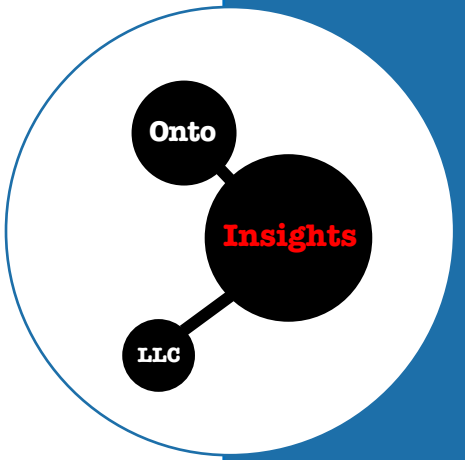
Populating Knowledge Graphs: The Confluence of Ontology and Large Language Models

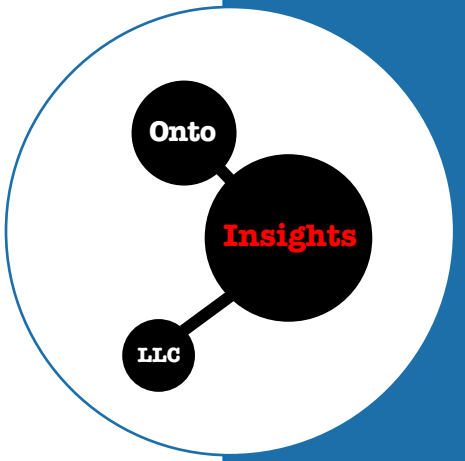


Andrea Westerinen
andrea@ontoinsights.com, arwesterinen@gmail.com

1 November 2023

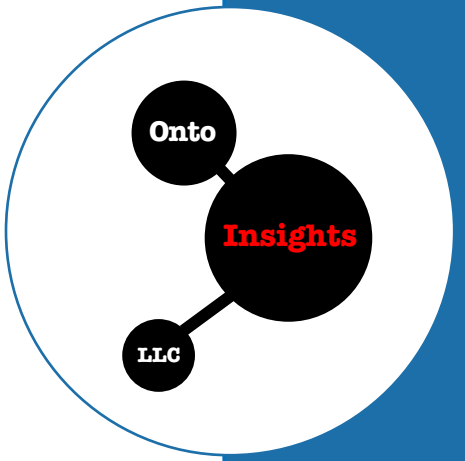
Abstract

- In the past, populating ontology-based Knowledge Graphs (KGs) from unstructured text involved convoluted natural language analyses and custom code
- World has changed with the use of Large Language Models (LLMs)
- This talk explores one use case: population of a knowledge database from news article texts
  - Evolution of the Deep Narrative Analysis (DNA) application from employing spaCy APIs to OpenAI

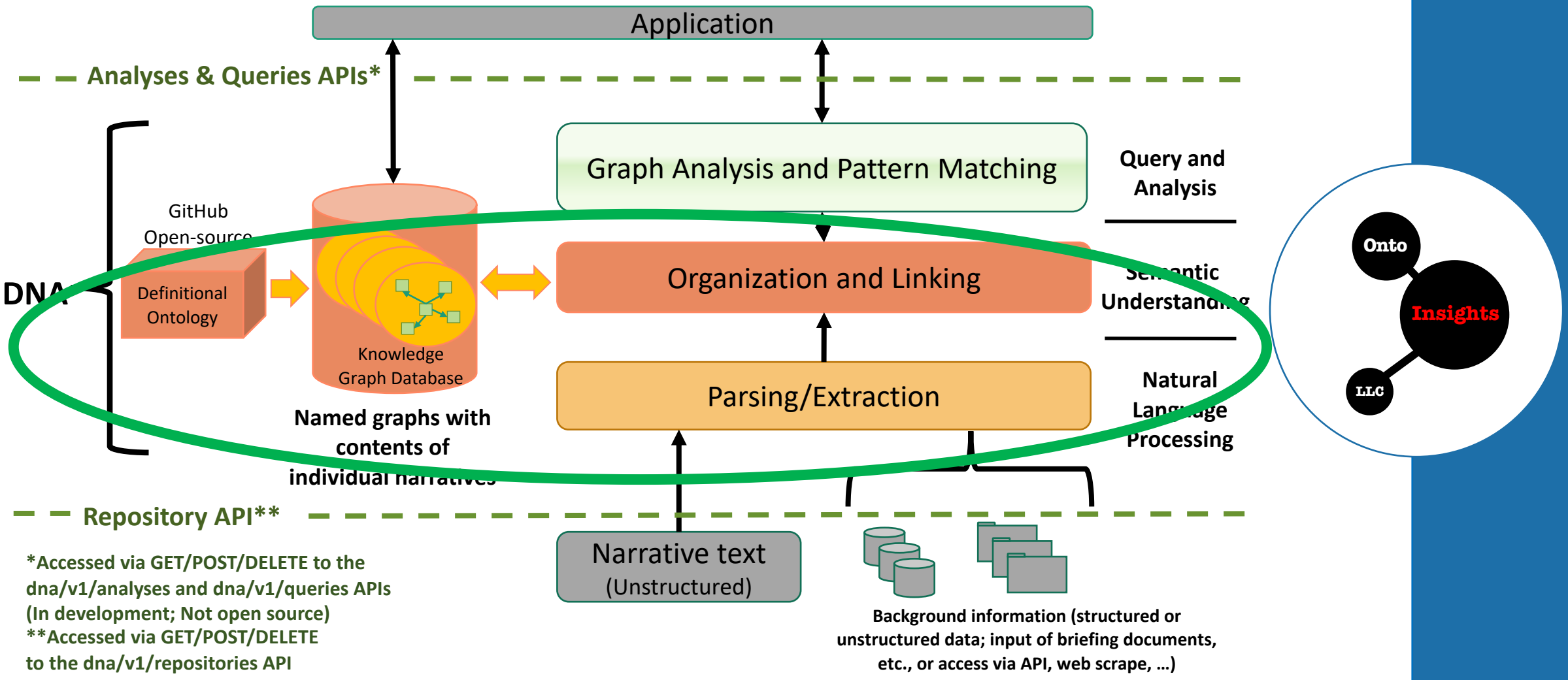


Introduction

- Deep Narrative Analysis (DNA)
  - Research prototype designed to:
    - Create knowledge stores with data from text stored in RDF graphs
    - Enabling aggregation of textual information within and across documents
    - To efficiently compare and analyze collections of text
    - To understand patterns, trends, …
  - Use cases:
    - Aid readers of news (in understanding how the reporting is “tuned”)
    - Investigate/discover mis-/dis-information “flags”
- Features:
  - Detailed text parse and graph storage for both drill-down and aggregate analyses
  - Inclusion of definitional and contextual background information to aid readers of news and for reasoning/inference



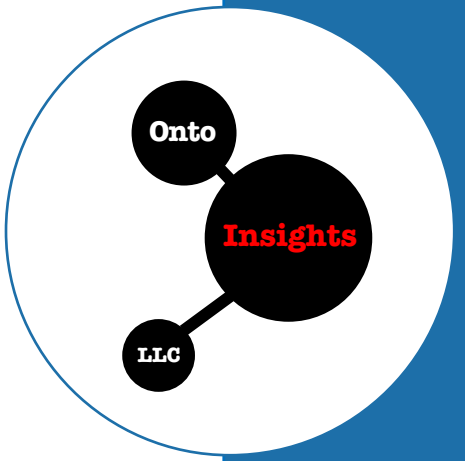
DNA Architecture



*Accessed via GET/POST/DELETE to the dna/v1/analyses and dna/v1/queries APIs (In development; Not open source)
**Accessed via GET/POST/DELETE to the dna/v1/repositories API

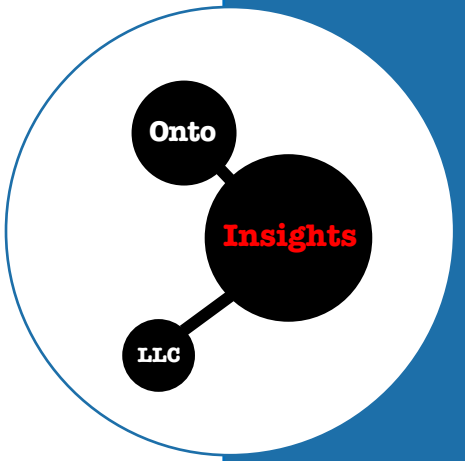
DNA Architecture – Foundational Technologies

- Linguistic event theory, built into the DNA Ontology
  - Closely aligned with linguistic and syntactic patterns underlying LLMs
- Natural language processing
  - spaCy for named entity recognition, and extraction of quotations and sentences
  - LLMs for analysis of basic linguistic details, use of rhetorical devices and semantic role labelling, alignment with the ontology concepts and interpretations of a narrative by viewpoint
- Other technologies, not further discussed:
  - Ontological reasoning and inference
  - Graph analysis and machine learning and pattern recognition

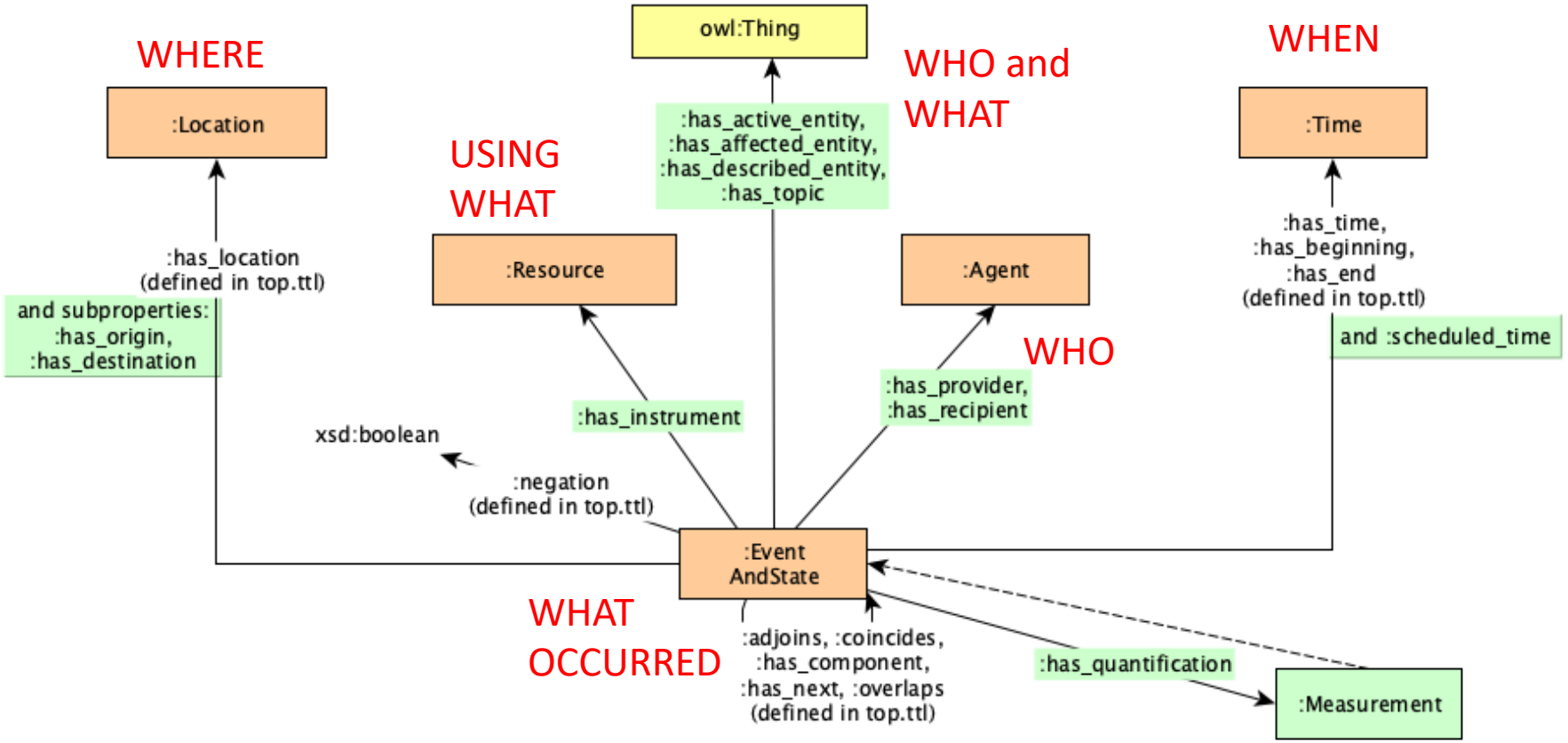


DNA Usage via REST APIs

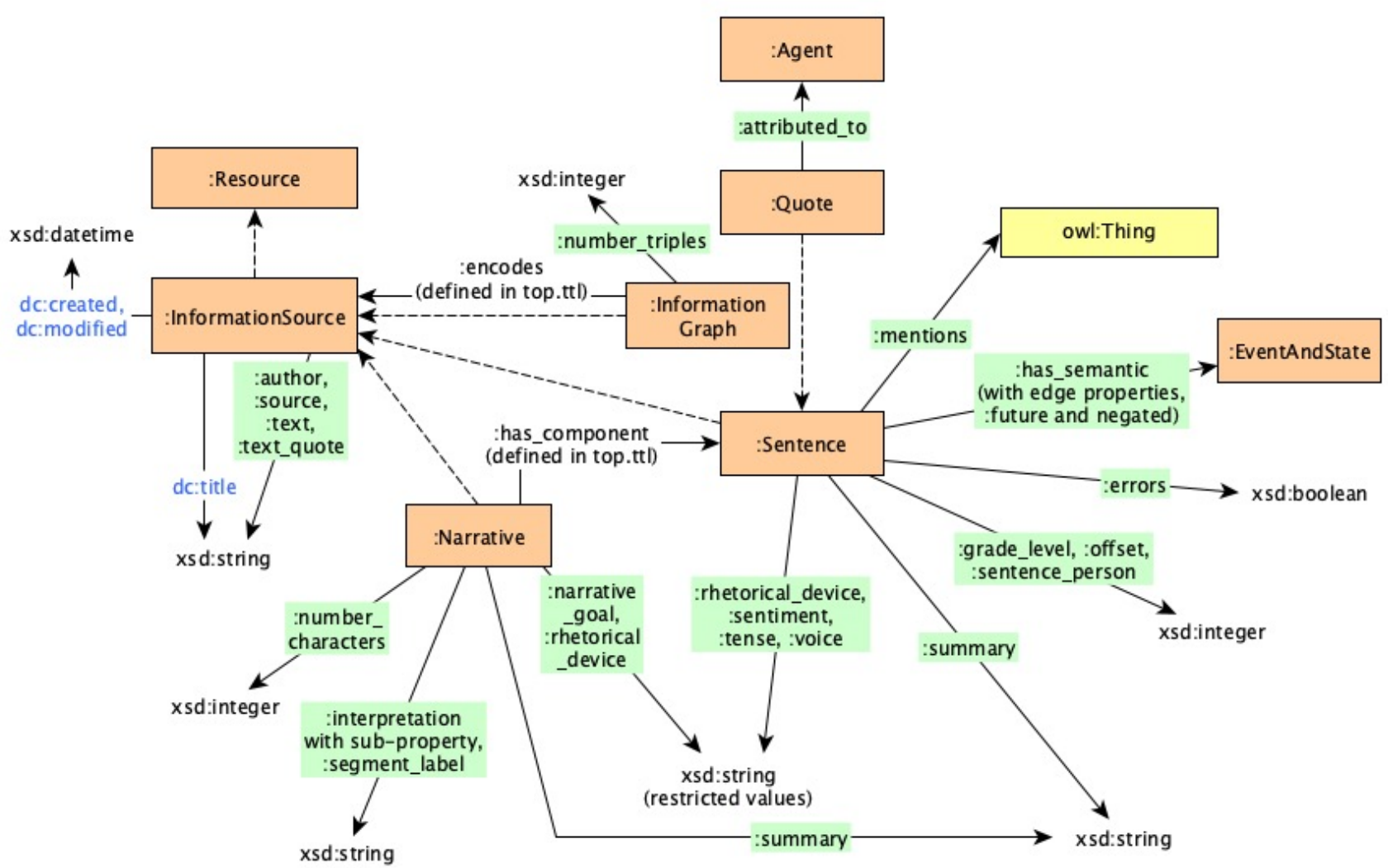
- DNA API definition
- dna/v1/repositories to create, delete or list the repositories of text in the database
- dna/v1/news to retrieve or retrieve/ingest a list of articles from newsapi.org
  - "Ingest" => parse, retrieve linguistic and ontology details, create KG and store the triples to specified repository
- dna/v1/narratives to ingest text provided in the API request
- dna/v1/narratives/graphs to retrieve the KG for an article/narrative for use in analysis or for review/edit, or to update one
- (In development) Analysis using graph/ML/pattern recognition tooling
  - Basic examples in the paper, Translating Narratives to Knowledge Graphs

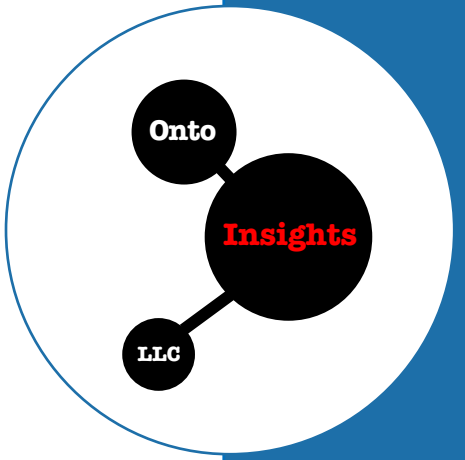


Core Concepts in the DNA Ontology



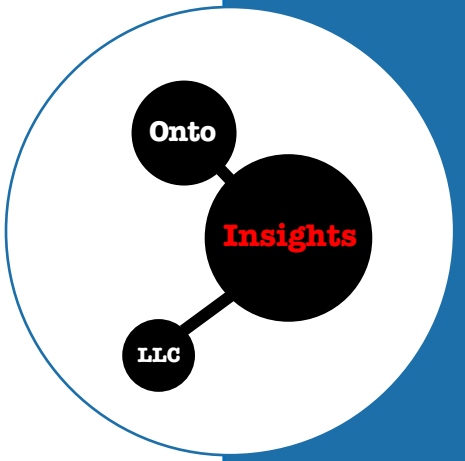
Narratives and News in the Ontology





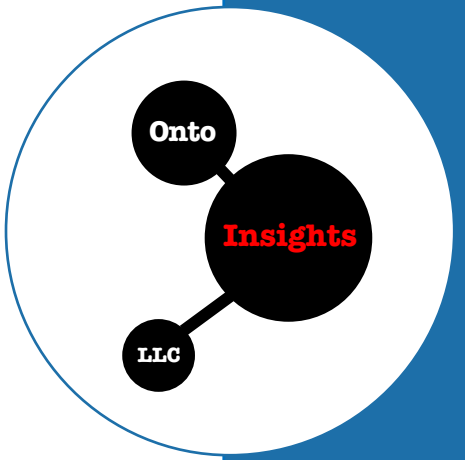
Development History 1

- Early versions based on spaCy, WordNet/FrameNet, dictionaries, …
- Supported basic parse, ontology alignment and KG creation/storage
- Sentiment analysis difficult
- Attempted to address multi-lingual WordNets, RDF language tags, …
- Problems:
  - Inability to address complexity of human language programmatically
  - Difficulty, time and expense of creating/customizing training data

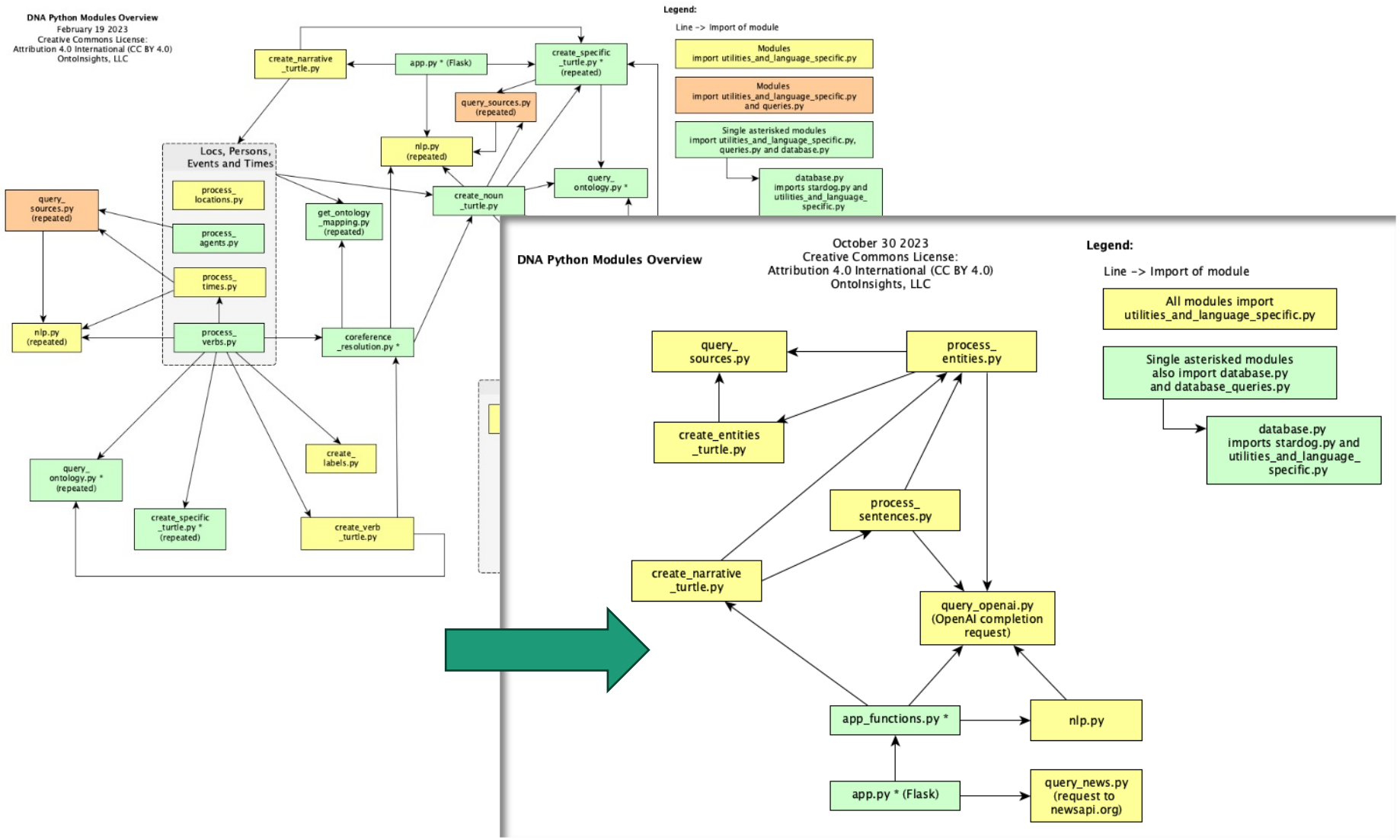


Development History 2

- Evolution to spaCy plus LLMs (OpenAI APIs)
- Value proposition:
  - Obtaining linguistic details (similar to DNA's basic parse) with sentiment analysis at the narrative and sentence levels
    - With much simpler code base! (>50% code size reduction with expanded capabilities)
  - Identifying rhetorical devices in text
  - Capturing entity relationships using semantic role labeling
  - Capturing text's possible goals and interpretations
  - Improving mapping of text to the ontology
    - No need to exhaustively incorporate and extend synsets and frames

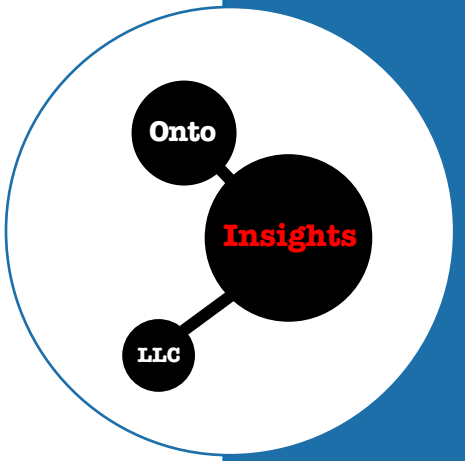


Code Evolution



Prompting

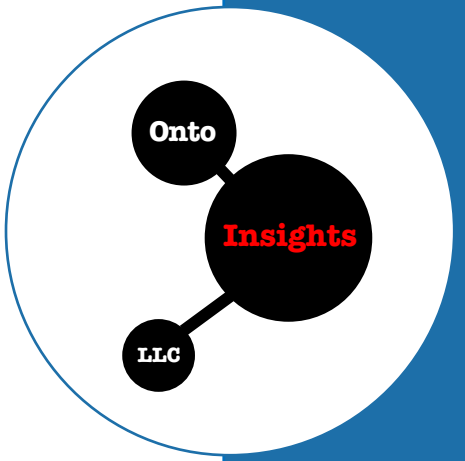
- All prompts in the file, query_openai.py, in dna directory
- Incorporates context, instruction and multiple-choice techniques
- Designed as a series of specific prompts (“chained” via the code)
  - Narrative analysis
    - Rhetorical devices and viewpoint interpretations
  - Sentence analysis
    - Linguistics (tense, voice, errors, …)
    - Rhetorical devices
    - Event ontology mapping
- Designed to return JSON results that are easily mapped to the “top elements” in the ontology hierarchy



Prompt Example

- Context: You are a linguist and NLP expert, analyzing quotations from news articles.
- Input: Here is the text of a quotation from an article (ending with the string "**" which should be ignored): {quote_text} **
  - ** distinguish where input to be analyzed ends
- Specific instruction: For the text, indicate its sentiment ("positive", "negative" or "neutral") and create a summary in 8 words or less. Indicate the grade level that is expected of a reader to understand its semantics.
- Output: Return the response as a JSON object with keys and values as defined by {quote_format1}.

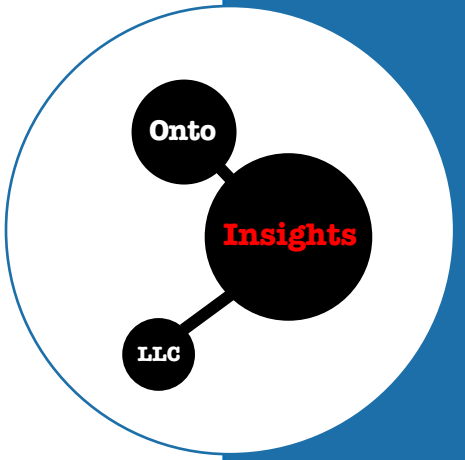
```
quote_format1 = '{"sentiment": "string", ' \
                '"grade_level": "int", ' \
                '"summary": "string"}'
```



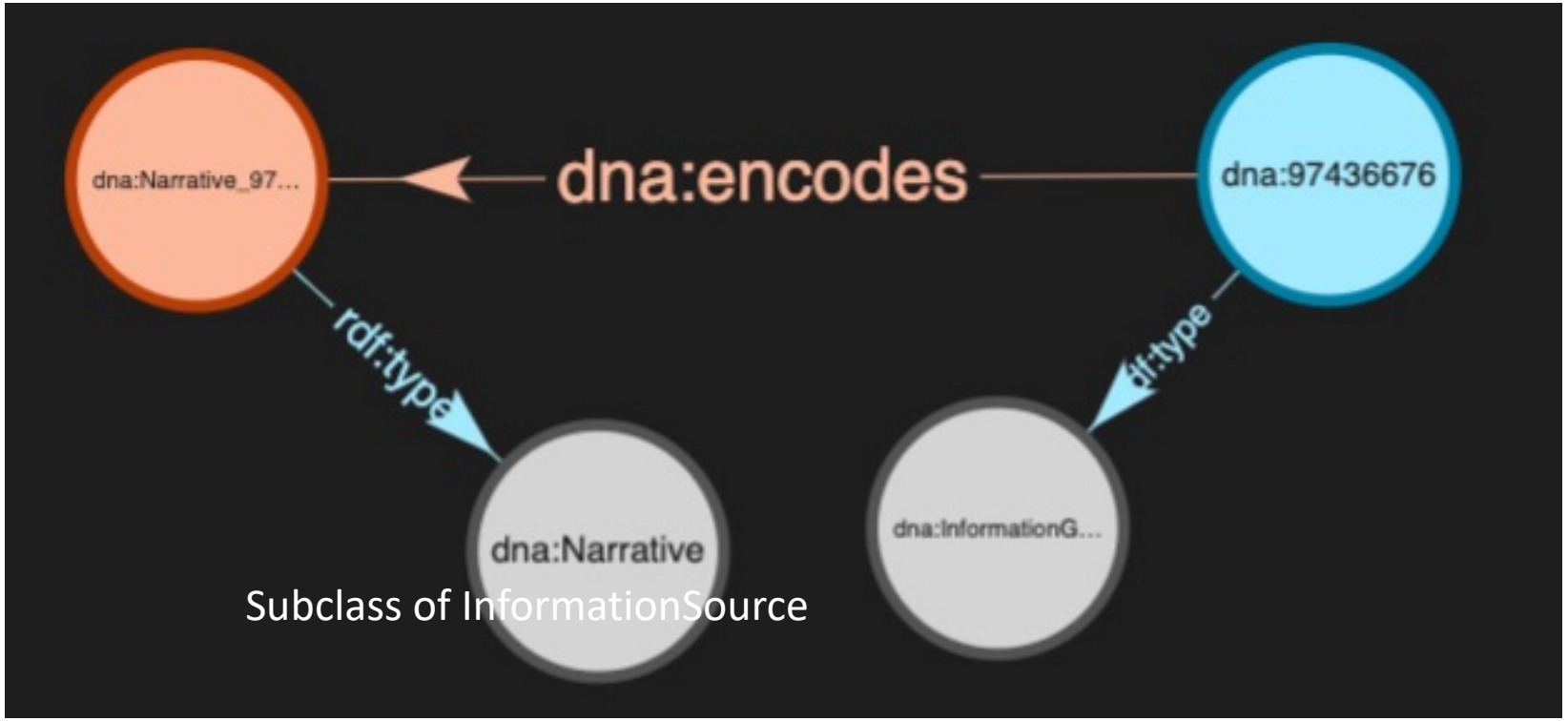
Example – Narrative Input

- Narrative input via POST to the DNA RESTful API

```
def test_narratives_post_ok1(client):
    article_text = "John is a musician. When Mary goes to the grocery store, John practices guitar."
    req_data = json.dumps({
        "narrativeMetadata": {
            "title": "A narrative title",
            "published": "2022-07-14T17:32:28Z",
            "source": "Wall Street Journal",
            "url": "http://some-site.com/narrative",
            "length": len(article_text)},
        "narrative": article_text
    })
```

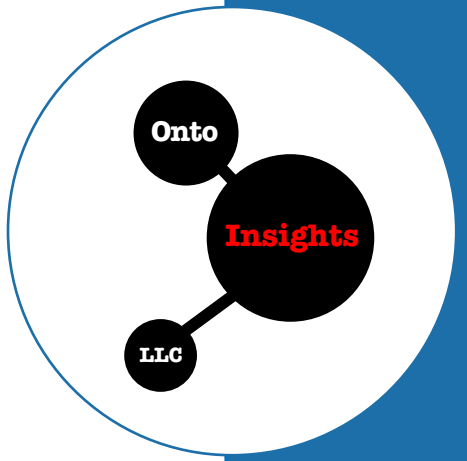


Example – Narrative-Level Graph



dna:97436676 **rdf:type** dna:InformationGraph **dna:number_triples** 47 **dc:created** 2023-11-01T03:18:43

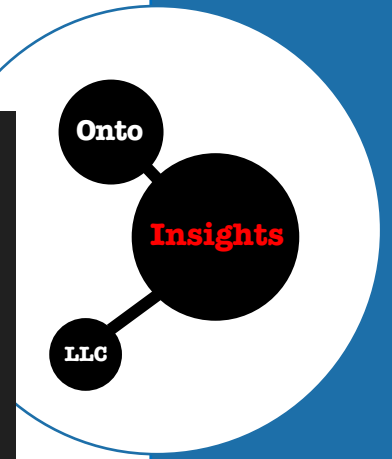
dna:Narrative_97436676 **rdf:type** dna:Narrative **dna:source** "Wall Street Journal"
dna:text "John is a musician. When Mary goes to the grocery store, John practices guitar." **dna:narrative_goal** "describe-current", "life-story"
dna:summary "The narrative describes the routine of John, a musician, who practices guitar when Mary goes to the grocery store."
dna:external_link "http://some-site.com/narrative" **dna:number_characters** 79 **dc:created** 2022-07-14T17:32:28Z **dc:title** "A narrative title"



Example – Sentence-Level Graph

- Sentences from the narrative stored in the InformationGraph
- Each narrative's graph is stored its own named graph
- Second sentence from the example: When Mary goes to the grocery store, John practices guitar.

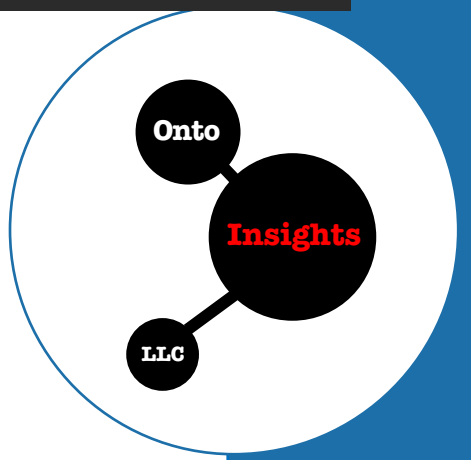
```
:Sentence_80c8d290-63b7 a :Sentence ; :offset 2 .
:Sentence_80c8d290-63b7 :text "When Mary goes to the grocery store, John practices guitar." .
:Mary a :Person .
:Mary rdfs:label "Mary" .
:Mary rdfs:comment "Needs disambiguation; See the web site, https://en.wikipedia.org/wiki/Mary" .
:Mary :gender "female" .
:Sentence_80c8d290-63b7 :mentions :Mary .
:Sentence_80c8d290-63b7 :mentions :John .
:Sentence_80c8d290-63b7 :sentence_person 3 ; :sentiment "neutral".
:Sentence_80c8d290-63b7 :voice "active" ; :tense "present" ; :summary "Mary shops, John practices guitar" .
:Sentence_80c8d290-63b7 :grade_level 2 .
:Sentence_80c8d290-63b7 :has_semantic :Event_cc50d721-d839 .
:Event_cc50d721-d839 a :MovementTravelAndTransportation .
:Event_cc50d721-d839 :has_active_entity :Mary .
:Event_cc50d721-d839 :has_location [ :text "grocery store" ; a :Location ] .
:Sentence_80c8d290-63b7 :has_semantic :Event_ea2c7644-3ff0 .
:Event_ea2c7644-3ff0 a :ArtAndEntertainmentEvent .
:Event_ea2c7644-3ff0 :has_active_entity :John .
:Event_ea2c7644-3ff0 :has_instrument [ :text "guitar" ; a :MusicalInstrument ] .
```



Example – Rhetorical Devices

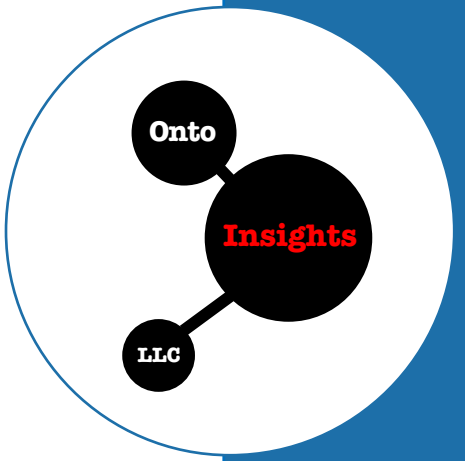
```
# Output:
# :Sentence_0139e4a2-94d0 a :Sentence ; :offset 1 .
# :Sentence_0139e4a2-94d0 :text "U.S. Rep. Liz Cheney conceded defeat Tuesday in the Republican primary
#      in Wyoming, an outcome that was a priority for former President Donald Trump as he urged GOP voters
#      to reject one of his most prominent critics on Capitol Hill." .
```

```
# :Sentence_0139e4a2-94d0 :mentions geo:6252001 .
# :Sentence_0139e4a2-94d0 :mentions :Liz_Cheney .
# :Sentence_0139e4a2-94d0 :mentions :Tuesday .
# :Sentence_0139e4a2-94d0 :mentions :Republican .
# :Sentence_0139e4a2-94d0 :mentions :Wyoming .
# :Sentence_0139e4a2-94d0 :mentions :Donald_Trump .
# :Sentence_0139e4a2-94d0 :mentions :GOP .
# :Sentence_0139e4a2-94d0 :mentions :Capitol_Hill .
# :Sentence_0139e4a2-94d0 :sentence_person 3 ; :sentiment "negative".
# :Sentence_0139e4a2-94d0 :voice "active" ; :tense "past" ; :summary "Cheney loses Wyoming primary, Trump\'s
#      priority achieved." .
# :Sentence_0139e4a2-94d0 :grade_level 12 .
# :Sentence_0139e4a2-94d0 :rhetorical_device {:evidence "The text refers to authority figures such as U.S. Rep.
#      Liz Cheney and former President Donald Trump."}  "ethos" .
# :Sentence_0139e4a2-94d0 :rhetorical_device {:evidence "The text refers to a specific event, the Republican
#      primary in Wyoming, to engage the reader."}  "kairos" .
# :Sentence_0139e4a2-94d0 :rhetorical_device {:evidence "The text uses wording that appeals to emotions
#      such as the defeat of Liz Cheney and the victory being a priority for Donald Trump."}  "pathos" .
# :Sentence_0139e4a2-94d0 :has_semantic :Event_088aad21-e7f0
```



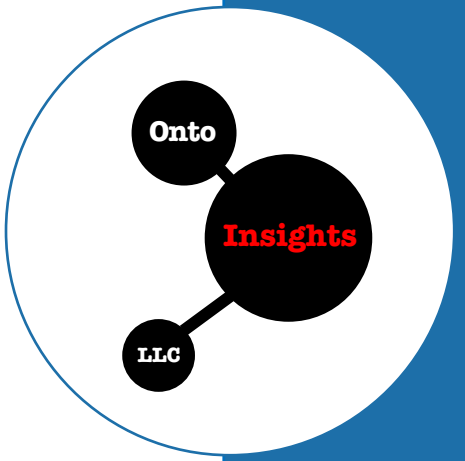
Findings

- Successful POC => Improved parse and ontology creation correctness with LLMs
  - With less code and complexity
- Importance of exception handling and programmatic validation of results
  - Expect strange behaviors such as the sentence, “Joe is conservative”, having no verb (!)
  - LLMs generate unexpected information that is not accounted for in the JSON format, and happily return text
  - => Requires further tuning of the prompts + validation
- When prompting:
  - Simplicity is best but not always possible
  - Keep all inputs/instructions/outputs together
  - Use consistent terminology



Findings and Research

- Inconsistencies occur but overall parse accuracy increased
  - Accuracy could be improved using self-consistency (repeated) prompting
  - At the expense of cost and performance time
- Large prompts certainly affect performance time
  - And, OpenAI load influences response time
  - Good news is that OpenAI does handle “large” prompts
- Continuing research and tuning improvements – for example:
  - Updated and expanded tests
  - Continued refinement of prompts and validation of results
  - Improved co-reference resolution across sentences
  - Move to a private/local LLM implementation
  - Deeper ontology drill-down
  - Investigation of analysis level (full article + sentences today)



Backup Slides

