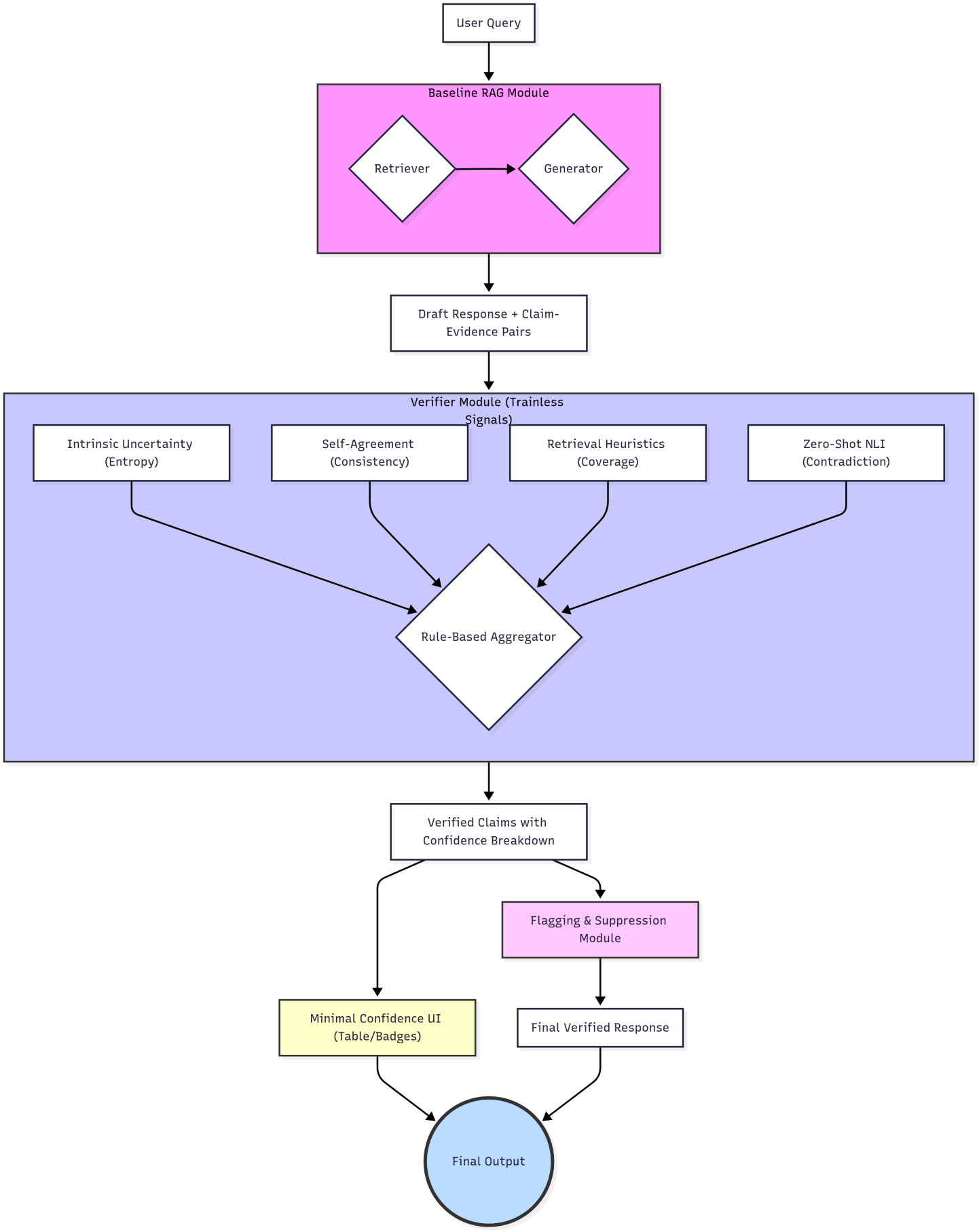
LWW2502 - Progress Presentation

Date: 2025-11-10

Part 1: System Architecture Design Snippets



Part 2: RAG Pipeline & Demo

- **Inputs:**
  - `user_query` : (string) The input prompt from the user.
- **Process:**
  i. **Retrieve:** The retriever fetches relevant documents.
  ii. **Generate:** The generator LLM produces a draft response, while capturing token-level metadata (e.g., logits for entropy calculation).
  iii. **Decompose & Pair:** The draft is decomposed into atomic claims, creating direct `(claim, evidence)` pairs.
- **Outputs:**
  - `draft_response` : (string) The full, unverified draft response.
  - `claim_evidence_pairs` : (List[dict]) A list where each dictionary contains the `claim` , the `evidence` document, and `generator_metadata` .
- **Demo**
  - Sample Query:

```
# Define sample queries
sample_queries = [
    "What is artificial intelligence?",
    "How do machines learn from data?",
    "What is deep learning?",
    "What is natural language processing?"
]
```

  - Retrieve

  i. Top k Evidence

```
Top Retrieved Evidence:
 [1] (Score: 0.8676, Rank: 1)
     Doc: wiki_00000278#0
     Text: Artificial intelligence (AI) is the intelligence of machines or software, as opposed to the intelligence of humans or animals....
```

```
  [2] (Score: 0.7846, Rank: 2)
      Doc: wiki_00002500#70
      Text: Artificial intelligence

Artificial intelligence (AI) involves the study of cognitive phenomena in machines....
```

```
[3] (Score: 0.7744, Rank: 3)
    Doc: wiki_00001201#0
    Text: AI is artificial intelligence, intellectual ability in machines and robots....
```

ii. Claims

```
Claim 1:
  ID: 36f6172a-b424-44b0-a11c-5324588175f4
  Top Evidence: wiki_00000278#0
  Evidence Candidates: 5 chunks
  Evidence Spans Available: 5 chunks
```

- Generate

```
📝 Generated Response:
  the intelligence of machines or software
```

Part 3: Verifier Module

- **Inputs:**
  - `claim_evidence_pairs` : (List[dict]) The list of `(claim, evidence)` pairs with generator metadata.
- **Process & Sub-components:**
  i. For each `(claim, evidence)` pair, the following sub-components run in parallel:
    - **Intrinsic Uncertainty:** Analyzes `generator_metadata` .
      - **Output:** `entropy_score` (e.g., length-normalized negative log-likelihood).
    - **Self-Agreement:** If enabled, generates `k` response samples.
      - **Output:** `consistency_score` (e.g., variance or disagreement across samples).
    - **Retrieval-Grounded Heuristics:**
      - **Process:** Calculates `evidence_coverage` (percentage of claim entities in evidence) and `citation_integrity` (token overlap for cited spans).
      - **Output:** A dictionary of heuristic scores.
    - **Zero-Shot NLI:** Uses an off-the-shelf NLI model.
      - **Process:** Labels each `(claim, evidence_sentence)` pair as Entail/Contradict/Neutral.
      - **Output:** Aggregated NLI scores (e.g., `max_contradiction_prob` , `entailment_ratio`).
  ii. **Rule-Based Aggregator:** Gathers all signals into a structured breakdown using explicit rules. No trainable fusion logic is used at this stage.
- **Outputs:**
  - `verified_claims` : (List[dict]) A list where each dictionary contains:

- `claim` : (string) The original atomic claim.
- `evidence` : (dict) The associated evidence.
- `confidence_breakdown` : (dict) A structured dictionary containing all raw signals (e.g., `entropy_score` , `nli_results` , `coverage_score`).
- `final_verdict` : (string) A final verdict (e.g., "Supported", "Contradictory", "Low Confidence") derived from the rule-based aggregator.