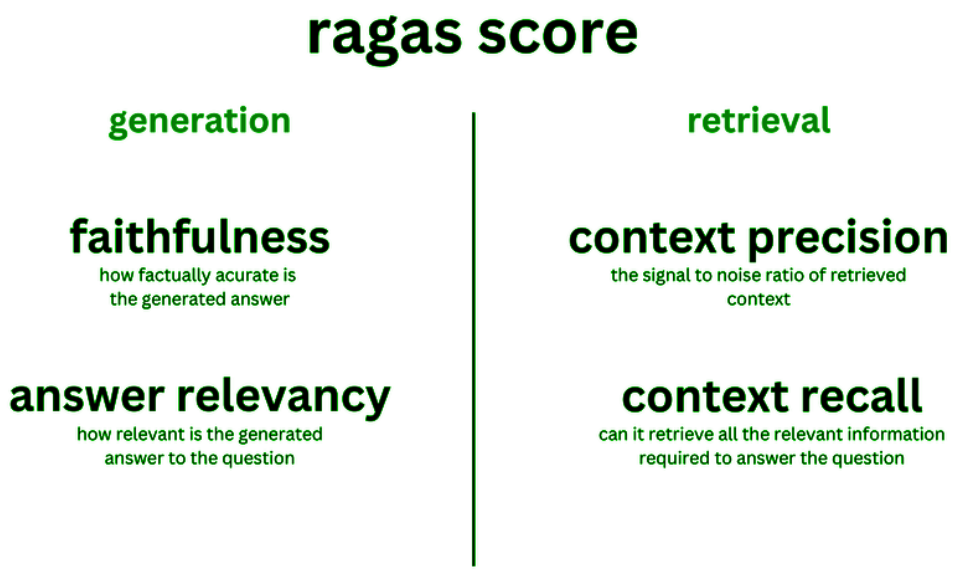
Evaluation the RAG

The Dataset will be the groundtruth for us:

I've created the evaluation data set, which will serve as the ground truth for our evaluation. I used GPT-4.0 to generate it (5 queries per country)

The Metrics:



Faithfulness:

‘how accurately the generated response reflects the content of the retrieved documents’

Suppose the user asks:

"What are the legal obligations of landlords in Scotland under the Housing (Scotland) Act 2006?"

The chatbot retrieves a section from the Housing (Scotland) Act 2006 and generates a response:

Retrieved Text:

> "Landlords are required to ensure that properties meet the repairing standard. This includes ensuring the structure is wind and watertight, installations are in proper working order, and the property has adequate smoke detectors and carbon monoxide alarms."

Generated Response:

> "Under the Housing (Scotland) Act 2006, landlords must make sure their properties meet the repairing standard. This includes ensuring the structure is wind and watertight, and that essential installations like water and heating are functional. The act also mandates smoke detectors and carbon monoxide alarms."

In this case, the **generated response is faithful** to the retrieved text because it accurately reflects the content without fabricating or omitting key information.

Faithfulness Equation

The faithfulness score can be calculated using an automated metric that compares the generated output to the retrieved documents.

$$\text{Faithfulness Score} = \frac{\text{Correct Information from Retrieved Documents}}{\text{Total Information in Generated Response}}$$

This ratio measures how much of the generated response is directly supported by the retrieved documents. The closer the score is to 1, the more faithful the generated response is.

Example:

Let's break down the response above. Suppose the following key pieces of information are extracted:

1. Properties must meet the repairing standard.
2. Structure must be wind and watertight.
3. Installations (like water and heating) must be functional.
4. The property must have smoke detectors and carbon monoxide alarms.

If all four points from the retrieved text are accurately reflected in the generated response, the score will be:

$$\frac{4}{4} = 1$$

However, if the chatbot incorrectly adds information or misses an essential point, the score would decrease.The answer is scaled to (0,1) range. Higher the better.

The Enhanced Faithfulness Metric

Our legal chatbot must deliver highly accurate responses when interpreting UK legislation. The current faithfulness metric lacks transparency and precision, making it less suited for evaluating complex legal answers. A new faithfulness metric offers better granularity and transparency, making it ideal for our needs.

Current Metric Limitations

1. **No Transparency**: The overall score hides inaccuracies in specific parts of the response.
2. **Overgeneralization**: A single score misses critical details in complex legal answers.
3. **Difficult Fine-tuning**: Without seeing where errors occur, it's harder to improve responses.

Benefits of the New Faithfulness Metric

1. **Granular Evaluation**: It evaluates each sentence separately, ensuring every part aligns with the legal text.
2. **Transparency**: Intermediate outputs show exactly where the response deviates from the retrieved legal information.
3. **Improved Accuracy**: Ensures small but important legal details are identified and corrected, vital for legal professionals and compliance tasks.

Example

If a chatbot response incorrectly says "similar job" instead of the legally correct "same job or suitable alternative," the new metric would flag this error, ensuring full alignment with the legal text.

The previous faithfulness metric was like a **black box** because it provided a single overall score without revealing how each part of the generated response was evaluated. This lack of transparency made it difficult to identify specific inaccuracies in the response, which is particularly problematic in complex legal contexts where precision is crucial.

Link for the study: https://arxiv.org/pdf/2407.12873

Answer relevance

Answer relevance is a crucial metric in evaluating how well the generated answer responds to a given question. It assesses whether the answer is directly related to the user's query and contains information that is relevant and useful. For our legal chatbot, this ensures that when users ask questions about UK legislation, the chatbot's responses are accurate and aligned with the specific request.

Answer relevance is measured using **cosine similarity** between the embedding of the original question and the embeddings of several reverse-engineered (or generated) questions based on the answer. The underlying idea is that if the answer correctly addresses the original question, it should be possible to generate new questions from that answer which align closely with the original query.

The generated questions are compared to the original question using **mean cosine similarity**. The result typically ranges between -1 and 1:

- **1** means the answer is perfectly aligned with the question. The angel is 0 and the cosine is 1
- **0** indicates no relationship.the angel is 90 and the cosine is 0
- **-1** suggests a contradictory relationship.

The Equation:

$$\text{answer relevancy} = \frac{1}{N} \sum_{i=1}^{N} \cos(E_{g,i}, E_o)$$

The modified answer relevance metric logs the generated questions, embeddings, and similarity scores for transparency, allowing users to inspect how the score was derived. This was not done in the original version, making the process more interpretable in the modified version.

Context Recall:

Context Recall measures how well the retrieved contexts align with the ground truth answer. It checks if all relevant claims or pieces of information present in the ground truth have corresponding chunks in the retrieved context.

Formula:

$$\text{Context Recall} = \frac{\text{Ground truth claims that can be attributed to retrieved context}}{\text{Total number of claims in the ground truth}}$$

In Practice: If the query is: “Under what conditions can personal data be processed according to the Data Protection Act 2018?”

Ground Truth:

Claim 1: "Personal data can be processed with the data subject's consent."

Claim 2: "Processing is necessary for the performance of a contract."

Claim 3: "Processing is necessary for compliance with a legal obligation."

Retrieved Contexts:

Chunk 1: “Personal data can only be processed if necessary for the public interest.” (Not relevant)

Chunk 2: “The Data Protection Act 2018 states that personal data can be processed with the data subject’s consent.” (Relevant, matches Claim 1)

Chunk 3: “Processing is necessary for the performance of a contract or for compliance with legal obligations.” (Relevant, matches Claim 2 and Claim 3)

Step 1: Break Ground Truth into Claims

Claim 1: "Consent is required for processing."

Claim 2: "Contractual necessity is a condition."

Claim 3: "Legal obligation is another condition."

Step 2: Check Attribution

Claim 1 is matched by Chunk 2.

Claim 2 and Claim 3 are matched by Chunk 3.

Thus, 3 out of 3 claims are covered by the retrieved contexts.

Step 3: Calculate Context Recall

$$\text{Context Recall} = \frac{3}{3} = 1.0$$

In this case, Context Recall is 1.0 (or 100%), meaning all relevant claims were retrieved by the RAG system.

Context Precision:

tells how good the system is at making sure that the most relevant information appears at the top of the results.

Question:

"What is the statutory maternity leave entitlement in the UK?"

Ground Truth:

In the UK, statutory maternity leave is 52 weeks, which consists of 26 weeks of ordinary maternity leave and 26 weeks of additional maternity leave.

Retrieved Contexts (from UK legislation):

1. "Statutory maternity leave in the UK is available for up to 26 weeks, and it can be extended under certain conditions." (partially relevant)
2. "Employees are entitled to 52 weeks of maternity leave, with the first 26 weeks being ordinary maternity leave and the remaining 26 weeks being additional maternity leave." (relevant)
3. "Statutory maternity pay is provided for up to 39 weeks, but only 90% of average weekly earnings are paid for the first six weeks." (irrelevant)

Step-by-Step Corrected Calculation:

Step 1: Check relevance for each chunk:

- Context 1: Partially relevant (mentions 26 weeks, not the full 52 weeks). For context precision, this still counts as irrelevant, so we assign 0.
- Context 2: Relevant (mentions the full 52-week entitlement). Assign 1.
- Context 3: Irrelevant (discusses statutory pay, not leave entitlement). Assign 0.

Step 2: Calculate Precision@k for each rank:

- Precision@1: The first retrieved chunk is irrelevant (partially relevant but doesn’t fully answer the query):

$$Precision@1 = \frac{0}{1} = 0$$

- **Precision@2**: The second retrieved chunk is **relevant**:

$$Precision@2 = \frac{1}{2} = 0.5$$

- **Precision@3**: The third chunk is **irrelevant**:

$$Precision@3 = \frac{0}{3} = 0$$

Step 3: **Calculate the overall Context Precision**:

$$ContextPrecision = \frac{(0 + 0.5 + 0)}{1} = 0.5$$

A score of 0.5 tells us that the system is only partially accurate in ranking the relevant information**.**

Answer Correctness:

Answer Correctness evaluates how factually accurate the generated answer is compared to the ground truth. It measures whether the key facts in the generated answer match the facts in the ground truth. The score ranges from 0 to 1:

- 1 means the generated answer is entirely correct.

- 0 means the answer is incorrect.

This evaluation is based purely on **factual correctness**.

Factual Correctness:

We measure **factual correctness** using the following concepts:

- **True Positive (TP)**: Facts that are present in both the generated answer and the ground truth.
- **False Positive (FP)**: Facts that are present in the generated answer but not in the ground truth (i.e., incorrect facts).

- **False Negative (FN)**: Facts that are present in the ground truth but missing from the generated answer.

Example Scenario

Ground Truth Answer:

1. Einstein was born in 1879.
2. Einstein was born in Germany.
3. Einstein developed the theory of relativity.

Generated Answer:

1. Einstein was born in 1879.
2. Einstein was born in Spain.
3. Einstein developed the theory of relativity.
4. Einstein won the Nobel Prize in Physics in 1921.

Step-by-Step Calculation

1. Identify True Positives (TP), False Positives (FP), and False Negatives (FN):
    - True Positives (TP): Facts present in both the ground truth and the generated answer.
        - "Einstein was born in 1879."
        - "Einstein developed the theory of relativity."
    - False Positives (FP): Facts present in the generated answer but not in the ground truth.
        - "Einstein was born in Spain."
        - "Einstein won the Nobel Prize in Physics in 1921."
    - False Negatives (FN): Facts present in the ground truth but missing from the generated answer.
        - "Einstein was born in Germany."
2. Count the number of TP, FP, and FN:
    - TP = 2 (Einstein was born in 1879, Einstein developed the theory of relativity)
    - FP = 2 (Einstein was born in Spain, Einstein won the Nobel Prize in Physics in 1921)
    - FN = 1 (Einstein was born in Germany)

F1 Score Formula:

$$\text{F1 Score} = \frac{\text{TP}}{\text{TP} + 0.5 \times (\text{FP} + \text{FN})}$$

Plugging in the Values:

$$\text{F1 Score} = \frac{2}{2 + 0.5 \times (2 + 1)}$$

Step-by-Step Breakdown:

1. Addition inside parentheses:

$$2 + 1 = 3$$

2. Multiplying by 0.5:

$$0.5 \times 3 = 1.5$$

3. Add TP to the result:

$$2 + 1.5 = 3.5$$

4. Final division:

$$\frac{2}{3.5} \approx 0.57$$

The old version:

```
answer_correctness
                0.0
```

The enhanced one:

```json
{
  "TP": [],
  "FP": ["The legal age for marriage in England is 16 years old"],
  "FN": ["The legal age for marriage in England is 18 years old"]
}
```

```json
{
  "num_tp": 0,
  "num_fp": 1,
  "num_fn": 1,
  "json": "{
    'TP': [],
    'FP': ['The legal age for marriage in England is 16 years old'],
    'FN': ['The legal age for marriage in England is 18 years old']
  }",
  "score": 0
}
```

The multiquery technique: