EXECUTIVE SUMMARY

Developing a Data Mining Repository to Accelerate Covid-19 Research for the Scientific Community and Public Policy Makers

LTS is founded by Amy Neustein, PhD, Series Editor of **Speech Technology and Text Mining in Medicine and Health Care** *(de Gruyter); Editor of* **Advances in Ubiquitous Computing: Cyber-Physical Systems, Smart Cities, and Ecological Monitoring** *(Elsevier, 2020); and co-author (with Nathaniel Christen) of* **Cross-Disciplinary Data Integration Models for the Emerging Covid-19 Data Ecosystem** *(Elsevier, forthcoming).*

Introduction

The LTS Cross-Disciplinary Repository for Covid-19 Research (**CR2**) is a collection of open-access research data sets related to SARS-COV-2 and Covid-19, which is being developed as a supplement to the forthcoming Elsevier volume examining Covid-19 research from the perspective of text and data mining technologies that cut across disiplines. The benefit of **CR2** is that it can accelerate Covid-19 research by (1) pooling a diverse collection of data sets into a single resource which scientists can utilize; (2) serving as the prototype for larger research portals that can aggregate new Covid-19 data that will emerge from hospitals, labs, and academic institutions in the future; (3) formalizing a framework for aggregating patient narratives to accurately capture first-hand subjective symptomatology of the patient suffering from Covid-19; and (4) accelerating the implementation of novel data-integration and software-development technologies which can contribute to scientific progress vis-à-vis Covid-19 in particular, and biomedical/scientific computing methodology in general. The software used to curate **CR2** data has diverse applications for software and database engineering, and provides solutions to technical problems with a broad reach in the private sector. Further documentation of the **CR2** technology and products may be found on the development repository for the aggregation of **CR2** data (Mosaic-DigammaDB/CRCR).

Background

The sudden emergence of Covid-19 as a global crisis has cast a spotlight on computational and technological challenges which, in the absence of a catastrophic pandemic, would rarely rise to public attention. In particular, an effective response to the dangers of SARS-COV-2 requires coordinated policy making integrating diverse modes of scientific inquiry. Genomic, biomolecular, epidemiological, socio-demographic, clinical, and radiological information are all pertinent to Covid-19. In this environment, it is important that the empirical foundations for expert recommendations — which in turn drive public policies of enormous social and economic consequence — be transparently documented and critically examined. The proper synergy between government and science depends on data centralization: given the gaps in our current Covid-19 knowledge, it is understandable that different jurisdictions will craft responses to the pandemic in different ways. There is no central authority with sufficient epistemic force to legitimize homogeneous mandates across the entire country. However, such policy differences should be a consequence of alternative interpretations of scientific knowledge or the diverse needs of local communities — rather than being a haphazard consequence of governments working with divergent, competing, and poorly integrated data.

The current administration, along with numerous corporate and academic entities, has clearly recognized the need for a more centralized paradigm for sharing Covid-19 data. For example, the White House spearheaded a scientific initiative to develop **CORD-19**, an open-access corpus of over 46,000 peer-reviewed publications related to Covid-19, which were transformed into a common machine-readable representation so as to promote text and data mining. Similarly, large institutions such as Google, Johns Hopkins, and Springer Nature have all implemented some form of coronavirus data-sharing platform targeted to both scientists and policy makers. However, these two aspects of the corporate/academic contributions to Covid-19 data sharing (exemplified by the **CORD-19** White House initiative and by institution-generated portals, respectively) have been incomplete, for opposite but complementary reasons. Specifically, **CORD-19** is highly structured and tightly integrated, but it focuses primarily on text mining and scientific documents, not *research* data. While it is possible to find data sets about Covid-19 through **CORD-19**, the techniques to do so are both cumbersome and non-scalable. On the other hand, projects such as the Johns Hopkins coronavirus "dashboard" provide accessible data sets, yet these projects are isolated and do not offer the level of structure and integration evinced by **CORD-19**. In short, an optimal Covid-19 research platform would merge the structural text-mining rigor of **CORD-19** with the data-centric focus of isolated projects that share Covid-19 data with the scientific community, policy makers, and the general public.

Description

The design of **CR2** derives from the principles outlined in the previous paragraph. In particular, an ideal data-sharing ecosystem should merge data from multiple sources, but should do so in a fashion which yields a machine-readable totality, analogous to **CORD-19**'s structuration with respect to text mining. The merit of **CR2** therefore lies not only in the data which it will encompass but also in novel technology that it will concretize for constructing data repositories adhering to these principles. Accordingly, **CR2** can provide value at different scales of realization. Relatively small data sets serve several scientific and computational purposes: (1) they can provide researchers with a mental picture of how data in different disciplines, projects, and experiments is structured; (2) they can serve as a prototype and testing kernel for technologies implemented to manipulate data in relevant formats and encodings; and (3) they can lay the foundation for data-integration strategies. For example, when designing a representation format and/or implementing code to merge different data formats into a single structure (or meta-structure), it is useful to work with small, representative examples of the data structures involved, so as not to complicate the integration logic with computational details solely oriented to scaling up the data-management logistics. As a result, **CR2** can provide a testbed for implementing data-integration technologies which can scale up as needed. To fulfill this mission, **CR2** can aggregate relatively small data sets which have previously been published on academic and research portals, such as Springer Nature, Dryad, and DataVerse. At the same time, a more substantial (and not necessarily fully open-access) Covid-19 data-set collection would also be beneficial to the scientific and policy-making community. Ideally, then, **CR2** will be paired with a larger technology which shares a similar implementational strategy but with different accession paradigms, allowing for an open-ended collection of Covid-19 data which users may selectively access (instead of a single package that users may acquire as an integrated resource). The common denominator in both cases (whether the focus is on relatively smaller or larger data sets) is the importance of deploying novel and contemporary data-integration techniques to centralize Covid-19 research as much as possible. Accordingly, this summary will briefly explain how **CR2** can accelerate Covid-19 data integration on both a practical and technological level.

Methodology for Covid-19 Data Integration

As indicated above, pertinent Covid-19 data is drawn from multiple scientific disciplines. On a technological level, Covid-19 data is documented via a wide array of file types and data formats. This diversity presents technological challenges: if a Covid-19 information space encompasses files representing 25 different incompatible formats, users would need 25 different technologies to fully benefit from this data. In many cases, however, data incompatibilities are merely superficial — an important subset of Covid-19 data, for example, has a common tabular meta-model, even if the data is realized in discordant technologies (spreadsheets, relational databases, comma-separated-value or Numeric Python files, and so forth). Applying **CR2**'s technology, one level of data integration can thus be achieved simply by encoding tabular structure into a common representation: any field in a table can be accessed via a record number and a column name and/or index. In some cases, more rigorous integration is also possible — for example, by identifying situations where columns in one table correspond semantically or conceptually to those in another table. In either case, it is reasonable to assume that a single abstract data format lies behind surface data-expression in patterns such as spreadsheets and comma-separated values (**CSV**), so that all files in an archive encoding spreadsheet-like data can be migrated to a common model.

Other forms of clinical and epidemiological inputs are often more amenable to graph-like representations. For instance, trajectories of viral transmission through person-to-person contact is obviously an instance of social network analysis. Similarly, models of clinical treatments and outcomes can take graph-like form insofar as there are causal or institutional relations between discrete medical events: a certain clinical observation *causes* a care team to request a laboratory analysis, which *yields* results that *factor* into the team's decision to *administer* some treatment (e.g., a drug *from* a particular provider *with* a specific chemical structure), which observationally *results* in the patient improving and eventually *being* discharged. In short, patient-care information often takes the form — at least conceptually — of a network comprised of different "events," each event involving some observation, action, intervention, or decision made by care providers, and where the important data lies in how the events are interconnected: both their logical relationships (e.g., cause/effect) and their temporal dynamics (how long before a drug leads to a patient's improvement; how much time elapses between admission to a hospital and discharge). These graph-like representations are a natural formalization of "patient-centered" data models.

Using **CR2** associated software, a higher level of data integration can then be achieved by merging tabular and graph-like models into a single *hypergraph* format. A significant subset of Covid-19 data (or, more generally, any clinical/biomedical information) conforms to either tabular or graph structures; thus it is feasible to unify all of this information into a common framework. A graph-plus-table architecture is generally considered some form of Hypergraph model, and indeed **CR2** adopts a hypergraph paradigm to merge many different sorts of information into a common structure. In particular, **CR2** introduces a new "Hypergraph Exchange Format" (**HGXF**) which can provide a text encoding of many files that, when originally published,

embodied a diverse array of file-types requiring a corresponding array of different technologies. **CR2** will include specialized computer code that would enable machine-readability of the **HGXF** files, and use them to create hypergraph-database instances. In short, **CR2** will promote Covid-19 data integration by translating a wide range of files into a common **HGXF** format, something that has not yet been done before.[1]

Hypergraph Data Models and Multi-Application Networks

As has been outlined thus far, via the **CR2** technology most Covid-19 data can be wholly or partially integrated into a single hypergraph framework, which accordingly simplifies the process of designing software applications and algorithms to analyze and manipulate this data. Specifically, software components can employ a single code library to obtain, read, consume, and store data, rather than needing to re-implement this logic for a large number of different file formats and/or database models.

Quality software (especially in the clinical and biomedical context) demands a balance between applications which are either too broad or too narrow in scope. On the one hand, doctors often complain that homogeneous Electronic Health Record systems (where every digital record or observation is managed by a single all-encompassing application) are unwieldy and hard to work with. This is understandable, because the clinical tasks of health care workers with different specializations can be very different. On the other hand, doctors also complain about software and information systems which are so balkanized that they must repeatedly switch between different, non-interoperable applications. In short, clinical, diagnostic, and research software should be neither too homogeneous nor too isolated; finding the proper balance between these extremes is, no doubt, a major challenge to the usability of electronic health systems going forward.

Against this background **CR2** demonstrates novel solutions to this problem: it focuses on the dimensions of data acquisition and management that are specific to individual scientific or medical specializations, while also identifying requirements that are consistent across domains. Scientific software generally needs to hone in on the data visualization and analytic requirements of particular disciplines; for example, biochemists use different programs than astrophysicists. However, much of the code underlying scientific applications has nothing to do with these high-level models or theories, but is simply a fulfillment of basic data-management functionality — data storage, accession, provenance, searching, user validation, and so forth. In effect, the computational requirements of scientific and biomedical software can be partitioned into two classes: (1) domain-specific logic which reflects the quantitative or theoretical models of narrow scientific fields; and (2) data-management logistics which can be realized within a central access hub, rather than being re-implemented by each application in isolation.

In short, the architecture enabled by **CR2** conceives of a central hub which is responsible for storing data and for serving as a common access point — providing the "gateway" where authorized users can gain access to heterogeneous information spaces utilized by an array of domain-specific software applications. Since peer applications would not be directly responsible for data persistence or user identity management, they can focus on their specific data analysis and visualization capabilities. The central hub, serving multiple peer applications, is then a heterogeneous data space managing information from multiple applications while also tracking information about the applications themselves: helping users to identify and launch the software which is most directly relevant to their clinical or research needs at the moment. Meanwhile, because peer applications are jointly connected to a central hub, it is possible to implement scientific workflows where one application may send and receive data from its peers, allowing applications to complement each others' capabilities.

This multi-application networking architecture has precedents in some of the current database and engineering technologies. For example, many hospitals and medical institutions employ some version of a "Data Lake," pooling disparate data sources into a heterogeneous aggregate which is then accessed by multiple client applications. Similarly, Machine Learning and Artificial Intelligence often adopts "software agents" or analytic modules in contexts such as Online Analytic Processing, which again represent semi-autonomous software components sharing an originary data hub. Web applications, too, often act as domain-specific subsidiaries deferring operational requirements, such as user authentication or transaction processing, to a central web service. The limitation of multi-application networks in these existing contexts are that the software agents involved are generally "lightweight," with relatively primitive user-interface design. By contrast, the hypergraph technology introduced with **CR2** will support multi-application networking in the context of more substantial desktop-style scientific applications. In sum, the novel hypergraph technology developed by LTS offers a hybrid of the development methodologies employed for desktop scientific software and those applicable to multi-agent heterogeneous data stores, like a Semantic Data Lake. To accomplish these goals, **CR2** will utilize a new hypergraph database engine, coded in the **C++** programming language,

---

[1] Not every format relevant to Covid-19 can be realistically translated to **HGXF**. In particular, scienctific fields requiring substantial quantitative analysis — e.g., biomechanics or genomics — express data via encodings optimized for relevant mathematical operations. In this scenario, **CR2** will not attempt to migrate *all* of a data file to **HGXF**. However, even for these files **CR2** will generally provide a supplemental **HGXF** encoding supplying data *about* the original file, with information about the file type, preferred software components for viewing/manipulating its data, and so forth. In this manner the contents of non-**HGXF** files can be indirectly included into the **CR2** hypergraph-based ecosystem.

which has a unique focus on supporting native **GUI** applications from the ground up, including persisting application state and storing application documentation within the database itself.

Data Microcitations

An additional feature of the **CR2** database engine, which will make it particular useful for curating research data, is support for annotating individual components of a database — a technology sometimes referred to as "micro-citation". Data microcitations are a references to integral parts of a data set; for instance, an individual table, or a single record or column in a table. Microcitations allow individual elements within a data set to be cited by and linked to publications. Microcitations are also a tool for data mining, because a data set can be analyzed by examining each annotated element within the data set.

In some contexts, microcitations can link data set elements to scientific concepts and controled vocabularies, which can clarify the scientific meaning attributed to the corresponding element. For example, microcitations allow table columns to be mapped to statistical parameters, enabling their empirical properties (such as min/max values and distribution) to be queried. Similarly, dimensional and measurement annotations describe the empirical and experimental significance of the measured or calculated quantities which are stored in a database. Quantity dimensions model the conceptual roles which particular parameters perform: the axiation "mJ/cm^2" (millijoule per square centimeter), for example, indicates the intensity of Ultraviolet light, which means that any table (or any other data aggregate) having a column or field with this dimension is intrinsically associated with observations or experiments pertaining to Ultraviolet light. Consequently, to locate data sets relevant for research about the antiviral Ultraviolet radiation, one method is to search for data fields dimensionalized in terms of joule or millijoule per square centimeter. As this example illustrates, data microcitaition — via annotations on data fields, statistical parameters, and table columns — is an important data-mining tool. Database-level support for microcitations therefore serves two distinct benefits: (1) to aid data mining, and (2) to enable granular links to be established between publications and data sets, making it easier for researchers to find the specific information most relevant to their own research.

Adding Patient Narratives to Covid-19 Data

In addition to aggregating published data sets, **CR2** may be used as a repository for collecting new Covid-19 information. With that in mind, we are prioritizing the design of a standard for storing and accessing natural-language text representing patients' subjective symptom descriptions, which is quite useful for diagnostic/prognostic assessments of patients infected by Covid-19.

Just as **CR2** envisions a curation of published data sets for data mining to improve machine-readability of Covid-19 research, LTS also sees the benefit of a repository of patient narratives prepared for text mining, to improve machine readability of the open-ended symptom descriptions offered by patients. While **CR2** does not need to specify how these narratives should be collected, it will implement a common representational format so that patient narratives can be pooled, similar to to how **CORD-19** research texts are merged and encoded with a system that permits annotation.

In modeling patient narratives, this technology will be oriented toward the scientific-computing ecosystem outlined in the previous section. In particular, we assume that **GUI**-based desktop applications will be the primary instruments for data collection and analysis; this means that the encoding of patient narratives may, at times, need to be paired with **GUI** or multi-media content. For example, the software for patients to submit medical history information could also allow them to pair (text-form) narratives with graphics indicating the location of their pain or discomfort. Furthermore, the software could allow narratives to be accompanied by an audio file where patients could cough/speak into a microphone. In light of this range of possible inputs, a patient-narrative encoding must, therefore, be flexible enough to include diverse multi-media content.

As described earlier, an information space adapted for multiple peer applications should encompass capabilities for saving application state (the current visual appearance of the program), which includes features for modeling instances of **GUI** classes. This technology provides the necessary infrastructure for managing patient narratives. For example, consider a multi-media intake form where patients may describe symptoms by placing icons (representing pain or discomfort) against anatomic silhouettes (head/body, back/front, extremities, and so forth). As patients use such a multi-media form, **GUI** application state corresponds to the patient's subjective symptomology; in this way the graphics-based represention of symptoms could then be incorporated into the overall patient narrative. This is an example of how application-persistence logic can be marshaled to the related project of curating patient narratives.

Further documentation of text-encoding methodology applicable to both patient narratives and publications associated with **CR2** research data is available on the **CR2** web site, such as here (this is a downloadable **PDF** link; visit the repository to see the larger archive structure).

For more information, please contact:

Linguistic Technology Systems

amy.neustein@verizon.net • (917) 817-2184