MIMIC-III to OMOP project<sup>30</sup> demonstrates the architecture featured on AWS for creating a healthcare data warehouse following OMOP CDM standards. This project can be accessed and processed with Apache Spark, facilitating efficient data management and interoperability in healthcare research.

Using these cloud-based solutions for MIMIC-III and MIMIC-IV analysis provides several benefits, including scalability, flexibility, and reduced infrastructure management. Researchers can focus on extracting insights and developing predictive models without the overhead of managing complex computing resources. Each cloud platform offers unique advantages, such as serverless querying, seamless integration with data lakes, and support for both SQL and Spark, satisfying various analytical needs. While cloud storage systems offer scalability and flexibility, they present several challenges concerning compliance with GDPR. Data sovereignty issues arise as cloud storage often involves storing data in data centers located in various countries, which may have a different level of data protection than the EU. Additionally, the risk of data breaches increases as sensitive information is transmitted over the Internet and stored on third-party servers. Ensuring that cloud providers comply with strict security measures is crucial, but managing GDPR compliance becomes more complex, demanding additional data processing agreements, data protection impact assessments, and the enforcement of the right to be forgotten. To mitigate these issues, using Apache Spark with local data storage can be a viable alternative, as it allows organizations to maintain greater control over their data, ensuring compliance with GDPR while benefiting from Spark's powerful data processing capabilities.

## 4.2 | Predictive Big Data Analytics

Predictive analytics uses statistical models and machine learning algorithms to forecast future events based on historical data. During the initial step, the data is split into training and testing sets. Typically, 70–80% of the data is used for training, and the remaining 20%–30% is set aside for testing. The training data is used to build the predictive model, while the test data is used to evaluate its performance.

Various algorithms, such as linear regression, Decision Trees (DT), or neural networks, are applied to the training data during training. Hyperparameters are tuned using cross-validation, a process where the training data is further split into smaller subsets to validate the model multiple times. Once the model is trained, it is evaluated on the test data using metrics like accuracy, precision, recall, F1 score, and ROC-AUC.

Several libraries and frameworks are commonly used for predictive analytics in big data:

- **Scikit Learn**<sup>31</sup> [71]: A comprehensive library in Python that provides simple and efficient tools for data mining and data analysis. It supports various supervised and unsupervised learning algorithms.
- **TensorFlow**<sup>32</sup> and **Keras**<sup>33</sup>: TensorFlow [72] and its high-level API, Keras [73], are widely used for the building and training of deep learning models.
- **PyTorch**<sup>34</sup> [95]: is known for its flexibility and ease of use, especially in research and prototyping.
- **Hugging Face Transformers**<sup>35</sup>: provide pre-trained models for a variety of tasks like text classification, named entity recognition, and question-answering.

Traditional predictive analytics libraries predominantly handle vast amounts of data through batching techniques. Batching involves dividing the dataset into smaller, manageable chunks called batches, which allows the models to process data in increments rather than all at once. This approach is crucial to efficiently managing memory usage and computational resources.

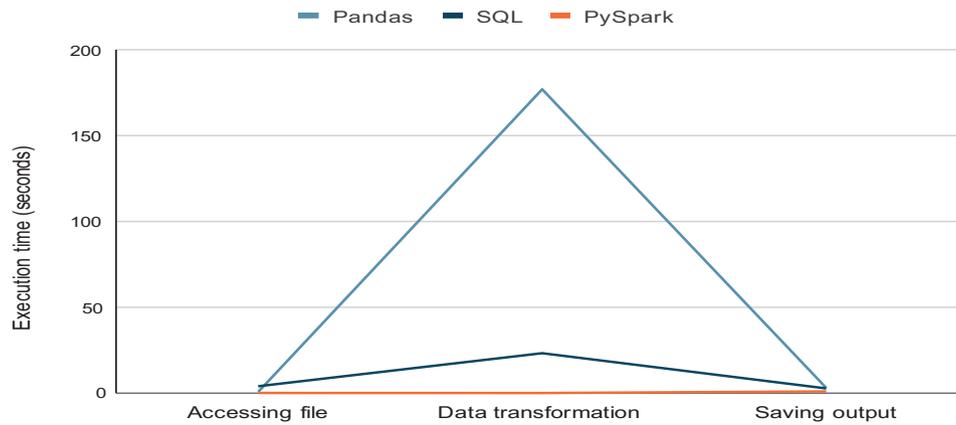
Spark MLlib<sup>36</sup> [74], Apache Mahout<sup>37</sup>, and Dask ML<sup>38</sup> are scalable machine learning libraries designed for processing large datasets efficiently. Spark MLlib leverages Apache Spark's distributed framework and DataFrames to support classification, regression, clustering, and pipeline workflows. Apache Mahout focuses on distributed algorithms for clustering, classification, and recommendation systems, often running on Hadoop. Dask ML enables parallel machine learning with familiar Scikit-Learn-like APIs, optimizing computation and memory usage through Dask's task scheduler and integration with Dask DataFrames.

Traditional machine learning libraries like Scikit-Learn are limited to single-machine use and may struggle with large datasets, lacking the inherent support for distributed computation in Spark ML and Dask ML. While traditional methods are easier to use for smaller datasets due to their simplicity, Spark ML and Dask ML offer significant performance benefits for large-scale data processing through scalability, parallel computing, and extensive ecosystem integration.

A practical application of big data predictive analytics in nephrology involves the integration of machine learning algorithms with scalable big data platforms like Apache Spark. Apache Spark MLlib is used with classification algorithms like Support Vector Machines (SVM), DT, and Gradient-Boosted Trees (GBT) to enhance prediction accuracy [75]. Apache Spark's in-memory processing capabilities efficiently handle large-scale CKD datasets, enabling rapid analysis and model training [69]. Similarly, the SRIdent pipeline utilizes Apache Spark for real-time identification of species from high-throughput sequencing reads, demonstrating Spark's utility in processing complex biological data in both metagenomics and clinical diagnostic settings [66]. Other scalable big data frameworks, such as Dask, are also employed in transcriptomics analysis, providing a parallel computing environment to manage the increasing dimensionality of genomics datasets [62]. These pipelines exemplify how big data platforms can be harnessed to improve the efficiency and accuracy of predictive models in medical research, particularly in the early detection and management of diseases like CKD.

## 4.3 | Benchmarking Scalable Processing Technologies for Kidney Proteomics Data

Data analytics relies on efficient data processing to make timely and accurate recommendations for high-risk patients. To evaluate different approaches for handling large-scale medical data, we compared three technologies—the standard Pandas python library, the SQL query language which is most often used to handle the data in production applications, and PySpark python API for Apache Spark for big data handling—on mass spectrometry laboratory results from approximately 600 CKD patients, totaling 822 GB of storage. The objective is to compare processing time between technologies while maintaining consistent output.



**FIGURE 3** | Visual comparison of performance across tools.

**TABLE 3** | Performance Comparison Between Pandas, SQL, and PySpark.

Task	Pandas	SQL	PySpark
Accessing raw data	0.6542	3.9784	0.0528
Transforming one file	177.2061	23.2552	0.0377
Saving results	3.0093	2.7126	0.9596

The data processing workflow is related to the calculation of spectra (common peaks) and consists of multiple steps, including data ingestion, filtering, aggregation, and transformation. When implementing this workflow, we observed significant differences in processing time across the three technologies.

The input raw data consisted of around 30 million rows per file with three columns of retention time, mass-to-charge ratio, and intensity. The statistics shown in Table 3 are measured as average values of processing 50 raw files for each step. Pandas times for accessing raw data are measured for accessing raw tsv files, as for SQL and PySpark, the initial data are accessed from a Postgre database. Furthermore, the measured times for accessing and saving data for all tools are measured with respect to 1 million rows.

The results shown in Figure 3 highlight the importance of choosing the appropriate technology for healthcare big data analytics. Although Pandas remains useful for small-scale exploratory analysis, SQL and PySpark offer significant advantages for handling larger datasets. Such insights underscore the necessity of selecting tools that balance efficiency and scalability to meet the demands of the provided data.

By ensuring efficient data processing, healthcare systems can take advantage of predictive and prescriptive analytics to improve patient outcomes. Predictive models can identify high-risk patients based on medical history, current health data, and lifestyle factors, while prescriptive analytics can recommend actionable interventions, such as personalized treatment plans, medication adjustments, or dietary changes. Faster data processing enables real-time decision making, allowing healthcare providers to optimize resource allocation, prioritize patient care,

and take proactive measures to prevent complications, ultimately enhancing both individual and systemic health outcomes.

#### 4.4 | Prescriptive and Autonomous Big Data Analytics

Prescriptive analytics is an advanced type of data analytics that not only predicts future outcomes but also suggests actions to achieve desired outcomes or optimize results. It uses a combination of machine learning, artificial intelligence, optimization algorithms, and business rules to recommend specific courses of action, considering the potential consequences of each decision. Prescriptive big data analytics focuses on understanding the contextual insights and underlying patterns within large datasets. It involves using advanced analytical techniques to interpret data meaningfully, providing a comprehensive view of the data's significance. This type of analytics helps identify trends, correlations, and anomalies that can inform strategic decision-making. Prescriptive analytics should recommend actions based on predictive insights to achieve desired outcomes. Prescriptive analytics integrates predictive modeling using machine learning, statistical techniques, and computational heuristics to formulate actionable recommendations.

It employs mathematical techniques to determine the best course of action by evaluating various possible scenarios. It simulates different situations and applies decision rules to assess how changes in variables and constraints can impact outcomes. This approach helps decision-makers choose strategies that effectively achieve their goals while considering factors like uncertainty, risk, and dynamic system behaviors. By using these techniques, organizations can gain a deeper perspective on their data, leading to more informed and effective decision-making processes [76].

Autonomous big data analytics is using artificial intelligence and machine learning. This approach minimizes human intervention, enabling systems to analyze data, detect patterns, and generate insights independently. These platforms, also referred to as Automated Machine Learning (AutoML), have gained popularity for simplifying the machine learning process by automating tasks like model selection, hyperparameter tuning, and feature engineering [77]. By combining prescriptive and autonomous big data analytics,

organizations can achieve a holistic understanding of their data while benefiting from automated processes' efficiency and agility.

#### 4.5 | Large Language Model (LLMs) and Their Relation to Prescriptive Analytics

LLMs [78] contribute to prescriptive big data analytics using their ability to process and summarize large volumes of text data to extract key themes and trends. LLMs have substantial background knowledge from their extensive training process, during which they have been exposed to numerous possible courses of action based on data analysis in many research papers. This repository of information allows LLMs to provide insightful summaries and identify significant patterns.

The key challenge for LLMs in prescriptive big data analytics is understanding the context of the data they are analyzing while being guided by domain-specific knowledge toward precise decision-making. One of the main techniques that addresses this challenge is prompt engineering, which involves prompts to direct LLMs to focus on specific data aspects or to follow a particular analytical approach. For example, in the context of literature review relation extraction, a well-crafted prompt can guide the LLM to identify and extract relationships between key concepts, such as the connection between biomarkers and diseases [79]. This approach allows for a more structured and comprehensive understanding of the research landscape. Here is a sample prompt for this case:

—Role: You are an expert Bioinformatician researcher with extensive experience in CKD research.

—Task: Given the following abstracts, extract and identify relationships between biomarkers and diseases. Focus on identifying specific biomarkers mentioned, the diseases they are associated with, and the roles or impacts of these biomarkers.

—Context: Abstract1: abstract1 | Abstract2. abstract2 | Abstract3. abstract3

—Guidelines: For each abstract, provide a structured output in the following CSV format: "Abstract Number," "Biomarker," "Disease," "Role," "Relationship." If multiple biomarkers and diseases are mentioned, list each one separately.

In this example<sup>39</sup>, the answer obtained from Chat GPT-4o using the abstracts of the articles with PubMed IDs 37356648, 37291728 and 37873853 identified 11 relationships. To ensure accurate and effective responses, several best practices should be followed when designing LLM prompts [80]. First, the LLM's **role** must be clearly defined to guide it in answering domain-specific tasks. For instance, in relation extraction, defining the LLM's role as a medical expert enables it to accurately identify facts, such as the connections between biomarkers, diseases, and other medical details within the provided abstracts. Second, the **task** the LLM is to perform should be explicitly described, ensuring

unambiguous instructions. Providing a relevant **context**, such as abstracts or key facts derived from them, is vital to direct the focus of the LLM on the specific medical topics being researched. Detailed **guidelines** should be provided to identify entities (e.g., biomarkers, diseases) and represent relationships (e.g., in CSV or JSON format). The generated results must be **validated** and **curated** by experts to ensure their accuracy, particularly in relation extraction, where identifying medically significant relationships is crucial. Finally, the prompt should be iteratively refined based on the quality of responses to further enhance accuracy. Adherence to these best practices ensures that the LLM produces accurate, relevant, and structured output, improving its ability to respond effectively to medical queries.

LLMs can also obtain structured output, such as extracting metadata or key relationships from a few rows of data. In relation extraction tasks, the desired metadata is often represented in JSON (JavaScript Object Notation), a lightweight data-interchange format that is easy for humans to read and write and for machines to parse and generate. JSON is commonly used to represent structured data as key-value pairs, arrays, and nested structures. When integrating LLM output into AI-powered applications, engineers often face challenges in ensuring that the generated output adheres strictly to the required format, ensuring consistency and accuracy throughout the system. Several innovative techniques have been developed to address the challenge of constraining LLM output to JSON format.

These techniques employ different strategies but converge on the goal of producing structured and predictable outputs.

- **Prompt Engineering:** This strategy involves guiding the model with carefully crafted input prompts, such as template-based prompts or clear instructions. It is effective for simple tasks, easy to implement, and does not require model retraining. However, it can struggle with complex constraints and relies on prompt quality. Although prompt engineering can guide LLMs to produce structured outputs (e.g., JSON metadata), inconsistencies can arise. For example, asking an LLM to follow a specific format may result in variations like `protein_name: "peptide_sequences"` or `Protein (protein_name): "peptide_sequences"`. The KOR framework<sup>40</sup> addresses this by wrapping LLMs with schema specification and example-based prompts, but further validation is still needed to ensure consistency.
- **Postprocessing Techniques:** This strategy involves applying constraints after text generation using techniques like rule-based filtering and validation/correction. It ensures output quality by checking and adjusting it against predefined rules, although it may require multiple iterations to obtain valid results. Libraries and frameworks such as **Instructor**<sup>41</sup>, **Haystack**<sup>42</sup>, **LangChain**<sup>43</sup>, and **LlamaIndex**<sup>44</sup> use **Pydantic**<sup>45</sup> objects for output validation. These objects define expected fields, descriptions, and examples, helping guide the prompt and validate results after generation.
- **Controlled Generation:** This strategy controls model output by integrating token-level and grammar constraints into the generation process. Techniques like token-level constraints within beam search help manage output diversity and coherence but are computationally intensive and may require

multiple runs. The **LM format enforcer library**<sup>46</sup> integrates token-level constraints with open-source models, enforcing output formats like JSON or regular expressions. Similarly, **Llama.cpp**<sup>47</sup> uses formal grammars to constrain open-source models. Recent advancements allow specifying response formats directly in the LLM query, simplifying the process and reducing the need for postprocessing.

- **Fine-Tuning and Transfer Learning:** This strategy adapts models to specific needs by fine-tuning them on relevant data. Domain-specific fine-tuning customizes the model for a particular domain, enhancing its relevance and accuracy in that context. Task-specific fine-tuning helps the model learn constraints naturally for particular tasks. Fine-tuning embeds an understanding of both the domain and output format, improving the model's ability to generate structured outputs reliably. However, this approach requires substantial computational resources and domain-specific data. Domain-specific models, such as BioMistral [81] and OpenBioLLMs [82], have been fine-tuned on extensive biomedical datasets, enabling them to generate highly accurate, domain-tailored text with exceptional precision [83].

Retrieval-Augmented Generation (RAG) [84] is a technique that combines retrieval-based and generation-based approaches to enhance the performance of language models. In RAG, a retrieval module first searches for relevant documents or pieces of information from a large corpus. Then, the generation module uses this retrieved information to produce more accurate and contextually relevant outputs.

RAG is highly valuable as it provides access to current information by retrieving the latest research and clinical guidelines, ensuring that the responses generated are based on the most current knowledge. Using external sources, RAG enhances response accuracy, offering detailed and context-specific information that purely generative models may overlook. Furthermore, RAG ensures that responses are evidence-based and supported by relevant and reliable sources, thereby improving the trustworthiness and reliability of medical communication. One such source is PubMed, a comprehensive repository of scientific papers. To demonstrate the practical utility of RAG in enhancing prescriptive analytics, we present a case study<sup>48</sup> involving the development and application of a custom RAG system designed for synthesizing medical knowledge. This system aims to address the challenges of providing accurate, up-to-date and evidence-based information for complex medical inquiries.

As shown in Figure 4, the LLM consults the PubMed repository to fetch relevant papers, ensuring that the latest information is available. However, the PubMed API operates on a keyword-based search principle, which makes it impractical to send the entire user query directly to the PubMed API. To optimize the retrieval process, a separate LLM prompt is used to refine the user's query by identifying and extracting the most specific and contextually relevant keywords. This LLM-enhanced search approach ensures that the keywords accurately represent the core elements of the user's question, allowing for a more targeted search. The extracted keywords are then used to query PubMed, retrieving relevant articles and abstracts that provide an up-to-date evidence-based response.

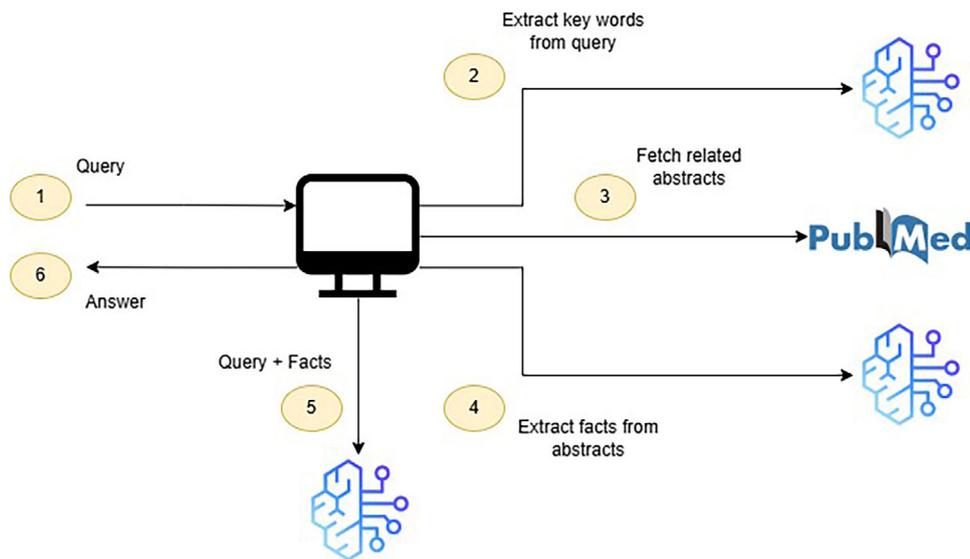
Once the latest scientific information has been obtained, it can be used as a context to provide a more precise response from the LLM. However, abstracts are often lengthy, and due to token limitations in LLMs, it is not feasible to use multiple long abstracts as context. Additionally, only a small portion of the abstract typically contains relevant information for the user's query. To overcome these issues, another LLM prompt is used to extract the most important medical facts from the abstracts and return them in the form of triples (e.g., ACE inhibitors → lower → blood pressure). This ensures that only the most relevant information is provided to the LLM for a more accurate and concise response.

In conclusion, the described process demonstrates how a RAG system can operate as an advanced agent, using a combination of tools to deliver accurate, up-to-date, and contextually relevant responses. The modularity of this system not only ensures access to the latest scientific knowledge but also offers flexibility, making it applicable to various medical contexts beyond just the retrieval of scientific papers. By focusing on extracting and utilizing the most relevant information, the system enhances the precision of responses, making it highly effective for complex medical inquiries and adaptable to other domains requiring evidence-based, context-specific knowledge.

## 5 | High-Performance Computing (HPC) for Big Data Analytics

HPC significantly advances data management and analysis in CKD research [85]. Initially, HPC systems managed and stored large datasets from various sources such as EHRs, biobanks, and omics studies. These datasets can include patient records, genetic data, and clinical trial results, among others. This capability ensures that different data sets can be accessed securely and efficiently [86]. During data processing, HPC accelerates data cleaning, normalization, and transformation tasks, thus ensuring high-quality data for subsequent analysis. For data integration, HPC enables the seamless merging of heterogeneous data types, creating uniform datasets essential for comprehensive analysis. In the analysis phase, HPC supports executing complex statistical models and machine learning algorithms, facilitating the identification of patterns and predictive factors related to CKD [85]. Furthermore, HPC improves data interpretation by providing the necessary computing power for high-speed data visualization and reporting [87]. This enables researchers and clinicians to quickly obtain actionable insights that are essential for clinical decision-making and improving patient outcomes.

HPC has emerged as a vital tool in the field of big data analytics, offering the necessary computing power to process the massive and ever-expanding datasets characteristic of modern research and industry [88]. HPC systems, designed to execute complex calculations and manage large-scale data processing tasks, deliver unprecedented efficiency and speed, providing a reliable, high-performance solution. HPC systems focus on executing complex computations at high speed using parallel processing and supercomputers, while big data systems like Hadoop, Data Lakes, and Spark are designed for handling and processing vast amounts of data efficiently, emphasizing storage, data integration and scalable data processing across distributed systems.



**FIGURE 4** | RAG process for medical knowledge synthesis: Query keywords extraction, PubMed retrieval, and fact extraction as triples.

Big data analytics platforms such as Hadoop [89] and Spark [90] are designed for distributed processing and analysis of data and leverage the power of cluster computing to manage large amounts of data effectively. These platforms achieve significant performance gains when integrated into HPC infrastructures. HPC architectures for parallel processing and distributed computing enable the simultaneous execution of numerous computing tasks, dramatically increasing data throughput and reducing processing times [88].

Moreover, HPC plays a fundamental role in optimizing machine learning algorithms, which are integral components of big data analytics [91]. These algorithms, which are used for tasks such as data training, model optimization, and real-time inference, require significant computing resources. HPC provides the necessary computing power to accelerate these processes and enable more sophisticated and accurate models [92]. For instance, in data training, HPC can significantly reduce the time required to train a model, thereby speeding up the entire process. This, in turn makes it easier to derive deeper insights from the data, promoting innovation and informed decision-making.

The integration of HPC and big data analytics platforms extends significantly into the domains of data storage and retrieval [93]. HPC systems can efficiently manage the huge amounts of data distributed across multiple storage nodes, ensuring fast access and data transfer. This capability is essential for applications that require real-time data analysis and visualization. For instance, in a healthcare setting, where real-time patient monitoring is crucial, HPC can ensure that the data is quickly retrieved and analyzed, enabling timely interventions.

## 6 | Conclusion

The integration of big data analytics into nephrology represents a paradigm shift, offering new opportunities for enhancing the understanding, prediction, and management of CKD and other related conditions. However, the complexity and heterogeneity of medical data pose significant challenges that must be addressed

to realize the potential of big data in this field fully. Our analysis underscores the importance of adopting a strategic combination of advanced tools and technologies to overcome these challenges effectively.

One of the key challenges in nephrology big data research is the vast diversity of data formats and standards, which complicates data integration and analysis. To address this, our approach emphasizes the use of standardized data models. These frameworks facilitate interoperability and enable researchers to harmonize data from multiple sources, thereby enhancing the accuracy and reliability of analytical outcomes.

Storage and management of large-scale data are equally important. Data lakes and cloud storage solutions provide scalable and flexible options for handling the extensive datasets generated in nephrology research. These platforms, combined with advanced data management techniques like the ELT process, allow for efficient storage, retrieval, and processing of diverse data types. Metadata management and data cataloging tools are also vital in ensuring data discoverability and usability, which are essential for large-scale analytics.

In terms of data analysis, our study highlights the critical role of HPC and distributed computing frameworks in managing the sheer volume and velocity of nephrology data. Tools such as Apache Spark and Dask provide the computational power needed to process large datasets and support complex machine learning algorithms. These platforms enable the development of predictive models that can forecast disease progression and treatment outcomes with greater accuracy, ultimately leading to more personalized and effective care for CKD patients.

Moreover, the integration of LLMs and techniques such as RAG and Chain-of-Thought prompting further enhances the analytical capabilities in nephrology. These advanced models allow for the extraction of meaningful insights from unstructured data, thus supporting more informed clinical decision-making. By incorporating these AI-driven techniques, nephrology research can leverage vast amounts of text and structured data to

uncover new relationships and patterns that were previously inaccessible.

Our analysis demonstrates that while no single tool can address all challenges, a strategic combination of technologies—ranging from data lakes and cloud storage solutions to advanced machine learning frameworks—can significantly enhance the capabilities of CKD research and treatment. By carefully selecting and integrating these tools, researchers and clinicians can develop more accurate predictive models, streamline data processing, and ultimately, deliver more personalized and effective care to patients.

The future of nephrology relies on the strategic use of data management platforms, standardized models, advanced analytics tools, and AI-driven frameworks to overcome the challenges of working with complex and diverse medical data. As we continue to refine and expand these approaches, the potential for big data to transform nephrology and improve patient outcomes becomes increasingly tangible. The ongoing collaboration between researchers, clinicians, and technologists will be key to driving these advancements forward, ensuring that the benefits of big data are fully realized in the quest to combat chronic kidney disease and enhance patient care.

## Acknowledgments

This work was supported by the COST Action Permedik (CA21165), Faculty of Computer Science and Engineering through the project KG-Enrich, the Slovenian Research and Innovation Agency through program grants No. P2-0098 and project grant No. GC-0001, European Union under grant agreement 101060712 and 101159214, the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through the Transregional Collaborative Research Centre SFB TRR219 (Project-ID 322900939) and the Collaborative Research Centre 1382 (Project-ID 403224013). Additional funding was provided by the DFG under INST 948/4S-1 and Clinical Research Unit (CRU) 5011 (Project No. 445703531). J.J. acknowledges support from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Innovative Training Networks (EU-ITN) INTRICARE (Grant No. 722609) and CaReSyAn (Grant No. 764474). This publication is based upon work from COST Action Permedik (CA21165), supported by COST (European Cooperation in Science and Technology).

## Conflicts of Interest

The authors declare no conflict of interest related to this work.

## Data Availability Statement

All data generated or analyzed during this study are included in this published article and Supporting Information.

## Endnotes

<sup>1</sup><https://www.hl7.org/fhir/>

<sup>2</sup><https://www.ohdsi.org/data-standardization/>

<sup>3</sup><https://www.biopax.org/>

<sup>4</sup><https://sbml.org/>

<sup>5</sup><https://gdpr-info.eu/>

<sup>6</sup><https://aws.amazon.com/s3/>

<sup>7</sup><https://cloud.google.com/storage>

<sup>8</sup><https://learn.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-introduction>

<sup>9</sup><https://www.hus.fi/en/research-and-education/tutkijan-palvelut/data-services/#principal-datasets-in-the-data>

<sup>10</sup><https://www.ohdsi.org/data-standardization/>

<sup>11</sup><https://www.ohdsi.org/ohdsi-hl7-collaboration/>

<sup>12</sup><https://www.ohdsi.org/software-tools/>

<sup>13</sup><https://github.com/banderlog/MIMICIII-to-FHIR>

<sup>14</sup><https://github.com/OHDSI/MIMIC>

<sup>15</sup><https://www.cdc.gov/php/php/resources/health-insurance-portability-and-accountability-act-of-1996-hipaa.html>

<sup>16</sup><https://pandas.pydata.org/>

<sup>17</sup><https://pola.rs/>

<sup>18</sup><https://www.dask.org/>

<sup>19</sup><https://www.jetbrains.com/lp/devecosystem-2023/big-data/>

<sup>20</sup><https://hive.apache.org/>

<sup>21</sup><https://spark.apache.org/>

<sup>22</sup><https://iceberg.apache.org/>

<sup>23</sup><https://kudu.apache.org/>

<sup>24</sup><https://cloud.google.com/bigquery>

<sup>25</sup><https://impala.apache.org/>

<sup>26</sup><https://aws.amazon.com/athena/>

<sup>27</sup><https://azure.microsoft.com/en-us/products/synapse-analytics>

<sup>28</sup><https://www.databricks.com/product/databricks-sql>

<sup>29</sup><https://aws.amazon.com/blogs/big-data/perform-biomedical-informatics-without-a-database-using-mimic-iii-data-and-amazon-athena/>

<sup>30</sup><https://github.com/amazon-archives/AWS-MIMIC-IIItoOMOP>

<sup>31</sup><https://scikit-learn.org/stable/>

<sup>32</sup><https://www.tensorflow.org/>

<sup>33</sup><https://keras.io/>

<sup>34</sup><https://pytorch.org/>

<sup>35</sup><https://huggingface.co/>

<sup>36</sup><https://spark.apache.org/mllib/>

<sup>37</sup><https://mahout.apache.org/>

<sup>38</sup><https://ml.dask.org/>

<sup>39</sup><https://chatgpt.com/share/35af1cc-f58c-4321-a72c-c401900c712b>

<sup>40</sup><https://eyurtsev.github.io/kor/index.html>

<sup>41</sup><https://python.useinstructor.com/>

<sup>42</sup><https://haystack.deepset.ai>

<sup>43</sup><https://python.langchain.com>

<sup>44</sup><https://docs.llamaindex.ai>

<sup>45</sup><https://docs.pydantic.dev/>

<sup>46</sup><https://github.com/noamgat/lm-format-enforcer>

<sup>47</sup><https://github.com/ggerganov/llama.cpp/tree/master/grammars>

<sup>48</sup> <https://colab.research.google.com/drive/1UUTBSeHTn8EawNmQSHJadohi8bp8Coy8?usp=sharing>

## References

1. B. Bikbov, C. A. Purcell, A. S. Levey, et al., "Global, Regional, and National Burden of Chronic Kidney Disease, 1990–2017: A Systematic Analysis for the Global Burden of Disease Study 2017," *Lancet* 395, no. 10225 (February, 2020): 709–733, [https://doi.org/10.1016/S0140-6736\(20\)30045-3](https://doi.org/10.1016/S0140-6736(20)30045-3).
2. C. P. Kovesdy, "Epidemiology of Chronic Kidney Disease: An Update 2022," *Kidney International Supplements* 12, no. 1 (April, 2022): 7–11, <https://doi.org/10.1016/j.kisu.2021.11.003>.
3. K. J. Foreman, N. Marquez, A. Dolgert, et al., "Forecasting Life Expectancy, Years of Life Lost, and All-Cause and Cause-Specific Mortality for 250 Causes of Death: Reference and Alternative Scenarios for 2016–40 for 195 Countries and Territories," *Lancet* 392, no. 10159 (November, 2018): 2052–2090, [https://doi.org/10.1016/S0140-6736\(18\)31694-5](https://doi.org/10.1016/S0140-6736(18)31694-5).
4. C.-M. Liao, Y. i-W. Kao, Y. i-P. Chang, and C.-M. Lin, "An Approach for Personalized Dynamic Assessment of Chronic Kidney Disease Progression Using Joint Model," *Biomedicines* 12, no. 3 (2024): 622, <https://doi.org/10.3390/biomedicines12030622>.
5. J. Siwy, H. Mischak, and P. Züribig, "Proteomics and Personalized Medicine: A Focus on Kidney Disease," *Expert Review of Proteomics* 16, no. 9 (September, 2019): 773–782, <https://doi.org/10.1080/14789450.2019.1659138>.
6. A. Kitcher, U. Ding, H. H. L. Wu, and R. Chinnadurai, "Big Data in Chronic Kidney Disease: Evolution or Revolution?" *BioMed-Informatics* 3, no. 1 (March, 2023): 260–266, <https://doi.org/10.3390/biomedinformatics3010017>.
7. J. Saez-Rodriguez, M. M. Rinschen, J. Floege, and R. Kramann, "Big Science and Big Data in Nephrology," *Kidney International* 95, no. 6 (June, 2019): 1326–1337, <https://doi.org/10.1016/j.kint.2018.11.048>.
8. C. Yang, G. Kong, L. Wang, L. Zhang, and M.-H. Zhao, "Big Data in Nephrology: Are We Ready for the Change?" *Nephrology* 24, no. 11 (2019): 097–1102, <https://onlinelibrary.wiley.com/doi/abshttps://doi.org/10.1111/nep.13636>.
9. R. Collins, "What Makes Uk Biobank Special?" *Lancet* 379, no. 9822 (2012): 1173–1174.
10. V. Jotwani, S. Y. Yang, H. Thiessen-Philbrook, et al., "Mitochondrial Genetic Variation and Risk of Chronic Kidney Disease and Acute Kidney Injury in UK Biobank Participants," *Human Genetics* 143, no. 2 (2024): 151–157, <https://doi.org/10.1007/s00439-023-02615-4>.
11. A. E. W. Johnson, T. J. Pollard, L. Shen, et al., "MIMIC-III, a Freely Accessible Critical Care Database," *Scientific Data* 3, no. 1 (May, 2016): 160035, <https://doi.org/10.1038/sdata.2016.35>.
12. C. Liu, S. Wang, and X. Wang, "Effect of Transthoracic Echocardiography on Short-Term Outcomes in Patients With Acute Kidney Injury in the Intensive Care Unit: A Retrospective Cohort Study Based on the MIMIC-III Database," *Annals of Translational Medicine* 10, no. 15 (2022): 826, [10.21037/atm-22-3158](https://doi.org/10.21037/atm-22-3158).
13. S. Martini, F. Eichinger, V. Nair, and M. Kretzler, "Defining Human Diabetic Nephropathy on the Molecular Level: Integration of Transcriptomic Profiles With Biological Knowledge," *Reviews in Endocrine and Metabolic Disorders* 9, no. 4 (December, 2008): 267–274, <https://doi.org/10.1007/s11154-008-9103-3>.
14. C. E. Gillies, R. Putler, R. Menon, et al., "An eQTL Landscape of Kidney Tissue in Human Nephrotic Syndrome," *American Journal of Human Genetics* 103, no. 2 (2018): 232–244, <https://doi.org/10.1016/j.ajhg.2018.07.004>.
15. S. Davis and P. S. Meltzer, "Geoquery: A Bridge Between the Gene Expression Omnibus (geo) and Bioconductor," *Bioinformatics* 23, no. 14 (2007): 1846–1847.
16. T. Schmidt, P. Samaras, M. Frejno, S. Gessulat, M. Barnert, and H. Kienegger, "ProteomicsDB," *Nucleic Acids Research* 46, no. D1 (2018): D1271–D1281, <https://doi.org/10.1093/nar/gkx1029>.
17. M. Sud, E. Fahy, D. Cotter, et al., "Metabolomics Workbench: An International Repository for Metabolomics Data and Metadata, Metabolite Standards, Protocols, Tutorials and Training, and Analysis Tools," *Nucleic Acids Research* 44, no. Database issue (2016): D463–D470, <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4702780/https://doi.org/10.1093/nar/gkv1042>.
18. D. S. Wishart, A. Guo, E. Oler, et al., "HMDB 5.0: The Human Metabolome Database for 2022," *Nucleic Acids Research* 50, no. D1 (January, 2022): D622–D631, <https://doi.org/10.1093/nar/gkab1062>.
19. R. Apweiler, A. Bairoch, C. H. Wu, et al., "UniProt: The Universal Protein Knowledgebase," *Nucleic Acids Research* 32, no. 1 (2004): D115–D119, <https://doi.org/10.1093/nar/gkh131>.
20. E. W. Deutsch, N. Bandeira, Y. Perez-Riverol, et al., "The ProteomeXchange Consortium at 10 Years: 2023 Update," *Nucleic Acids Research* 51, no. D1 (January, 2023): D1539–D1548, <https://doi.org/10.1093/nar/gkac1040>.
21. À. Argilés, J. Siwy, F. Duranton, et al., "CKD273, a New Proteomics Classifier Assessing CKD and Its Prognosis," *PLoS ONE* 8, no. 5 (2013): 62837, <https://doi.org/10.1371/journal.pone.0062837>.
22. D. M. Good, P. Züribig, A. Argilés, et al., "Naturally Occurring human Urinary Peptides for Use in Diagnosis of Chronic Kidney Disease," *Molecular & Cellular Proteomics* 9, no. 11 (November, 2010): 2424–2437, <https://doi.org/10.1074/mcp.M110.001917>.
23. T. Liu, X. X. Zhuang, X. J. Qin, L. B. Wei, and J. R. Gao, "Identifying Effective Diagnostic Biomarkers and Immune Infiltration Features in Chronic Kidney Disease by Bioinformatics and Validation," *Frontiers in Pharmacology* 13 (2022): 1069810, <https://doi.org/10.3389/fphar.2022.1069810>.
24. D. Dai, P. J. Alvarez, and S. D. Woods, "A Predictive Model for Progression of Chronic Kidney Disease to Kidney Failure Using a Large Administrative Claims Database," *ClinicoEconomics and Outcomes Research* 13 (2021): 475–486, <https://doi.org/10.2147/CEOR.S313857>.
25. K. Takkavatakarn, W. Oh, E. Cheng, G. N. Nadkarni, and L. Chan, "Machine Learning Models to Predict End-Stage Kidney Disease in Chronic Kidney Disease Stage 4," *BMC Nephrology* 24, no. 1 (December, 2023): 376, <https://doi.org/10.1186/s12882-023-03424-7>.
26. G. Gigliotti, R. Joshi, A. Khalid, D. Widmer, M. Boccellino, and D. Viggiano, "Epigenetics, Microbiome and Personalized Medicine: Focus on Kidney Disease," *International Journal of Molecular Sciences* 25, no. 16 (August, 2024): 8592, <https://doi.org/10.3390/ijms25168592>.
27. L. Usvyat, L. S. Dalrymple, and F. W. Maddux, "Using Technology to Inform and Deliver Precise Personalized Care to Patients With End-Stage Kidney Disease," *Seminars in Nephrology* 38, no. 4 (2018): 418–425, <https://doi.org/10.1016/j.semnephrol.2018.05.011>.
28. N. Kaur, S. Bhattacharya, and A. J. Butte, "Big Data in Nephrology," *Nature Reviews Nephrology* 17, no. 10 (October, 2021): 676–687, <https://doi.org/10.1038/s41581-021-00439-x>.
29. B. S. Glicksberg, K. W. Johnson, and J. T. Dudley, "The Next Generation of Precision Medicine: Observational Studies, Electronic Health Records, Biobanks and Continuous Monitoring," *Human Molecular Genetics* 27, no. R1 (May, 2018): R56–R62, <https://doi.org/10.1093/hmg/ddy114>.
30. F. Chen, P. Kantagowit, T. Nopsopon, A. Chuklin, and K. Pongpirul, "Prediction and Diagnosis of Chronic Kidney Disease Development and Progression Using Machine-Learning: Protocol for a Systematic Review and Meta-Analysis of Reporting Standards and Model Performance," *PLoS ONE* 18, no. 2 (February, 2023): 0278729, <https://doi.org/10.1371/journal.pone.0278729>.

31. J. Miao, C. Thongprayoon, S. Suppadungsuk, O. A. Garcia Valencia, and W. Cheungpasitporn, "Integrating Retrieval-Augmented Generation With Large Language Models in Nephrology: Advancing Practical Applications," *Medicina* 60, no. 3 (March, 2024): 445, <https://doi.org/10.3390/medicina60030445>.
32. C. Delrue, S. De Bruyne, and M. M. Speeckaert, "Application of Machine Learning in Chronic Kidney Disease: Current Status and Future Prospects," *Biomedicines* 12, no. 3 (March, 2024): 568, <https://doi.org/10.3390/biomedicines12030568>.
33. F. Fang, G. Gao, Q. Wang, Q. Wang, and L. Sun, "Combining SDS-PAGE to Capillary Zone Electrophoresis-Tandem Mass Spectrometry for High-Resolution Top-Down Proteomics Analysis of Intact Histone Proteoforms," *Proteomics* 24, no. 17: 2300650, <https://doi.org/10.1002/pmic.202300650>.
34. H. Helskyaho, L. Ruotsalainen, and T. Männistö, "Defining Data Model Quality Metrics for Data Vault 2.0 Model Evaluation," *Inventions* 9, no. 1 (2024): 21.
35. C. Giebler, C. Gröger, E. Hoos, H. Schwarz, and B. Mitschang, "Leveraging the Data Lake: Current State and Challenges," in *Big Data Analytics and Knowledge Discovery: 21st International Conference, Dawak 2019, Linz, Austria, August 26–29, 2019, Proceedings 21*, 179–188.
36. D. Borthakur, et al., "HDFS Architecture Guide," *Hadoop Apache Project* 53, no. 1–13 (2008): 2.
37. E. Zdravetski, P. Lameski, A. Dimitrievski, M. Grzegorowski, and C. Apanowicz, "Cluster-Size Optimization Within a Cloud-Based Etl Framework for Big Data," in *2019 IEEE International Conference on Big Data (Big Data)* (IEEE, 2019), 3754–3763.
38. M. Grzegorowski, E. Zdravetski, A. Janusz, P. Lameski, C. Apanowicz, and D. Slezak, "Cost Optimization for Big Data Workloads Based on Dynamic Scheduling and Cluster-Size Tuning," *Big Data Research* 25 (2021): 100203.
39. A. Maunula, J. Martola, S. Atula, S. M. Laakso, and P. J. Tienari, "Incidental Demyelination in Magnetic Resonance Imaging and 10-Year Risk of Multiple Sclerosis: A Data Lake Cohort Study," *European Journal of Neurology* 30, no. 8 (2023): 2376–2384, <https://doi.org/10.1111/ene.15849>.
40. A. Nelde, M. G. Klammer, C. H. Nolte, et al., "Data Lake-Driven Analytics Identify Nocturnal Non-Dipping of Heart Rate as Predictor of Unfavorable Stroke Outcome at Discharge," *Journal of Neurology* 270, no. 8 (August, 2023): 3810–3820, <https://doi.org/10.1007/s00415-023-11718-x>.
41. P. Sawadogo and J. Darmoni, "On Data Lake Architectures and Metadata Management," *Journal of Intelligent Information Systems* 56, no. 1 (2021): 97–120.
42. R. Blasini, K. M. Buchowicz, H. Schneider, B. Samans, and K. Sohrabi, "Implementation of Inclusion and Exclusion Criteria in Clinical Studies in OHDSI ATLAS Software," *Scientific Reports* 13, no. 1 (December, 2023): 22457, <https://doi.org/10.1038/s41598-023-49560-w>.
43. M. Jovanovic and D. Trajanov, "Consolidating Drug Data on a Global Scale Using Linked Data," *Journal of Biomedical Semantics* 8 (2017): 1–24.
44. M. Ashburner, C. A. Ball, J. A. Blake, et al., "Gene Ontology: Tool for the Unification of Biology," *Nature Genetics* 25, no. 1 (2000): 25–29.
45. L. M. Schriml, C. Arze, S. Nadendla, et al., "Disease Ontology: A Backbone for Disease Semantic Integration," *Nucleic Acids Research* 40, no. D1 (2012): D940–D946.
46. K. Donnelly, "SNOMED-CT: The Advanced Terminology and Coding System for eHealth," *Studies in Health Technology and Informatics* 121 (2006): 279.
47. M. Ivanovic and Z. Budimac, "An Overview of Ontologies and Data Resources in Medical Domains," *Expert Systems With Applications* 41, no. 11 (2014): 5158–5166.
48. H. Liyanage, P. Krause, and S. De Lusignan, "Using Ontologies to Improve Semantic Interoperability in Health Data," *BMJ Health & Care Informatics* 22, no. 2 (2015).
49. I. Harrow, E. Jiménez-Ruiz, A. Splendiani, et al., "Matching Disease and Phenotype Ontologies in the Ontology Alignment Evaluation Initiative," *Journal of Biomedical Semantics* 8 (2017): 1–13.
50. A. Kiourtis, A. Mavrogiorgou, A. Menychtas, I. Maglogiannis, and D. Kyriazis, "Structurally Mapping Healthcare Data to HL7 FHIR Through Ontology Alignment," *Journal of Medical Systems* 43 (2019): 1–13.
51. A. Hogan, E. Blomqvist, M. Cochez, et al., "Knowledge Graphs," *ACM Computing Surveys (CSUR)* 54, no. 4 (2021): 1–37.
52. B. C. M. Fung, K.e Wang, R. Chen, and P. S. Yu, "Privacy-Preserving Data Publishing," *ACM Computing Surveys* 42, no. 4 (2010): 1–53, <https://doi.org/10.1145/1749603.1749605>.
53. T. Neubauer and J. Heurix, "A Methodology for the Pseudonymization of Medical Data," *International Journal of Medical Informatics* 80, no. 3 (2011): 190–204.
54. J. F. Marques and J. Bernardino, "Analysis of Data Anonymization Techniques," in *Proceedings of the 12th international joint conference on knowledge discovery, knowledge engineering and knowledge management (ic3k 2020) - KEOD, SciTePress* (2020), 235–241, <https://doi.org/10.5220/0010142302350241>.
55. L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," *International Journal on Uncertainty, Fuzziness and Knowledge-Based Systems* 10, no. 5 (October, 2002): 557–570, <https://doi.org/10.1142/S021848850201648>.
56. A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkatasubramanian, "L-Diversity: Privacy Beyond k-Anonymity," *ACM Transactions on Knowledge Discovery from Data* 1, no. 1 (March, 2007): 3–es, <https://doi.org/10.1145/1217299.1217302>.
57. N. Li, T. Li, and S. Venkatasubramanian, "t-Closeness: Privacy Beyond k-Anonymity and L-Diversity," in *2007 IEEE 23rd International Conference on Data Engineering*, (IEEE, 2007), 106–115, <https://doi.org/10.1109/ICDE.2007.367856>.
58. R. Chevrier, V. Foufi, C. Gaudet-Blavignac, A. Robert, and C. Lovis, "Use and Understanding of Anonymization and De-Identification in the Biomedical Literature: Scoping Review," *Journal of Medical Internet Research* 21, no. 5 (2019): 13484, <https://www.jmir.org/2019/5/e13484/> <https://doi.org/10.2196/13484>.
59. K. Munjal and R. Bhatia, "A Systematic Review of Homomorphic Encryption and Its Contributions in Healthcare Industry," *Complex & Intelligent Systems* 9 (2023): 3759–3786, <https://doi.org/10.1007/s40747-022-00756-z>.
60. N. Rieke, J. Hancox, W. Li, et al., "The Future of Digital Health With Federated Learning," *NPJ digital medicine* 3, no. 1 (2020), <https://arxiv.org/abs/2003.08119>.
61. O. Gupta and R. Raskar, "Distributed Learning of Deep Neural Network Over Multiple Agents," *Journal of Network and Computer Applications* 116 (2018): 1–8, <https://api.semanticscholar.org/CorpusID>.
62. M. Moreno, R. Vilaça, and P. G. Ferreira, "Scalable Transcriptomics Analysis With Dask: Applications in Data Science and Machine Learning," *BMC Bioinformatics [Electronic Resource]* 23, no. 1 (November, 2022): 514, <https://doi.org/10.1186/s12859-022-05065-3>.
63. M. Rocklin, "Dask: Parallel Computation With Blocked Algorithms and Task Scheduling," in *Proceedings of the 14th Python in Science Conference-SCIPY* (2015), 126–132.
64. A. Thusoo, J. S. Sarma, N. Jain, et al., "Hive," *Proceedings of the VLDB Endowment* 2, no. 2 (2009): 1626–1629.
65. S. Salloum, R. Dautov, X. Chen, P. X. Peng, and J. Z. Huang, "Big Data Analytics on Apache Spark," *International Journal of Data Science and Analytics* 1 (2016): 145–164.
66. R. Karimi and A. Hajdu, "SRIdent: A Novel Pipeline for Real-Time Identification of Species From High-Throughput Sequencing Reads in Metagenomics and Clinical Diagnostic Assays," *Annual International*

- Conference of the IEEE Engineering in Medicine and Biology Society. *IEEE Engineering in Medicine and Biology Society. Annual International Conference* 2015 (2015): 6481–6484, <https://doi.org/10.1109/EMBC.2015.7319877>.
67. J. Fan, S. Dong, and B.o Wang, “Variant-Kudu: An Efficient Tool Kit Leveraging Distributed Bitmap Index for Analysis of Massive Genetic Variation Datasets,” *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology* 27, no. 9 (September, 2020): 1350–1360, <https://doi.org/10.1089/cmb.2019.0344>.
68. J. M. Abuin, N. Lopes, L. Ferreira, T. F. Pena, and B. Schmidt, “Big Data in Metagenomics: Apache Spark vs MPI,” *PLoS ONE* 15, no. 10 (2020): 0239741, <https://doi.org/10.1371/journal.pone.0239741>.
69. R. Guo, Y. Zhao, Q. Zou, X. Fang, and S. Peng, “Bioinformatics Applications on Apache Spark,” *GigaScience* 7, no. 8 (2018): giy098.
70. A. Johnson, L. Bulgarelli, T. Pollard, S. Horng, L. A. Celi, and R. Mark, “Mimic-IV,” *PhysioNet* (2020): 49–55, accessed August 23, 2024, <https://physionet.org/content/mimiciv/1.0/>.
71. O. Kramer, “Scikit-Learn,” in *Machine Learning for Evolution Strategies* (Cham: Springer International Publishing, 2016), 45–53, [https://doi.org/10.1007/978-3-319-33383-0\\_5](https://doi.org/10.1007/978-3-319-33383-0_5).
72. M. Abadi, A. Agarwal, P. Barham, et al., “TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems,” In *Proceedings of the 12th usenix conference on operating systems design and implementation*, USENIX Association (2016), 265–283.
73. N. Ketkar, “Introduction to Keras,” *Deep Learning With Python: A Hands-on Introduction* (Berkeley, CA: Apress, 2017): 97–111.
74. X. Meng, J. Bradley, B. Yavuz, et al., “MLlib: Machine Learning in Apache Spark,” *Journal of Machine Learning Research* 17, no. 34 (2016): 1–7.
75. M. A. Abdel-Fattah, N. A. Othman, and N. Goher, “Predicting Chronic Kidney Disease Using Hybrid Machine Learning Based on Apache Spark,” *Computational Intelligence and Neuroscience* 2022, no. 1 (2022): 9898831, <https://doi.org/10.1155/2022/9898831>.
76. F. Al Zoubi, G. Khalaf, P. E. Beaulé, and P. Fallavollita, “Leveraging Machine Learning and Prescriptive Analytics to Improve Operating Room Throughput,” *Frontiers in Digital Health* 5 (2023): 1242214, <https://doi.org/10.3389/fdgh.2023.1242214>.
77. X. He, K. Zhao, and X. Chu, “AutoML: A Survey of the State-of-the-Art,” *Knowledge-based Systems* 212 (2021): 106622.
78. W. X. Zhao, K. Zhou, J. Li, et al., “A Survey of Large Language Models,” (November, 2023), accessed July 29, 2024, <http://arxiv.org/abs/2303.18223>.
79. E. Filipovska, A. Mladenovska, M. Bajrami, et al., “Bench-Marking OpenAI’s Apis and Large Language Models for Repeatable, Efficient Question Answering Across Multiple Documents,” in *19th Conference on Computer Science and Intelligence Systems* (FedCSIS, 2024).
80. G. Marvin, N. Hellen, D. Jjingo, and J. Nakatumba-Nabende, “Prompt Engineering in Large Language models,” in *Data Intelligence and Cognitive Informatics*, ed. I. J. Jacob, S. Piramuthu, and P. Falkowski-Gilski, (Springer Nature Singapore, 2024), 387–402.
81. Y. Labrak, A. Bazoge, E. Morin, P. A. Gourraud, M. Rouvier, and R. Dufour, “Biomistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains,” (2024). *arXiv preprint arXiv:2402.10373*.
82. M. S. Ankit Pal, *Openbiollms: Advancing Open-Source Large Language Models for Healthcare and Life Sciences* (Hugging Face, 2024), <https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B.HuggingFace>.
83. F. J. Dorfner, A. Dada, F. Busch, et al., “Evaluating the Effectiveness of Biomedical Fine-Tuning for Large Language Models on Clinical Tasks,” *Journal of the American Medical Informatics Association* (April, 2025), ocaf045, <https://doi.org/10.1093/jamia/ocaf045>.
84. Y. Gao, Y. Xiong, X. Gao, et al., “Retrieval-Augmented Generation for Large Language Models: A Survey,” *arXiv preprint arXiv:2312.10997*, 2(2024), <https://arxiv.org/abs/2312.10997>.
85. S. S. Abdullah, N. Rostamzadeh, F. T. Muanda, et al., “High-Throughput Computing to Automate Population-Based Studies to Detect the 30-Day Risk of Adverse Outcomes After New Outpatient Medication Use in Older Adults With Chronic Kidney Disease: A Clinical Research Protocol,” *Canadian Journal of Kidney Health and Disease* 11 (2024): 20543581231221891.
86. C. Guo and J. Chen, “Big Data Analytics in healthcare,” in *Knowledge Technology and Systems: Toward Establishing Knowledge Systems Science*, ed. Y. Nakamori (Springer Nature, 2023), 27–70, <https://doi.org/10.1007/978-981-99-1075-52>.
87. A. F. Szczepanski, J. Huang, T. Baer, Y. C. Mack, and S. Ahern, “Data Analysis and Visualization in High-Performance Computing,” *Computer* 46, no. 5 (2013): 84–92, <https://doi.org/10.1109/MC.2012.192>.
88. T. Sterling, M. Brodowicz, and M. Anderson, *High Performance Computing: Modern Systems and Practices* (Elsevier Science, 2017), <https://books.google.si/books?id=qOHIBAAQBAJ>.
89. T. White, *Hadoop: The Definitive Guide* (O’Reilly Media, Inc., 2012).
90. H. Karau, A. Konwinski, P. Wendell, and M. Zaharia, *Learning Spark: Lightning-Fast Big Data Analysis*. (O’Reilly Media, Inc., 2015).
91. E. Wulff, M. Gironé, and J. Pata, “Hyperparameter Optimization of Data-Driven AI Models on HPC Systems,” *Journal of Physics: Conference Series* 2438, no. 1, (February, 2023): 012092, <https://doi.org/10.1088/1742-6596/2438/1/012092>.
92. R. Lim, *Methods for Accelerating Machine Learning in High Performance Computing* (University of Oregon, 2019), Area-2019-01.
93. P. Raj, A. Raman, D. Nagaraj, and S. Duggirala, *High-Performance Big-Data Analytics*, Vol. 1 (Springer, 2015).
94. A. R. Jones, M. Eisenacher, G. Mayer, et al., “The mzIdentML Data Standard for Mass Spectrometry-Based Proteomics Results,” *Molecular & Cellular Proteomics* 11, no. 7, (2012) M111.014381–1–M111.014381-10, <https://doi.org/10.1074/mcp.M111.014381>.
95. S. Imambi, K. B. Prakash, and G. Kanagachidambaresan, “Pytorch,” *Programming with TensorFlow: solution for edge computing applications* (2021): 87–104.