**🧠 Knowledge Graph + Personality Modelling Assessment Report**

**1️⃣ Synthetic Data Generation**

Since no real data was provided, **synthetic data** was programmatically generated to emulate a 2-page narrative of professional collaboration between domain experts (Dr. Maya Thompson, Dr. Omar Khalid, Prof. Aisha Mwangi, etc.).

**How & What Generated:**

* Context-rich text with realistic entities (PERSON, ORG, LOC, DATE, EVENT) and personality cues (adjectives like *analytical*, *pragmatic*).
* Linguistic variation and co-occurrence patterns ensured natural entity diversity and relationship density.

**Justification:**  
This synthetic dataset captures *semantic richness* and *context diversity* essential for validating entity extraction, coreference resolution, and trait attribution — without using private data. It ensures **reproducibility** and **ethical safety** for LLM training or graph construction.

**2️⃣ Model Evaluation and Metrics**

The pipeline can be evaluated using **hybrid linguistic + structural metrics**:

| **Stage** | **Metric** | **Purpose** |
| --- | --- | --- |
| Entity Recognition | F1-score, Precision, Recall | Accuracy of extracted entity boundaries |
| Relation Extraction | F1-score / Micro-average F1 | Relation labeling consistency |
| Graph Quality | Graph Density, Modularity, Avg. Degree | Structural coherence of KG |
| Personality Attribution | Lexicon Match Accuracy, Human Consistency | Reliability of hybrid rule-based + lexicon approach |

**Justification:**  
Combines **linguistic accuracy** (for NER/RE) with **graph topological soundness**, giving both semantic and structural validity.

**3️⃣ Implementation and Personality Representation**

**Implementation Steps:**

1. Preprocessing — normalization, lemmatization, and sentence segmentation.
2. Entity & Relation Extraction — spaCy + dependency-based heuristics.
3. Entity Normalization — alias merging via Levenshtein similarity.
4. Personality Extraction — hybrid rule-based + lexicon-based (e.g., NRC, LIWC).
5. Graph Construction — NetworkX nodes (entities/traits) + edges (relations / has\_trait).
6. Validation — schema, connectivity, duplication checks.

**Representation Choices:**

* Each **person** node has attribute edges to **trait nodes** via has\_trait.
* Traits modeled as separate nodes, not attributes, to support graph reasoning.

**Justification:**  
This design ensures **interpretability**, **queryability**, and **future extensibility** (e.g., personality clustering, trait influence analysis).

**4️⃣ LLM Workflow Pipelining**

The workflow was designed as a **multi-prompt chain**:

1. **Prompt 1 – Text Preprocessing & Context Understanding**  
   → LLM cleans and segments document into coherent sentences.
2. **Prompt 2 – Entity & Relation Extraction**  
   → Uses transformer / spaCy pipeline for structured triples.
3. **Prompt 3 – Personality Inference**  
   → Extracts adjectives & lexicon matches describing entities.
4. **Prompt 4 – Graph Construction & Validation**  
   → Builds and verifies schema-consistent graph in NetworkX.
5. **Prompt 5 – Reporting & Visualization**  
   → Outputs normalized entities, relations, and traits for export.

**Justification:**  
Chaining prompts modularizes the reasoning steps, improving **traceability**, **error recovery**, and **explainability** — essential for complex reasoning workflows.

**5️⃣ Data Processing and Normalization**

**Normalization Steps:**

* Case folding + punctuation removal.
* Stopword preservation (to retain context for relations).
* Alias merging using SequenceMatcher similarity ≥ 0.75.
* Entity-type standardization (PERSON, ORG, LOC, etc.).

**What Not Normalized:**

* Proper nouns and personality phrases (to maintain semantic richness).

**Justification:**  
Selective normalization ensures data is **consistent** yet **contextually meaningful** — maintaining linguistic fidelity critical for NLP/LLM downstream tasks.

**✅ Demonstrated Competencies**

* **LLM Workflow Understanding:** Sequential modular pipeline with justifications.
* **Data Science Integration:** Preprocessing, normalization, metric-based evaluation.
* **Personality Modelling:** Adoption of hybrid lexicon + rule approach linked to KG graph.

**🔍 Limitations and Future Work**

**Limitations:**

* The **coreference resolution** used was heuristic (coreferee), which performs well for short texts but may miss long-range entity references in multi-page documents.
* The **rule-based personality extraction** depends on limited lexicon coverage (NRC, LIWC-style lists). It might miss implicit traits (e.g., inferred from actions rather than adjectives).
* Relation extraction relied on **syntactic heuristics** (verb-based dependency pairs), which can produce incomplete edges in complex or nested clauses.
* Some **isolated nodes** and **nonstandard entity types** (like time phrases) require manual post-validation to maintain schema consistency.

**Future Work:**

* Replace heuristic coref with a **transformer-based model** (e.g., SpanBERT or AllenNLP coref) for long-context resolution.
* Extend the personality lexicon using **LLM-based embedding similarity** to dynamically infer personality traits beyond fixed rule sets.
* Integrate a **relation classification model** (e.g., spacy-REBEL, T5-RE) for more accurate triplet extraction.
* Automate KG validation with ontology constraints (e.g., Person → has\_trait → Trait only).
* Deploy the workflow as a **multi-stage LLM chain** using LangChain / Azure PromptFlow to handle reasoning, extraction, and validation sequentially.