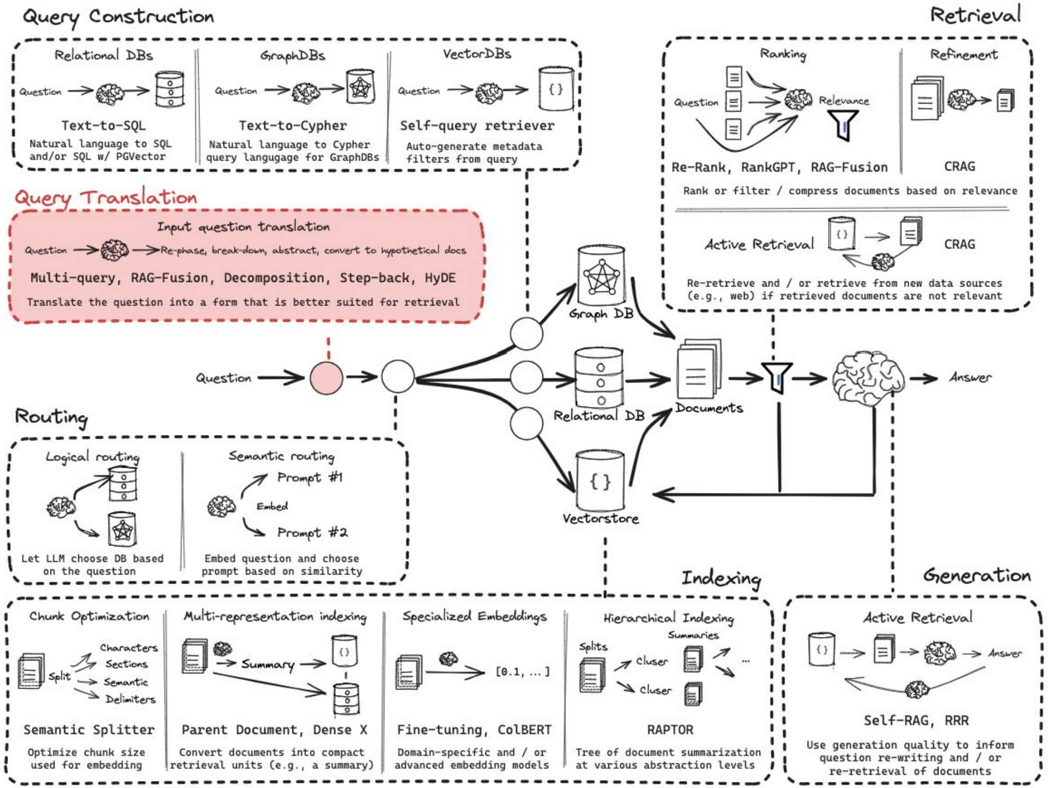
🦜🔗 LangChain

RAG from scratch: Query Translation (HyDE)

Lance Martin
Software Engineer, LangChain
@RLanceMartin

LangChain

Query Translation



General approaches to transform questions

3.1 Preliminaries

Dense retrieval models similarity between query and document with inner product similarity. Given a query q and document d, it uses two encoder function enc_q and enc_d to map them into d dimension vectors $\mathbf{v_q}, \mathbf{v_d}$, whose inner product is used as similarity measurement.

$$\mathrm{sim}(\mathrm{q}, \mathrm{d}) = \langle \mathrm{enc}_q(\mathrm{q}), \mathrm{enc}_d(\mathrm{d}) \rangle = \langle \mathbf{v_q}, \mathbf{v_d} \rangle \quad (1)$$

For zero-shot retrieval, we consider L query sets $Q_1, Q_2, ..., Q_L$ and their corresponding search corpus, document sets $D_1, D_2, ..., D_L$. Denote the j-th query from i-th set query set Q_i as q_{ij}. We need to fully define mapping *functions* enc_q and enc_d without access to any query set Q_i, document set D_i, or any relevance judgment r_{ij}.

The difficulty of zero-shot dense retrieval lies precisely in Equation 1: it requires learning of two embedding functions (for query and document respectively) into the *same* embedding space where inner product captures *relevance*. Without relevance judgments/scores to fit, learning becomes intractable.

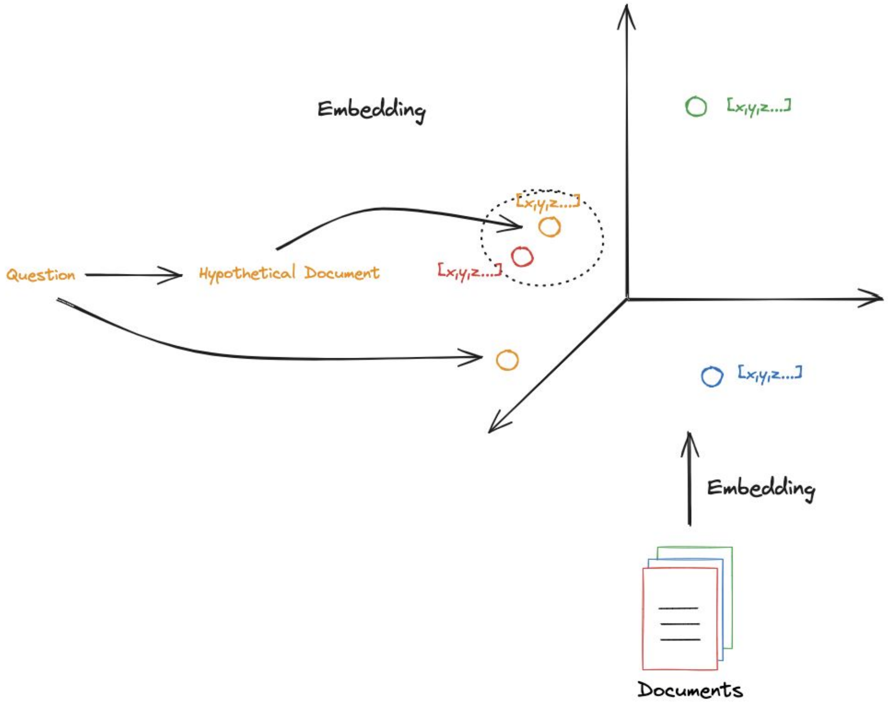
3.2 HyDE

HyDE circumvents the aforementioned learning problem by performing search in document-only embedding space that captures document-document similarity. This can be easily learned using unsupervised contrastive learning (Izacard et al., 2021; Gao et al., 2021; Gao and Callan, 2022). We set document encoder enc_d directly as a contrastive encoder $\mathrm{enc}_{\mathrm{con}}$.

https://arxiv.org/pdf/2212.10496.pdf

LangChain

Intuition



🦜🔗 LangChain

Code walk-through