Abstract

This chapter will take a quick look at certain technological requirements associated with "precision," "personalized," or "patient-centered" medicine. We will emphasize how the goals of precision medicine are advanced through bioimaging, and also how precision medicine may influence trial design. We use features if image-analysis and clinical trials to consider our overview of biomedical research methods initiated in Chapter 2. We also discuss biomedical text mining as a further dimension in research methodology, summarizing the CORD-19 corpus on Covid-19 related research as a case-study.

Chapter 3: Patient-Centered Research Methodology for Bioimaging and Covid-19

1 Introduction

This chapter will touch on research methodology and clinical trial design in light of Patient-Centered paradigms and Precision Medicine. We will focus in particular on bioimaging/radiology, on trial architecture (particularly vis-à-vis Covid-19) and on text and data mining.

After examining how Precision Medicine effects requirements for bioimaging software, we will consider patient-centered priorities in the context of clinical trials. The goal of incorporating more granular patient-specific data within clinical observations in general translates over to clinical trials in particular, with trial-specific data models needing to incorporate patient-centered details at several stages, including recruitment of patients for trials, information management while trials are being conducted, and subsequent follow-up studies and/or analyses.

This chapter will conclude by examining text and data mining techniques oriented toward biomedical publications and data sets, on the premise that text and data mining represent a computationally sophisticated methodology in their own right. We will focus on the case-study of **CORD-19**, a large corpus of open-access Covid-related articles.

2 Precision Medicine and Bioimaging

Patient-centered research in the radiological context focuses on improving the precision of diagnostic-imaging techniques and corresponding clinical interventions. Indeed, the goal of contemporary radiology is not only to confirm a diagnosis, but also to extract cues from medical images that suggest which course of treatment has the highest probability of favorable outcomes. A related goal is curating collections of diagnostic images so as to improve our ability to identify such diagnostic and prognostic cues, potentially using Machine Learning and/or Artificial Intelligence applied to large-scale image repositories.

The goal of building "searchable" image repositories has inspired projects such as the Semantic Dicom Ontology (**SEDI**)[1] and the **VISION** "structured reporting" system.[2] As explained in the context of **SEDI**: "if a user has a CT scan, and wants to retrieve the [corresponding] radiation treatment plan ... he has to search for the RTSTRUCT object based on the specific CT scan, and from there search for the RTPLAN object based on the RTSTRUCT object. This is an inefficient operation because all RTSTRUCT [and] RTPLAN files for the patient need to be processed to find the correct treatment plan" [14, page 1]. Even relatively simple queries such as "display all patients with a bronchial carcinoma bigger than 50 cm^3" cannot be processed by **PACS** systems: "although there are various powerful clinical applications to process image data and image data series to create significant clinical analyses, none of these analytic results can be merged with the clinical data of a single patient."[3] These limitations partly reflect the logistics of how information is transferred between clinical institutions and radiology labs. In response, and in an effort to advance the science of diagnostic image-analysis, organizations such as the Radiological Society of North America (**RSNA**) have curated open-access data sets encompassing medical images as well as image-annotations (encoding feature vectors) that can serve as reference sets and test corpora for investigating analytic methods. Such repositories are designed to integrate data from multiple hospitals and multiple laboratories — bypassing the conventional data flows wherein radiological information is shared between clinicians and radiologists, but is not also merged into broad-spectrum corpora.

This renewed focus on patient outcomes has important consequences for the scope and requirements of diagnostic-imaging software. In particular, the domain of radiological applications is no longer limited to **PACS** workstations where pathologists perform their diagnostic analysis, with the results transferred back to the referring institution (and subsequently available only through that institution's medical records, if at all). In the older, conventional workflow, radiographic images are requested by some medical institution for diagnostic purposes. Relevant information is therefore shared between two end-points: the institution which prescribes a diagnostic evaluation and the radiologist or laboratory which analyzes the resulting images. Building radiographic data repositories complicates this workflow because a third entity becomes involved — the organization responsible for aggregating images and analyses is generally distinct from both the prescribing institution and the radiologists themselves. As a result, both radiologists and prescribing institutions, upon participation in the formation of the target repository, must identify which image series and which patient data are proper candidates for the relevant repository.

For a concrete example, **RSNA** has announced the forthcoming publication of an open-access image repository devoted to Covid-19.[4] This repository is being curated in collaboration with multiple European, Asian, and South American organizations so as to collect data from hospitals treating Covid-19 patients. Such a collaboration requires protocols both for data submission and for patient privacy and security. As this example demonstrates, these kinds of data-sharing initiatives present new requirements for radiological software, which must not only allow for the presentation, annotation, and analysis of medical images, but also for participation in data-sharing initiatives adhering to rigorous modeling and operational protocols.

Simultaneously, the science of diagnostic imaging is also expanding as new image-analytic techniques prove to be effective at detecting signals within image data, often complementing the work of human radiologists. The proliferation of image-analysis methodologies places a new emphasis on *extensibility,* where radiological software becomes more powerful and flexible because new analytic modules may be plugged in to a central **PACS** system.

[1] See https://bioportal.bioontology.org/ontologies/SEDI.

[2] See https://epos.myesr.org/esr/viewing/index.php?module=viewing_poster&task=&pi=155548.

[3] See https://semantic-dicom.com/starting-page/.

[4] See https://www.rsna.org/covid-19.

A good example of this new paradigm is CaPTk, which we will discuss below. The CaPTk project provides a central application which supplies a centralized User Interface and takes responsibility for acquiring and loading radiographic images. The CaPTk core application is then paired with multiple "peer" applications which can be launched from CaPTk's main window, each peer focused on implementing specific algorithms so as to transform and/or to extract feature vectors from images sent between CaPTk and its plugins.

Both the patient-outcomes focus in building image repositories and the integration of novel Computer Vision algorithms depend, at their core, on rigorous data sharing. Taking the RSNA Covid-19 repository as a case study for promoting research into post-diagnostic outcomes, this repository is possible because an international team of hospitals and institutions have agreed to pool radiological data relevant to SARS-CoV-2 infection according to a common protocol. Taking CaPTk as a case-study in multi-modal image analysis, this system is likewise possible because analytic modules can be wrapped into a plugin mechanism which allows many different algorithms to be bundled into a common software platform. Of course, these two areas of data-sharing overlap: one mission of repositories such as the RSNA's is to permit many different analyses to be performed on the common image assets. The results of these analyses then become additional information which enlarges the repository proportionately. If CaPTk modules are used to analyze the RSNA Covid-19 images, for example, there needs to be a mechanism for exporting the resulting data outside the CaPTk system, so that the analyses may be integrated into the repository either directly or as a supplemental resource.

This example demonstrates how software such as CaPTk may be extended to support the curation of image repositories dedicated to Patient Outcomes and Comparative Effectiveness Research (CER), insofar as analytic data generated by (for example) CaPTk components can acquire the capability to share data according to repository protocols. A further level of integration between CaPTk and CER initiatives (again, staying with CaPTk as a representative example of bioimaging applications in general) can be achieved if one observes that clinical outcomes may be part of the analytic parameters used by CaPTk modules. As presently constituted, CaPTk analytic tools are focused on extracting quantitative (or quantifiable) features from image themselves, without considering additional patient-centered context. There is no technical limitation, however, which would prevent the CaPTk system from sharing more detailed clinical information with its modules, allowing these analytic components to cross-reference image features with clinical or patient information. This then raises general questions about sharing clinical data *as well as* information derived from bioimage analysis, which we will review over the course of this chapter.

2.1 The Basic Synthesis Between Bioimaging and Precision Medicine

Patient-centered research in the radiological context has several dimensions, including analysis of the rationales for diagnostic imaging in the first place: how well does image-based diagnostics actually correlate with improved patient outcomes? How can we quantify the experience of image-based testing itself (e.g., to identify factors such as cost, discomfort, and radiation danger which can rank some tests as less desirable than others, as one parameter to consider when deciding whether to prescribe imaging, and which modality)? These were among the questions addressed by the Patient-Centered Outcomes Research Institute's (PCORI's) "Patient-Centered Research for Standards of Outcomes in Diagnostic Tests" (PROD) study, which also established a useful protocol for how medical institutions could provide reports on the experience and effectiveness of imaging from a patient's as well as clinician's perspective.

Assuming diagnostic imaging of a given modality *is* appropriate, an additional priority should then be to structure the diagnostic workflow — the image procurement, analysis, and data/metadata sharing protocols — to maximize the probability that subsequent clinical interventions are chosen which promote favorable outcomes on multiple patient-centered criteria, including quality of life and patient engagement. In the best case scenario, the goal of radiology is not only to confirm a diagnosis, but also to extract cues from medical images that suggest which course of treatment has the highest probability of favorable treatment outcomes.

While most radiologists and pathologists would probably agree with this assessment — that Precision Medicine can be a "game-changer" in bioimaging fields such as radiology — there are technological and operational challenges to making patient-centered perspectives a central feature of diagnostic-imaging methodology. Effectively cross-referencing imaging and outcomes data requires integrating heterogeneous information obtained at different times and places. Some clinical data is associated with each patient at the time that a radiological (or other imaging) study is prescribed. The images themselves, and subsequent diagnostic reports, provide layers of data that exist prior to the initiation of a course of treatment (at least insofar as a clinical intervention is chosen in response to radiological findings). Moreover, a rigorous data-integration protocol would need to incorporate information emerging *after* the treatment starts: descriptions and assessments of clinical outcomes, and, potentially, new data garnered by applying different image-analysis techniques. Ideally, image analysis can be powerful enough not only to identify a pathology, but to classify diagnoses into clusters based on recommended course of treatment. In order for analytic techniques to achieve this level of detail, however, it is necessary to arrive at a feedback loop where known clinical outcomes are associated with prior images, so that developers have those outcomes available as a further dimension of clinical data that may be statistically cross-referenced with image analysis.

The PROD study demonstrates that experiential factors should be evaluated — both in terms of testing itself (and its risks/costs) and in terms of post-diagnostic quality of life — as part of the data modeling treatment outcomes, and the comparative effectiveness of a selected course of treatments compared to alternative diagnostic methods and/or clinical interventions. From the perspective of standard data models, initiatives to cross-reference imaging and outcomes data include several Semantic Web ontologies, such as the Semantic Dicom Ontology (SeDI) and the ViSion "structured reporting" system (both referenced earlier). The purpose of these ontologies is to standardize the terms through which radiographic

procedures, analyses, and recommendations are described — more precisely or predictably than older technologies such as **DICOM** headers, **DICOM-RT**, and the **RadLex** lexicon. By properly aligning image metadata spanning multiple patients, it is possible to create "searchable" image archives such that images can be selected or classified within a larger image collection, yielding image series or patient cohorts that can be studied through the lens of predictive modeling or patient-centered outcomes. Projects such as **SeDI** implement "semantic" **PACS** workstations where the space of known images is defined by a particular **PACS** system, but analogous techniques could be used to construct larger-scale image corpora as well, for research purposes, data mining, or as test-beds for code and algorithms.

However, making interop work across distinct software ecosystems add development complexity: the requirements for implementing novel analytic methods are not only to compose executable code making the methods computationally realizable, but to package that code into a functional unit that can interoperate with other clinical and imaging software. This problem, in turn, engenders various software-engineering techniques and frameworks. A good example is the Cancer Imaging Phenomics Toolkit (**CaPTk**), developed at the Center for Biomedical Image Computing and Analytics (**CBICA**) at the University of Pennsylvania Perelman School of Medicine, which is an extensible platform for implementing analytic modules as peers to a central **PACS** system. In particular, **CaPTk** provides an implementation (apparently the only **C++**-based implementation) of **CWL**, using this workflow model in conjunction with the **Qt** Reflective Programming system to implement workflows connecting the central **CaPTk** application with its analytic extensions.[5] In effect, **CaPTk** achieves a workflow and messaging protocol for what they term "native," "standalone" applications, yielding an extensible architecture through which new image-analysis techniques can be integrated into an underlying **PACS** system.

Certain comparisons can be made between **CaPTk**, whose architectural innovations are centered on workflow management and multi-application networking, and **SeDI**, whose novel features focus on data alignment and integration. Both of these projects expand the analytic capabilities of diagnostic-imaging systems by promoting common data and code representations, enlarging the space of metadata available for query-evaluation and/or the range of quantitative techniques available for image analysis. Both rely on a canonical description format (**CWL**, in the case of **CaPTk**, and a novel **RDF** ontology, in the case of **SeDI**) which is not widely implemented by other **PACS** systems. Their concerns also overlap insofar as different analytic methods generate different kinds of image data, which need to be integrated into the total space of data available for a **PACS** system and/or an image repository. While the data-integration approaches chosen by these two projects are specific to the respective software applications which is their main result, both **SeDI** and **CaPTk** point to evident limitations in the scope of current diagnostic imaging software: failure to properly integrate image metadata (including clinical and outcomes data) into a multi-patient space optimized for query evaluation and data mining; and failure to integrate many diverse image-analysis methodologies into a common execution framework.

One take-away from this overview of **SeDI** and **CaPTk** is that new diagnostic imaging software can incorporate some version of the data models and protocols implemented by these two projects. On a broader level, however, the concrete examples of **SeDI** and **CaPTk** point to limitations in current frameworks such as **DICOM**. The integrative logic of **SeDI** and **CaPTk** is based on specific data structures — **DICOM** headers and **CWL**, respectively — and arose out of practical limitations in existing **DICOM** software. The **SeDI** and **CaPTk** solutions may be a methodological guide for related projects, but neither is a general-purpose framework for solving all data and code integration problems (not to imply that such a general-purpose solution exists). Integration problems are not exhausted by deriving solutions in one specific area: for example, merging **DICOM** metadata into **RDF** graphs may successfully align data structures conforming to current **DICOM** specifications, but does not guarantee integration of novel extensions or supplements to **DICOM**. Much as **DICOM-RT** extended **DICOM** to incorporate radiation-therapy recommendations, predictive modeling and patient-centered Comparative Effectiveness reach could easily lead to new data standards as researchers seek to integrate imaging data with treatment plans and outcomes evaluation.

Because **CaPTk** is extensible as part of its essential design, it is both technologically and philosophically consistent with **CaPTk**'s structure to introduce extensions which focus on the analytic convergence or cross-referencing between image-analysis and outcomes data; and to implement data-sharing protocols among **CaPTk** modules which model clinical and outcomes information alongside image data. In this sense, **CaPTk** is a logical starting point for the design and situating of hybrid radiological/outcomes software in a larger computational context, as will be detailed below.

2.2 Multi-Application Networks in the Context of Scientific Research Data

Architecturally, the pattern of organization just described — semi-autonomous applications linked together (often by virtue of being common extensions to an overarching "core" software platform) is analogous to the collection of software components that may share access to a data repository or a research-data corpus, include a corpus of medical/diagnostic images. The purpose of research data archives — particularly when they embrace contemporary open-access standards such as **FAIR** (Findable, Accessible, Interoperable, Reusable) [13] and the Research Object Protocol[6] — is to promote reuse and reproduction of published data and findings, such that multiple subsequent research projects may be based on data originally published to accompany one book or article. As a result, it is expected that numerous projects may overlap in their use of a common underlying data set, which potentially means a diversity of software components implementing a diversity of analytic techniques, each offering a unique perspective on the underlying data. In addition to providing diverse analytic methods, research-oriented software

[5] No native **C** or **C++** libraries are described on the **CWL** website among the tools and parsers available for **CWL**, but **CaPTk** is mentioned on a corresponding discussion thread concerning **C++** libraries. It seems therefore that the **CaPTk** "utilities" repository provides the de-facto standard **C++** implementation of **CWL**, at least according to the **CWL** group themselves.

[6] see `http://www.researchobject.org/scopes/`

transforms the ecosystem where scientific data and academic books/articles are published and explored. From a reader's point of view, open-access data sets present an interactive, multi-media experience which supplements reading article texts. Indeed, in recent years, publishing houses have embraced the notion (albeit more of a future vision than a present reality) that conventional publications are only one part of a larger package (e.g. a "Research Object Bundle"), which may contain data, computer code, and/or interactive graphics alongside text-based formats such as **PDF**. At the same time, from a scientist's point of view, open-access research data offers a starting point for their own investigations, or a contretemps with which to reinforce or contrast their own findings.

All of this means that scientists preparing academic papers are not only finalizing text descriptions but also, often, curating data, graphics, or code that demonstrates their work in an interactive, multi-media fashion (such assets are often presented to readers via "supplemental material," "additional files," or "data availability" sections on web pages showing article texts or abstracts). Scientists can support multi-media exploration of their research by presenting data in file formats used by data-visualization software; they can also assert more fine-grained control over data visualization by implementing custom software. One benefit of custom data-set software is the possibility of using *microcitations* to connect research data (and the **GUI** components where this data is visualized) to publication texts.[7] Microcitations enable application-level interoperability between document viewers and data-set applications. For instance, individual table columns can be associated with specific scientific concepts or statistical parameters that are described in the article text. This interop between data sets and **PDF** viewers is an example of multi-application networking — both the data-set application and the **PDF** software need to be implemented or extended with the capacity to execute microcitation-based workflows. Moreover, data-set applications can provide (through their type system and implementation protocol) one form of structural model and formal elaboration of research theories, methodology, or experimental design. Defining an annotation schema for data sets can potentially be an organic outgrowth of software-development methodology — viz., the engineering steps, such as implementing unit tests, which are essential to deploying a commercial-grade application.

Implementing a robust research-data software framework involves integrating multiple scientific applications, but also (ideally) extending these applications with features specifically of interest to those conducting or reviewing research using published data sets and/or described in academic literature: for instance, capabilities to download data sets from open-access scientific portals; to parse microcitation formats; and to interoperate with document viewers. This review of data-publishing technology is relevant to radiology and to Patient-Centered Outcomes because it typifies the emerging ecosystem where scientific research and open-access data is being disseminated. Open-access data publishing — especially within the emerging frameworks being advocated and developed by publishers and research institutions themselves — is conducive to a software engineering paradigm that favors the implementation of autonomous software agents which, their autonomy notwithstanding, can interoperate to the degree that they are collectively oriented to a shared data source. Such a multi-application network requires integration not only at the level of data structures, but also at the level of Human-Computer Interaction and inter-application messaging — in the optimal case users can switch between applications based on each components' respective capabilities. In sum, the **CaPTk** architecture — consisting of numerous autonomous, stand-alone, native applications federated into a decentralized but unified platform — is similar logistically to the kinds of application networks appropriate for the technology supporting archives of research data (including diagnostic-imaging repositories).

Given this architectural correlation, we would argue that the **CaPTk** architecture can be used as a prototype for implementing multi-application networks such as those applicable to research data repositories. Initiatives such as Research Objects and **FAIR** advocate for a technological infrastructure characterized by a diverse software ecosystem paired with open-access research data sets. Scientific data repositories, linked to academic publishing platforms, have been engineered to help scientists locate and re-use data sets which are relevant to their research projects. Although formats such as Research Objects have been standardized over the last decade, there has not been a comparable level of attention given to formalizing how multiple software applications should interoperate when manipulating overlapping data. The Common Workflow Language (**CWL**), which has been explicitly included in the Research Object model, documents one layer of inter-application messaging, including the encoding of parameters via command-line arguments; **CaPTk** provides a **C++** implementation of **CWL** and uses it to pass initial data between modules. Serializing larger-scale data structures is of course a generic task of canonical encoding formats such as **JSON**, **XML**, **RDF**, and Protocol Buffers — not to mention text or binary resources serialized directly from runtime objects via, for instance, **QTextStream** and **QDataStream.** This means that some level of inter-application communications is enabled via **CWL**, and that essentially any computationally tractable data structure can be encoded via formats such as **XML**. These solutions, however, are sub-optimal: **XML** (as well as **JSON** and analogous formats) is limited because it takes additional development effort to compose the code that marshals data between runtime and serial formats. Similarly, although **CWL** can model information passed between applications, it provides only an indirect guide for programmers implementing each application's "operational semantics" — viz., the procedures which must be executed before and after the event wherein data is actually passed between endpoints.

In the context of **CaPTk**, for example, integrating peer modules with the **CaPTk** core application involves more than simply ensuring that these endpoints communicate via a standardized data-serialization format: the plugins must be *registered* with the core application, which affects the core in several areas, including the build/compile process and construction of the main **GUI** window. Modeling the interconnections between semi-autonomous modules, as **CaPTk** demonstrates, therefore requires more detail than simply modeling their shared data encodings; it is furthermore necessary to represent all procedural and User Interface requirements in each component that may be affected by the others. Despite the standardization efforts that have been invested in Research Objects and the

[7] Microcitations are independently citeable smaller units of a data set, such as an individual record, a table column, or one table in a multi-table collection; we will discuss microcitations further in the next section.

Common Workflow Language, we contend that this fully detailed protocol for multi-application interop has not yet been rigorously formalized.

Rigorous models of application networks among semi-autonomous components acquire an extra level of complexity precisely because of this intermediate status: protocol definitions have to specify both the functional interdependence and the operational autonomy of different parts of the application network. Although one application does not need detailed knowledge of the other's internal procedure signatures (which would break encapsulation), a rigorous messaging protocol can be developed by specifying requirements on the relevant procedures implemented by each application. Developers can then explicitly state that a given set of procedures implements a given protocol (the multi-application documentation thereby has more detailed information about application-specific procedures than each application has vis-à-vis its peers). The functional interdependence between applications can accordingly be modeled by defining protocols which must be satisfied by procedure-sets internal to each end-point — the relevant information from an integrative standpoint is not the actual procedures involved, but confirmation that the relevant procedure sets adhere to the relevant multi-procedural protocol.[8]

3 Precision Medicine in Trial Design

Data-sharing initiatives need to pay particular attention to the logistics of hosting/accessing data in contexts where granular patient-specific information is important, such as immunoprofiling. Questions which should be addressed include (1) where data is to be hosted; (2) how participating institutions should submit data to a central repository; (3) how participating institutions and/or outside investigators should access previously-deposited data; (4) how to ensure anonymization of patient-specific records; (5) how to ensure that different labs used by different hospitals are utilizing compatible protocols, so that results from multiple labs/hospitals can be seamlessly merged in a shared data commons; (6) how to ensure proper alignment between software employed at different institutions; and (7) how to incorporate data curated within the context of a multi-institutional data-sharing initiative into scientific papers documenting research findings. Each of these areas of concern are technically demanding because of the complex and heterogeneous nature of immunological profiles.

As a case-study in clinical-trial software engineering, consider again the proposals in Shrestha *et al.* [12] which we reviewed last chapter. As these authors recommend, Covid-19 trials should be designed to focus on specific patient groups which are more likely to benefit from the interventions that form the basis of the relevant clinical trials. Moreover, toward the goal of applying precision medicine to Covid-19 clinical practice, it should be possible to construct a quantifying domain of patient-profile signals (antecedent to trial commencement) to quantify the statistical probability that a given treatment will have a favorable patient outcome in relation to all the prior data in a patient's profile. Since researchers assume that certain factors in a patient's profile will be statistically correlated with favorable outcomes in conjunction with specific treatment plans, part of the trial's purpose is to determine which parts of the patient profile are, in fact, statistically relevant.

In practical terms, then, setting up Covid-19 trials would involve defining patient-selection criteria and implementing systems to screen for patients who may be good candidates for different trials. This would require two steps: (1) constructing a format where trial criteria can be rigorously notated; and (2) implementing software at each participating hospital to search for good candidates to register in each trial. The trial-specific software would need to query each hospital's clinical and/or diagnostic records. Because patient-specific factors for determining trial eligibility would cover a broad spectrum of data types — from sociodemographics and medical history to domain-specific lab/image results — the screening software would, therefore, need to integrate such heterogeneous data sources.

Given the sheer scale of the SARS-CoV-2 pandemic, there are likely to be many candidates for almost any Covid-19 trial. However, because of the extent of the pandemic, governments and hospitals have had to establish treatment protocols rather haphazardly. As a result, there has been a fair amount of trial and error as opposed to a *systematic* framework for evaluating diagnostic regimens and treatment protocols. For example, rather than have one or two canonical SARS-CoV-2 antigen tests, used to measure virus antibody levels in patients, the US Centers for Disease Control recommends or has authorized a large list of assays performed by many different companies, using many different biochemical methods.[9]

Because the format of data resulting from immunoassays depends on the specific biochemical mechanism which (within each assay) yields quantitative data, a broad spectrum of antigen tests requires a diverse array of data formats which need to be integrated. As such, whenever Covid-19 immunoassays are considered as critical factors in immunological profiles for mapping patients to appropriate Covid-19 clinical trials, querying for good trial candidates means querying across a wide spectrum of structurally different data types that correspond to this broad array of antigen tests — specifically to the mechanisms through which laboratory instruments generate quantifiable data and to the computational procedures which process such data. Here is a good example of why specialized custom software is necessary: given the heterogeneity of Covid-19 data, synthesizing all this information calls for integrative procedures implemented directly in programming languages such as **C++** (rather than query languages such as **SQL**).

3.1 Software Alignment for Covid Phylogeny Studies

Similar to immunological studies, discussed above, analysis of SARS-CoV-2 mutations is yet another area where software alignment can prove to be important. Studies suggest that variations in the viral strain causing Covid-19 symptoms may be partly responsible for divergent immunological responses to the virus across

[8] Reviewing the source code and documentation for CaPTk confirms that multi-application messaging along these lines is implicitly adopted by CaPTk; see for example https://www.med.upenn.edu/cbica/assets/user-content/images/captk/2018_ISBI_CaPTk.0404.Part2.pdf, particularly the material starting on the 30th slide of that presentation.

[9] *See*, for example, https://www.fda.gov/medical-devices/coronavirus-disease-2019-covid-19-emergency-use-authorizations-medical-devices/vitro-diagnostics-euas#individual-antigen.

the patient population.[10] If one patient responds either less favorably or more favorably than the average patient-response to a given treatment, clinicians need to assess whether this difference can be explained solely by the patient's prior immunological profile or whether the patient has been exposed to a genetically divergent viral strain.

Modeling SARS-CoV-2 evolution across the globe is a massive project. There have been over 41 million Covid-19 cases worldwide, in virtually every nation on earth, so a complete phylogenetic picture of SARS-CoV-2 in humans would need to pool data from many different healthcare systems. Yet, even technically detailed analysis of the phylogeny of SARS-CoV-2, such as that conducted by Dearlove *et al.* (as reported at the end of 2020) only considered 27,977 patients (about 0.1% of global cases), with almost half from the United Kingdom [3]. Hence, achieving something resembling a holistic picture of SARS-CoV-2 mutations and how they might affect clinical treatments, would require many parallel studies analogous to that of Dearlove *et al.*.

This then raises questions of study alignment: because calculating viral phylogeny requires making technical decisions about how genetic sequences should be acquired and analyzed, decisions which may vary among research teams, spanning many healthcare institutions. For example, Dearlove *et al.* describe several computational steps which they had performed both to normalize each SARS-CoV-2 genome sequence in their data set for cross-comparison and to run predictive simulations (used to estimate whether divergence between sequence-pairs are the result of localized, random mutations or, conversely, an indication that SARS-CoV-2 is evolving into further distinct strains). Clustering SARS-CoV-2 genomes into variants — that is, identifying which mutations are random and which appear to be propagating to subsequent viral generations — involves making computational and biological assumptions, such as how to statistically marshal genomic data so as to quantify the prevalence of a mutation, and how to estimate whether a particular mutation confers an adaptive benefit to the viral agent (e.g., an ability to elude antibodies targeting structural proteins).

Given that modeling viral phylogeny requires certain computational assumptions and biological guesswork, data from multiple studies can only be reliably integrated if there is some degree of alignment across their methodology. As such, research teams should document their protocols in a manner that permits assessment as to whether protocol differences might compromise the resulting data. One way to achieve this is to model the protocol itself as a datatype in a programming language such as **C++**. For each study, such as Dearlove *et al.* cited above, there would then be a **C++** object encapsulating all details of the researchers' protocols and computational workflows. Assuming that research into SARS-CoV-2 phylogeny could be centralized into a publication archive, then their associated protocol-objects could be aggregated into a protocol database. The purpose of such a database would be to establish a composite picture of SARS-CoV-2 evolution by synthesizing data from multiple studies and/or multiple countries. Protocol-alignment would be one part of a common framework to quantify the epidemiological significance of SARS-CoV-2 mutations.

In short, a holistic global picture of SARS-CoV-2 must represent SARS-CoV-2 mutations which have been deemed phylogenetically and/or clinically significant (i.e., having potential either to influence the overall evolution of Covid-19 and/or to have some bearing on clinical treatments), and must *also* represent divergent SARS-CoV-2 strains. These data-points would then be the basis of further details such as: when did a given strain and/or mutation first appear? Is the strain/mutation geographically localized? What is the proportion of different strains/mutations in a geographic area? Is there evidence that different strains/mutations affect a patient's immunological response to Covid-19 and/or the effectiveness of vaccines, antibody regimens, steroids, or other clinical interventions? How can genetic mutations within the SARS-CoV-2 virus be correlated with structures in the spike proteins encoded by the viral genes? This last question points to the importance of integrating genomic data with **3D** molecular models. Whereas data structures modeling the viral genome are composed of nucleotides — and, at a higher scale, Open Read Frames (**ORF**s) — data structures describing the biophysics of glycoproteins involve **3D** geometry and chemical bonds. Analyzing how SARS-CoV-2 genes affect the production of glycoproteins, therefore, requires annotating and cross-referencing nucleotide/**ORF** data structures with molecular models encoded in formats such as Protein Data Bank (**PDB**).

Assuming that a synthesis of SARS-CoV-2 genomic research is based on a **C++** protocol object model, as suggested above, one can similarly develop **C++** data types to model SARS-CoV-2 strains and mutations. The trio of protocol, strain, and mutation objects could then serve as the basis of a unified Object-Oriented framework for representing SARS-CoV-2 evolution across the world. This would allow the results of different studies to be compared, so as to build a composite global profile of SARS-CoV-2 phylogeny. We are not aware of Object-Oriented models proposed as representational devices for Covid Phylogeny, but this form of software design would be consistent with code developed in contexts such as oncology, for example the computational simulation of tumor growth, which we will discuss next chapter. Covid-Phylogeny Object Models could serve as a nexus for merging temporal and geographical data concerning the epidemiology of SARS-CoV-2 mutations with genomic data demonstrating which mutations are significant, as well as clinical data tracking correlations between mutations and treatment outcomes. Supporting large-scale multi-trial data integration would, therefore, also introduce new requirements for clinical trial software.

3.2 Customizing Clinical Trial Management Software

Once trials embrace heterogeneous data models (which require special-purpose software for accessing some of the trial data), Clinical Trial Management Systems (**CTMS**) requirements become more complex. In these situations, **CTMS**s may need to model and in some cases replicate complex computational workflows, such as those employed by Dearlove *et al.* for calculating SARS-CoV-2 genomic sequences from patients' blood samples. The **CTMS** software may also need to interoperate with domain-specific ap-

[10] *See* for instance Osman Shabir, M.Sc., "The Phylogenetic Tree of the SARS-CoV-2 Virus," News Medical, https://www.news-medical.net/health/The-Phylogenetic-Tree-of-the-SARS-CoV-2-Virus.aspx; the author identifies one region in particular where "[c]hanges within this region may account for the differences in immune responses to SARS-CoV-2."

plications, as in bioimaging and image analyses, signal processing (e.g., for **EKG** analysis), Flow Cytometry, biochemical assays, genomic analysis, epidemiological modeling and so forth. If possible, such applications should be configured or extended to work with the clinical trial software. For instance, if a **DICOM** (Digital Imaging and Communications in Medicine) client is used to study an image derived from a specific trial — e.g., a radiological scan of a Covid-19 patient's lungs — the **DICOM** software could be provided with a plugin that would show trial information in a separate window, which could then be juxtaposed with the main-image view. In this context, software alignment means that all institutions participating in a trial could use the *same* plugins, so that the trial's central **CTMS** system could interoperate with special-purpose software in a consistent manner. This would also aid in establishing, as part of the trial design, protocols for depositing special-purpose data assets (such as **FCS** or **DICOM** files) alongside clinical data and **ECRF**s.

A further benefit of **CTMS** customization is that custom software adds flexibility for trial design. By definition, trials allow researchers to test biomedical hypothesis in a controlled manner. Trials are, therefore, defined around the premise that observational information resulting from the trial is empirically significant, revealing something new about what the trial was designed to investigate. For instance, a Covid-19 trial might assess how well patients with varying prior immunological profiles respond to monoclonal antibody (**MAB**) treatments. The relevant observations in this case derive from the subsequent course of the disease for each patient, as well as potential adverse reactions, but there are, however, inherent complicating factors, such as: (1) were patients receiving other treatments as well? (2) for patients who recover, how do we know that the antibodies expedited that recovery? (3) how quickly was the recovery? and (4) did patients continue to suffer from Covid-related symptoms even when they were no longer infectious?

In addition to these post-treatment observations, situational details specific to the trial — such as each patient's **MAB** dosage level, prior Covid-19 risk factors, or viral-load change over time — need to be incorporated into the trial's unique data models. Moreover, a comprehensive investigation could well incorporate both information about the patient's unique immunological profiles and the nature of the SARS-CoV-2 variant/strain found in the patient.

3.2.1 Toward Fine-Grained Sociodemographic Models

Patient-centric data could likewise include sociodemographic information about the patient, supplemented by epidemiological metrics, such as contact tracing: was the patient living or working in an environment where they were likely to have been exposed to the virus? If not, how did they contract it?

Flexibility in trial design is important because the correct protocols for modeling and integrating both pre-treatment and post-treatment data need to be worked out for each trial, depending on the trial's goals and logistics. Designers need to identify, for example, what dimensions of patients' immunological and sociodemographic profiles are likely to be consequential when analyzing treatment outcomes. It is important for trial designers to model sociodemographic data attentively. Indeed, biomedical research has increasingly been criticized in recent years for bias toward certain populations (e.g., white, middle-class non-seniors), leading to the unavoidable consequence that certain racial, age, and socioeconomic categories will be vastly under-represented in trial cohorts. This results in uncertainties as to how well trial results carry over to populations at large — that is, populations characterized by a heterogeneous mix of demographic factors. Trial designers can mitigate these concerns by demonstrating sociodemographic diversity among trial participants.

Demonstrating sociodemographic diversity, however, calls for transparency about how sociodemographic details are represented. The process of grouping patients into ethnic/racial and/or socioeconomic strata can be equivocal at times. For example, if a trial participant is a graduate student at the University of Chicago, should their socioeconomic status be assessed on the basis of their own income or that of their parents? If their zip code places them on the South Side of Chicago, a region with both a prestigious campus and pockets of extreme poverty, should they be demographically classified alongside residents of that neighborhood? What about a varsity linebacker who appears to be in excellent health before a Covid-19 infection? Should his status as an athlete be taken to indicate being extremely fit prior to the disease, or might his background as a football player intimate a potential history of brain trauma which may compound his neurological damage due to Covid-19?

In short, sociodemographic data can be notoriously imprecise. It is well-known that patients of lower socioeconomic status have higher Covid-19 infection rates and mortality rates than patients of higher socioeconomic status, but this disparity may be explained by several causal factors: more infectious workplaces, inferior post-infection treatment, poorer state of health before the onset of Covid, and so forth. Teasing apart these factors demands fine-grained analysis of the individual patient's pre-Covid history; sociodemographic generalizations can exclude important details. For example, for purposes of analysis, in the case of a graduate student with middle-class parents but no health insurance of their own, how should we quantify their degree of access to health care? How well does geographic location serve as a proxy for socioeconomic status?

The case of a middle-class student living in a well-off near-campus corner of an otherwise impoverished neighborhood suggests that geospatial metrics (such as zip code or congressional district) are imperfect proxies for wealth; but other factors — such as air/water pollution or the risk of being the victim of a crime — may be statistically correlated among geographically proximate residents even if they have otherwise divergent sociological profiles. These examples illustrate that the value of sociodemographic data is proportionate to the level of statistical detail with which the data may be analyzed. Instead of a broad and vague designation, such as "low income," one may want to derive more detailed subclasses, incorporating information about patients' employment, access to health care, physical fitness, and so forth. Two patients with similar income levels, for instance, may have different levels of access to health care (depending on factors such as whether the patients have employer-based insurance or geographic proximity to healthcare facilities) or different levels of exposure to Covid-19 in the workplace, depending on the nature of their job. Even if a trial does not quantify granular sociodemographic assessments — such as

patients' workplace conditions and access to health care, which might serve as a more accurate estimation of how socioeconomic status causally affects disease outcomes — subsequent researchers may determine ways to analyze or add on to data generated by a trial (e.g., through follow-up studies of enrolled patients), so as to make such sociodemographic granularity part of the quantifiable framework.

3.2.2 Measuring Cognitive and Neurological Effects

Similar interpretive issues of interpretation also apply to post-treatment observations. How should researchers decide which observations qualify as clinically significant consequences of Covid-19? We have seen that as the pandemic has unfolded, a fair number of cases have been cited in the professional literature describing Covid-19 patients who suffer certain cognitive/neurological effects, such as muscular fatigue or weakness, mental confusion or poor concentration (sometimes referred to as "brain fog"), or symptoms of Guillain–Barré syndrome spectrum. Almost certainly, some of these symptoms may be the product of cognitive/neurological effects due to SARS-CoV-2. At the same time, Covid-19 patients — even those who fight off the infection successfully or who test positive but remain asymptomatic — may find their lives so disrupted by the pandemic that this may (indirectly) cause cognitive and neurological problems. For instance, prolonged inactivity (for a typically active person), which commonly occurs as the result of a quarantine, may contribute to poor concentration and other diminished forms of mental acuity. Given the fact that most people's lives during the pandemic are not "normal," it may be difficult to establish which symptoms experienced by a patient are actually biological effects of the disease itself or, alternatively, indirect consequences of lifestyle restrictions. This sort of ambiguity also applies to potential adverse side-effects of Covid-19 treatment. How long after a treatment is administered should a patient's symptoms be considered potential side-effects of the therapy itself or, alternatively, the result of nagging uncertainties and a disrupted lifestyle imposed by the pandemic?

The fact that these questions have no predetermined answers indicates that reporting protocols, which give some structure to this kind of amorphous and *ad hoc* patient data, need to be considered as an integral part of trial design. Specific parameters should be established for the post-treatment and post-recovery window of time, where patients' symptoms might be noted as potential effects of the disease or of the administered therapies themselves; moreover these symptoms should be verified and classified according to the trial design itself lest they could be lost altogether from the medical reporting process. The methodology for doing so should be rigorous but also flexible, because it is hard to predict *a priori* the range of patients' responses to both diseases and treatments. For example, because SARS-CoV-2 was initially believed to affect lung functioning primarily, the risk of long-term cognitive/neurological damage was not widely anticipated when considering treatment over the course of Covid-19 infection. Consequently, because tests of a cognitive/neurological/physiological/radiographic (except for basic lung scans, with respect to radiology) had not been a common facet of early Covid-19 trials/observational studies, there were no corresponding data structures included in those studies for capturing this full spectrum of neuropsychological and neurological data.

Systematically tracking Covid-19's cognitive/neurological deficits would entail neurological, laboratory, neuropsychological and radiographic tests to understand the full extent of their cognitive and neurological impairments. For example, the use of s when considering Covid-related ischemic stroke [7], [5], the use of electrophysiological tests, cerebrospinal fluid tests (**CSF**), or the **MRC** (Medical Research Council) Scale for muscle-strength test when considering Covid-related Guillain-Barré symptoms [9], or the use of neuropsychological tests, such as Trail Making Test (**TMT**), Sign Coding Test (**SCT**), Continuous Performance Test (**CPT**), and Digital Span Test (**DST**) — which measure a patient's executive abilities (letter and number recognition mental flexibility, visual scanning, and motor function) and sustained and selective attention, along with other cognitive and neurological functioning — when considering cognitive/neuropsychological impairments after a serious bout with Covid-19 [16].

These cognitive, neurological, physiological, laboratory, and radiographic data structures thereby become an integral part of the information relevant to trial evaluation, because they document symptoms which are presumptively attributable to Covid-19. However, in practice, prior to clinicians having been alerted to the fact that Covid-19 may cause lingering cognitive/neurological damage, Covid-19 trials were not designed to incorporate neurological or cognitive data in a systematic manner. This scenario points to how trial data models in Covid-19 studies can benefit from a built-in capacity to be redesigned impromptu while the trial is ongoing so as to accommodate new and emerging information, such as debilitating cognitive or neurological impairment, as discussed above.

The example of neurological/cognitive damage attributable to Covid-19 illustrates the larger point that clinical trials' data models should be flexible and suitable to become more fine-grained over time. Fully accounting for all relevant observational details, such as impaired cognitive and neurological functioning as in the case above, may not be feasible during an initial analysis of trial data. Trial data should, therefore, be modeled within a framework which supports the relatively free-form aggregation of new information and new observations (such as a graph database) rather than the rigid schema of relational tables *ab initio*. In keeping with the way in which clinical data emerges in an *ad hoc* way during the longevity of a clinical trial, custom **CTMS** software would then allow each trial to expose its data models in a procedural, Object-Oriented fashion, using trial-specific software as a Reference Implementation illustrating how trial data is structured and curated.

3.2.3 Aggregating Trial Data via Graph Models

Consider how patient profiles usually consider the medications each patient is taking — any patient (a node) could in principle be connected to any medication (another node); some patients may be taking *no* medications, others may take just *one*, and some may take *two or more*. Also, connections between patients and medications can be the basis of further details that emerge over time (and are registered in the graph) inasmuch as medications are prescribed to patients by a specific doctor at a specific time, in a specific dosage, in response to specific diagnostic tests, and so forth. In short, infor-

mation can "fan out" from the patient-to-medication connection in a relatively free-form manner. In general, then, as a subset of overall patient profiles, information about medication evinces the structural features which are, in many contexts, optimally represented via free-form labeled graphs. Meanwhile, patient profiles may also consider medical history, which can be modeled as a graph with detailed logical and temporal inter-node connections. According to this representational strategy, each patient's history is a series of events and observations which are temporally ordered — it is possible to query or traverse the graph in a manner which takes before/after relations into account — and where there are also logical or causal connections defined between nodes. For instance, an edge might assert that a given medication was prescribed to a patient (an event) *because of* the results of a given lab test (an observation). In these examples, different sorts of clinical data — sociodemographic, pharmacological, medical-history — are modeled according to different sorts of graph structures (hypernodes, nonschematic labeled edges, temporalized graphs, and so forth).

A useful data-integration case-study is the University of Pennsylvania's Carnival project (which achieves data integration by adopting property-graph databases, illustrating the flexibility of graph models in a way that we can also apply to trial design). Carnival synthesizes heterogeneous biomedical data by translating information from disparate sources into a common property-graph representation and then querying this data with the Gremlin Virtual Machine. Gremlin is a "step-based" virtual machine where "steps" between potential focus elements in a property graph play the role of primitive processing instructions; querying and traversing property graphs involves executing a series of Gremlin steps.[11] Most Gremlin implementations are based on the Java programming language and the Java Virtual Machine (**JVM**), so that queries themselves are written in a **JVM** language (Groovy, in the case of Carnival). The challenge for any database engine which employs a relatively complex data-representation strategy — such as a hypergraph, property-graph, tuple-store (a collection of records with varying numbers of fields) or a multi-dimensional (possibly sparse) array — is to efficiently map the high-level data structures manipulated within the database itself to the lower-level memory units which are stored to disk. Lower-level data structures are typically modeled via simpler database constructions such as key-value stores, memory cells, or relational tables, so there must be a translation pipeline between high-level structures (properties, hypernodes, hyperedges, and so forth) and low-level points (record cells, shared memory address, key-value values, etc.)

3.3 Representing Trial Data via Object Models

As a concrete example of data-modeling principles proposed in the prior section, consider how the Object-Oriented models for Covid Phylogeny could serve as a nexus for integrating SARS-CoV-2 phylogenetic data across multiple studies and healthcare systems. Such an Object Model may be extended in different ways for different clinical trials examining a whole range of Covid-19 treatments. For a given antibody regimen, for instance, scientists need to quantify how well the antibodies disable Covid-19 spike proteins directly and/or how well the antibodies block the virus's ability to attach to human cells. These measurements generate data which indicate how well a patient's immune response to Covid-19 is boosted by the administered antibodies, information which is usually delivered in special-purpose formats such as **FCS** or fluorescence images. A Covid-19 software ecosystem would then need to ensure that such immune-response data can be effectively parsed and integrated into Object Models describing how SARS-CoV-2 is evolving around the globe. Such data integration modeling would allow researchers to reliably assess in each individual patient their immune response to the particular acquired SARS-CoV-2 strain/variant vis-à-vis their personal immunological profile, and to competently track both current and emerging SARS-CoV-2 strains throughout the population.

These sorts of flexible Object Models can then facilitate trials designed according to the novel protocol proposed by Shrestha *et al.*, discussed above, where each trial would study a preselected (non random) cohort of patients for whom both pre-treatment immunoprofiling data and post-treatment outcomes data would be available, so as to compare multiple trials against one another. Object Models customized for each trial would generate a data framework through which the causal relations between patient profiles and treatment outcomes would be investigated. Customized trial software would, accordingly, provide a Reference Implementation demonstrating each trial-specific Object Model. If adopted for multiple trials conducted across multiple clinics/hospitals, trials' data models may help doctors better understand which aspects of patient profiles are particularly significant when matching Covid-19 patients to the most salutary treatment.

4 Text and Data Mining via CORD-19

CORD-19 is a collection of research articles about Covid-19 which was developed (starting in Spring 2020) in conjunction with a White House "call to action" to spur Covid-19 research. This White House initiative was described as a "call to action ... to develop new text and data mining techniques that can help the science community answer high-priority scientific questions related to COVID".[12] As raw data for this initiative, the US government helped spearhead a consortium of industry and academic institutions, headed by the Allen Institute for AI Research, who curated a "machine-readable Coronavirus literature collection" which includes article metadata and (in most cases) publication text for over 56,000 coronavirus research papers. This corpus is paired with links to publisher portals (including Springer Nature, Wiley, Elsevier, the American Society for Microbiology, and the New England Journal of Medicine) providing full open access to Covid-19-related literature; these resources collectively constitute **CORD-19** (the "Covid-19 Open Research Dataset").

The **CORD-19** collection was formulated with the explicit goal of promoting both text mining and data mining solutions to advance coronavirus research. This means that **CORD-19** is intended

[11] The theoretical foundations of step-based Virtual Machines are presented in Marko Rodriguez, "Stream Ring Theory," February 14, 2019 (https://zenodo.org/record/2565243#.X3vzqS4pDeQ).

[12] See https://www.whitehouse.gov/briefings-statements/call-action-tech-community-new-machine-readable-covid-19-dataset/

to be used both as a document archive for text mining and as a repository for finding and obtaining coronavirus data for subsequent research. The White House announcement directly requests institutions to develop *additional* technologies which would help scientists and jurisdictions to take advantage of **CORD-19** as it was initially published. In short, **CORD-19** was released with the explicit anticipation that industry and academia would augment the underlying data by layering on additional software.

Despite the obvious benefit to researchers, the health-care community, and the public at large in publishers choosing to release a substantial quantity of Covid-19 related literature in Open-Access fashion, **CORD-19** is not without certain limitations. These largely stem from how the articles are encoded into an ostensibly (**JSON**-based) machine-readable format. To be fair, the problems we identify here reflect the current authors' own personal assessments of the **CORD-19** corpus; they are not broad criticisms which have been asserted by researchers working directly with **CORD-19** or discussed in peer-reviewed literature. With that caveat, however, we assert that certain issues deserve mention:

Transription Errors Transcription errors can cause the machine-readable text archive to misrepresent the structure and content of documents, hindering text-mining technology that targets the archive. In **CORD-19**, for instance, there are cases of scientific notation and terminology being improperly encoded. As a concrete example, "**2'-C-ethynyl**" is encoded in **CORD-19** as "**2 0 -C-ethynyl**", which could stymie text searches against the **CORD-19** corpus (see [4] for the human-readable publication where this error is observed; the corresponding index in the corpus is 9555f44156bc5f2c6ac191dda2fb651501a7bd7b.json).

Poorly Indexed Research Data Although **CORD-19** provides a structured representation of a large collection of research *papers*, there is no easy-to-use tool for finding research *data* through **CORD-19**.

Poorly Integrated Research Data The research data which *can* be accessed through **CORD-19** evinces a wide variety of technical fields and formats, with distinct software requirements; as a result, it is a difficult task to merge and integrate different data sets related to Covid-19. At present, **CORD-19** does not include any software tools or computer code that would facilitate data integration.

Disconnect Between Text Data and Publisher Portals Although most of the **CORD-19** manuscripts represent peer-reviewed literature which can be accessed through document portals (for instance, the National Center for Biotechnology Information website), the **CORD-19** archival schema does not represent these links (except indirectly via Document Object Identifiers). As such, there is no easy way for researchers to find and read publications which have been flagged by text-mining algorithms as being potentially of interest to them. Furthermore, there is no direct mechanism to enlarge the **CORD-19** corpus with papers newly added to publisher portals.

To clarify the final comment: the Allen Institute for AI, which curated **CORD-19**, encourages publishers to contribute new (or newly-available) articles to the corpus. However, integration with **CORD-19** is not developed as a formal step in the publication workflow. In particular, publishers are not themselves generating machine-readable document infosets that can be integrated with the **CORD-19** schema (which, in turn, causes transcriptions errors and other problems as just outlined).

With respect to text mining, an immediate problem arises in **CORD-19**'s archive-construction methodology: especially, how the text was parsed from **PDF** files. This is a process which almost inevitably causes imprecise or inaccurate text representation, which can degrading the quality of the archive unless manual or automated corrections are made. In particular, the **CORD-19** library evinces transcription errors, as mentioned above (especially in relation to technical or scientific phrases and terminology); scientific notation in particular may be improperly encoded. Moreover, there is no semantic marking identifying that (say) the "**2 0 -C-ethynyl**" text segment has a specific technical meaning. These errors or limitations arise in part from unavoidable anomalies which occur when reading texts from **PDF** files rather than from machine-readable, structured formats such as **XML**.

It is also worth observing that the **JSON** format used for encoding **CORD-19** manuscripts presents some logistical challenges for any operations related to text-mining or to cross-referencing publications and data sets. In particular, **CORD-19** makes partial use of "standoff annotation"; specifically, document features such as citations and references are notated through character offsets into the paragraph where they appear. As a result, accurately reading these document elements requires synthesizing data points parsed from several distinct objects in the **JSON** code, which is only feasible given a client library built to interface with the **CORD-19** files in accord with their specific schema. Such a client library would implement convenience procedures to handle recurring tasks, such as obtaining the full bibliographic reference affixed to a given location in a manuscript.

With respect to *data* mining in the **CORD-19** context, the limitations in the currently available raw **CORD-19** data are even more pronounced than in the context of text mining. In particular, neither the article metadata nor the full open-access document collections have any mechanism for actually obtaining data published alongside research papers, or even identifying which papers have accompanying data in the first place. The Springer Nature collection which was originally one component within **CORD-19** illustrates the limitations of this relatively unstructured data-publishing approach (this following analysis will focus on Springer Nature, but the problems identified are no less pronounced on the other **CORD-19** portals — if anything, because Springer Nature allows readers to browse articles in **HTML** within the web portal directly, one can ascertain whether research data exists for an article without downloading and reading it; with other **CORD-19** resources it is actually harder to locate supplemental data when available). Initially, the Springer Nature portal encompassed 43 articles, of which 15 were accompanied by research data that could be separately downloaded (this number does not include papers that document research findings only indirectly, via tables or graphics printed inline with the text). Collectively these articles referenced over 30 distinct data sets (some papers were linked to multiple data sets), forming a data collection

which could be a valuable resource for Covid-19 research — not only through the raw data made available but as a kernel around which new coronavirus data could accumulate. However, there is currently no mechanism to make this overall collection available as a single resource.[13]

This problem demonstrates, among other things, how document-metadata formats such as the Research Object protocol are limited in applying only to *single* articles. As a result, there is no commensurate protocol for publishing *groups* of articles which are tied to groups of data sets unified into an integral whole. Open-access Covid-19 papers also reveal limitations of exiting online document portals, especially with respect to how publications are linked to data sets. In particular, there is no clear indication that a given paper is associated with downloadable data; usually readers ascertain this information only by reading or scrolling down to a "supplemental materials" or "data availability" addendum near the end of the article. Moreover, because the Springer Nature portal (and similar publisher resources) aggregates papers from multiple sources, there is no consistent pattern for locating data sets; each journal or publisher has their own mechanism for alerting readers to the existence of open-access data and allowing them to download the relevant data sets.

4.1 Data Integration within CORD-19

Aside from the issues which are likely to hinder text and data mining across **CORD-19**, the collective group of Covid-19 data sets also illustrates the limitations of information spaces pieced together from disconnected raw data files with little additional curation. The files included in this group of data sets encompass a wide array of file types and formats, including **FASTA** (which stands for Fast-All, a genomics format), **SRA** (Sequence Read Archive, for **DNA** sequencing), **PDB** (Protein Data Bank, representing the **3D** geometry of protein molecules), **MAP** (Electron Microscopy Map), **EPS** (Embedded Postscript), and **CSV** (comma-separated values). There are also tables represented in Microsoft Word or Excel formats. Although these various formats are reasonable for storing raw data, not all of them are actually machine-readable; in particular, the **EPS**, Word, and Excel files need manual processing in order to use the information they provide in a computational manner. A properly curated data collection would need to unify disparate sources into a common machine-readable representation (such as **XML**).

Going further, productive data curation should also aspire to *semantic* integration, unifying disparate sources into a common data model. For example, multiple spreadsheets among the Springer Nature Covid-19 data sets hold sociodemographic and epidemiological information relevant to modeling the spread of the disease. These different sources could certainly be integrated into a canonical social-epidemiology-based representational paradigm which recognizes the disparate data points which might be relevant for tracking the spread of Covid-19 (with the potential to unify data from many countries and jurisdictions).

This is not only an issue of data *representation* (viz., how data is physically laid out), but also of data types and computer code. According to the Research Object protocol, data sets should include a code base which provides convenient computational access to the published data. In the case of Covid-19, this entails creating a sociodemographic and epidemiological code library optimized for Covid-19 information, which would be the primary access point for researchers seeking to use the data which has been published in conjunction with the 43 manuscripts examined here that were aggregated on Springer Nature, along with any other coronavirus research which comes online. Similar comments apply not only to tabular data represented in spreadsheet or **CSV** form, but to more complex molecular or microscopy data that needs specialized scientific software to be properly visualized.

Considering the overall space of Covid-19 data, it is unavoidable that some files require special applications and cannot be directly merged with the overall collection. For instance, there is no coherent semantics for unifying Protein Data Bank files with sociodemographics and epidemiology; files of the former type have specific scientific uses and can only be understood by special-purpose software. Nevertheless, a well-curated data-set collection can make using such special-purpose data as convenient as possible. In the case of Protein Data Bank, a downloadable Covid-19 archive can include source code for **IQMOL**, a molecular-visualization application that supports **PDB** (among other file formats) and has few external dependencies (so it is relatively easy to build from source).

Indeed, a curated Covid-19 archive might include an enhanced version of software such as that **IQMOL** prioritizes Covid-19 research, with the goal of integrating biomolecular and social-epidemiological data as much as possible. For example, as Covid-19 potentially mutates in different ways in different geographic areas, it will be important to model the connections between "hard" scientific Covid-19 information and sociodemographics. As the pandemic evolves, genomic and biochemical information may be linked to particular strains of virus, which in turn are linked to sociodemographic profiles: certain strains will be more prevalent in certain populations. Consequently, models of Covid-19 variants will need to be formulated and then integrated with both chemical/molecular data and sociodemographic/epidemiological data. Different Covid-19 strains then form a bridge linking these different sorts of information; researchers should be able to pass back and forth from molecular or genomic visualizations of Covid-19 to social-epidemiological charts and tables based on viral strains. Ideally, capabilities for this sort of interdisciplinary data integration would be provided by a Covid-19 archive as enhancements to applications, such as **IQMOL**, that scientists would use to study the published data.

It is worth noting that a data-mining platform requires *machine-readable* open-access research data, which is a more stringent requirement than simply publishing data alongside which can be understood by domain-specific software. For example, radiological imaging can be a source of Covid-19 data insofar as patterns of lung scarring, such as "ground-glass opacity", is a leading indicator of the disease. Consequently, diagnostic images of Covid-19 patients

[13] As **CORD-19** has evolved, the publisher-specific sections therein appear to be merged into portals such as Springer Nature directly, so our above comments based on isolating Springer Nature articles are probably more applicable to the original archive design than the current technology. However, insofar as the current portal simply defers to publisher-specific search features, we would argue that accessing Covid-19 data sets through **CORD-19** is if anything more difficult than before.

are a relevant kind of content for inclusion in a Covid-19 data set (see [11] as a case-study). However, diagnostic images are not in themselves "machine readable." When medical imaging is used in a quantitative context (e.g., applying Machine Learning for diagnostic pathology), it is necessary to perform Image Analysis to convert the raw data (viz., in this case, radiological graphics) into quantitative aggregates (for instance by using image segmentation to demarcate geometric boundaries and then defining diagnostically relevant features, such as opacity, as a scalar field over the segments). In short, even after research data is openly published by article authors, it may be necessary to perform additional analysis on the data for it to be a full-fledged component of a machine-readable information space.[14]

Another concern in developing an integrated **CORD-19** data collection is that, logistically speaking, not all Covid-19 data is practical to reuse as a downloadable package. This is especially true for genomics; several of the aforementioned 43 coronavirus papers included data published via online data banks capable of hosting data sets that are too large for an ordinary computer. In these situations scientists formulate queries or analytic scripts that are sent remotely to the online repositories, so that researchers access the actual published data only indirectly. Nevertheless, access to these data sets can still be curated as part of a Covid-19 package; in particular, computer code can be provided which automates the process of networking with remote genomics archives through the accession numbers and file-formats which those archives recognize.

As a final point on the topic of integrating disparate **CORD-19** research data, note that an overarching framework for indexing Covid-19 data sets would also facilitate the process of cross-referencing article text and research data. In particular, the annotation system employed for **CORD-19** could profitably be enhanced by a system of *microcitations* that apply to portions of manuscripts *as well as* data sets. In the publishing context, a microcitation is defined as a reference to a partially isolated fragment of a larger document, such as a table or figure illustration, or a sentence or paragraph defining a technical term, or (in mathematics) the statement/proof of a definition, axiom, or theorem. In data publishing, "data citations" are unique references to data sets in their entirety or to smaller parts of data sets. A data microcitation is then a fine-grained reference into a data set: for example, "the precise data records actually used in a study" (as defined by the Federation of Earth Science Information Partners; see [8]), one column in a spreadsheet, or one statistical parameter in a quantitative analysis.

Ideally, the text-mining and data-mining notions of microcitation should be combined into a unified framework. In particular, text-based searches against the **CORD-19** corpus should also try to find matches in the data sets accompanying articles within the corpus. As a concrete example, a concept such as "expiratory flow" appears in **CORD-19** both as a table column in research data and as a medical concept discussed in research papers; a unified microcitation framework should therefore map *expiratory flow* as a keyphrase to both textual locations and data set parameters. Similarly, a concept such as *2′-C-ethynyl* (mentioned earlier in the context of transcription errors) should be identified both as a phrase in article texts and as a molecular component present within compounds whose scientific properties are investigated through **CORD-19** research data, so that a search for this concept can trigger both publication and data-set matches. Implementing this kind of unified search mechanism requires that data sets be *annotated* with techniques similar to those used for marking up Natural Language techniques.

Considering the inter-disciplinary nature of Covid-19 research, it is unavoidable that different scientists will need different sorts of specialized software to analyze the kinds of information relevant to their research. For instance, the computational techniques applicable to diagnosing coronavirus infection are scientifically very different from those used for genomic or epidemiological studies of the disease; it is impractical to expect pathologists to use the same software as bioinformaticians studying the pathogen, or for either to use the same software as virologists modeling the (potential or observed) spread of the disease. In short, even if scientists from disparate disciplines start with a common pool of raw data, they will need to analyze this data through a diverse set of supplemental computational tools, which will vary not only across disciplines but also in terms of the software and laboratory facilities available to different researchers through their institutions. In this sense it is impossible to unify all **CORD-19** data into a *fully* self-contained information space.

Nevertheless, committing to "standalone" data publishing remains a valuable goal even in a context where published data sets will invariably migrate to different digital ecosystems. Although scientists may use external digital tools *when necessary* to perform certain calculations, or when interfacing with laboratory equipment, we strongly recommend that the degree of variation across different domain-specific extensions to **CORD-19** be greatly minimized. Ideally, that is, the version of **CORD-19** (along with its supporting technology) found in a biomedical setting should be as similar as possible to that found in a biochemical context, or a health-policy context. In the absence of any initiative to limit this drift, **CORD-19** could easily devolve into a federation of separate resources which have no interconnection apart from their nominal focus on Covid-19.

This section has highlighted limitations of data sets published in conjunction with coronavirus articles made available as open-access resources on Springer Nature (and, by extension, **CORD-19**). The central point here is to argue for a distinct data-curation stage in the publication process, with data curators playing a role distinct from that of both authors and editors.[15] Moreover, the discussion has hopefully highlighted problems with current data-sharing paradigms, even those such as the Research Object and **FAIR** initiatives which are explicitly devoted to improving how open-access data sets are published. **CORD-19** exposes several lacunae in the Research Object protocol, for example, which point to the need for a more detailed extension of this protocol. In particular, an enhanced protocol should encompass:

[14] This does not mean that diagnostic images (or other graphical data) should not be placed in a data set; only that computational reuse of such data will usually involve certain numeric processing, such as image segmentation. Insofar as this subsequent analysis is performed, the resulting data should wherever possible be added to the underlying image data as a supplement to the data set.

[15] The point here is not to critique the work of individual authors; curating data sets according to exacting scientific standards demands a separate vein of expertise which typically lies outside researchers' disciplinary scope. The point is rather that publishers should recognize data curation as a distinct process and skill-set complementary to both writing and editing research works.

1. A canonical framework for archiving collections of data sets, not only single data sets (and not only groups of data sets published with a single research paper). For example, all data sets published alongside the 43 Springer Nature articles could be unified into a single collection.

2. A code base accompanying data-set collections designed to help research unify the information provided. Curating the overall collection would involve pooling disparate data into common representation, and implementing computer code which deserializes and processes the unified data accordingly. For instance, **CSV**, **EPS**, and Microsoft Word/Excel tables could be migrated to **XML**, **JSON**, or a more complex common format. Customized computer code could then be implemented specifically to parse and merge the information present in single data sets within the overall collection. This implementation would reciprocate the Research Object goal of unifying code and data, but again would operate at the level of an aggregate of research projects rather than a single Research Object.

3. A unified data-set collection should be self-contained as much as possible, and should be built around a foundation where advanced computing capabilities are available in a transparent, standalone fashion, without requiring tools outside the collection itself. One way to achieve this is by gravitating toward components that can provide features such as scripting and data persistence through components that can be shared in pure source-code fashion, such as the WhiteDB database engine [10] and the AngelScript scripting language [6].

4. A unified data-set collection should also provide prototyping and remote-access tools to interface with web-based information spaces that host data sets too large to be individually downloaded. Ideally, these would include simulations of remote services, which would help scientists understand the design of the remote archives and how to interface with them. These simulations could function analogously to (for instance) PurpleData, a prototyping tool for Google's BigData developed by Verily (the Alphabet subsidiary developing an online Covid-19 portal for remote diagnostics and disseminating public information about the pandemic).

5. Finally, a unified research portal should influence the design of the web portals where associated texts are published. It should be easy for readers to identify which articles have supplemental data files and to download those files if desired. Moreover, textual links should be established between publication content and data sets — for instance, a plot or diagram illustrating statistical or equational distributions should link to the portion of the data set from which that quantitative data is derived.

This discussion has used the Covid-19 crisis as a lens through which to examine data-publishing limitations in general. Such limitations are not specific to coronavirus in particular. However, the nearly unprecedented urgency of this epidemic reveals how both the scientific and publishing industries are still struggling to develop technologies and practices which keep pace with the intersecting needs of systematic research and public policy. An optimistic projection is that the crisis will spur momentum toward a more sophisticated data-sharing paradigm — perhaps a generalization of the Research Object protocol toward data-set collections.

4.2 Reviewing the CORD-19 Document Model

In order to discuss the possibilities and limitations of **CORD-19** (and potentially other document corpora with a similar design) it is worth examining how **CORD-19** encodes textual data in greater detail. This discussion has ramifications outside of **CORD-19** itself, insofar as **CORD-19** hopefully points to gaps in current publishing technologies. These gaps need to be addressed if publishers are to curate open-access corpora which truly leverage the digital and interactive technology available to us with modern software.

The basis of **CORD-19**'s infrastructure is a **JSON** scheme which describes the document hierarchy of research articles encoded within the corpus. Apart from metadata (consisting of basic details such as document title and authors' names) and bibliographic entries, all document content according to this schema is divided into paragraphs (implicitly the documents are divided into sections as well, but sections are notated as properties of the paragraphs they contain, not as a separate level in the hierarchy). Each paragraph encoding contains an underlying string vector (a stream of characters) and, separate and apart from that, character "spans" which point to references (such as Named Entites), citations, and equations. This indicates that the **CORD-19** encoding uses "standoff annotation," where any content modifying the interpretation assigned to portions of the main text is notated with a series of data structures described apart from the main text itself.

Standoff text-encoding systems may be contrasted with **XML** or **HTML**, where "tags" are mixed with character data. For example, consider a span of text which quotes from another document: in **HTML**, the special status of the quoted text may be marked by surrounding the text with **<quote>** start and end tags. Syntactically, this markup system has the effect that tags and text are seen side-by-side: any content governed by the **<quote>** (i.e., the text of the quote itself) is printed immediately after the begin-tag, and the quotation ends when the last character is followed by an end-tag (i.e., **</quote>**).

Apart from such syntactic details, the distinction between tag-based markup and standoff annotation determines the "semantics" of the document, insofar as tags form a document hierarchy. Continuing the **<quote>** example, the text-span inside the quote tags is represented as a *child element* of the quote, whereas the quote itself may be a child element of a larger-scale entity (such as a paragraph). In effect, the paragraph *contains* a quote, and the quote *contains* a string of characters. Such nested levels of containment provide the structure through which hierarchical documents (formats such as **XML** and **HTML**) are interpreted.

To see the contrast with *standoff* annotation, if one were to describe a document using a standoff annotation system, the notation that a particular span of characters belongs to a quotation would not be marked-up amidst the characters themselves. Instead, the quotation-designation would use numeric indices to declare that the character at a certain position in the main text begins a quotation,

and some later character in the text ends that quotation. When serializing documents with standoff annotation, all the characters in a document are typically represented as one character-stream, and any notation describing markup applied to spans within that character stream is asserted afterward, using indexes into the stream to demarcate element boundaries.

The **JSON** schema used for **CORD-19** is not entirely standoff, because there is a document hierarchy (for example, a publication's abstract is modeled as a sibling element to the main body text, so abstracts and the main text represent an intermediate hierarchical level, contained within the overall document and containing individual paragraphs). However, **CORD-19** uses in effect a standoff-annotation system for each paragraph, so there is no hierarchical level smaller than paragraphs themselves, except implicitly; after the text (viz., the character stream) there are subsequent notations of spans within the paragraph (each span description is considered a child of the paragraph itself, as is the paragraph text).

This arrangement has consequences for text mining algorithms, which may be strengths or weaknesses in different contexts. One consequence is that the raw text is all grouped together in one place — algorithms do not have to tie together child nodes of disparate **XML** elements to derive a beginning-to-end sequence of the text belonging to any paragraph. Instead, it is simply necessary to read all data in the "text" field of the relevant "paragraph" object. The character-sequence in this text may contain words and sentences, but potentially other strings of symbols (such as chemical formulae) which are not explicitly marked. This may or may not be desirable. It could potentially complicate **NLP** tasks, because the language-processing components will be fed not only sequences of English words but also, sometimes interspersed among ordinary words, technical symbol-sequences such as "2'-C-ethynyl" (an example used earlier in this chapter). Standoff annotations may or may not be effective in marking the boundaries of such extra-lexical sequences; certainly we cannot rely on Named Entity detectors to properly identify and demarcate the boundaries of all uses of technical terminology or special symbols (again, the limitations of automated annotation are discussed earlier in this chapter).

In discussing standoff annotation it is also worth considering how the text of **CORD-19** publications was obtained. According to **CORD-19** documentation, most full-text transcriptions in the corpus were obtained from **PDF** files, via a pipeline using (Text Encoding Initiative) **XML** as an intermediate representation. Necessarily, then the encoded text is only an approximate representation of the original:

> To provide accessible and canonical structured full text, we parse content from PDFs and associated paper documents. The full text is presented in a JSON schema designed to preserve most relevant paper structures such as paragraph breaks, section headers, and inline references and citations. ... We recognize that converting between PDF or XML to JSON is lossy. However, the benefits of a standard structured format, and the ability to reuse and share annotations made on top of that format have been critical to the success of CORD-19. ... Though we have made the structured full text of many scientific papers available to researchers through CORD-19, a number of challenges prevent easy application of NLP and text mining techniques to these papers. First, the primary distribution format of scientific papers — PDF — is not amenable to text processing. The PDF file format is designed to share electronic documents rendered faithfully for reading and printing, not for automated analysis of document content. Paper content (text, images, bibliography) and metadata extracted from PDF are imperfect and require significant cleaning before they can be used for analysis. Second, there is a clear need for more scientific content to be made easily accessible to researchers. Though many publishers have generously made COVID-19 papers available during this time, there are still bottlenecks to information access. ... Lastly, there is no standard format for representing paper metadata. Existing schemas like ... JATS[,] Crossref [or] Dublin Core have been adopted as representations for paper metadata. However, there are issues with these standards; they can be too coarse-grained to capture all necessary paper metadata elements, or lack a strict schema. ... Without solutions to the above problems, NLP on COVID-19 research and scientific research in general will remain difficult. [15, page 6]

As an example of these **NLP** issues, consider the challenge of demarcating all named entities, particularly technical character-sequences (such as chemical formulae) which are not ordinary lexemes. Whether or not authors explicitly mark up such sequences (they may well do so in that formulae or equations are often typeset differently than normal text) this markup is not preserved in **PDF** versions of articles. As the authors of the last-cited article point out, many (roughly 38%) of papers in **CORD-19** are also available in the **JATS** (Journal Article Tag Suite) format, which is a more precise text encoding than **PDF**. However, even in this context **JATS** does not compel authors to explicitly notate textual entities such as special terms or character-sequences — in fact **JATS** does not truly have an obvious structure or set of alternative structures for identifying what would normally be considered annotation-worthy text spans or named entities; the closest correlates are probably the generic **<kwd>** (keyword) and **<abbrev>** (abbreviation) tags as well as discipline-specific options such as **<chem-struct>** (for chemical structures) and **<disp-formula>** (for mathematical expressions). In short, building a corpus such as **CORD-19** for rigorous text-mining is made more difficult because authors and publishers do not publish texts in formats which are optimized for text mining in the first place; the acknowledged limitations of **CORD-19** reflect problems of industry practice, not programming lacunae that could be alleviated with more sophisticated **NLP** algorithms.

Having acknowledged these limitations, a discussion of document corpora could then reasonably pivot from the empirical goal of curating useful text archives from currently published text to examining how more sophisticated corpora may be published in the future. It is reasonable, for example, to propose that full-text publications be released *both* in reader-friendly **PDF** form *and* in machine-readable forms such as **JATS**. This is not just an abstract proposal; indeed, the text of this very book has been prepared using a novel document-generation system which creates both machine-readable structured text and **PDF** output, moreover

with cross-referencing between them; notably, the positions of discursively important textual markers, such as sentence boundaries, are mapped to **PDF** screen coordinates (the code library for the book includes document-generation code as well as the data set of coordinate positions generated as part of the book's publication workflow). In particular, it is reasonable for authors and editors to manually introduce textual annotations for content such as named entities, keywords, important technical terms, and other content which should be targeted by **NLP** engines separate and apart from ordinary lexemes with their conventional natural-language semantics. Typically such specialized terms/lexemes would be marked up in any case because they may require distinct fonts or styling than their surroundings. It is also reasonable to manually define sentence boundaries via simple rules (e.g. two following spaces mark the end of a sentence; a single space, such as that following an abbreviation, indicates situations where a character such as a period, which could potentially mark the end of a sentence, is actually playing a different discursive role).

By following simple rules of document content-entry and lexicography, certain **NLP** tasks, such as sentence-boundary and Named Entity recognition, can be optimized — eliminating the need for probabilistic algorithms and relying instead on much less sophisticated, but more accurate, markup-parsing logic. If sentence boundaries and Named Entities are explicitly annotated in machine-readable text encodings, then extracting these features is not really an issue of "Natural Language Processing" as such. On the other hand, **AI**-driven analysis of document corpora would still require **NLP** for other aspects of parsing documents; it is unreasonable to expect authors, for instance, to manually notate sentence parse-graphs. This then suggests the question of where the boundary lies between discursive structures which might reasonably be left to authors or editors to manually notate (e.g. sentence boundaries) and those which in practice could only be obtained via **NLP** (such as part-of-speech tags). Related to this question is how best to model **NLP** structures, such as the trees or graphs representing the syntax of natural-language sentences. We will consider this question in subsequent chapters in the context of Conceptual Space Theory.

References

1 Khalid Belhajjame, *et al.*, "Workflow-centric research objects: First class citizens in scholarly discourse". https://pages.semanticscholar.org/coronavirus-research

2 "COVID-19 Open Research Dataset (CORD-19)". 2020. Version 2020-03-13. Retrieved from https://pages.semanticscholar.org/coronavirus-research. Accessed 2020-03-20. doi:10.5281/zenodo.3715506 https://pages.semanticscholar.org/coronavirus-research

3 Bethany Dearlove, *et al.*, "A SARS-CoV-2 vaccine candidate would likely match all currently circulating variants View ORCID Profile". *Proceedings of the National Academy of Sciences*, Volume 117 (2020), pages 23652-23662. https://www.pnas.org/content/117/38/23652

4 Ludëk Eyer, *et al.*, "Nucleoside analogs as a rich source of antiviral agents active against arthropod-borne flaviviruses". https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5890575

5 Johanna T Fifi and J Mocco, "COVID-19 related stroke in young individuals". *The Lancet*, Volume 19 (2020). https://www.thelancet.com/journals/laneur/article/PIIS1474-4422(20)30272-6/fulltext

6 Andreas Jönsson, "AngelCode Scripting Library", www.AngelCode.com/AngelScript/

7 Ross W Paterson, *et al.*, "The Emerging Spectrum of COVID-19 Neurology: Clinical, radiological and laboratory findings", *Brain*, Volume 143, Number 10 (2020), pages 3104-3120. https://academic.oup.com/brain/advance-article/doi/10.1093/brain/awaa240/5868408

8 Mark A. Parsons and Ruth Duerr, "Data Identifiers, Versioning, and Micro-citation", https://www.thelancet.com/action/showPdf?pii=S1473-3099%2820%2930086-4

9 Samir Abu-Rumeileh, *et al.*, "Guillain-Barré Syndrome Spectrum Associated with COVID-19: An up-to-date systematic review of 73 cases", *Journal of Neurology*, Volume 268, Number 4 (2020), pages 1133-1170. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7445716

10 Enar Reilent, "Whiteboard Architecture for the Multi-agent Sensor Systems", https://www.thelancet.com/action/showPdf?pii=S1473-3099%2820%2930086-4

11 Heshui Shi, *et al.*, "Radiological findings from 81 patients with COVID-19 pneumonia in Wuhan, China: a descriptive study". https://www.thelancet.com/action/showPdf?pii=S1473-3099%2820%2930086-4

12 Gentle Sunder Shrestha, *et al.*, "Precision Medicine for COVID-19: A call for better clinical trials". *Critical Care*, Volume 24 (2020). https://ccforum.biomedcentral.com/articles/10.1186/s13054-020-03002-5

13 Alina Trifan and José Luís Oliveira, "FAIRness in Biomedical Data Discovery", *12th International Conference on Health Informatics*, Proceedings (2019), pages 159-166. https://www.scitepress.org/Papers/2019/75764/75764.pdf

14 Johan Van Soest, *et al.*, "Towards a semantic PACS: Using Semantic Web Technology to Represent Imaging Data". *Studies in Health Technology and Informatics*, Volume 205 (2014), pages 166-179. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5119276

15 Lucy Lu Wang, *et al.*, "CORD-19: The Covid-19 Open Research Dataset". *ArXiv*, 2020. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7251955

16 Hetong Zhou, *et al.*, "The Landscape of Cognitive Function in Recovered COVID-19 Patients". *Journal of Psychiatric Research*, Volume 128 (2020), pages 98-102. https://www.sciencedirect.com/science/article/abs/pii/S0022395620308542