Gpt2 - **0.84** (evaluate yg buat sendiri)

{'rouge1': 0.08033525853808088,

'rouge2': 0.03827190014059329,

'rougeL': 0.07886812502420418,

'rougeLsum': 0.07812743678967579}

{'precision': 0.7681157910823821,

'recall': 0.8902929902076722,

'f1': 0.823948609828949}

Qwen - unsloth/Qwen2.5-1.5B-Instruct - 0.46

{'rouge1': 0.19563380353338553,

'rouge2': 0.04879885701157348,

'rougeL': 0.18651224417416795,

'rougeLsum': 0.19022926820454467}

{'precision': 0.817308007478714,

'recall': 0.8378127408027649,

'f1': 0.8266139924526215}

Llama - unsloth/Llama-3.2-1B-Instruct - 0.34

{'rouge1': 0.06393612279206115,

'rouge2': 0.011435593592275622,

'rougeL': 0.060936961055714944,

'rougeLsum': 0.06066050118386275}

{'precision': 0.7838149988651275,

'recall': 0.8105958688259125,

'f1': 0.7958067572116851}

Seq2seq T5 - 0.**74**

{'rouge1': 0.7461016949152542, -> ini jg tinggi ini tertinggi

'rouge2': 0.18561403508771931,

'rougeL': 0.75,

'rougeLsum': 0.7533333333333333}

{'precision': 0.9521969032287597,

'recall': 0.9520050120353699,

'f1': 0.9515127944946289}

Bert - bert-large-uncased-whole-word-masking-finetuned-squad - **0.68**

{'rouge1': 0.5259939435233552, -> wow ini tinggi bgt :v

'rouge2': 0.1630626509086288,

'rougeL': 0.5238832866479926,

'rougeLsum': 0.5297824161922835}

{'precision': 0.7990534424781799,

'recall': 0.8371058690547943,

'f1': 0.8162295854091645}

Medical Assist

Inputnya apa aja?

1. **Symptom detection** (nigel) and **Precaution** (hanif)

<https://arxiv.org/pdf/2304.04052>

<https://medium.com/@qmsoqm2/auto-regressive-vs-sequence-to-sequence-d7362eda001e>

<https://arxiv.org/pdf/2312.10997>

<https://huggingface.co/medicalai/ClinicalBERT>

<https://github.com/saverymax/qdriven-chiqa-summarization/blob/master/data_processing/data/medinfo_collection.json>

Datasets:

* [www.kaggle.com/datasets/itachi9604/disease-symptom-description-dataset?select=dataset.csv](http://www.kaggle.com/datasets/itachi9604/disease-symptom-description-dataset?select=dataset.csv) (main)
* <https://github.com/itachi9604/healthcare-chatbot>
* [huggingface.co/datasets/duxprajapati/symptom-disease-dataset](http://huggingface.co/datasets/duxprajapati/symptom-disease-dataset)
* <https://huggingface.co/datasets/ruslanmv/ai-medical-chatbot> (main)
* <https://github.com/jind11/MedQA?tab=readme-ov-file> -> Open QA medical dataset, menarik buat diexplore. Ini juga ada referensi ke dataset medical sebelumnya yang isinya buat QA juga, tapi bukan open domain gt. Dokumennya beneran dari textbook medical gt.
* <https://www.kaggle.com/datasets/jpmiller/layoutlm?select=medquad.csv> -> QA dataset yang per topik kesehatannya udah rapih, enak buat dipake
* <https://github.com/abachaa/Existing-Medical-QA-Datasets?tab=readme-ov-file> -> existing med qa datasets
* <https://github.com/saverymax/qdriven-chiqa-summarization/tree/master> -> extractive QA dataset

Model yang sudah di fine tuned

<https://huggingface.co/ruslanmv/Medical-Llama3-8B>

<https://huggingface.co/sethuiyer/Medichat-Llama3-8B>

<https://www.kaggle.com/datasets/padmajabuggaveeti/disease-and-its-symptoms-precautions-riskfactors?select=disease_medicine.csv>

<https://github.com/mila-iqia/ddxplus>

Ide kalau symptom nya di luar scope bisa di tune question dengan symptom yg di luar scope trs jawabannya “itu di luar informasi..”

Approach:

* Model yg beda ? BERT, GPT, (decoder, encoder only or both)
* Teknik tuning

Evaluasi:

Cek ke data itachi setelah train dari ruslan.. Bisa pakai similarity, embedding, ..

1. Intent Classification **(akmal)**

[github.com/rikhuijzer/nlu\_datasets](http://github.com/rikhuijzer/nlu_datasets)

[paperswithcode.com/dataset/vimq](http://paperswithcode.com/dataset/vimq)

[github.com/ensembles4612/medical\_intent\_detector\_using\_BERT](http://github.com/ensembles4612/medical_intent_detector_using_BERT) (ini yang merujuk ke health)

1. Evaluation answer to the real data (bisa dokumen RAG atau data csv nya yg awal aja) **(akmal)**

Gmn cara tau answer nya ga halusinasi

<https://medium.com/@1048463053/how-to-evaluate-chatbots-1a3ca919c01d>

Possible query, label

Guardrail:

* Diluar scope medis (bisa di awal)
* Diluar scope pengetahuan si modelnya (di akhir)

Metode-metode RAG:

<https://www.datacamp.com/blog/rag-advanced>

### **Retrieval Mechanisms**

* **Sparse Retrieval**:
  + TF-IDF or BM25-based retrievers.
  + Focuses on exact word matches and keyword-based search.
* **Dense Retrieval**:
  + Uses vector embeddings for semantic search.
  + Models like **DPR (Dense Passage Retriever)** encode queries and documents into a shared vector space.
* **Hybrid Retrieval**:
  + Combines sparse and dense retrieval for better recall and precision.
* **Neural Search**:
  + Leverages models like ColBERT to efficiently handle large document collections using multi-vector retrieval.

### **Integration with Generation**

* **Direct Augmentation**:
  + Retrieved passages are directly prepended to the input of a generative model like GPT or T5.
* **Intermediate Summarization**:
  + Summarize retrieved passages before feeding them to the generator.
* **Iterative Augmentation**:
  + Perform retrieval in multiple rounds, refining the query or response with each iteration.

### **Query Reformulation**

* **Static Reformulation**:
  + Modify the query once before retrieval (e.g., using templates).
* **Dynamic Reformulation**:
  + Use feedback from the generative model or user interactions to refine queries.
* **Backwards Query Generation**:
  + Generate pseudo-queries from the target answer to enrich the retrieval dataset during training.

### **Combining Retrieval and Generation**

* **Two-Step Pipelines**:
  + Separate retriever and generator models, each trained independently.
* **End-to-End RAG**:
  + Jointly train retrieval and generation components for better optimization (e.g., through reinforcement learning).
* **Multi-Modal RAG**:
  + Incorporates non-textual data (e.g., images, tables) alongside text during retrieval.

### **Feedback Loops**

* **Re-Ranking**:
  + Rank retrieved passages based on relevance scores or contextual importance.
* **Retrieval-Enhanced Training**:
  + Use retrieved documents to provide external context during model fine-tuning.
* **Active Learning**:
  + Use human feedback to iteratively improve retrieval quality and model performance.

### **Retrieval Model Training**

* **Supervised Learning**:
  + Train with labeled pairs of queries and relevant documents (e.g., MS MARCO, Natural Questions datasets).
* **Contrastive Learning**:
  + Train encoders to maximize similarity for relevant query-document pairs while minimizing it for irrelevant pairs.
* **Few-shot Retrieval**:
  + Use few-shot or zero-shot learning for domains with limited labeled data.

### **Post-Processing**

* **Fact Verification**:
  + Use models to verify the accuracy of generated content against retrieved passages.
* **Answer Consolidation**:
  + Summarize or rank the outputs to ensure coherence and factuality.
* **Confidence Scoring**:
  + Attach confidence levels to generated responses based on retrieval relevance and model certainty.

Final rag

unsloth/Qwen2.5-1.5B-Instruct pubmed

{'rouge1': 0.2585072440045757, 'rouge2': 0.19045328808073408, 'rougeL': 0.23320736638659278, 'rougeLsum': 0.2347544991793698}

precision 0.8995576477050782

recall 0.8216160869598389

f1 0.8580327290296554

unsloth/Qwen2.5-7B-Instruct pubmed

{'rouge1': 0.40562025150831615, 'rouge2': 0.3077225985085137, 'rougeL': 0.35678570458031067, 'rougeLsum': 0.3585127015544004}

{'precision': 0.8977643990516663, 'recall': 0.8571103662252426, 'f1': 0.8763365137577057}

unsloth/Qwen2.5-7B-Instruct l12 embedding

{'rouge1': 0.43526846438459527, 'rouge2': 0.35640740351454137, 'rougeL': 0.4006309632506977, 'rougeLsum': 0.40051583434498483}

{'precision': 0.90550368309021, 'recall': 0.8657482069730759, 'f1': 0.8843280780315399}

unsloth/Qwen2.5-1.5B-Instruct l12

{'rouge1': 0.27205284772784816, 'rouge2': 0.20762285941477904, 'rougeL': 0.2453838030616751, 'rougeLsum': 0.24519062705638217}

{'precision': 0.9023079472780228, 'recall': 0.8250430124998093, 'f1': 0.8610310572385788}

unsloth/Llama-3.2-1B-Instruct l12

{'rouge1': 0.15601349701791334, 'rouge2': 0.06766384979667514, 'rougeL': 0.12873589888148262, 'rougeLsum': 0.13011079993055114}

{'precision': 0.8372294980287552, 'recall': 0.8025684636831284, 'f1': 0.8190993303060532}

unsloth/Llama-3.2-1B-Instruct pubmedbert

{'rouge1': 0.172824729164122, 'rouge2': 0.07921485047816285, 'rougeL': 0.13561465134966538, 'rougeLsum': 0.1365700001658004}

{'precision': 0.8585503697395325, 'recall': 0.8060175126791, 'f1': 0.831032492518425}

unsloth/Llama-3.2-1B-Instruct l12 rawretrieve

{'rouge1': 0.23712686278901987, 'rouge2': 0.1567717741250017, 'rougeL': 0.20954514562225698, 'rougeLsum': 0.21097153100824007}

{'precision': 0.8581778132915496, 'recall': 0.8192421263456344, 'f1': 0.8375813966989517}

unsloth/Qwen2.5-1.5B-Instruct l12 rawretrieve

{'rouge1': 0.31183621722749355, 'rouge2': 0.26305386000938225, 'rougeL': 0.2993876681941014, 'rougeLsum': 0.2994808240534926}

{'precision': 0.9162831246852875, 'recall': 0.8312226647138595, 'f1': 0.87085364818573}

unsloth/Qwen2.5-7B-Instruct l12 rawretrieve

{'rouge1': 0.4822937036065301, 'rouge2': 0.39793676599171257, 'rougeL': 0.4390692068372347, 'rougeLsum': 0.4396613947647122}

{'precision': 0.9091486430168152, 'recall': 0.8751699328422546, 'f1': 0.8909201967716217}

Seq2seq T5 rawretrieve

{'rouge1': 0.18139600679389078, 'rouge2': 0.12678342497693768, 'rougeL': 0.16700154310682092, 'rougeLsum': 0.16698942939897787}

{'precision': 0.8992286944389343, 'recall': 0.8084751230478286, 'f1': 0.8508381897211075}

Bert - bert-large-uncased-whole-word-masking-finetuned-squad rawretrieve

{'rouge1': 0.21821748669072363, 'rouge2': 0.17783350128898257, 'rougeL': 0.20682712033807743, 'rougeLsum': 0.20500099878871691}

{'precision': 0.7628444594144821, 'recall': 0.701669204235077, 'f1': 0.7298660361766816}

unsloth/Qwen2.5-1.5B-Instruct tfidf

{'rouge1': 0.2204602320067319, 'rouge2': 0.14974799933812422, 'rougeL': 0.19699418533445434, 'rougeLsum': 0.19764215294114165}

{'precision': 0.8774100482463837, 'recall': 0.8049841976165771, 'f1': 0.8390253877639771}

unsloth/Qwen2.5-1.5B-Instruct bm25

{'rouge1': 0.24803964229608283, 'rouge2': 0.17546547627594652, 'rougeL': 0.21840803882543697, 'rougeLsum': 0.21869028878116364}

{'precision': 0.8868778449296951, 'recall': 0.8193204778432847, 'f1': 0.8510477501153946}