Presented by John Snow LABS

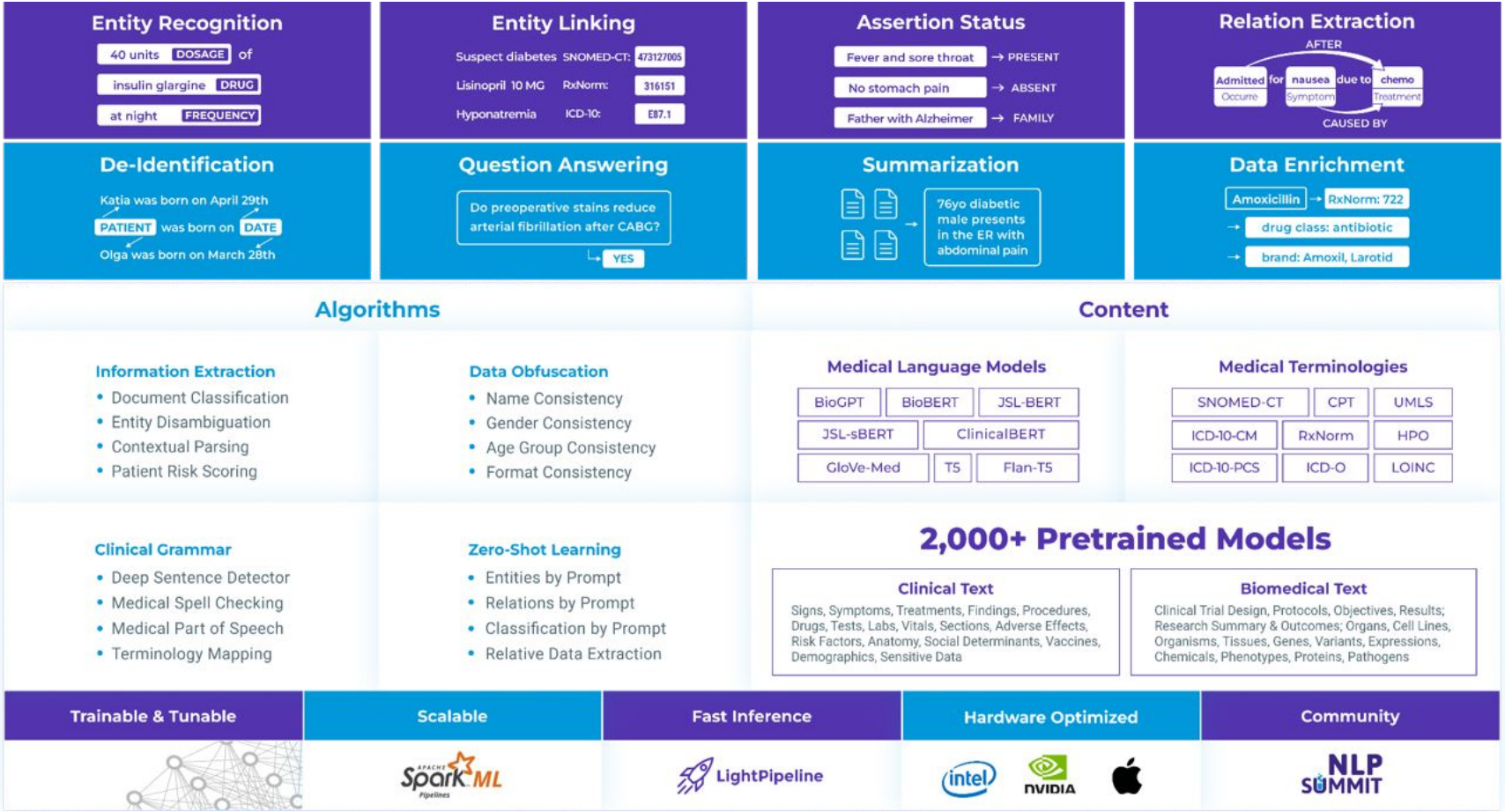
NLP SUMMIT

Building Healthcare NLP Agents with LLMs

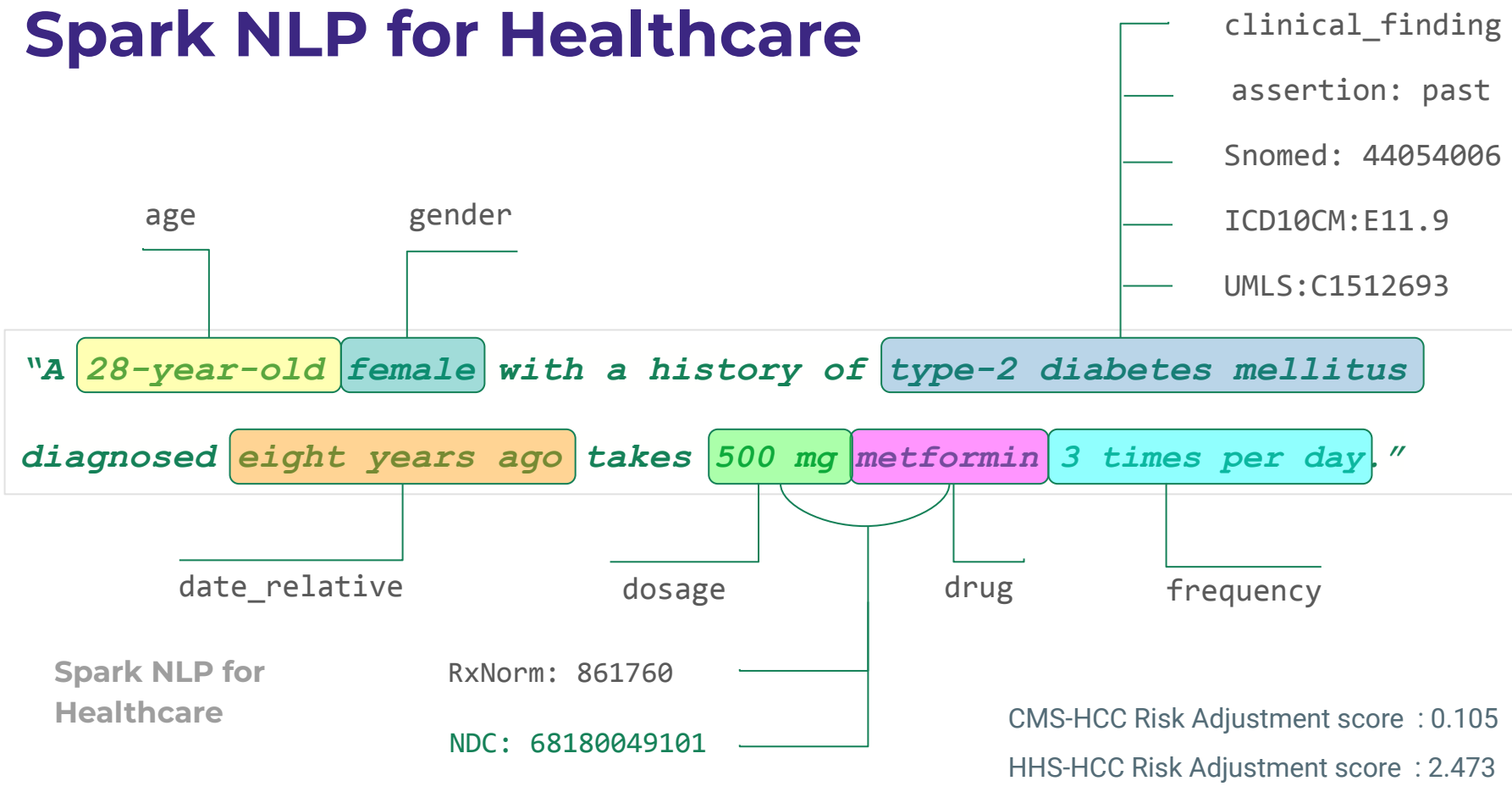
Certification Trainings,
John Snow Labs
July 17th, 2024

Veysel Kocaman, PhD
Head of Data Science
John Snow Labs

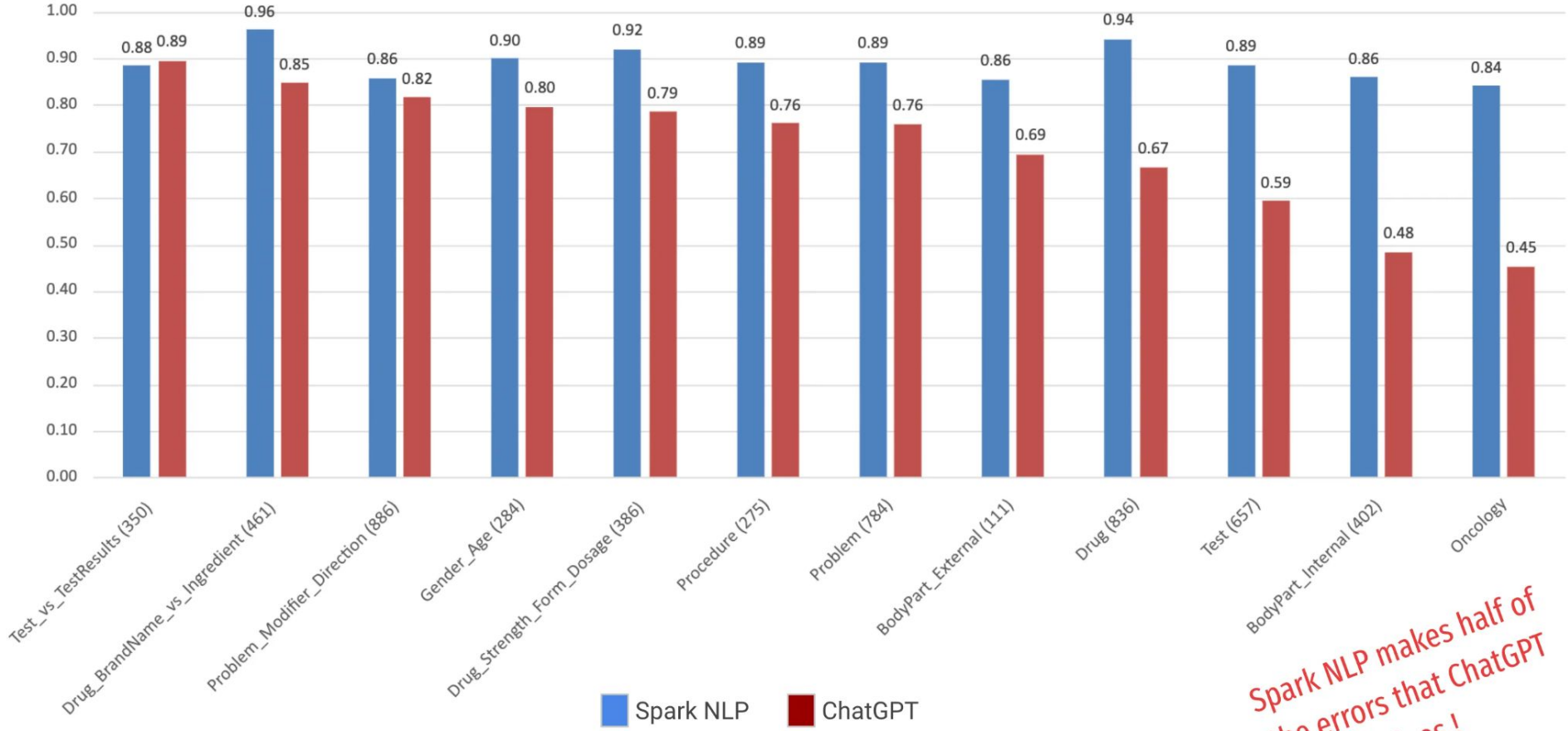
Healthcare NLP by John Snow Labs



Spark NLP for Healthcare



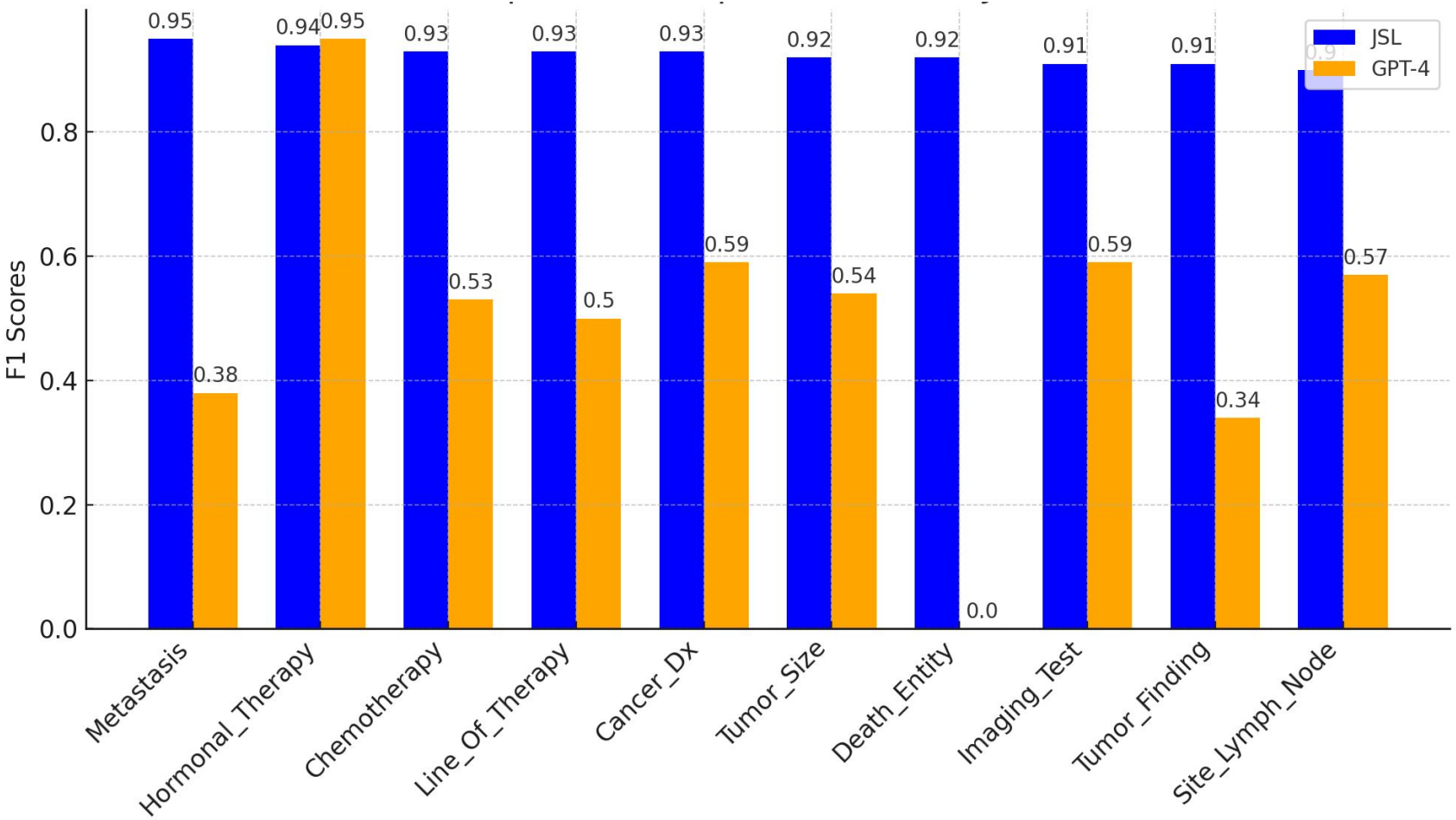
Spark NLP for Healthcare vs ChatGPT (GPT 3.5) on Clinical Entities



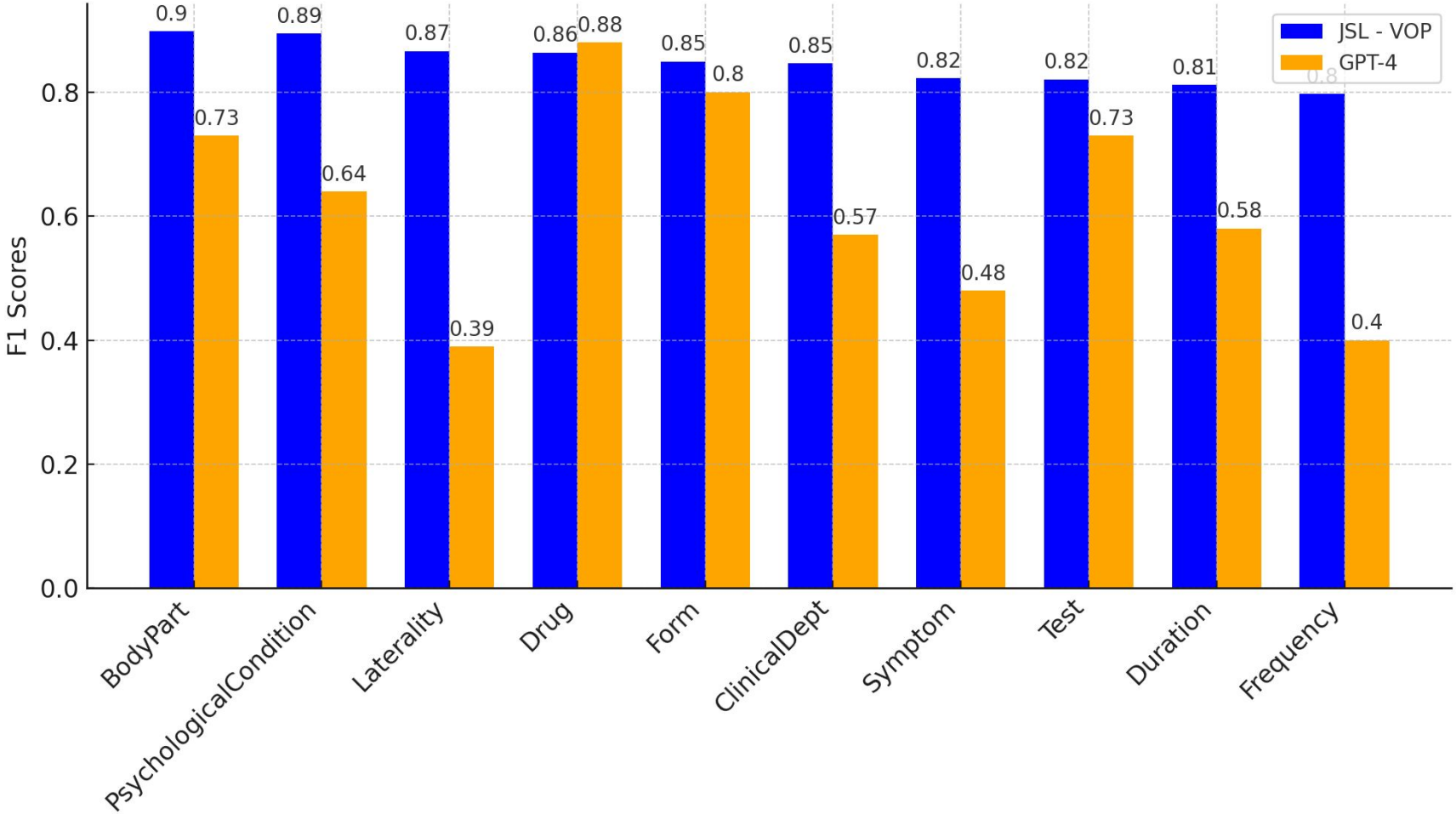
Spark NLP makes half of the errors that ChatGPT does !

https://github.com/JohnSnowLabs/spark-nlp-workshop/tree/master/tutorials/academic/LLMs_in_Healthcare

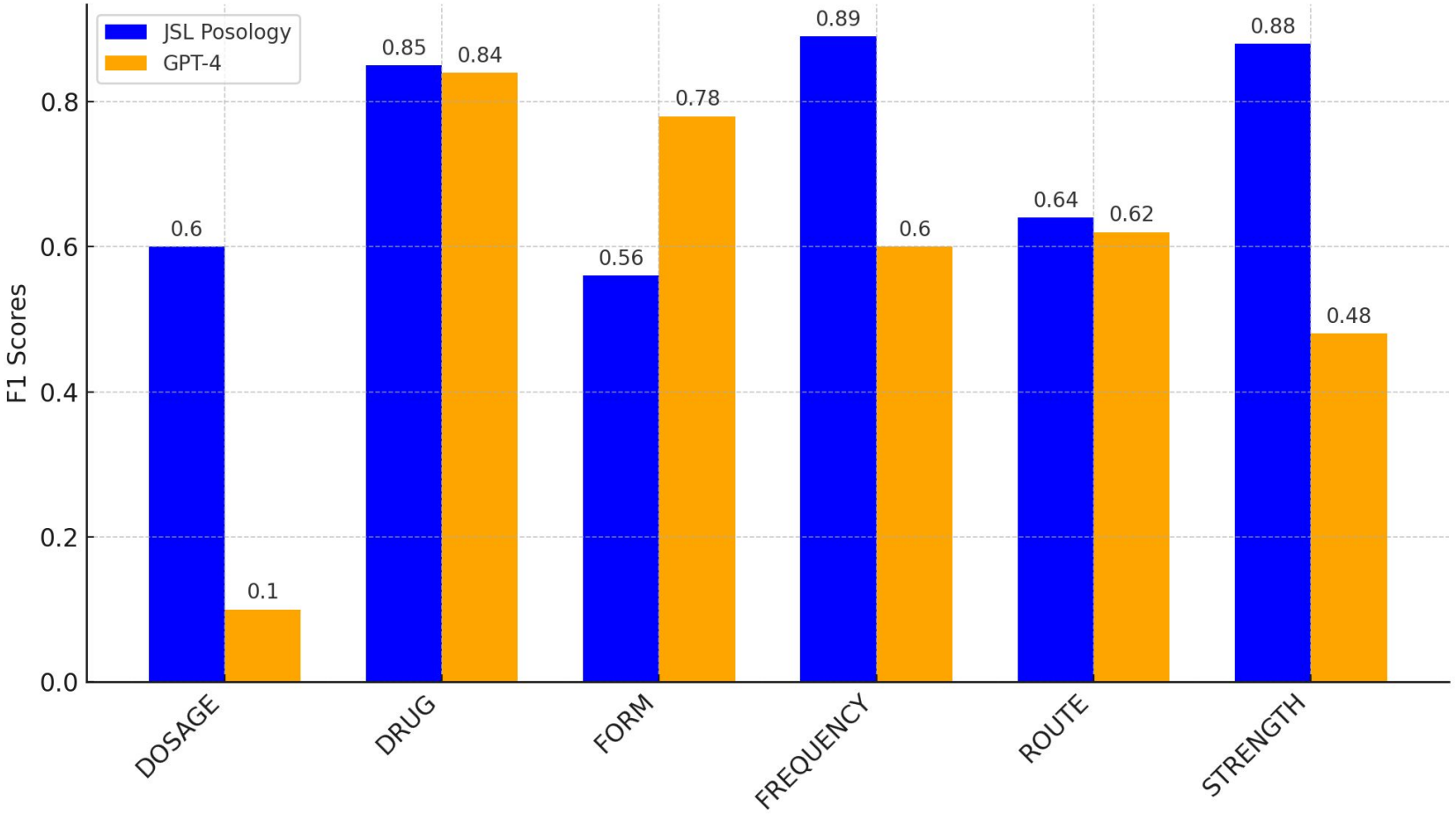
Oncology NER (JSL vs GPT-4)



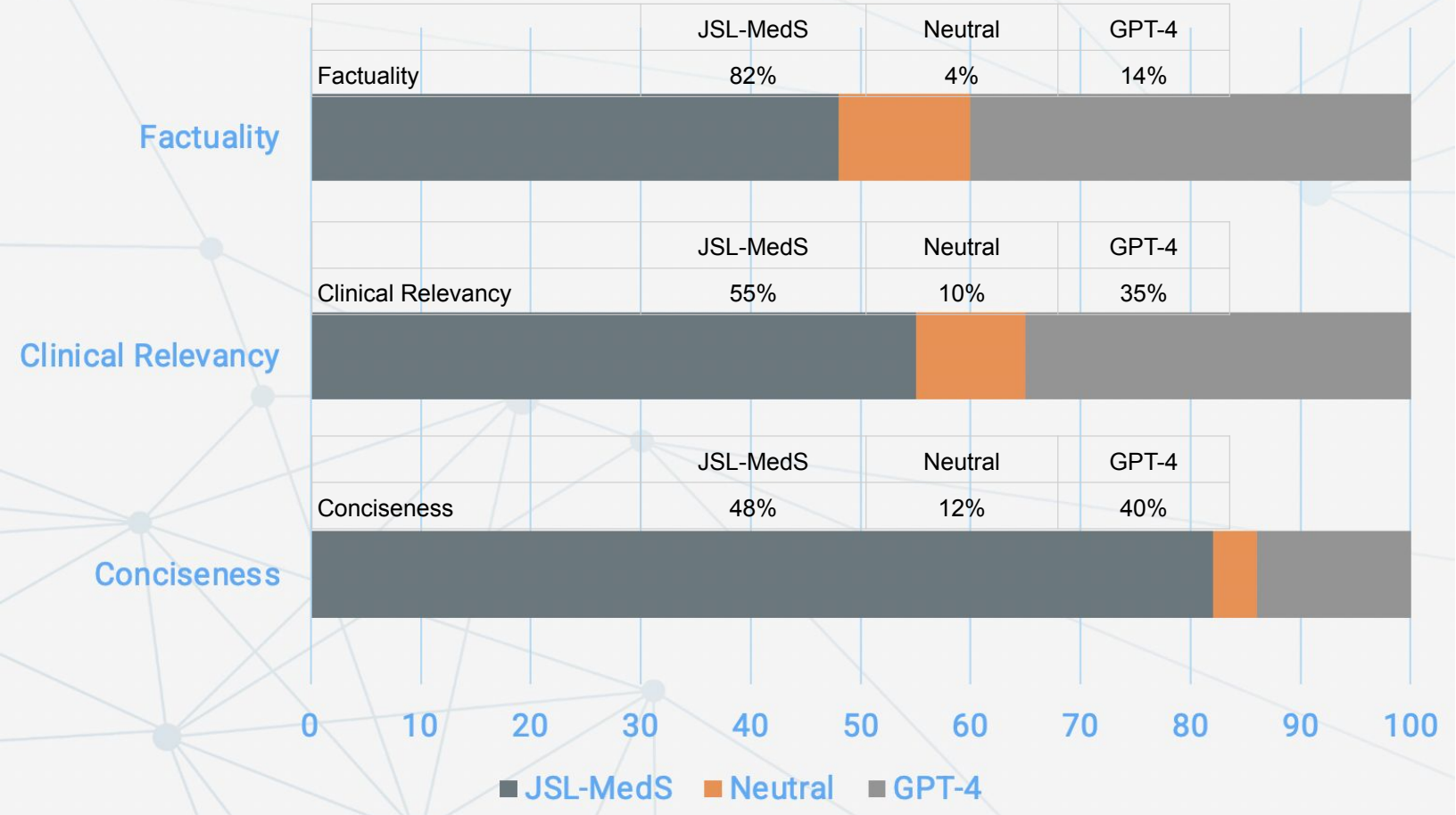
VOP (Voice of Patient) NER (JSL vs GPT-4)



Medication NER (JSL vs GPT-4)



JSL Clinical Summarizer & QA Comparison with GPT-4



John Snow Labs' small language models' (SLMs) preference rate over GPT-4 over 210 questions/ tasks/ interactions by physicians (randomized blind trials)

Healthcare NLP by John Snow Labs

RxNorm Resolver

ICD10 Resolver

Snomed Resolver

MedDRA Resolver

Assertion Status Detection

Risk Adj. Module

Named Entity Recognition

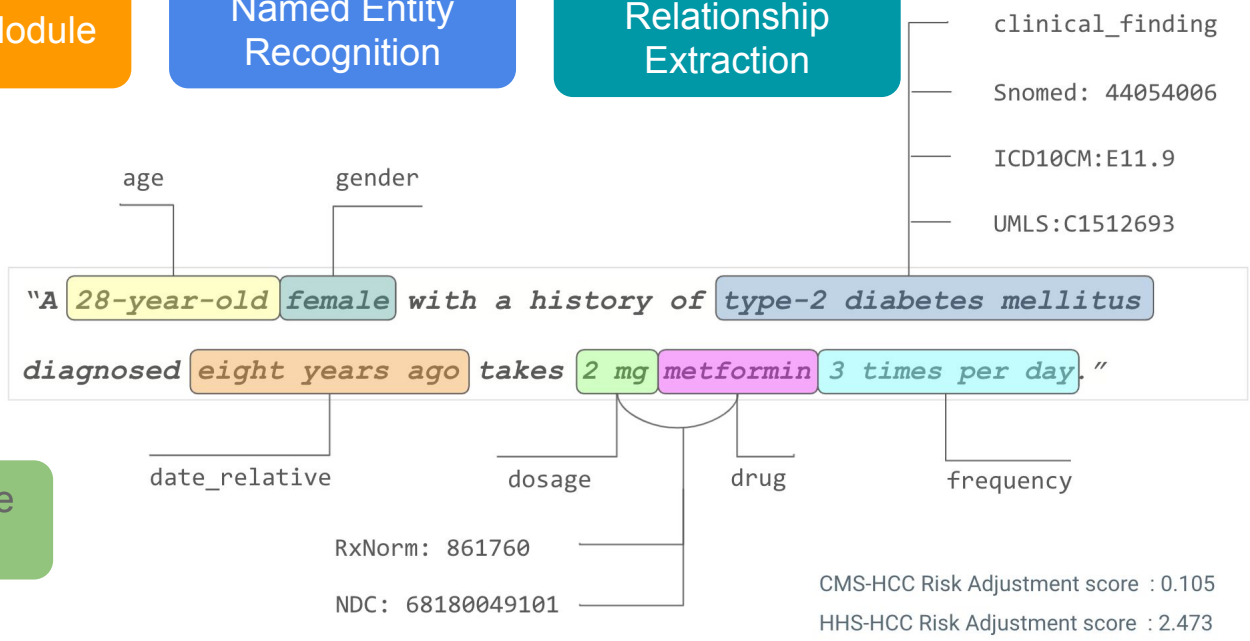
Relationship Extraction

Sentence Splitter

Tokenizer

Bert Embeddings

Small Language Models (SLM)



```python
# Annotator that transforms a text column from dataframe into an Annotation ready for NLP
documentAssembler = DocumentAssembler()\
    .setInputCol("text")\
    .setOutputCol("document")

sentenceDetector =
SentenceDetectorDLModel.pretrained("sentence_detector_dl_healthcare","en","clinical/models")\
    .setInputCols(["document"])\
    .setOutputCol("sentence")

# Tokenizer splits words in a relevant format for NLP
tokenizer = Tokenizer()\
    .setInputCols(["sentence"])\
    .setOutputCol("token")

# Clinical word embeddings trained on PubMED dataset
word_embeddings = WordEmbeddingsModel.pretrained("embeddings_clinical","en","clinical/models")\
    .setInputCols(["sentence","token"])\
    .setOutputCol("embeddings")

# NER model trained on a medical dataset
clinical_ner = MedicalNerModel.pretrained("ner_clinical_large","en","clinical/models")\
    .setInputCols(["sentence","token","embeddings"])\
    .setOutputCol("ner")\
    .setLabelCasing("upper") #decide if we want to return the tags in upper or lower case

ner_converter = NerConverterInternal()\
    .setInputCols(["sentence","token","ner"])\
    .setOutputCol("ner_chunk")

nlpPipeline = Pipeline(
    stages=[
        documentAssembler,
        sentenceDetector,
        tokenizer,
        word_embeddings,
        clinical_ner,
        ner_converter
        ])


empty_data = spark.createDataFrame([[""]]).toDF("text")

model = nlpPipeline.fit(empty_data)
```

```python
text = '''
A 28-year-old female with a history of gestational diabetes mellitus diagnosed eight years prior to presentation and subsequent type two diabetes mellitus (T2DM), one prior episode of HTG-induced pancreatitis three years prior to presentation, and associated with an acute hepatitis, presented with a one-week history of polyuria, poor appetite, and vomiting.'''

agent_result = process_command_SingleAgent(f"Can you extract Problem, Test and Treatment entities from the following text: {text}")
```

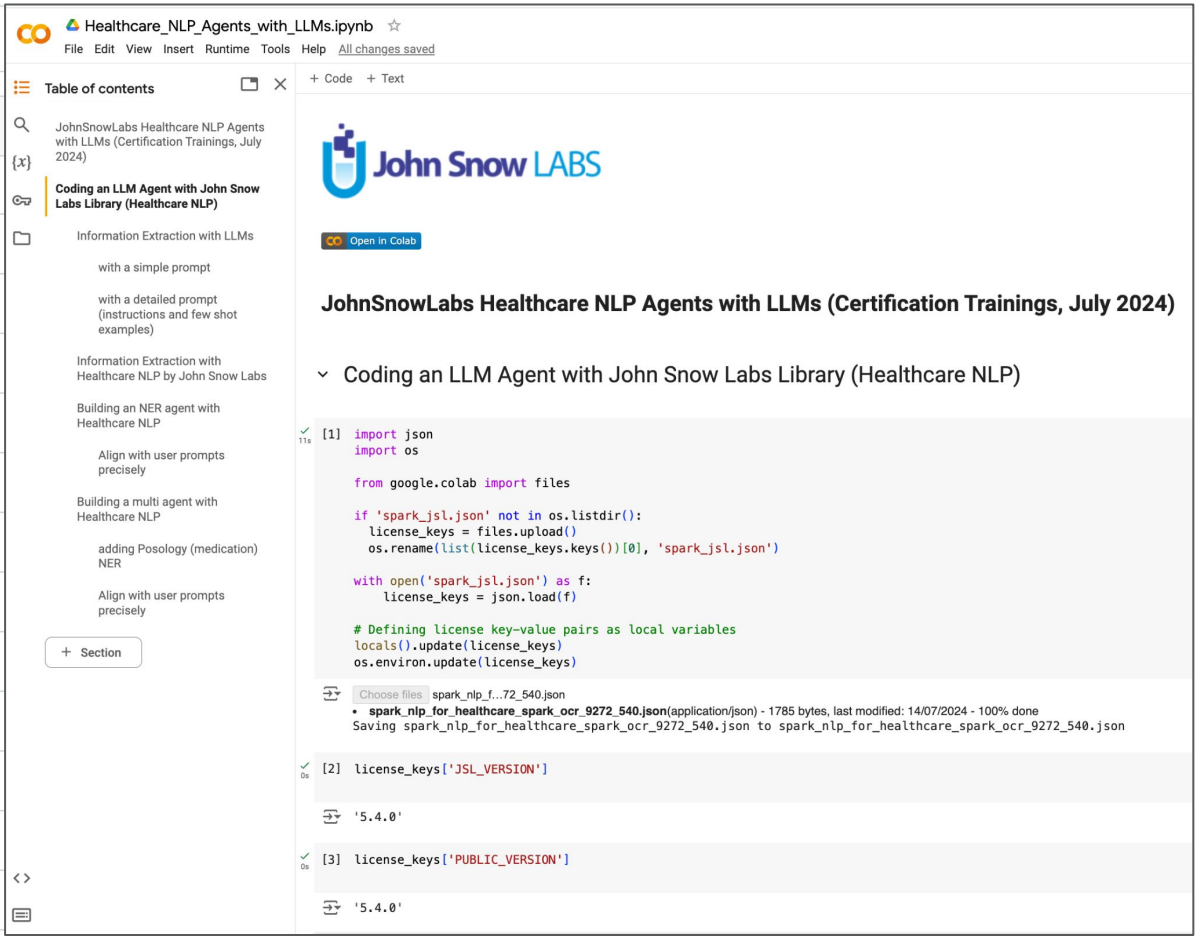
```
Agent found: SNLP4HC_general_Tool_func
```

	chunk	begin	end	entity_label	confidence
0	gestational diabetes mellitus	39	67	PROBLEM	0.91976666
1	subsequent type two diabetes mellitus	117	153	PROBLEM	0.75924003
2	T2DM	156	159	PROBLEM	0.9917
3	HTG-induced pancreatitis	184	207	PROBLEM	0.97535
4	an acute hepatitis	264	281	PROBLEM	0.9440667
5	polyuria	321	328	PROBLEM	0.9728
6	poor appetite	331	343	PROBLEM	0.9934
7	vomiting	350	357	PROBLEM	0.9854

spark-nlp-workshop / tutorials / Certification_Trainings / **Applied_Generative_AI** /

vkocaman Add files via upload

Name
..
data
slides
Domain_Specific_Language_Models.ipynb
Healthcare_NLP_Agents_with_LLMs.ipynb
Integrated Data Science Workflows.ipynb
Medical_Chatbot_API_Example.ipynb
Multimodal_Pipelines_ASR.ipynb
Multimodal_Pipelines_Visual_NLP.ipynb
Oncology_Use_Cases.ipynb
Open_Source_Language_Models.ipynb
Pipelines_LLM.ipynb
README.md



Presented by John Snow LABS

NLP SUMMIT

Thank you !

Veysel Kocaman, PhD

Head of Data Science

John Snow Labs

Understanding OMOP CDM

(Observational Medical Outcomes Partnership - Common Data Model)

Enhancing Healthcare through Data

Foundation: Part of the Observational Health Data Sciences and Informatics (OHDSI) initiative.

Objective: Utilize open-source data solutions to improve human health via large-scale analysis.

Purpose: Standardize the structure and content of observational healthcare data.

Features:

- Enables efficient, reliable evidence production through analysis.
- Incorporates a common vocabulary and standards for clinical data management.

Focus: Centered on patient outcomes and includes recorded healthcare events.

Community: An open community data standard, fostering collaboration and innovation in healthcare data utilization.

OHDSI
OBSERVATIONAL HEALTH DATA SCIENCES AND INFORMATICS

OOP-CDM is a data model that allows clinical information to be presented in a standardized and reusable way for research