Slide 1 — Project definition

Motivation

Clinical notes embed follow-up instructions in free text.
Goal: convert follow-up items into machine-readable JSON
so completion can be checked downstream (EHR / workflows).

Task (input → output)

Input: outpatient visit note text (includes visit date).
Output: list of scheduled follow-ups:
[{ action, period_date }]
Internally: extract period text (“two weeks”) span then normalize to date.

What we compare

- BioBERT pipeline: NER(ACT/TIME) + learned linking head + dateparser
- LLaMA fine-tune (LoRA): direct JSON extraction baseline
- ChatGPT zero-shot: prompt-only baseline

Key metrics

- Action span F1, Time span F1
- End-to-end Action+Date F1
- Date MAE (days) on matched actions

Datasets

Synthetic diverse doctor visit notes for closed set action of 28 actions.

Problem

Why extract follow-up instructions?

Clinical notes are rich—but not machine-readable.

Joint IE task

What becomes possible

- Automatically check if patients completed ordered follow-ups (via EHR queries)
- Trigger reminders / outreach for overdue items
- Support triage and care coordination
- Reduce manual chart review and quality monitoring effort

“MRI in 2 weeks”->

{"action": "MRI", "period_date": "2025-03-08"}

Why it is hard

- Multiple actions and multiple time phrases in the same note
- Past history vs future plan time expressions
- Action and time can be non-adjacent (time-first, split lines, parentheses)
- Linking ambiguity when 2 actions + 2 time phrases appear together

Dataset

Goal: generate realistic, diverse outpatient notes with controlled ground truth

What we generated & labeled

- 2,000 notes across 5 specialties (Orthopedic, Cardio/Pulm, GI, Neuro, General).
- Each note contains a visit date and optionally a plan header; 0–2 scheduled items per note.
- Ground truth per scheduled item: action (closed set), period_text, computed period_date, plus character spans (action/time) for BIO training.

Generation technique

- Skeleton-first: sample specialty/topic/styles, visit date, actions, and deterministic timing phrase templates.
- LLM writes the narrative note; plan section uses a random header and includes exactly the allowed scheduled items.
- Deterministic period_date label computed with dateparser using RELATIVE_BASE = visit date .

Synthetic Dataset Generation

- To ensure the downstream NLP models (BioBERT, Llama) generalize effectively to real-world clinical noise, the dataset generation process was engineered to maximize variance across four key axes: **Clinical Domain**, **Linguistic Style**, **Temporal Expression**, and **Structural Formatting**.
- The goal was to move beyond simple template-filling and create a dataset that mimics the "long tail" of irregularity found in genuine Electronic Health Records (EHRs).

Clinical Domain Diversity

- To prevent the model from overfitting to specific medical vocabularies, we implemented a broad coverage of clinical scenarios.
- **Breadth of Specialties:** The generator rotates through 5 distinct medical domains: *Orthopedics, Cardiovascular/Pulmonary, Gastroenterology, Neurology, and General Medicine*.
- **Condition Granularity:** Each specialty draws from a pool of specific pathologies (e.g., *ACL Tear* vs. *Lumbar Herniated Disc* in Orthopedics; *Atrial Fibrillation* vs. *COPD* in Cardio). This ensures the model encounters diverse symptom descriptions and medical reasoning.
- **Action Variety:** The target labels (actions) range from imaging (*MRI, CT Scan*) to procedures (*Endoscopy, Joint Injection*) and consultations (*Physical Therapy, GI Consult*).

Dataset

Generating diverse dataset

```
SPECIALTY_MAPPING = {
    "Orthopedic": [
        "MRI", "Physical Therapy", "CT Scan", "X-Ray", "Orthopedic Consult", "Joint Injection"
    ],
    "Cardiovascular / Pulmonary": [
        "Echocardiogram", "Stress Test", "Holter Monitor", "Pulmonary Function Test",
        "Cardiac MRI", "Cardiology Consult"
    ],
    "Gastroenterology": [
        "Endoscopy", "Colonoscopy", "Stool Antigen Test", "Abdominal Ultrasound",
        "GI Consult", "Breath Test"
    ],
    "Neurology": [
        "CT Scan", "EEG", "MRI Brain", "EMG", "Neurology Consult", "Sleep Study"
    ],
    "General Medicine": [
        "Blood Test", "X-Ray", "Vaccination", "Annual Physical", "Urinalysis", "Lipid Panel"
    ]
}
```

Dataset

```
TOPICS = {
    "Orthopedic": [
        "ACL Tear", "Rotator Cuff Injury", "Lumbar Herniated Disc", "Ankle Sprain",
        "Carpal Tunnel Syndrome", "Hip Osteoarthritis", "Meniscus Tear", "Tennis Elbow",
        "Plantar Fasciitis", "Distal Radius Fracture",
        "Sciatica", "Scoliosis", "Bunion (Hallux Valgus)", "Patellar Tendonitis",
        "Shoulder Dislocation"
    ],
    "Cardiovascular / Pulmonary": [
        "Atrial Fibrillation", "COPD Exacerbation", "Acute Bronchitis", "Hypertension",
        "Mitral Valve Prolapse", "Pneumonia", "Congestive Heart Failure", "Deep Vein Thrombosis",
        "Asthma Attack", "Pericarditis",
        "Pulmonary Embolism", "Coronary Artery Disease", "Aortic Stenosis", "Bradycardia",
        "Syncope/Fainting"
    ],
    "Gastroenterology": [
        "GERD", "Irritable Bowel Syndrome", "Crohn's Disease", "Ulcerative Colitis",
        "Gallstones", "Celiac Disease", "Peptic Ulcer", "Diverticulitis",
        "Hemorrhoids", "Liver Cirrhosis",
        "Barrett's Esophagus", "Acute Gastritis", "Hiatal Hernia", "Chronic Pancreatitis",
        "Hepatitis C"
    ],
    "Neurology": [
        "Migraine with Aura", "Epilepsy/Seizure", "Multiple Sclerosis", "Parkinson's Disease",
        "Ischemic Stroke", "Carpal Tunnel (Neuro view)", "Vertigo/BPPV", "Alzheimer's/Dementia",
        "Bell's Palsy", "Neuropathy",
        "Myasthenia Gravis", "Trigeminal Neuralgia", "Huntington's Disease", "Guillain-Barre Syndrome",
        "Restless Leg Syndrome"
    ],
    "General Medicine": [
        "Type 2 Diabetes", "Seasonal Influenza", "Hypothyroidism", "Urinary Tract Infection",
        "Anemia", "Vitamin D Deficiency", "Hypertension", "Annual Physical",
        "Lyme Disease", "Gout",
        "Hyperlipidemia (High Cholesterol)", "Fibromyalgia", "Chronic Fatigue Syndrome", "Infectious Mononucleosis",
        "Osteoporosis"
    ]
}
```

Linguistic Diversity (Stylistic Variance)

- A major innovation in this pipeline is the STYLE_FEATURES_DICT, which randomly injects stylistic constraints into the LLM prompt. This prevents the "synthetic distinctness" often seen in AI-generated text.
- **Narrative Voice:** Shifts between *First Person* ("I recommend...") and *Third Person/Passive* ("Patient to return...").
- **Tone & Persona:** Simulates different clinician archetypes, from the *Educator* (who explains the 'why') to the *Burnout/Hasty* clinician (telegraphic, minimum viable documentation).
- **Simulated Noise:**
  - **Typos:** Simulated fast-typing errors (e.g., "pateint").
  - **Dictation Artifacts:** Run-on sentences and missing punctuation typical of voice-to-text software.
  - **Shorthand:** Heavy use of clinical abbreviations ("c/o CP, SOB" vs. "complains of chest pain, shortness of breath").

Dataset

```
STYLE_FEATURES_DICT = {
    "Narrative Voice": [
        "First Person ('I examined the patient', 'I recommend')",
        "Third Person ('Patient presents with', 'It is recommended')"
    ],
    "Temporal Precision": [
        "Vague ('History of surgery years ago', 'pain for a while')",
        "Precise ('Surgery on 12/05/2020', 'pain started at 2 PM')"
    ],
    "Certainty": [
        "Definitive ('Patient has pneumonia')",
        "Hedging/Uncertain ('Findings suggestive of possible pneumonia', 'cannot rule out')"
    ],
    "Formality": [
        "Formal/Academic (Complete sentences, proper grammar)",
        "Casual/Direct (Conversational, simple sentence structures)",
        "Standard Clinical (Professional but concise)"
    ],
    "Detail Level": [
        "Highly Detailed/Verbose (Explains rationale, describes scene)",
        "Abbreviated/Telegraphic (Notes style, fragments)",
        "Standard (Balanced)"
    ],
    "Tone": [
        "Polite/Empathetic ('Patient is a pleasant 45yo...')",
        "Clinical/Detached (Just the facts)",
        "Direct/Urgent ('Patient in distress, immediate action required')"
    ],
    "Terminology": [
        "Simple Language (Patient-friendly terms like 'heart attack')",
        "Heavy Medical Jargon (terms like 'myocardial infarction')",
        "Mixed (Standard EHR style)"
    ],
```

Dataset

```
"Terminology": [
    "Simple Language (Patient-friendly terms like 'heart attack')",
    "Heavy Medical Jargon (terms like 'myocardial infarction')",
    "Mixed (Standard EHR style)"
],
"Structure": [
    "Standard SOAP Headers (Subjective: ... Objective: ...)",
    "Minimal Headers (HPI: ... PE: ... Imp: ...)",
    "Run-on Paragraph (No clear section breaks, one block of text)",
    "Bullet Points (Heavy use of lists for symptoms/plan)"
],
"Imperfections": [
    "Perfect Grammar (textbook quality)",
    "Slight Shorthand (Standard abbreviations like 'pt', 'yo', 'hx')",
    "Heavy Shorthand (Aggressive abbrv: 'c/o CP, SOB, N/V, rec MRI')",
    "Dictation Style (Occasional missing punctuation, run-on sentences)",
    "Minor Typos (Simulated fast typing errors like 'pateint' or 'swelng')"
],
"Clinician Persona": [
    "Defensive Medicine (Over-explaining rationale to justify decisions)",
    "Action-Oriented (Brief history, very detailed plan)",
    "Burnout/Hasty (Minimum viable documentation, very brief)",
    "Educator (Explaining the 'why' behind the diagnosis)"
]
```

Temporal Expression Diversity

- **Temporal Expression Diversity**
- Since the core task is **Time Extraction**, the dataset rigorously varies how time is encoded. The model must learn semantic equivalence across these formats:
- **Units & Numerals:** Random mixing of digits vs. words ("7 days" vs. "seven days") and unit variations ("wks", "weeks", "mo").
- **Absolute vs. Relative:** While the output is normalized dates, the input text varies between relative durations ("in 2 weeks") and fuzzy approximations ("in about a week“)
- **Phrasing Templates:** We utilize 10+ weighted templates to vary the lead-in phrase:
  - *Standard:* "Return in 2 weeks"
  - *Passive:* "Follow up in 2 weeks"
  - *Noun-Phrase:* "2-week follow-up"
  - *Shorthand:* "2wk f/u", "in ~2wks"

Temporal Expression Diversity

```
NUM_WORD = {
    1: "one", 2: "two", 3: "three", 4: "four", 5: "five",
    6: "six", 7: "seven", 8: "eight", 9: "nine", 10: "ten",
    11: "eleven", 12: "twelve", 13: "thirteen", 14: "fourteen"
}
```

```
TIME_TEMPLATE_WEIGHTS = [
    ("in {n} {unit}", 1.8),
    ("return in {n} {unit}", 1.2),
    ("follow up in {n} {unit}", 1.2),
    ("within {n} {unit}", 2.0),
    ("over the next {n} {unit}", 2.0),
    ("in about {n} {unit}", 1.8),
    ("in approximately {n} {unit}", 1.6),
    ("{n}-{unit_singular} follow-up", 2.0),
    ("{n} {unit_abbrev}", 1.6),
    ("in {n} {unit} time", 1.0),
]
```

```
HARD_TEST_TIME_TEMPLATES = [
    ("{n}wk f/u", 1.2),
    ("{n} wks f/u", 1.0),
    ("{n}{unit_abbrev} f/u", 1.0),
    ("f/u in {n} {unit}", 1.1),
    ("in {n}{unit_abbrev}", 1.0),          # e.g., in 6wks / in 3mos
    ("in ~{n} {unit}", 0.9),               # e.g., in ~8 weeks
    ("in approx. {n} {unit}", 0.9),
    ("in about {n}{unit_abbrev}", 0.8),
    ("in {n}-{m} weeks", 0.9),             # range
    ("in {n} to {m} weeks", 0.8),          # range
]
```

Structural & Layout Diversity

- To prevent the model from learning positional heuristics (e.g., "the date is always at the end of the line"), we strictly controlled the structural layout via PLAN_VARIANTS.
- **Format A/B (Inline):** ACTION - TIME vs TIME: ACTION (Testing attention direction).
- **Format C (Narrative):** The action and time are embedded in a full sentence. ("Plan: We will schedule an MRI in 2 weeks time.")
- **Format D (Parenthetical):** ACTION (TIME) — A common shorthand pattern.
- **Format E (Grouped):** Diagnostics: ACTION (TIME); ACTION (TIME) — Testing the ability to parse list delimiters.
- **Format F (Split Line):** The action and time are separated by newlines, forcing the model to aggregate information across line breaks.

Doctor Visit Note Example

Date of Visit: 2024-03-26

Patient is a pleasant 45-year-old male presenting with weakness on the right side and difficulty speaking, which began approximately **2 weeks ago**. He reports that these symptoms intensified over the **past few days**. Neurological examination reveals right-sided hemiparesis and expressive aphasia. His past medical history is significant for hypertension, which has been well controlled. A **prior CT scan**, **performed** during his last visit, **was negative** for acute changes.

After discussing his symptoms, I believe he has experienced an ischemic stroke. I emphasized the importance of immediate management to prevent further complications, especially since he has been feeling progressively worse and may need a more comprehensive evaluation.

INSTRUCTIONS:

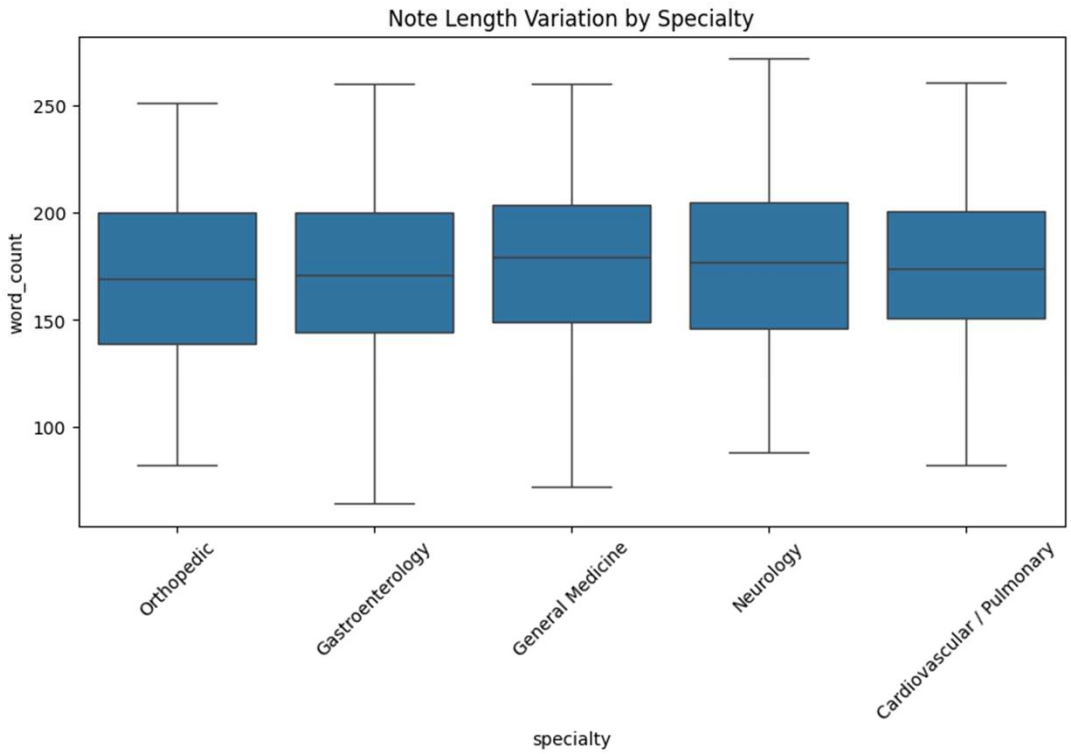
I **recommend** an **MRI Brain**. This will be **scheduled in 12 days**. Additionally, we should conduct a **Sleep Study** to assess any potential underlying sleep disorders. This **will** take place **over the next three days**. I advised the patient to monitor his symptoms closely and to seek emergency care if not improved **in 48 hours** or if he experiences any worsening of his condition. We will follow up on the results of these evaluations at his next appointment.

Dataset

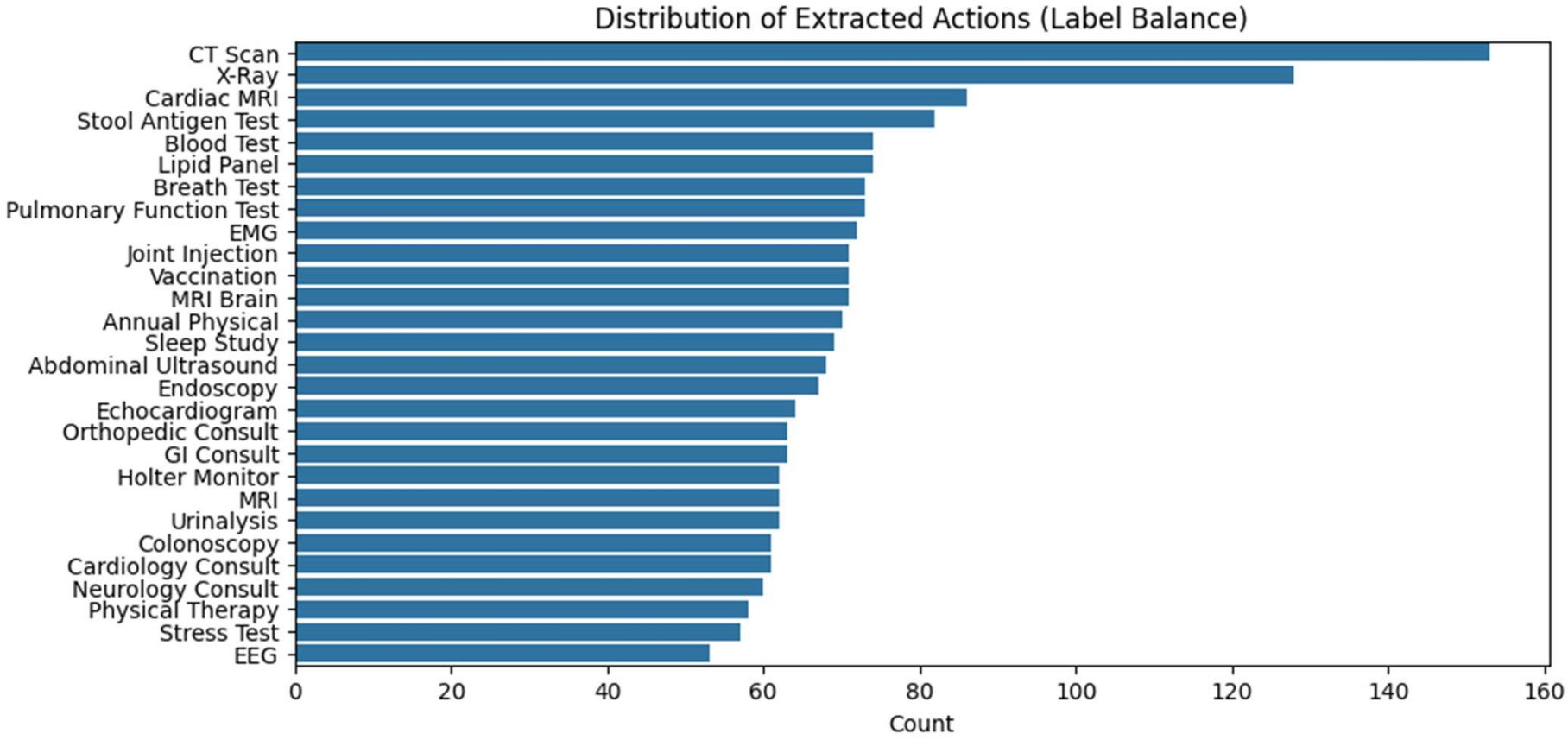
Style Prompt Affect

```
--- Validation: Does the Style Prompt affect Word Count? ---
style_category
Detailed       196.067616
Standard       173.623885
Telegraphic    143.858779
Name: word_count, dtype: float64
```

Note Length Per Specialty



Actions Distribution



Approaches compared

Structured pipeline vs generative baselines.

Baseline B — Fine-tuned LLaMA (generative)

- Prompted to output JSON directly
- Trained on synthetic dataset with closed action set in prompt
- Main risk: date arithmetic & formatting errors

Baseline C — Zero-shot ChatGPT (generative)

- No fine-tuning; prompt asks for strict JSON only
- Strong at finding actions/time phrases
- Still makes relative→absolute date mistakes

Head A: Named Entity Recognition (NER)

- This component is responsible for identifying the *actions and dates*
- **Scheme:** We employ a **BIO Tagging Scheme** with 5 classes: O (Outside), B-ACT (Begin Action), I-ACT (Inside Action), B-TIME (Begin Time), and I-TIME (Inside Time).
- **Class Imbalance Handling:** To prevent the model from bias toward the majority class (O), we apply weighted Cross-Entropy Loss, assigning a **10x penalty** for missing valid entity tokens compared to background tokens.

Head A: BIO TAGGING

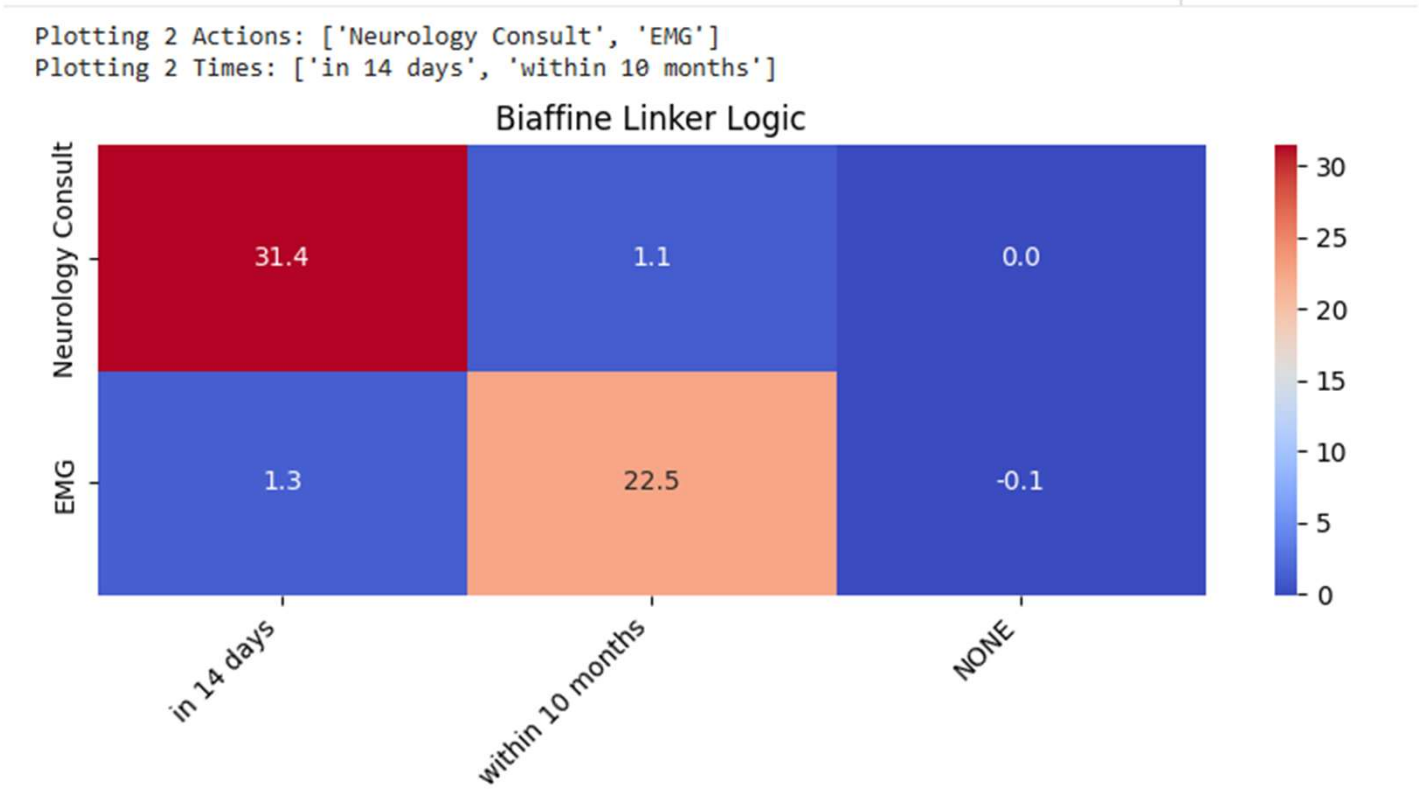
```
"""
Date of Visit: 2025-11-01

History: Patient is seen 2 weeks post-op from laparoscopic appendectomy. Incisions are well-healed.
No signs of erythema or drainage. Bowel function has returned to normal.

Plan:
- order an Abdominal Ultrasound this Friday in 6 weeks.
- If the ultrasound is negative resume gym activity.
"""
```

TOKEN	TAG ID	DECODED TAG
abdominal	1	B-ACT
ultra	2	I-ACT
##sound	2	I-ACT
in	3	B-TIME
6	4	I-TIME
weeks	4	I-TIME

Head B: Biaffine Linker

ASSESSMENT & PLAN:

- Carpal tunnel syndrome likely contributing to patient's symptoms.
- Neurology Consult (in 14 days) to further evaluate and discuss potential interventions.
- EMG (within 10 months) to assess the severity of median nerve involvement and confirm diagnosis.



Training Methodology

The model is trained using a **Joint Loss** function that optimizes both tasks simultaneously:

$L_{total} = L_{NER} + \alpha * L_{Linking}$

(Where α *=1.0 balances the importance of both tasks).*

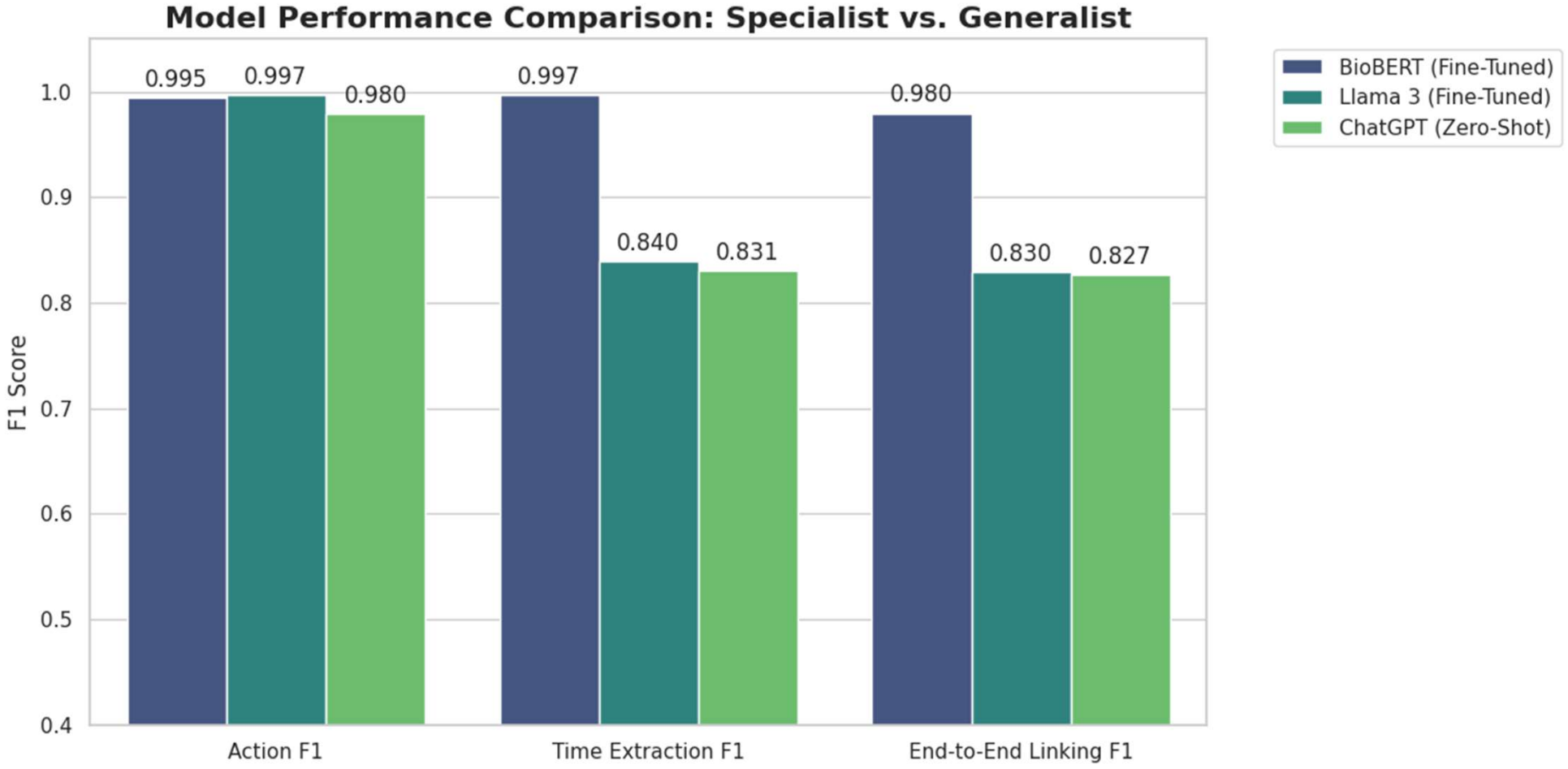
- **Early Stopping:** To prevent overfitting to the synthetic data patterns, we monitor validation loss with a patience of 4 epochs. (In our experiments, the model converged to optimal performance at **Epoch 7**).

LLaMA baseline-direct JSON extraction

```
[
  {"action": "MRI Brain", "period_date": "2025-02-15"}
]
```

```
system_prompt = """You are an expert clinical information extraction system.
Task:
Extract ONLY scheduled follow-up items from the clinical note (tests/labs/imaging/referrals/therapy explicitly planned for the future).
Ignore history/past tests/symptom duration.

Return ONLY valid JSON.
"""“user_prompt = f"""Rules:
- Output must be a JSON array (possibly empty).
- Each item must be a scheduled follow-up item from the note (not history).
- action must be EXACTLY one of the allowed actions below (no new actions).
- period_date must be computed relative to the visit date line in the note (YYYY-MM-DD).
{"- period_text must be copied verbatim from the note.\n" if LLAMA_TARGET_MODE != "date_only" else ""}
- If no scheduled items: return [].
- Output JSON only (no markdown, no extra text).
Allowed actions:
{allowed_actions}
Clinical note: {note}"""
```

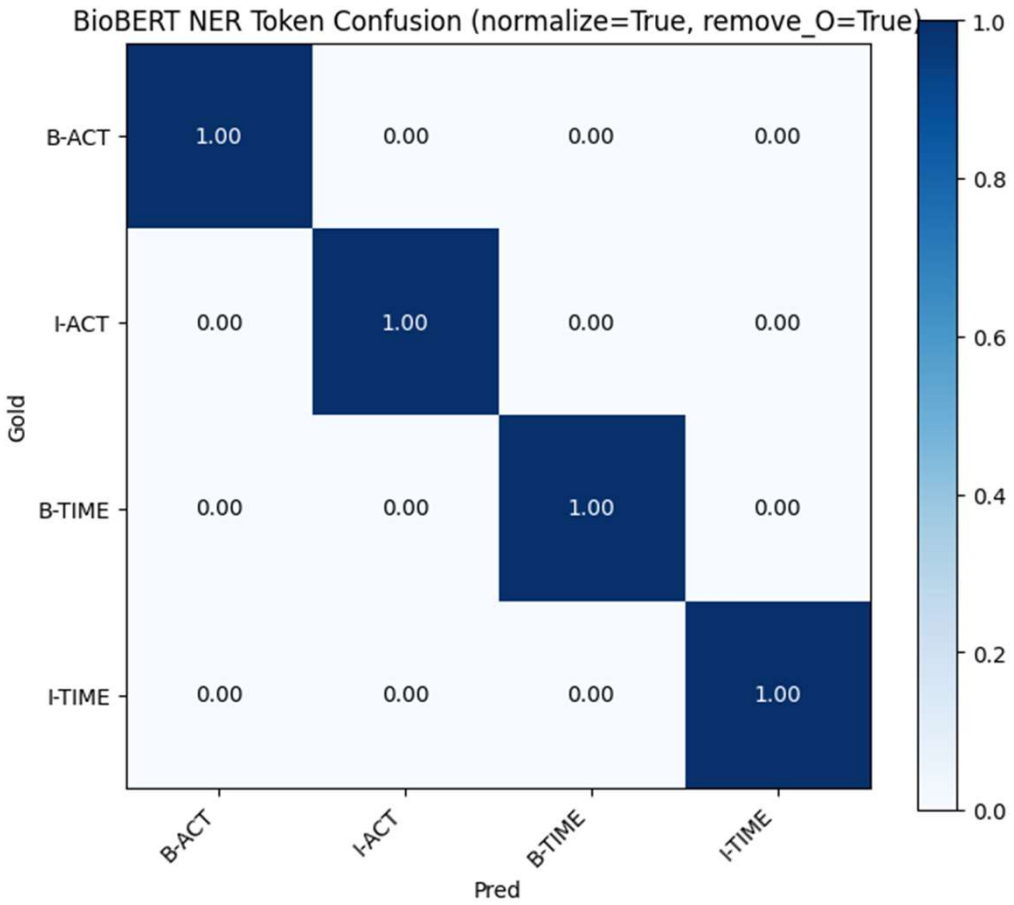
Comparing metrics



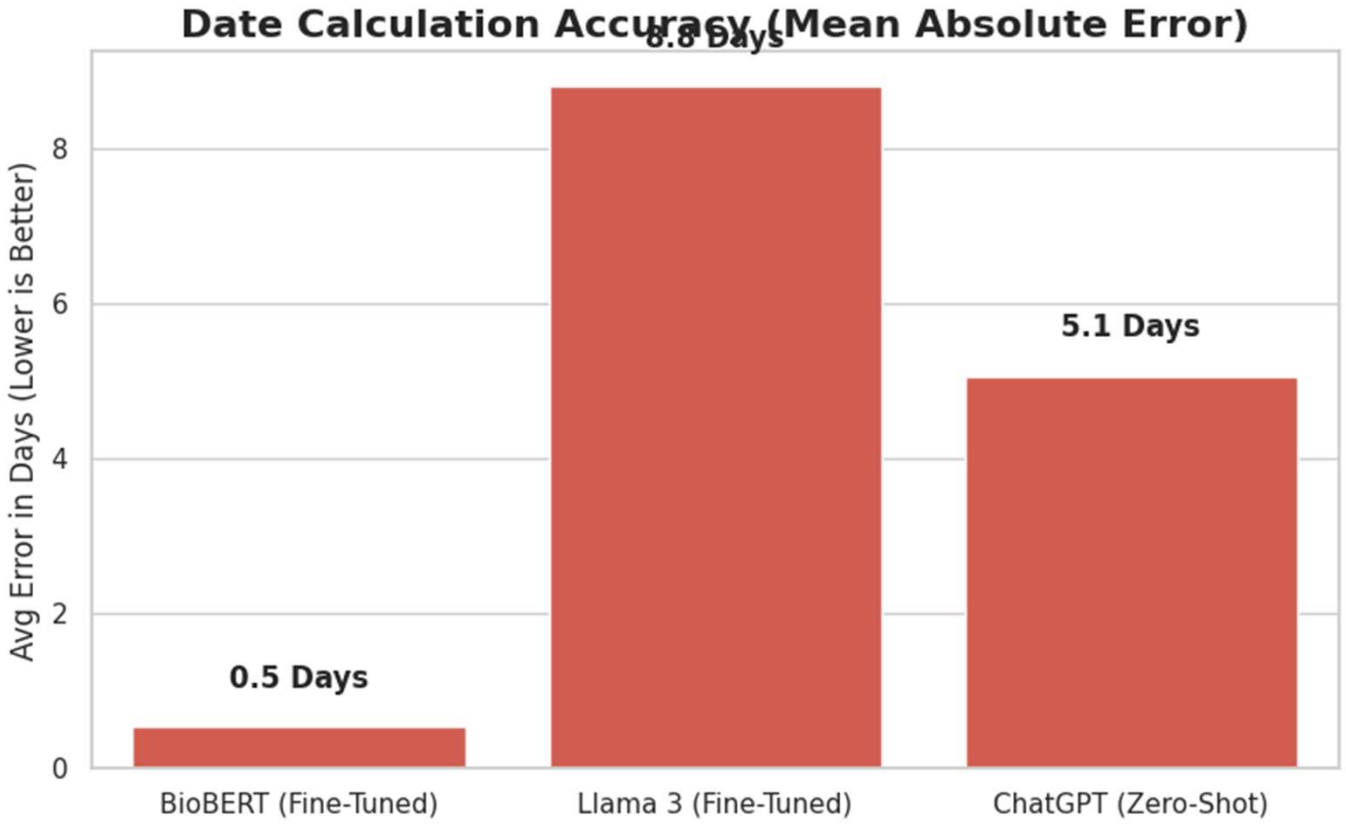
Model Performance Comparison Table

Metric	**BioBERT Pipeline** (Neuro-Symbolic)	**Llama 3** (Fine-Tuned)	**ChatGPT** (Zero-Shot)
Action Extraction F1	**0.995**	**0.997**	0.980
(Finding the text span)			
Time Extraction F1	**0.997**	0.840	0.831
(Finding the date phrase)			
End-to-End Pair F1	**0.980**	0.830	0.827
(Action + Calculated Date)			
Date Error (MAE)	**0.53 days**	8.82 days	5.07 days

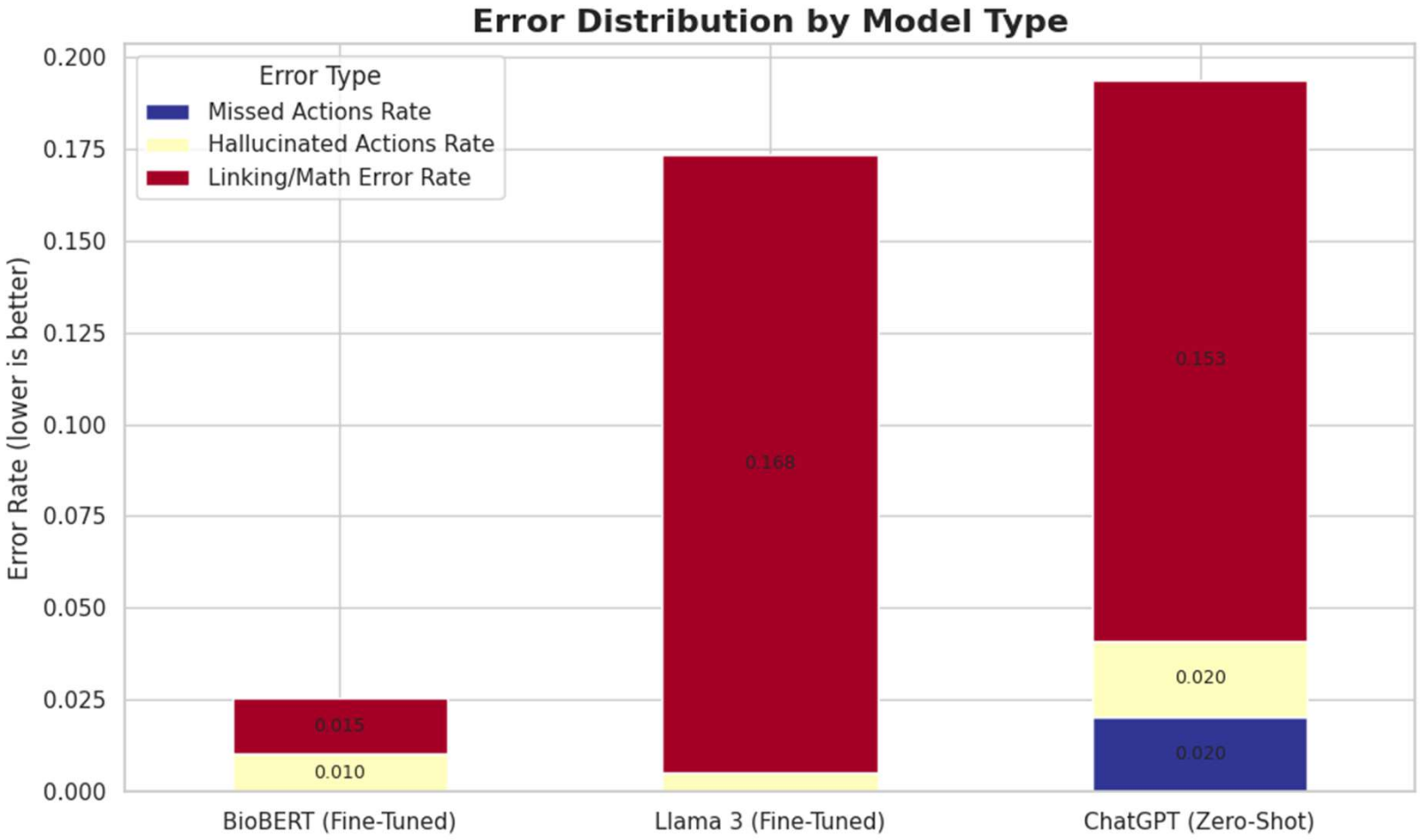
BioBERT Confusion Matrix



Date Calculation Accuracy (Mean Absolute Error)



Error Distribution by Model Type



Conclusion

Did we achieve the goal?

Yes. BioBERT pipeline reliably outputs follow-up actions with normalized dates (≈0.98 F1)

Limitations

- Synthetic data is still cleaner than real EHR notes.

Key takeaways

- LLMs can extract actions, but direct date generation is brittle.
- Extracting period_text as evidence + dateparser improves consistency.

Future work

- Extend the problem beyond 28 actions
- Allow more actions per note
- Evaluate pipeline on real-note benchmark