MediSage: An AI Assistant for Healthcare via Composition of Neural-Symbolic Reasoning Operators

Sutanay Choudhury
sutanay.choudhury@pnnl.gov
Pacific Northwest National Lab
Richland, WA, USA

Khushbu Agarwal
khushbu.agarwal@pnnl.gov
Pacific Northwest National Lab
Richland, WA, USA

Colby Ham
colby.ham@pnnl.gov
Pacific Northwest National Lab
Richland, WA, USA

Suzanne Tamang
stamang@stanford.edu
Stanford University
Stanford, CA, USA

ABSTRACT

We introduce MediSage, an AI decision support assistant for medical professionals and caregivers that simplifies the way in which they interact with different modalities of electronic health records (EHRs) through a conversational interface. It provides step-by-step reasoning support to an end-user to summarize patient health, predict patient outcomes and provide comprehensive and personalized healthcare recommendations. MediSage provides these reasoning capabilities by using a knowledge graph that combines general purpose clinical knowledge resources with recent-most information from the EHR data. By combining the structured representation of knowledge with the predictive power of neural models trained over both EHR and knowledge graph data, MediSage brings explainability by construction and represents a stepping stone into the future through further integration with biomedical language models.

ACM Reference Format:
Sutanay Choudhury, Khushbu Agarwal, Colby Ham, and Suzanne Tamang. 2018. MediSage: An AI Assistant for Healthcare via Composition of Neural-Symbolic Reasoning Operators. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX)*. ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Healthcare and artificial intelligence (AI) are two fields whose association goes back more than half a century. Deep learning-based advancements in AI inspired thousands of applied research efforts in healthcare, often demonstrating superior performance over traditional approaches. However, improved predictive performance alone is not enough to demonstrate clinical utility. Trust that rises from the explainability of AI tools to communicate an answer in a way that supports the cognitive processes of a domain expert is essential.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

Conversational AI tools have gained attention in the broader technology industry over the past few years, inspiring new opportunities for breakthroughs in healthcare. Staying at home with caregiver support is increasingly seen as a long-term solution for elderly individuals with health challenges. Also, as the COVID-19 pandemic underscored, in a time of strained capacity, the "hospital at home" movement is figuring out how to create an inpatient level of care anywhere. Therefore, successful solutions to establish AI-mediated natural language conversations with doctors, caregivers, and family is a critical need today.

Large-scale language models, most notably ChatGPT [11], have raised expectations of today's users for what conversational AI systems can deliver. However, ChatGPT-like systems are intended to work with textual data, and not with all the modalities of electronic healthcare record datasets. In this paper, we introduce MediSage, a knowledge graph-driven system that is derived from both EHR data and traditional clinical KBs. We demonstrate how MediSage complements the capabilities provided by language models [10, 13] and can be easily combined with them for providing greater user experience. However, the goal of this demonstration is to highlight the structure-driven explainability that graph-based approaches provide, and present it to the broader community as a basis for addressing the limitations of state-of-the-art large-scale language models for deriving situational-awareness from EHR data.

2 MEDISAGE ARCHITECTURE

2.1 Data Layer

The MediSage architecture has two major data components: a temporal database containing multimodal patient information and a graph that combines a medical knowledge graph with additional clinical information extracted from the patient database (see Figure 1). In the remainder of the paper, we will refer to these two data sources as PatientDB and ClinicalKG respectively.

(1) **PatientDB** is a temporal patient database that stores information as a collection of patient visits, with a patient visit being defined as a sequence of snapshots of the patient over time as indicated in Figure 1. Formally, each patient visit can be defined as a key-value pair with the key being a composite of a patient identifier and a visit timestamp, and the value being a sequence of patient states, referred to as a PatientTrajectory. Formally, we refer to each patient state as a

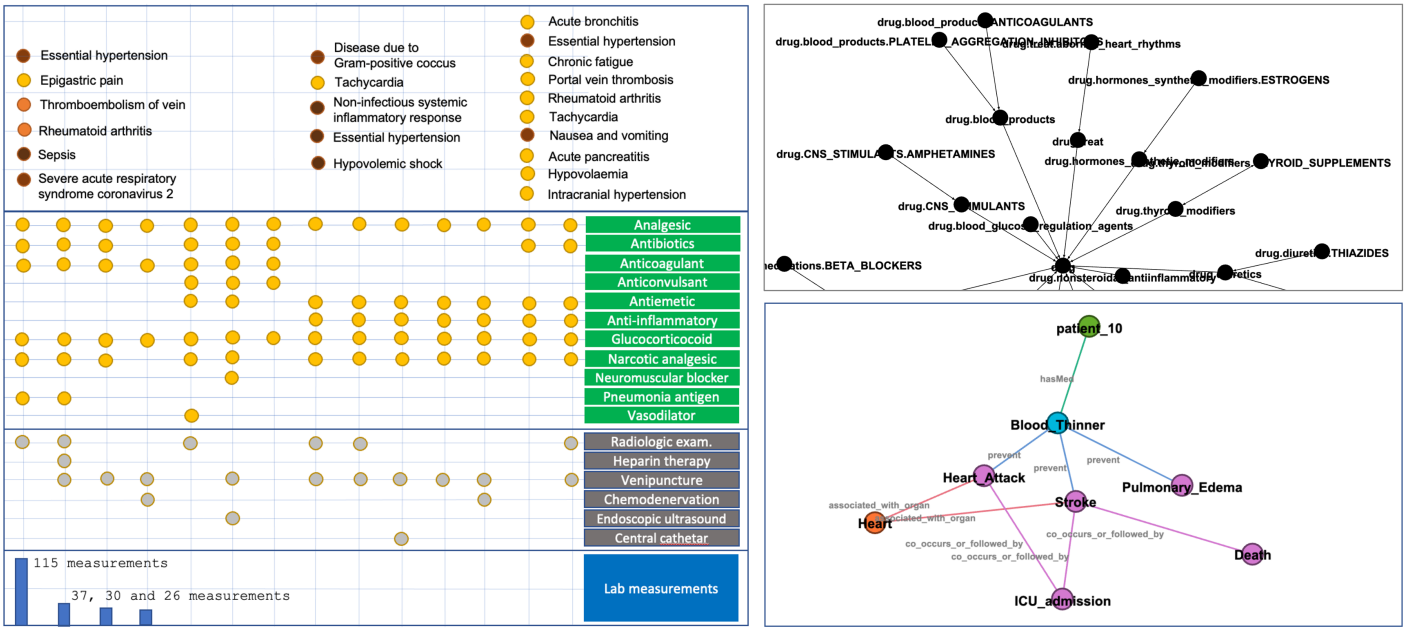

Figure 1: The image on the left shows a patient's EHR information over time. Each blue vertical line represents a timestep and each horizontal box shows information from a specific data type (medical conditions, drugs, etc.). The top right picture shows an ontology subgraph (extracted only for drugs to aid with readability). The bottom right subgraph is an illustration of how we connect information from an EHR database (that contains data from left) with a clinical knowledge graph that contains ontological hierarchy information (top right) as well as multi-relational edges such as ones between drugs and medical conditions (such as "blood thinners" prevent X), and edge weights that are learned from the patient database.

PatientContext object, which is a collection of labeled sets that represents patient's medical conditions, medications, procedures, numeric measurements at any point of time.

(2) **ClinicalKG** stores two types of edges (or triples). The first set of triples represent ontological properties for each (typed) entity present in the PatientDB. Given an entity such as a medical condition or drug, their ontological information captures the membership to various categories. It is important to note that an entity can belong to multiple categories. Therefore, the ontological structure of the knowledge graph often reflects a lattice structure as opposed to a perfect tree hierarchy (as shown in Figure 1). The second set of triples represents lateral relationships between the typed entities such as medical conditions, drugs, etc. and is learnt from both PatientDB and traditional clinical KBs. The resultant ClinicalKG is represented as a multi-graph with directed edges to allow presence of multiple directed edges between the same pair of nodes.

We provide support to ingest multiple clinical knowledgebases (KBs) such as OMOP [6], a subset of SNOMED-CT [2], and PrimeKG [3]. Each of these KBs offers a tradeoff in terms of scale, richness/information coverage, and noise. We use the OMOP-KB for this demonstration, primarily for logistical and data availability reasons. Data integration across PatientDB and ClinicalKG is a critical step for applying the knowledge contained in ClinicalKG to reason about latest updates in PatientDB. Our original implementation is based on mapping of EHR vocabularies, in which EHR data represented in OMOP Common Data Model is mapped to SNOMED/UMLS. However, in cases when a direct mapping is not available, we use an entity-linking approach with the BioLinkBERT language model [15].

2.2 Building Blocks: Reasoning Operators

PatientDB and ClinicalKG are supported by following suites of methods and models that serve as building blocks for composing methods for clinical decision support and reasoning tasks.

(1) Methods to discover comorbidity patterns via frequent pattern mining [4] from PatientDB, and queries to retrieve specific entities, associated patterns, and/or their aggregate counts distributions.
(2) Machine-learning models to predict patient outcomes from PatientDB using transformer architectures [1] and explanation of patient prediction via SHAP scores [9].
(3) Learning an Odds-Ratio Network Model [7] from PatientDB and storing causal odds-ratios as edge weights in ClinicalKG.
(4) Graph neural network models for link prediction [14] and multi-hop reasoning [12] over ClinicalKG, and methods for vector-space based k-Nearest Neighbor search [8] and reachability queries.

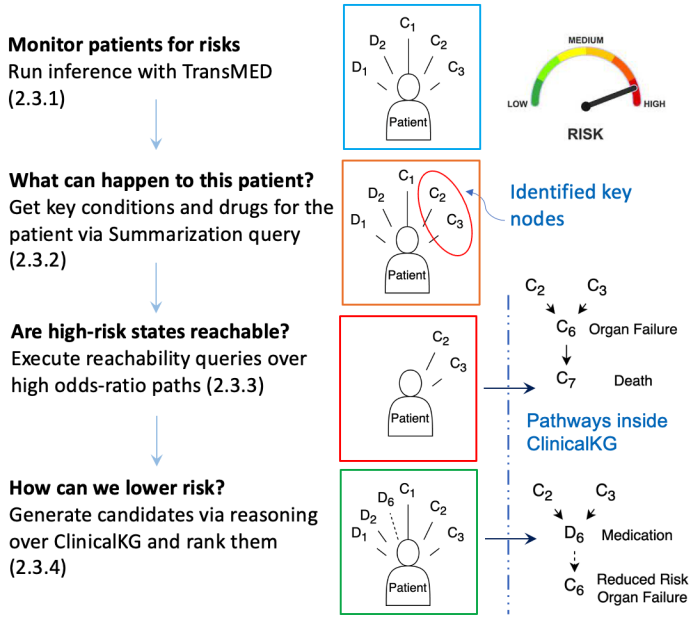

Figure 2: Diagram showing the mapping of various reasoning steps. We only use medical conditions and drugs (referred to as C_i and D_i respectively) for illustration purposes.

2.3 Decision Support and Explanation through Composition of Reasoning Operators

End-users for EHR systems express their needs in the form of abstract reasoning tasks than operations on knowledge graphs. Given a patient whose medical conditions changed recently, they want to reason about the severity of the patient's current state, the evolution of the patient's state into the future, and learn about potential interventions. This section describes how we translate human-level reasoning operations via neural-symbolic reasoning operators as described earlier.

2.3.1 Risk monitoring via Prediction Models. The goal of the risk monitoring task is to periodically execute a set of prediction models for each patient. Each prediction model is designed to accept the PatientContext objects as retrieved by a query window, and predict an outcome (such as admission to ICU, dependence on supplemental oxygen, or day to discharge) as specified by a look-ahead day offset. We provide the TransMED model [1] that integrates multimodal information and is suited for complex prediction tasks that require modeling of fine-grained temporal variations. Let $P_t = (C_t, M_t)$ be the patient state at time t, where $C_t \subseteq C$ is a set of observed codes and $M_t \in \mathbb{R}^{|M|}$ is a vector of lab values. Each element in the C_t is mapped to a node in the ClinicalKG, so we referred to it as node set. Given demographics d, risk factors r, and a sequence of T_h historical states $P_{t-T_h+1}, P_{t-T_h+2}, ..., P_t$, TransMED predicts a clinical outcome of interest T_f steps forward into the future.

2.3.2 Summarization. The goal of the summarization task is to return the most salient information for a patient's current state. Given a patient $P_t = (C_t, M_t)$ defined above, our goal is to filter the node set C_t and return top-K most significant medical conditions, drugs, and procedures. We implement this via a map-reduce operation in which each element of the node set C_t is mapped to a category. The reduce function sorts all nodes that belong to the same group via a ranking function and emits the top-ranked entry per group. Finally, the top-K ranking entries across all groups are returned. We provide implementation for two functions for node-level categorical mapping, and two functions for ranking nodes. The first mapping function is ontology-based, and it returns all categories a node is associated with. The second mapping function maps each node to the ids of frequent patterns or clusters it is associated with. For ranking functions, we support measures that cover commonness of a condition (computed by degree of the node in the graph), and its importance through co-occurrence with others (as computed by the PageRank centrality).

2.3.3 Explanation of future outcomes via Graph. Some clinical KBs provide information about disease co-occurrences, or association between a disease and a drug. However, most KBs suffer from information incompleteness, which is addressed by link prediction/KB completion methods. However, as the recent pandemic underscored, it is valuable to detect novel comorbidity or treatment patterns as soon as they manifest in the data. Such trends are observable in the EHR data, but clinical KBs such as OMOP or SNOMED may or may not reflect such changes soon. We propose a simple solution to bridge this gap by constructing an odds-ratio network [7] from PatientDB. Every node in the constructed odds-ratio network represents a medical concept such as a medical condition, and maps to a node in ClinicalKG. The edge weight in the odds-ratio network is computed to be the causal odds ratio between these nodes, and finally stored in ClinicalKG as edge weights.

Given the current state of a patient, we formulate the problem of finding potential transition to a high-risk state via reachability queries over the odds-ratio network. We designate nodes that represent organ failures as the target nodes for such queries. Given the node set C_t, we perform a search from each node in C_t to a set of target nodes specified by a maximum hop count L and minimum odds ratio w_{min} on each traversed edge. We use $L = 3$ and $w_{min} = 3$ for this demonstration.

2.3.4 Recommendation of interventions. Our recommendation algorithm uses a two-step approach. The first step performs candidate generation and the second step focuses on ranking. To generate treatment candidates, we retrieve node ids from C_t (set of observed codes for a patient at time t) that correspond to medical conditions, but only select ones whose treatment options are not present in C_t. This filtering is done via checking for known edge patterns between a medical condition and it's associated treatment. Candidate generation follows a similar approach, in which we query the graph based on the single-edge treatment pattern. If no known candidate is found by querying the ClinicalKG, we use a link prediction approach to infer them. We train a model that given a "head" node and an edge type, returns a set of candidate "tail" nodes. Given a ClinicalKG, we train link-prediction models using TransE, CompGCN and Query2Box methods and select the one with best performance.

We use a perturbation-based approach for ranking the interventions. Drawing inspiration from the work by [5], given a current

node set C_t and a recommended drug u, we denote the new patient context as $C_t^{'} = C_t \cup u$. Next, we compute the risk scores by evaluating both C_t and $C_t^{'}$ through the TransMED prediction model and compute the risk reduction $\delta_r(u) = TransMED(C_t) - TransMED(C_t^{'})$. Given all candidates R_c, we select the one providing maximal risk reduction: $\arg\max_{u \in R_c}(\delta_r(u))$.

2.4 Conversational Interface

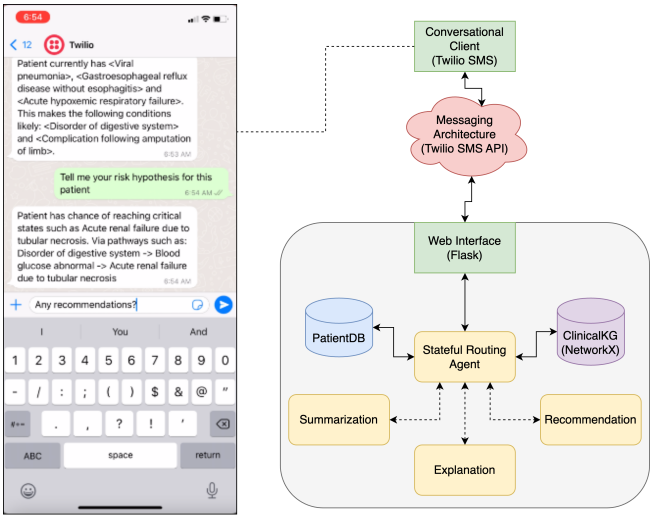

Figure 3: Illustration of MediSage user-interface and its communication with the server architecture.

We provide a conversational interface via WhatsApp that communicates with the MediSage server by a cloud-based messaging interface (Twilio). The messages are sent through a Flask-based web interface to the stateful AI agent who infers the current user intent and routes necessary calls to the backend's appropriate reasoning operator(s).

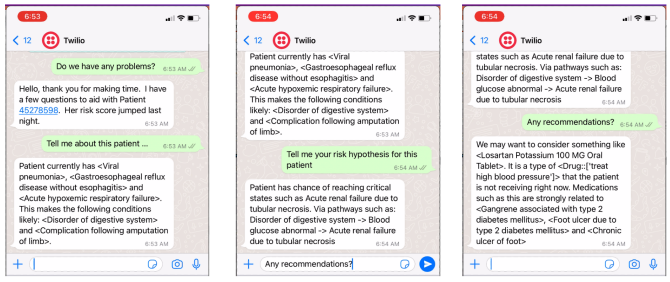

Figure 4: Illustration of how a user will interact with MediSage during the demonstration via a conversational interface.

3 DEMONSTRATION PLAN

Our demonstration will allow users to interact with the AI assistant and execute various reasoning queries via a WhatsApp-based conversational interface shown in (Figure 4). Our demo will be restricted to a single-user mode as we do not support multiusers or concurrent queries at this point.

The demonstration will be based on de-identified EHR data of all patients treated at Stanford Hospital between January 1, 2015, and March 19, 2021. This dataset was provided via Stanford Research Repository [6] and was used under approval by Stanford University Institutional Review Board (IRB) protocol: 50033 (Machine Learning of Electronic Medical Records for Precision Medicine). Patient-informed consent was waived by the IRB for this protocol. Full details on construction of the dataset are available at [1].

4 CONCLUSION

We present MediSage, a system that provides a conversational interface to perform step-by-step reasoning about a patient's conditions, her evolution into future and potential interventions for treatment. MediSage provides such capabilities through novel integration of a clinical knowledge graph with the EHR database, which allows it to combine individual patient-level information with recent patterns emerging in the entire database, all the while leveraging on the knowledge provided by traditional clinical KBs. Last but not the least, its composition of human-level reasoning operations via neural-symbolic operators grounds its responses into familiarity, and provides recommendations that are explainable-by-design.

REFERENCES

[1] Khushbu Agarwal, Sutanay Choudhury, Sindhu Tipirneni, Pritam Mukherjee, Colby Ham, Suzanne Tamang, Matthew Baker, Siyi Tang, Veysel Kocaman, Olivier Gevaert, et al. 2022. Preparing for the next pandemic via transfer learning from existing diseases with hierarchical multi-modal BERT: a study on COVID-19 outcome prediction. *Scientific Reports* 12, 1 (2022), 10748.
[2] Khushbu Agarwal, Tome Eftimov, Raghavendra Addanki, Sutanay Choudhury, Suzanne Tamang, and Robert Rallo. 2019. Snomed2Vec: Random Walk and Poincar\'e Embeddings of a Clinical Knowledge Base for Healthcare Analytics. *arXiv preprint arXiv:1907.08650* (2019).
[3] Payal Chandak, Kexin Huang, and Marinka Zitnik. 2022. Building a knowledge graph to enable precision medicine. *BioRxiv* (2022), 2022–05.
[4] Sutanay Choudhury, Sumit Purohit, Peng Lin, Yinghui Wu, Lawrence Holder, and Khushbu Agarwal. 2018. Percolator: Scalable pattern discovery in dynamic graphs. In *WSDM*.
[5] Ian C Covert, Scott Lundberg, and Su-In Lee. 2021. Explaining by removing: A unified framework for model explanation. *The Journal of Machine Learning Research* 22, 1 (2021), 9477–9566.
[6] Somalee Datta et al. 2020. A new paradigm for accelerating clinical data science at Stanford Medicine. (2020). https://arxiv.org/abs/2003.10534
[7] Jiří Gallo, Eva Kriegová, M Radvansky, Matúš Sloviak, and M Kudelka. 2022. Odds-ratio network for postoperative factors revealing differences in the 2-year longitudinal pattern of satisfaction between women and men after total knee arthroplasty. *Scientific Reports* 12, 1 (2022), 17470.
[8] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data* 7, 3 (2019), 535–547.
[9] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems* 30 (2017).
[10] Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics* 23, 6 (2022).
[11] Long et al. Ouyang. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155* (2022).
[12] Hongyu Ren, Weihua Hu, and Jure Leskovec. 2020. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. *arXiv preprint arXiv:2002.05969* (2020).
[13] Karan et al. Singhal. 2022. Large Language Models Encode Clinical Knowledge. *arXiv preprint arXiv:2212.13138* (2022).
[14] Ping Wang, Khushbu Agarwal, Colby Ham, Sutanay Choudhury, and Chandan K Reddy. 2021. Self-supervised learning of contextual embeddings for link prediction in heterogeneous networks. In *Proceedings of the Web Conference 2021*.
[15] Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. Linkbert: Pretraining language models with document links. *arXiv preprint arXiv:2203.15827* (2022).