< Preprocessing Unstructured Data for LLM Applications >

Data Preprocessing and LLMs

Retrieval Augmented Generation (RAG)

→ A technique for grounding LLM responses on validated external information

Contextual Injection

→ RAG apps load context into a database, then retrieve context to insert into a prompt

Preprocessing Output

Document Content : Text content from the documents. Used for keyword or similarity search in RAG apps.

< Document Elements >

The basic building blocks of a document. Useful for various RAG tasks, such as filtering and chunking.

- Title
- Narrative Text
- List Item
- Table
- ~~Image~~

< Element Metadata >

Additional information about an element. Useful for filtering in hybrid search and for identifying the source of a response

- Filename
- Filetype
- Page number
- Section

Why is Data Preprocessing hard?

- Content Cues → (html, visual, metadata)

  Different document types → different cues for element types

- Standardization Need

  To process content from different document types

  ↳ they need to be standardized

- Extraction Variability

  Different document formats may require different extraction approaches (forms vs journal articles)

- Metadata Insight

  In many cases, extracting metadata requires an understanding of document structure.

Normalizing Diverse Document

Format Diversity : Documents come in a variety of formats

Common Format : Preprocessing's first step : Convert raw document → Common format → available to be identify Common document elements.

Normalization Benefit : Normalized format allows any document to be processed in the same way, regardless of the source format.

- Filtering out unwanted elements, like headers & footers
- chunk document : by section.

Reduced Processing Section : The initial document preprocessing step is the most expense part of the process

- downstream tasks like chunking are inexpensive operations on normalized outputs
- Enables experiments with many different chunking techniques without reprocessing documents

Data Serialization

Serialization Benefits : Serializations allow the results of documents preprocessing to be used again later

Advantages of JSON

- Structure is common and well understood
- Is a standard HTTP response
- Able to be used in multiple programming L
- Can be converted to JSONL for Streaming use cases

What is Metadata?

Metadata: Is additional information that we extract while we're pre-processing the document

can be at the document level

at the element level

or can be information itself we extract.

Documents details : provides additional information about content extract from source - document.

Source Identification : One type of metadata is information about the document itself.

Structural Metadata : Metadata be Constructed from the structure of the document.

Search Enhancement : In RAG systems, metadata provides filtering options for hybrid search.

Semantic Search for LLMs

Semantic search with vector Databases

- Goal: Given an input text, find semantically similar context from a corpus of documents for use in prompt templates
- Embedding: Convert text to vectors that can be compared through a similarity function, such as cosine similarity.
- Vector DB: A DB optimized for performing similarity search.
- Prompt Template: Insert relevant contexts into a template to generate a prompt for the LLM. often context semantically similar to an input query.

Load: Insert the vectors into the DB, along with the source documents or a pointer to the source documents

Query Embed: Embed the input for the similarity search

Compare and Retrieve: Compare the query embeddings to documents in the vector DB; retrieve the k most similar documents.

Document Image Analysis

Document Image Analysis (DIA)

: Allows us to extract formatting information and text from the raw image of a document

Preprocessing with Rules - based Parsers

: Many document types (html, pdf, word Docs) include formatting information ⇒ these documents can be preprocessed with rules - based parsers.

Visual Information: For other documents, such as pdfs and images, the formatting information is visual

DIA Methods

1) DLP (Document Layout Detection)

2) Vision Transformers (ViT)

DLP : Uses an object detection model to draw and label bounding boxes around layouts elements on a document images.

ViT : input : document image.

output : text representation of a structured output.

Document Layout Detection

- Vision Detection : Identify and classify bounding box using a computer vision model
- Text Extraction : Extract text from the bounding box using object character recognition (OCR) when necessary
- Direct Extraction : For some documents like pdfs, text can be extracted directly from the document without OCR.

Vision Transformer

- Visual Transition : input → encoder → decoder → output
- DONUT Architecture : Document understanding Transformer is common Architecture
- Direct Conversion : OCR : Not Required, "image → text"
- Structured Transf : output : JSON Str

Tabel Extraction

Text Analysis : Most RAG1 use cases focus on text contents within documents.

Structured Data : Some Industry deal heavily with structured data embedded in unstructured documents.

Table Extraction : To support use cases such as question answering over tables, it is helpful to extract tables from documents.

Inherit Strudures : Some documents (HTML, Word docs) : contain table - structure info

Inference Required : Some documents (PDF, images) : table information needs to be inferred.

Techniques : Table-Transformer, ViT, OCR Postprocessing

Table - Transformer : model that identifies bounding boxes for table cells and converts the output to HTML

Two - Steps : 1) Identify tables using the document layout model

2) run the table through the table transformer.

⊕ : Can trace cells back to the original bounding boxes

⊖ : multiple expensive model calls

Vision Transformer

⊕ : allows for pompting, more flexible, one model call

⊖ : generative and prone to hallucination : no bounding boxes

OCR Postprocessing

OCR the table, then build the table structure based on patterns in the OCR output.

⊕ : fast; accurate for well behaved tables

⊖ : requires statistical or rules based parsing
less flexible than other approaches : no link back to bounding boxes in images