CS 6120: Natural Language Processing

Nurse Practitioner Copilot for Remote Patient Monitoring (RPM)

Presented by Team 102:

Abhinav Dholi
Feng Hua Tan

Guided by:

Prof. Amir Tahmasebi

Table of Content

Problem Statement & Motivation

What is RPM?

- Remote Patient Monitoring (RPM) uses devices like smartwatches and wearable heart monitors to track patients’ health at home
- Sends real-time data to nurses for continuous monitoring

Problem Statement

- Too much real time data -> RPM devices stream nonstop health data
- Heavy workload -> nurses spend a lot time writing & reviewing notes which means less time for patient care
- Patient data is spread across multiple platforms (ie. device portals) -> hard to quickly get overall picture
- Nurses have to read through long clinical notes to find key info which is time-consuming

Motivation

- Nurses need faster, smarter tools
- Patient benefits from quicker, informed care
- Our project fills that gap

Contribution

Abhinav Dholi:

- Implemented SQL QA Agent with security validation
- Designed and trained BiLSTM neural network for NER
- Implemented clinical note summarization with chunk processing
- Developed Flask web application backend with RESTful API
- Developed data preprocessing workflow for NER training
- Architected MySQL database with 11 interconnected tables
- Engineered regex-based text extraction pipeline for clinical notes

Feng Hua Tan:

- Tested ClinicalBERT for risk scoring; deprioritized due to noisy, non-specific entities (ie. “issue” as “other_event”, “sprain” as “condition” but without location like ankle)
- Explored keyword-based classification; ruled out due to ambiguous risk thresholds
- Chose MEWS (Modified Early Warning Score) for patient risk triaging
- Applied BiLSTM-based NER model to tag key medical entities in clinical notes
- Created triage dashboard and visualization of NER in clinical notes in Flask web application

Datasets

Synthea Clinical Notes

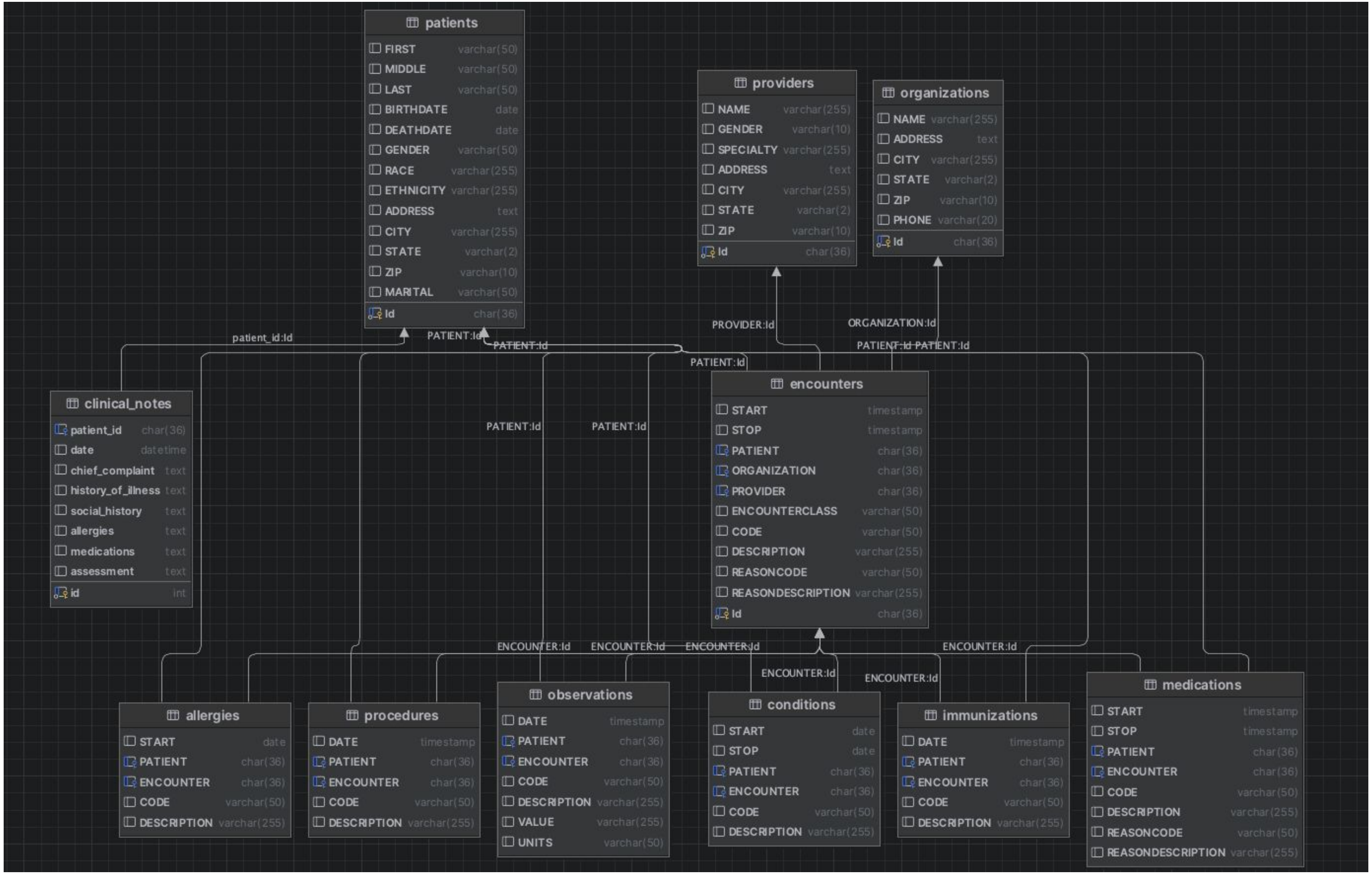
- Synthetic clinical notes for 65 patients across lifetime (total 4789 notes)
- Average 450 words per note
- Used to populate the RPM database

Database Schema (Synthea Generated)

- 11 interconnected tables: patients, providers, encounters, conditions, medications, observations, procedures, immunizations, allergies, clinical_notes, organizations
- Diagram shown on next slide

MACCROBAT 2018 Dataset (Used for NER training)

- 200 clinical documents with annotations
- 17,010 clinical entities across 78 classes
  - ibuprofen -> medication
  - chest pain -> symptoms
- 3,000-5,000 characters per document average
- Data split: 70% train, 10% validation, 20% test

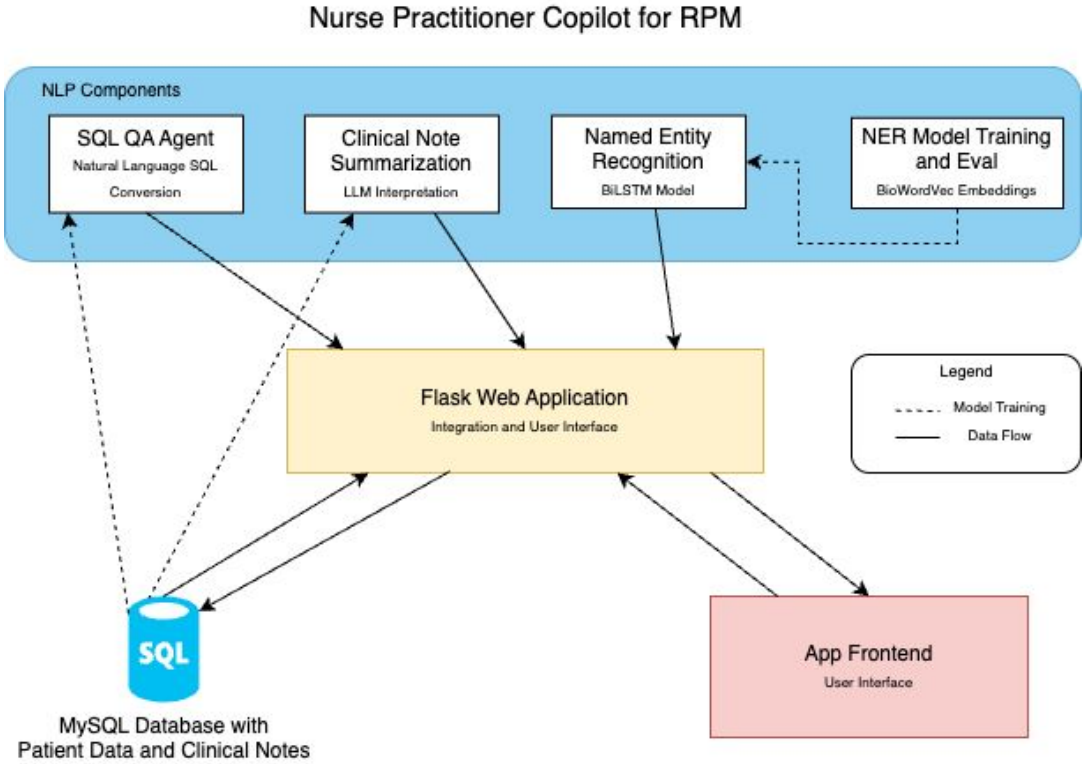


Database Schema

System Architecture

<u>Key Components</u>

- SQL QA Agent for natural language database queries
- Clinical Note Summarization for documentation efficiency
- Named Entity Recognition (NER) for extracting clinical entities
- NER Model Training and Evaluation Pipeline
- Patient Risk Assessment using Modified Early Warning Score (MEWS)
- Flask Web Application for unified interface



Demo

SQL QA Agent

Functionality: Translates natural language questions into patient-specific SQL queries

Implementation:

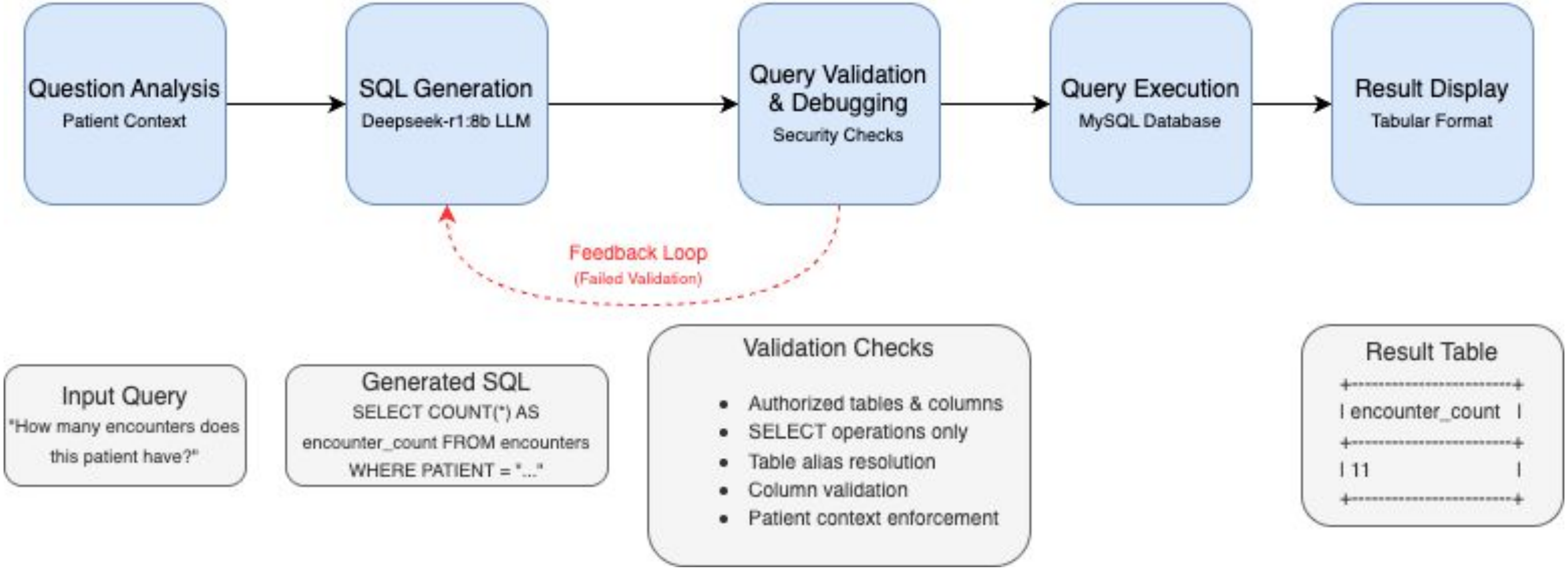
- Built with LangChain and Ollama frameworks using DeepSeek-r1:8b LLM
- 4-step process:
    1. Question Analysis & Context Preparation
    2. SQL Generation
    3. Query Validation & Debugging
    4. Query Execution & Result Display

Security Features:

- Custom SQL validator with AST-based parsing
- Ensures only authorized tables/columns are accessed
- Verifies only SELECT operations (no data modifications)
- Enforces patient context to maintain privacy

Performance:

- 90% accuracy on nurse-generated queries
- 100% recall for patient-specific information
- 90% of initially failed queries successfully corrected through feedback



SQL QA Agent Pipeline

Note Summarization

Purpose: Efficiently process lengthy clinical notes while preserving critical information

Approach:

- Chunking methodology to handle documents exceeding LLM context windows
- Multi-stage processing:
    1. Text chunking (3,000 character segments)
    2. Partial summarization of each chunk
    3. Combine summaries
    4. Temporal analysis (for yearly summaries)

Implementations:

- Single-note summarization
- Yearly summarization across multiple notes
- Preserves clinical context across sections (complaints, history, medications, etc.)
- See example on the right

Original clinical note

DATE: 2024-02-16 00:00:00
Chief Complaint: No complaints.
History of Illness: Isaiah615 is a 5 year-old nonhispanic white male. Patient has a history of sprain (morphologic abnormality), medication review due (situation), primary dental caries (disorder), acute viral pharyngitis (disorder), sprain of ankle (disorder), viral sinusitis (disorder).
Social History: Patient has never smoked. Patient comes from a middle socioeconomic background. Patient currently has UnitedHealthcare.
Allergies:
Medications: amoxicillin 250 mg / clavulanate 125 mg oral tablet; acetaminophen 160 mg chewable tablet
Assessment: Patient is presenting with acute bronchitis (disorder).

Generated summary

A 5-year-old nonhispanic white male presents with no complaints. He has a documented history of multiple conditions, including sprain (morphologic abnormality), primary dental caries, acute viral pharyngitis, sprain of the ankle, and viral sinusitis. Currently, he is prescribed amoxicillin/clavulanate and acetaminophen. The assessment reveals that his current condition is acute bronchitis.

Clinical Notes Named Entity Recognition

Purpose: Identify and classify clinical entities within nurse practitioner notes

Entity Types:

- Medications, vital signs, symptoms, diagnoses, procedures, treatment plans, etc.
- 78 clinical entity classes using Begin-Inside-Outside (BIO) tagging scheme

Model Architecture:

- BiLSTM neural network with BioWordVec embeddings
- 1.83M parameters (1.69M non-trainable, 149K trainable)
- Preprocessing: tokenization, cleaning, lemmatization

Training Details:

- Adam optimizer with categorical cross-entropy loss
- Early stopping based on validation F1 score
- Learning rate reduction on plateau
- ~20 epochs with batch size of 32

Model: "functional"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 100)	0
embedding (Embedding)	(None, 100, 200)	1,685,800
bidirectional (Bidirectional)	(None, 100, 128)	135,680
time_distributed (TimeDistributed)	(None, 100, 64)	8,256
dropout (Dropout)	(None, 100, 64)	0
time_distributed_1 (TimeDistributed)	(None, 100, 79)	5,135

Total params: 1,834,871 (7.00 MB)

Trainable params: 149,071 (582.31 KB)

Non-trainable params: 1,685,800 (6.43 MB)

Clinical note with NER entities

Isaiah615, a 2–3-year-old **(Age)** nonhispanic **(Sex)** white **(Personal_background)** male **(Sex)** with a middle **(Detailed_description)** socioeconomic **(History)** background and UnitedHealthcare insurance **(History)**, had 4 **(Duration)** visits **(Clinical_event)** in 2021.
His medical history includes acute **(Detailed_description)** viral **(History)** pharyngitis **(Disease_disorder)**, viral **(History)** sinusitis **(Disease_disorder)**, an ankle **(Biological_structure)** sprain **(History)**, and primary dental **(Nonbiological_location)** caries.
During these visits **(Clinical_event)**, he was prescribed amoxicillin/clavulanate for infections **(Disease_disorder)** and acetaminophen **(Medication)** for pain **(Sign_symptom)**, with no specific allergies **(Disease_disorder)** noted.
The year saw the development of dental **(Nonbiological_location)** caries in May 2021 and consistent use of antibiotics **(Medication)** and analgesics **(Medication)**.
Key observations highlight his growing health concerns, particularly related to dental **(Therapeutic_procedure)** issues.

Results

SQL QA Agent:

- Precision: 89%
- Recall: 100%
- F1-Score: 94%
- Accuracy: 90%

NER Model Performance:

- Token-Level Metrics
  - Precision: 97.3%
  - Recall: 93.7%
  - F1 Score: 95.4%
  - Accuracy: 95.0%

NER Model Performance continued:

- Entity-Level Performance
  - Best entities: Age (F1 95%), Sex (F1 97%)
  - Moderate entities: Medication (F1 77%), Clinical Event (F1 77%)
  - Challenging entities: Sign/Symptom (F1 62%)
  - Overall weighted F1: 71%
- Performance Analysis
  - Notable disparity between token-level metrics (95% F1) and entity-level metrics (71% F1)
  - B-I tag performance gap indicates challenges with entity continuation (ie. shortness of breath)
  - Shorter, standardized entities perform better than complex, variable expressions

Discussion

Interesting findings:

- NER accuracy varies by entity type and tag position (B vs I tags)
  - Short, simple terms (ie. age) are easier to detect than longer, complex ones (ie. sign/symptom)
  - B-I tag performance gap show that detect the beginning of a term but miss the continuation
- SQL QA agent achieved perfect recall (1.0) despite complex natural language queries
- Chunking approach for notes summarization effectively mitigates LLM context limitations

Challenges faced:

- Training with limited clinical data (200 documents) required specialized embeddings and architecture
- Striking a balance between making notes shorter & easier to read while preserving medical details
- Generated SQL queries from natural language raised concerns for correctness so we implemented several validation checks
- Ensuring safety from prompt injection attacks

Limitations and Conclusion

Limitations:

- Limited training data (only 200 annotated documents)
- Entity class imbalance causing varied performance
- SQL query complexity for certain edge cases
- LLM context limitations requiring chunking approach

Future Work:

- Transfer learning and few-shot adaptation for rare entities
- Enhanced temporal reasoning for patient trends
- Multi-modal integration with wearable device data
- Explainable AI for clinical decision support
- User interface improvements based on real-world usage

Conclusion: The Nurse Practitioner Copilot demonstrates a promising approach to streamlining clinical workflows, improving documentation efficiency, and enhancing clinical decision-making in remote patient monitoring scenarios.