

1. Deskripsi Tugas Besar

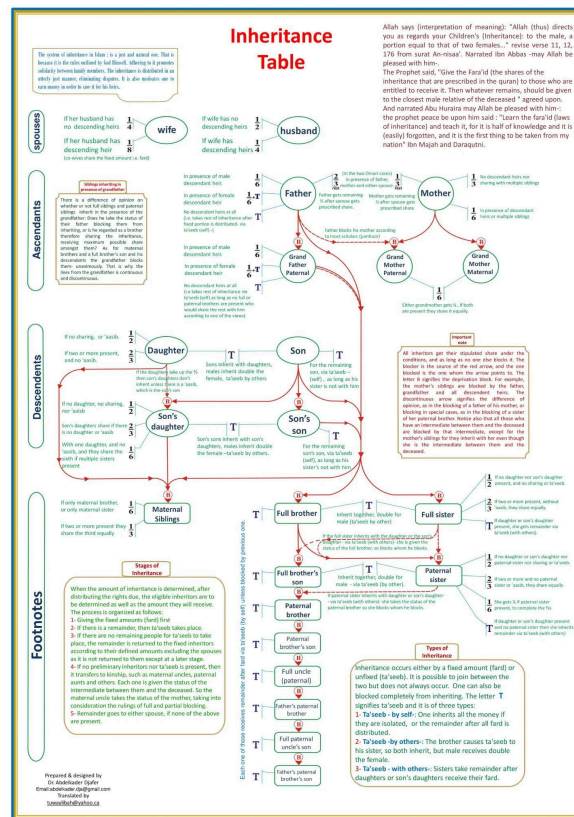
Secara garis besar, tugas besar ini dibagi menjadi dua, yaitu pengembangan sistem pipeline chatbot LLM berbasis RAG dan pengembangan/desain database berbasis graf untuk desain graf pengetahuan yang digunakan sebagai pengetahuan dasar dari RAG.

2. Kontribusi

Saya akui pada tugas besar ini **saya hampir tidak berkontribusi** ke aspek pengembangan sistem utamanya, tapi saya **hanya kontribusi ke interface frontend** sehingga saya kurang mendapatkan pengalaman dalam pengembangan sistem pipeline RAG pada LLM dan desain graf pengetahuan menggunakan database berbasis graf.

3. Penjelasan Perbaikan Kontribusi

Tentunya saya akan mencoba melakukan desain/pengembangan ulang sistem pipeline RAG pada LLM dengan batasan tertentu agar tugas perbaikan ini tidak begitu kompleks. Batasan pada implementasi ini pada proses desain graf pengetahuan dan desain queries yang akan digunakan. Referensi utama domain dalam pengembangan graf pengetahuan dan queries tertera pada gambar Inheritance Table berikut.



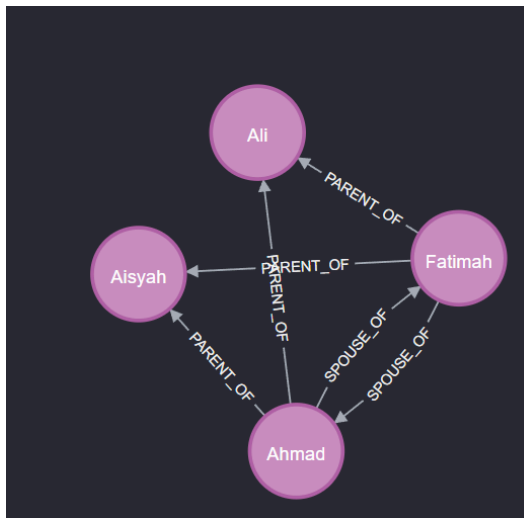
4. Batasan Perbaikan

- a. Desain graf pengetahuan digunakan untuk merepresentasikan pohon keluarga saja.
- b. untuk query dan pohon keluarga bersifat tetap berdasarkan database graf yang didesain.
- c. Tugas LLM hanya untuk mencari X mewariskan berapa dari Y, di luar itu LLM tidak bisa menjawab. Y adalah entitas berdasarkan relasi singular, jadi cuma bisa "children", "spouse".
- d. Input yang diterima sistem adalah pertanyaan dari user, konteks restriksi tugas LLM, jenis query untuk konteks tertentu, gambaran singkat terkait graf pengetahuan.
- e. Output hanya berupa informasi terkait pohon keluarga dan warisan persentase saja tanpa kalkulasi jumlah uang.
- f. Pohon keluarga terbatas hanya untuk keluarga kecil (keluarga utama) dan tidak ada handling kasus kompleks seperti poligami ataupun perceraian.
- g. Inferensi yang dilakukan hanya untuk kasus descendant (tidak termasuk cucu dan cicit) dan spouse saja. Tidak menangani kasus anak meninggalkan warisan ke siapa saja.
- h. Program hanya berbasis pada CLI.

5. Penjelasan Implementasi

a. Graf Pengetahuan (Pohon Keluarga)

Pada implementasi ini, saya menggunakan neo4j community edition karena open source dan setup cukup simpel dilakukan secara lokal. Saya melakukan desain pohon keluarga dengan basis node person {id, name} dan relasi berupa SPOUSE_OF dan PARENT_OF. Ilustrasi implementasi graf tertera pada gambar berikut.



Berdasarkan graf tersebut, pohon keluarga terdiri atas Ahmad (Ayah), Fatimah (Ibu), Ali (Anak laki-laki), dan Aisyah (Anak perempuan). Ini menjadi basis graf

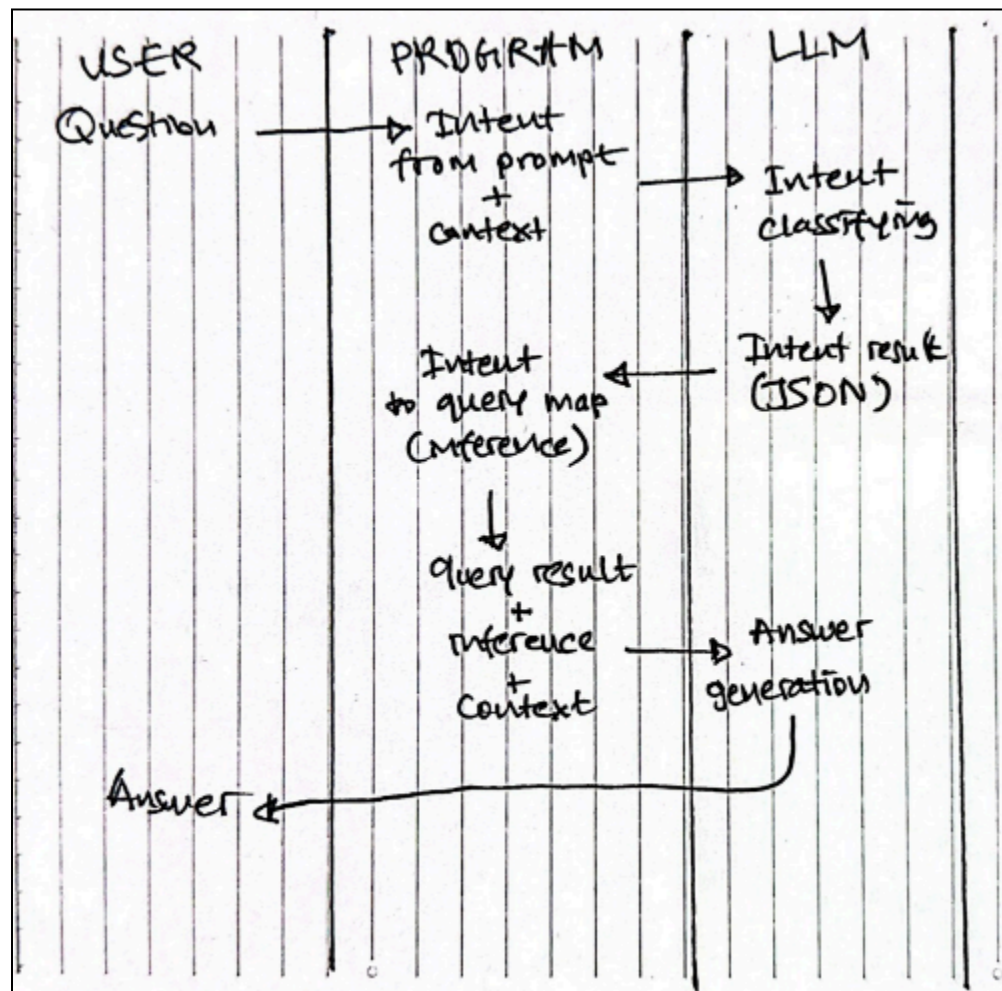
pengetahuan yang akan digunakan pada proses inferensi. Graf pengetahuan ini digunakan untuk mendapatkan konteks pohon keluarga dari X yang telah wafat yang tentunya akan ditentukan oleh LLM, dari konteks tersebut LLM akan membuat *intent* yang digunakan untuk menentukan predetermined queries.

Query yang didesain berdasarkan *intent* ini, kembali lagi, untuk mendapatkan data keluarga Y dari X. Contoh kasus yang ditangani:

1. "Ahmad (X) passed away, how much share does his spouse (Y) get?"
2. "Ahmad (X) passed away, how much share does his children (Y) get?"

b. Pipeline RAG

Pada implementasi ini, saya menggunakan openAI sebagai basis LLM. Desain alur simpel dari mekanisme pipeline RAG tertera pada gambar diagram berikut.



Terdapat 3 elemen utama pada diagram ini, user, program, dan LLM. Program dimulai dengan menerima input dari user, melakukan formatting pertanyaan dari user dengan tambahan konteks keperluan untuk intent classifying oleh LLM. Lalu, LLM melakukan intent classifying berdasarkan prompt sebelumnya dan menjawab dengan format strict JSON. Setelah program menerima jawaban intent

dari LLM, program melakukan pemetaan intent terhadap query yang sudah didesain untuk berbagai kasus intent. Lalu setelah itu, berdasarkan query dan intent dilakukan inferensi selanjutnya untuk menentukan persentase warisan dari X terhadap Y. Hasil query dan inferensi ini diberikan kepada LLM lagi untuk melakukan generasi jawaban ke user.