LLMs in Insurance: from main concepts to deployment of solutions for industrial risks

A. Couloumy , A. Grondin, A. Waswate

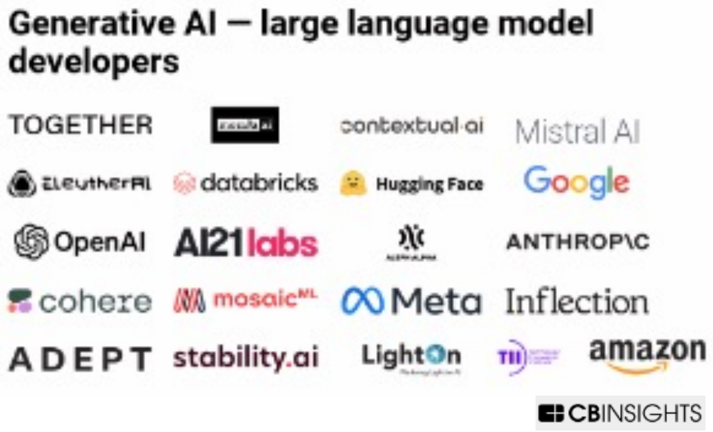
1. Introduction
GenAI & LLMs

- **Generative AI** (GenAI) is a type of artificial intelligence that aims to create new data or original content rather than simply analysing or reproducing existing data. It uses models and algorithms to autonomously generate information: Text, Images, Videos, Sounds, etc.

- Unlike other types of AI, generative AI is capable of producing creative and innovative results using techniques such as **generative neural networks** (GANs) or adversarial generative models. These models are trained on large amounts of data to learn the underlying patterns and structures, enabling them to generate new, realistic and consistent data.

- A **Large Language Model** (LLM) is an advanced language model that:
  - uses artificial intelligence techniques to generate text autonomously
  - is trained on large corpus to learn linguistic structures, patterns and relationships
  - is capable of understanding and generating text in different languages
  - uses probabilistic models to predict the probability of a given word sequence

- Since 2020, the LLM proposals on the market **have followed one another**:
  - Proprietary solutions: GPT 3.5, GPT 4, PaLM 2, CLAUDE 2, etc.
  - Open source: Llama 2, OpenLLama, MPT, T5, Alpaca, Bloom, Falcon, Mistral, Alfred, etc.



1. Introduction
Usage & Business tasks

- **Basic tasks**
  - **Generate text**: create coherent, relevant text such as an article, email, report, etc.
  - **Editing text**: correcting spelling, grammar, replacing or deleting key words, etc.
  - **Translate** into another language for multilingual communication
  - **Summarising** a text: extracting the key information from a text in a concise manner
  - **Classify** a text: assign a category to the text (spam/non-spam, positive/negative sentiment)

- **Advanced tasks**
  - **Extract structured** information from a text (names of people, dates, places, organisations, etc.) - Parsing
  - **Assess the similarity** between two texts. Dealing with plagiarism, searching for similar texts, etc.
  - **Change the writing style** of a text. Remaining neutral when disseminating information, using humour, etc.
  - **Synthesise** a document with rules and constraints and feed other tasks via LLMs
  - **Create interactive** dialogues between two agents/persons

- **Business tasks**
  - **Cause extraction** and details, regularization for non-life property pricing
  - **Risk assessment** (scoring and summarization) for energy underwriting reports
  - **Q&A on General terms and conditions** on custom Belgium insurance motor pdfs
  - **Wording comparisons** of reinsurance treaty clauses, (entity, clauses) and reasoning
  - **Internal control definition** using natural language and tabular analysis application

- **For actuaries:**

Data generation, augmentation, labelling, quality assessment, etc.

Modelling with feature extraction, sequential reasoning, advanced clustering, etc.

Coding with documentation management, migration facilitation, code review, etc.

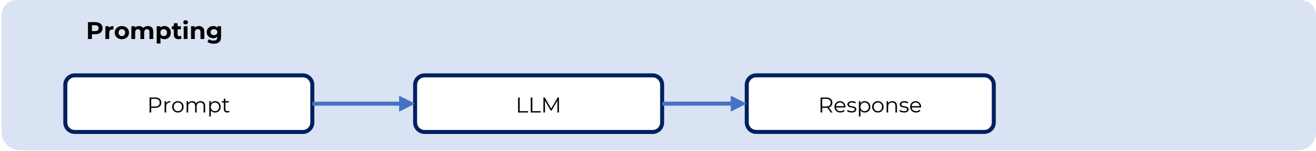
1. Introduction
Methods

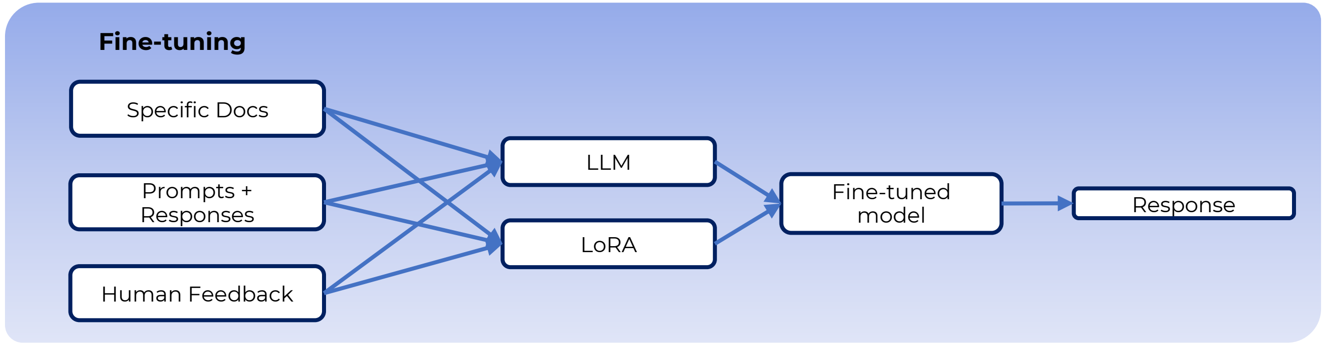
- **Technical landscape** to keep in mind:

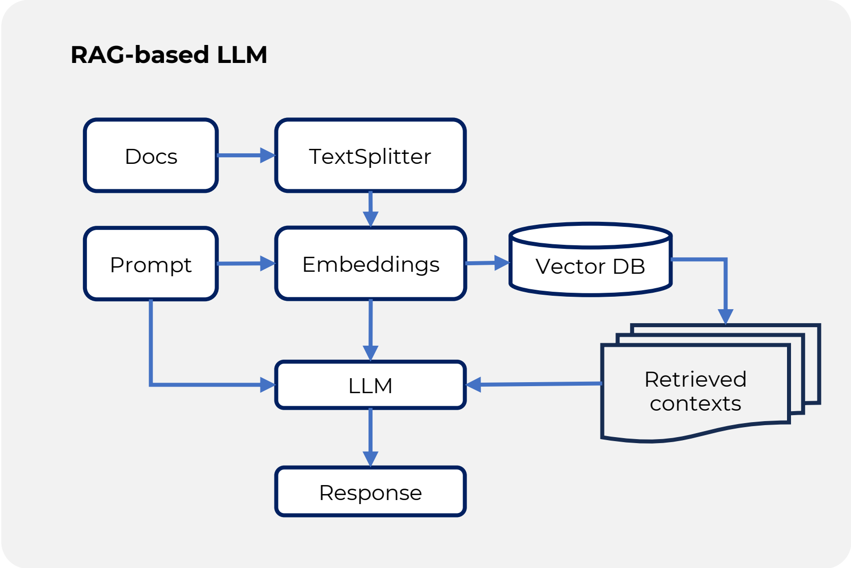
Annotation Data curation & preprocess Fine-tuning Metrics

Licensing Performance/Size Comparison Production run Memory/Caching

Inference length Hallucination Quantization/distillation Monitoring vulnerabilities

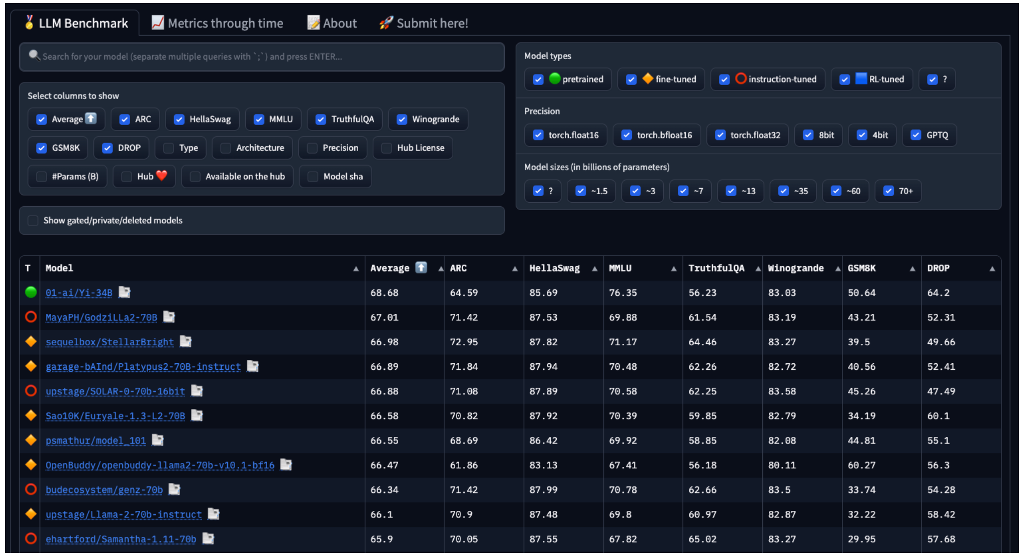






1. Introduction
Evaluation

- Several methods and metrics exist.
- **Word or Character Accuracy:** This metric measures the proportion of words or characters correctly predicted by the model in relation to the reference text.
- **BLEU (Bilingual Evaluation Understudy):** The BLEU score is a popular measure for evaluating the quality of machine translations. It compares the sentences generated by the model with the reference sentences, taking into account the correspondence of n-grams (sequences of words or characters).
- **ROUGE (Recall-Oriented Understudy for Gisting Evaluation):** ROUGE is a metric used to evaluate text summaries generated by LLMs. It compares words and phrases in the generated text with those in the reference text, focusing on similarity and recall.
- **METEOR (Metric for Evaluation of Translation with Explicit ORdering):** METEOR measures the similarity between a generated text and a reference text using word matches, synonyms, paraphrases and ordered words.
- Are you looking for a multi-tasking model or a model for a very specific task? Opt for your **own metrics** according to use



T	Model	Average	ARC	HellaSwag	MMLU	TruthfulQA	Winogrande	GSM8K	DROP
🟢	01-ai/Yi-34B	68.68	64.59	85.69	76.35	56.23	83.03	50.64	64.2
⭕	MayaPH/GodziLLa2-70B	67.01	71.42	87.53	69.88	61.54	83.19	43.21	52.31
🔶	sequelbox/StellarBright	66.98	72.95	87.82	71.17	64.46	83.27	39.5	49.66
🔶	garage-bAInd/Platypus2-70B-instruct	66.89	71.84	87.94	70.48	62.26	82.72	40.56	52.41
⭕	upstage/SOLAR-0-70b-16bit	66.88	71.08	87.89	70.58	62.25	83.58	45.26	47.49
🔶	Sao10K/Euryale-1.3-L2-70B	66.58	70.82	87.92	70.39	59.85	82.79	34.19	60.1
🔶	psmathur/model_101	66.55	68.69	86.42	69.92	58.85	82.08	44.81	55.1
🔶	OpenBuddy/openbuddy-llama2-70b-v10.1-bf16	66.47	61.86	83.13	67.41	56.18	80.11	60.27	56.3
⭕	budecosystem/genz-70b	66.34	71.42	87.99	70.78	62.66	83.5	33.74	54.28
🔶	upstage/Llama-2-70b-instruct	66.1	70.9	87.48	69.8	60.97	82.87	32.22	58.42
⭕	ehartford/Samantha-1.11-70b	65.9	70.05	87.55	67.82	65.02	83.27	29.95	57.68

1. Introduction
IT Stack

Wide ecosystem

Doc. transformers & vector store

unstructured
ChromaDB
Pgvector
Weaviate

Training & quant. libraries

Peft
BitesandBytes
Transformers
Lamini
GPTQ
AutoTrain
Setfit

Agent chaining & flow tool

LangChain
LlamaIndex
LangFlow
LangSmith

Cloud services

AWS services (Sagemaker, Bedrock), Azure services, etc.
Inference Endpoints

Related NLP libraries

Kor
UniversalNER
Spacy
SpanMarker
Doctran

Integrations

SerpAPI
Google Search API

LLM repositories or APIs

OpenAI (gpt3.5 turbo, gpt4, gpt4 turbo, etc.)
Cohere
HF hub (falcon, llama2, dolly, etc.

Annotation & Curation

Arguilla
LabelBox

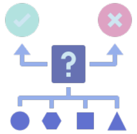
Vulnerabilities & security

Giskard
Lakera

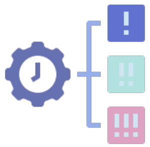
2. Context
Business case

- Competition to **innovate under pressure** (in terms of time and human resources) : 2 weeks during summer 2023.
- The case study aims at **facilitating understanding of risk engineering reports** provided by brokers during an industrial / energy underwriting process (dense, long, highly technical reports).
- More than automation, the work is useful for better understanding the **underlying risk**.
- In details:

Information **extraction**



Risk **identification**



Criticality **assessment**

2. Context
Data & strategy

ACTUAIRES 100% DATA SCIENCE BY Institut A

- Key figures to introduce the work:

100	20	12	6	5
pdf reports	annotated reports	info to extract	levels of criticality	risk families

Extraction

Indicators (plant details, events, recommendations, number/amount of losses, maintenance budget, sum insured, etc.), Point of interests

Identification

Risk exposures, Layout & construction, Control and safeguarding, Management systems, Loss mitigation.

Assessment

Score, plus positive and negative point of interest, ordered qualitatively according: Critical, intermediary, low importance.

- Solution pipeline:

Data processing ❯ LLM chaining ❯ Post processing ❯ Reporting

3. Approach
Data processing

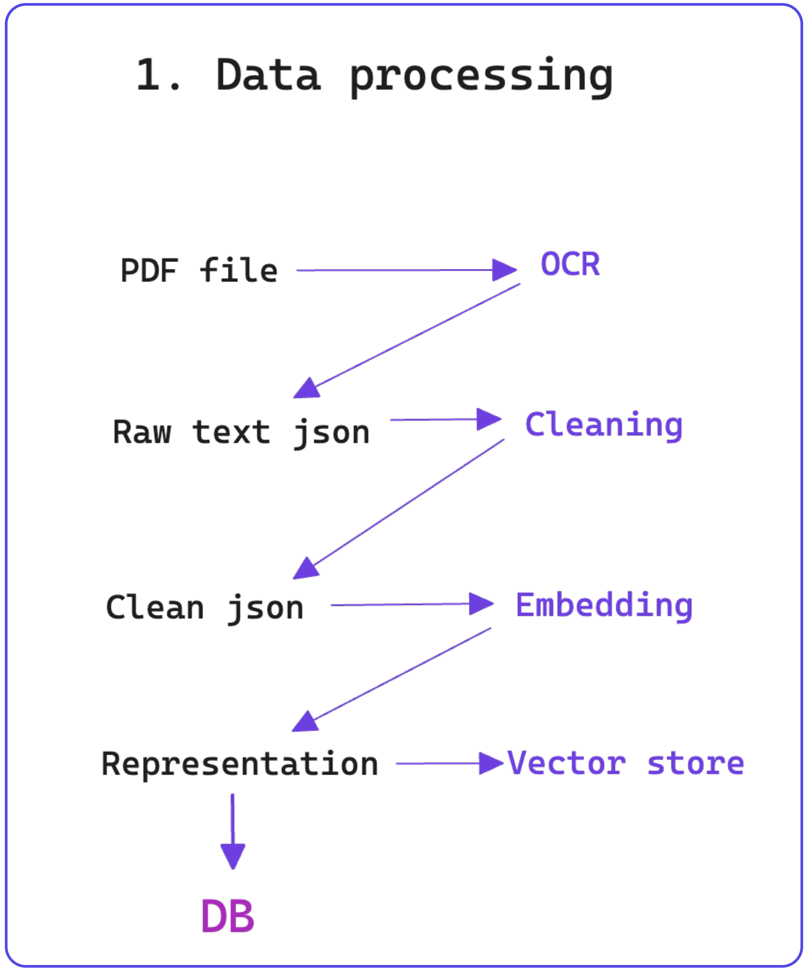
- Each file is pre-processed using an **OCR** (AWS Textract). JSONs text results are used at lines level and reshaped to fit with next retrieval tasks.
- Text is **cleaned** (deduplicate, gibberish, footer, etc.), ordered and each line is indexed using a sliding window (recursive contextual technique).
- **Sentence transformers** embeddings have been used to represent texts (for cost, simplicity, integration and speed reasons).
- Specifically, "all-mpnet-base-v2" * was used (English top 30 MTEB leader board – no custom training).
- Representations and metadata are stored in **vector store.**

Amazon Textract

LangChain

Hugging Face

chroma



* See appendix for more information

3. Approach
LLM - Selection

- We have selected LLM according 4 criteria:

Performance **Inference** **Cost** **Context length**

- **4 foundation models*** have been shortlisted to be used into the process:

Llama 2

HF Llama 2 13B Orca 8k
Open Source, Apache 2.0, small size model, long context length (with quantization)

FalconLLM

HF Falcon 40B
Open Source, Apache 2.0, Medium size model, regular context length (with quantization)

OpenAI

OpenAI GPT 3.5 turbo 16k
Open access, proprietary, Large size model, very long context length

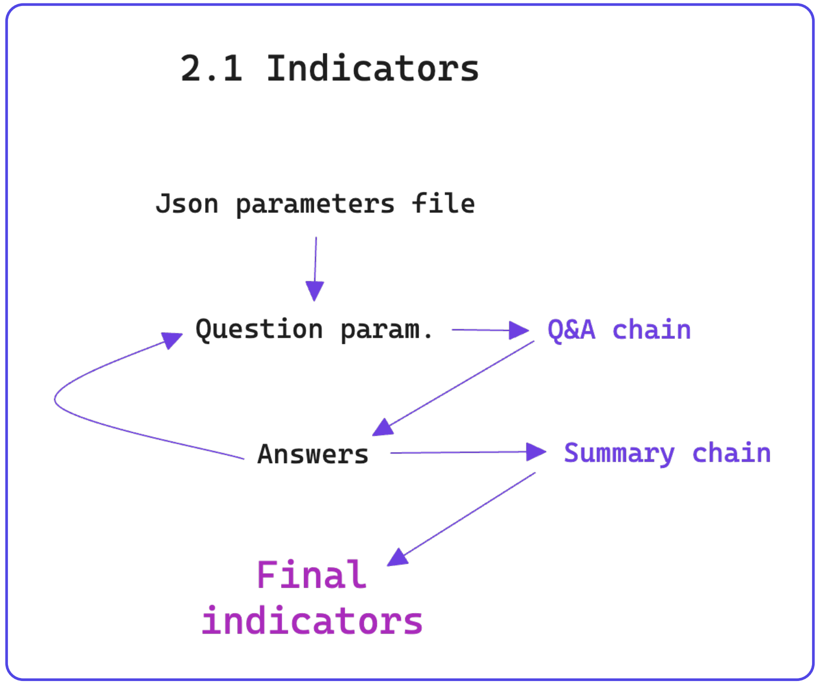
OpenAI

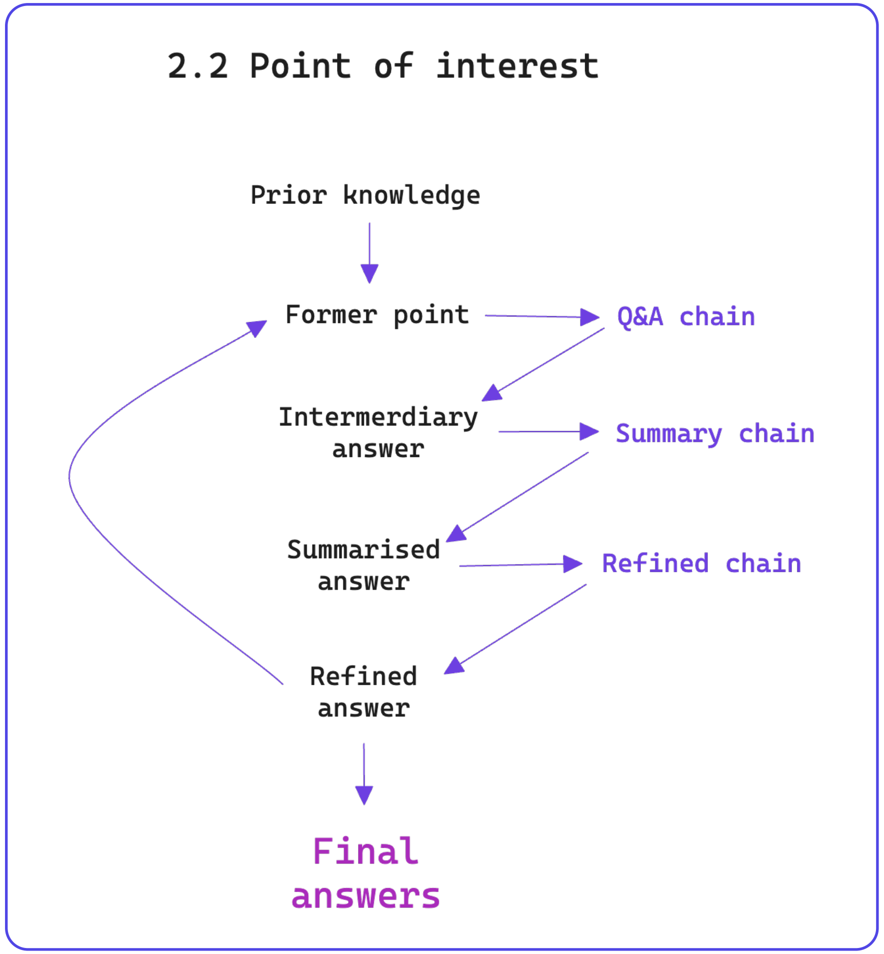
OpenAI GPT 4
Open access, proprietary, Very large size model, long context length

- Further details:
  - We finally select Open AI services because of contest context (decision in production would be different)
  - Any consideration of fine-tuning because of time constraints
  - May other models would have been considered

* See appendix for more information

3. Approach
LLM - Chaining

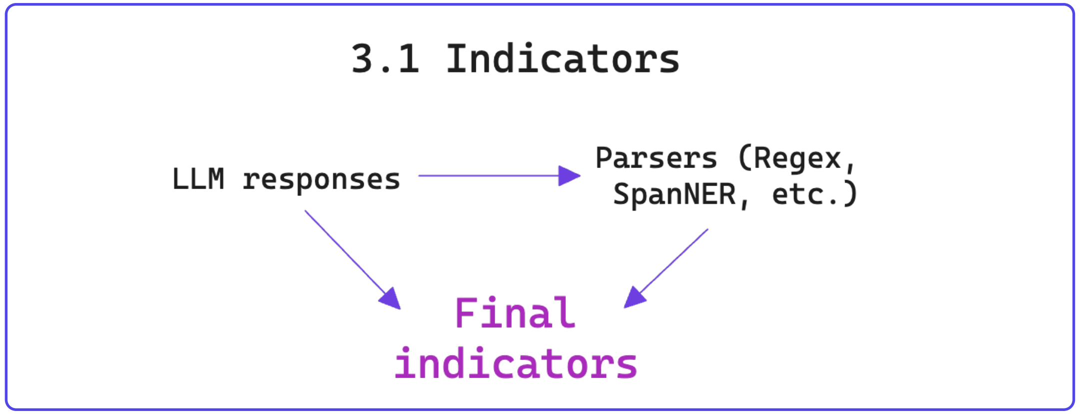


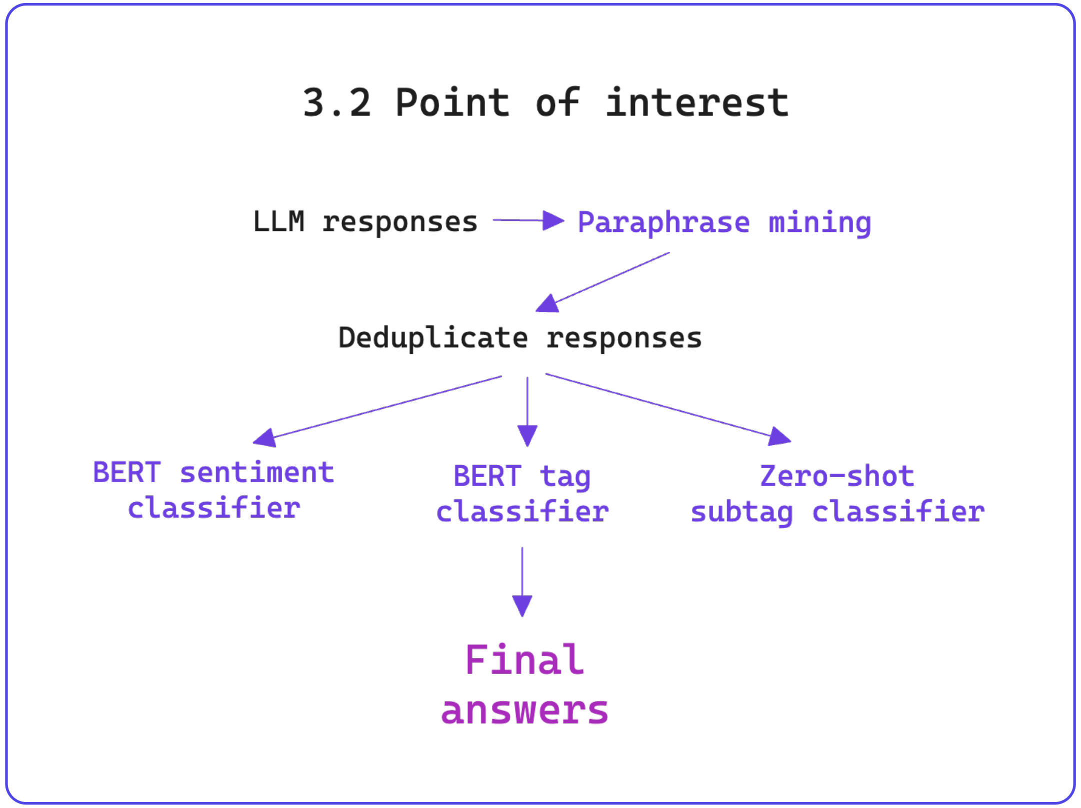


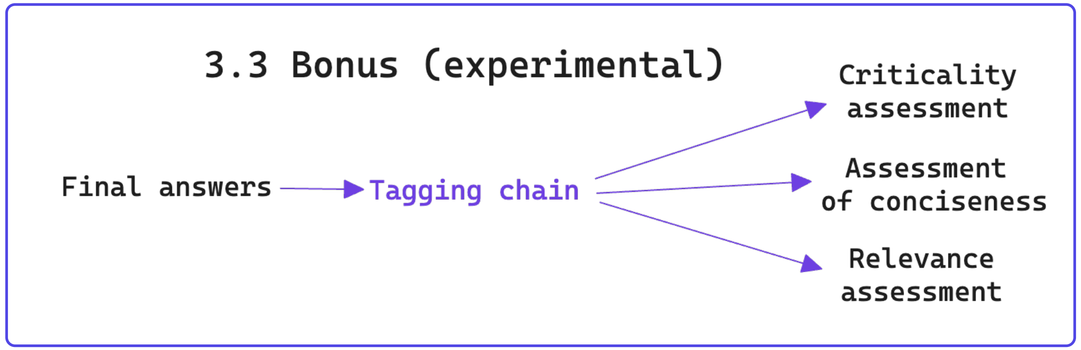
- Comments:
  - **Strategies are different** depending on information gathered
  - LLM prompts have been optimized regarding dedicated tasks
  - **RAG fusion** (summary part) allow to smooth the approach
  - Refine part aims at “formatting” results

3. Approach
Post processing

- It aims at providing **further insights** related to LLM responses while guaranteeing a more effective and relevant experience for users. Several techniques* have been developed





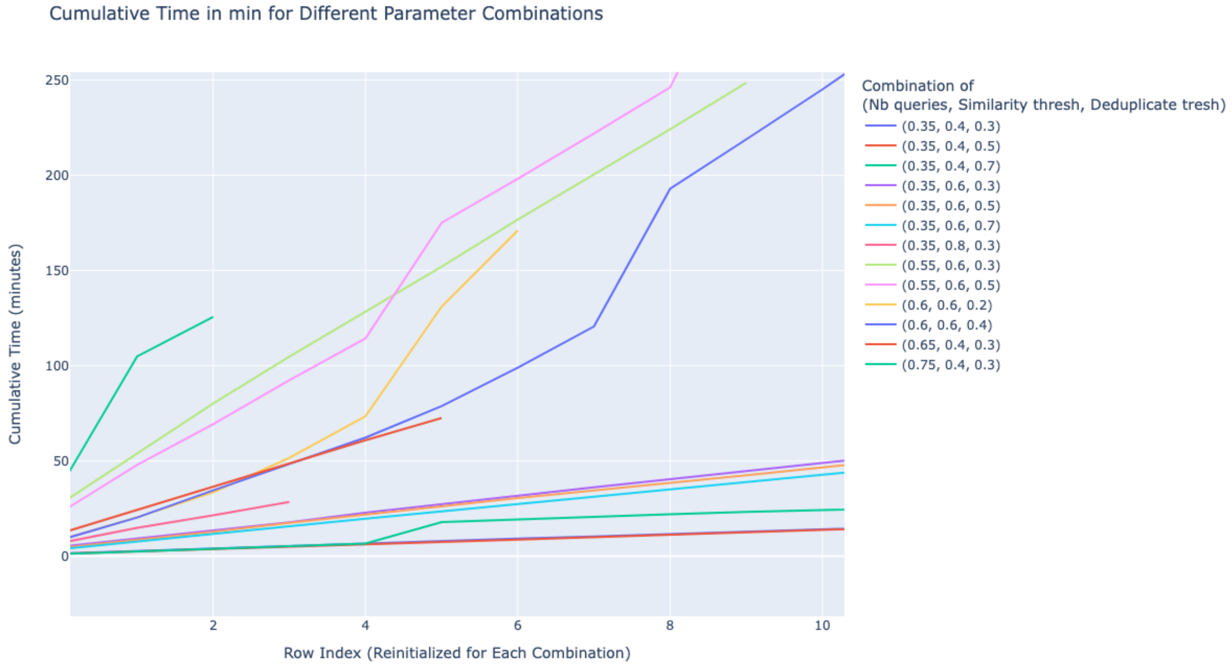


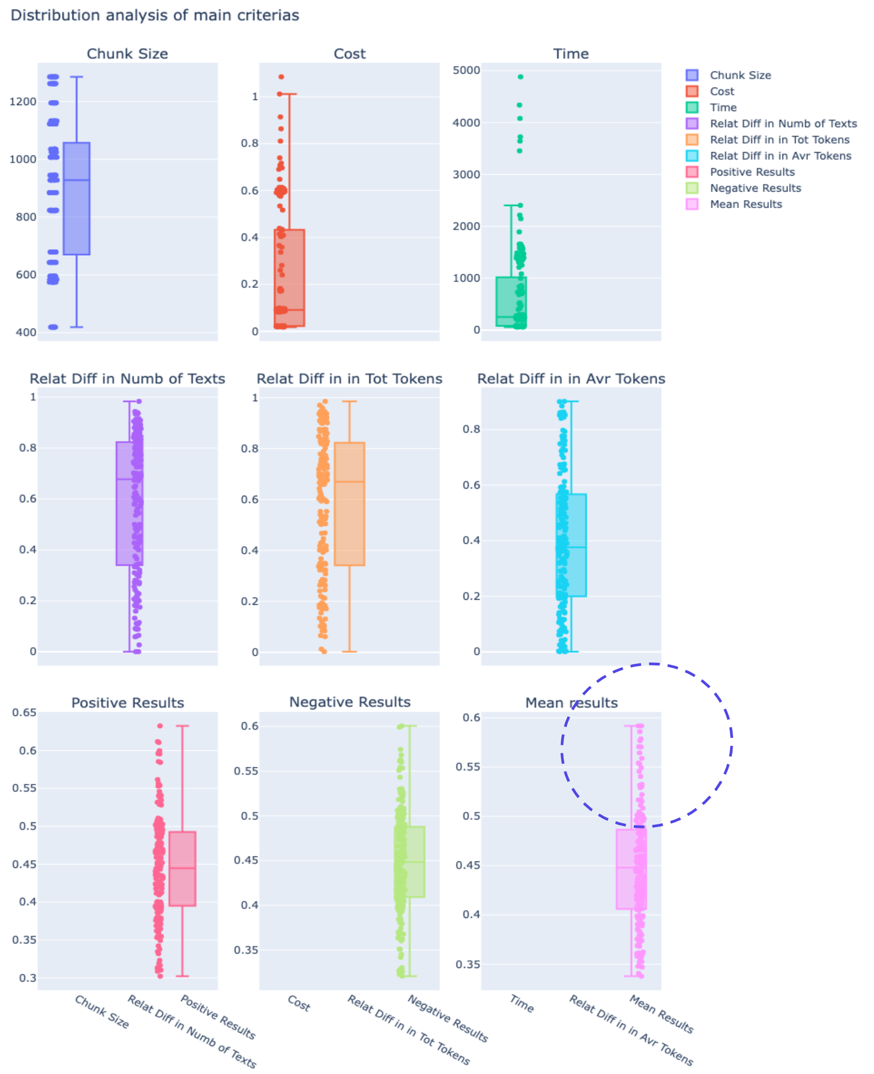
* See appendix for more information

4. Results

Sensitivity test (1/2)

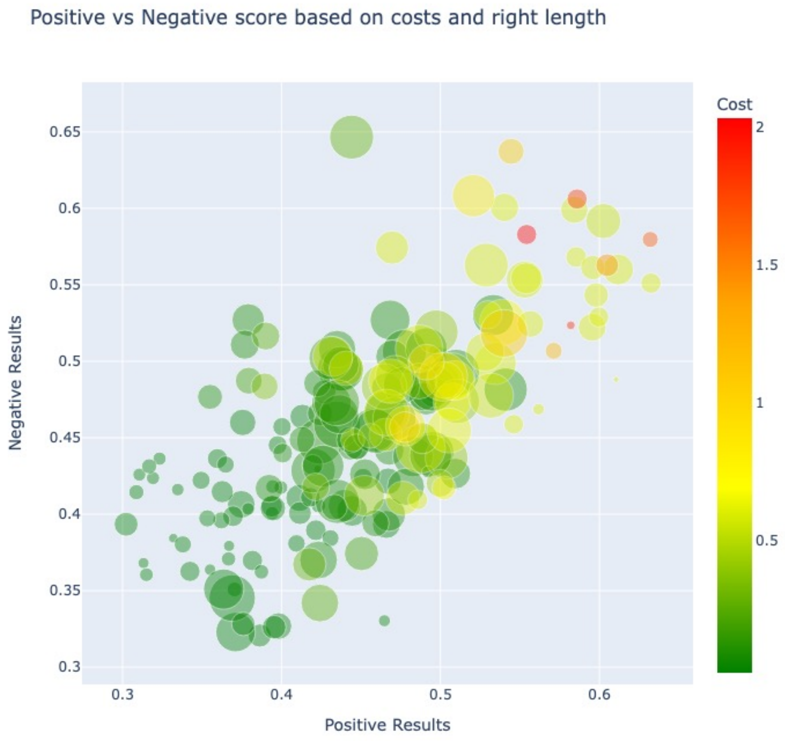
- Sensitivity tests have been ran **more than 200 times** (10 scenarios x 20 documents), mixing LLM parametrizations, prior knowledge queries , paraphrase mining threshold, prompt tuning, etc.

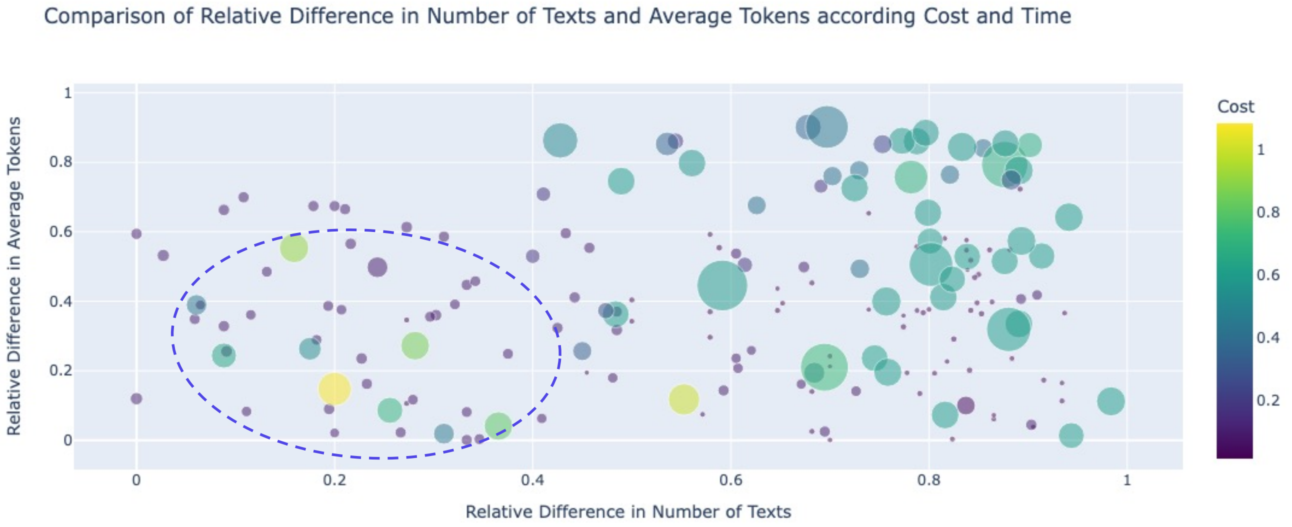




4. Results
Sensitivity test (2/2)

- Focus was on process **time and the quality of the results.** The aim was to find scenarios offering the best indicators:





- **Time** per task, batch, locally, in production to assess scalability, replicas needs, hardware requirements.

- **Quality** using error measures: length comparison, number of result comparison, dissimilarity, subtext pairwise comparison.

4. Results
Performance & Integration

- Main **performance** indicators:

+28% relative diff. of nb. of results

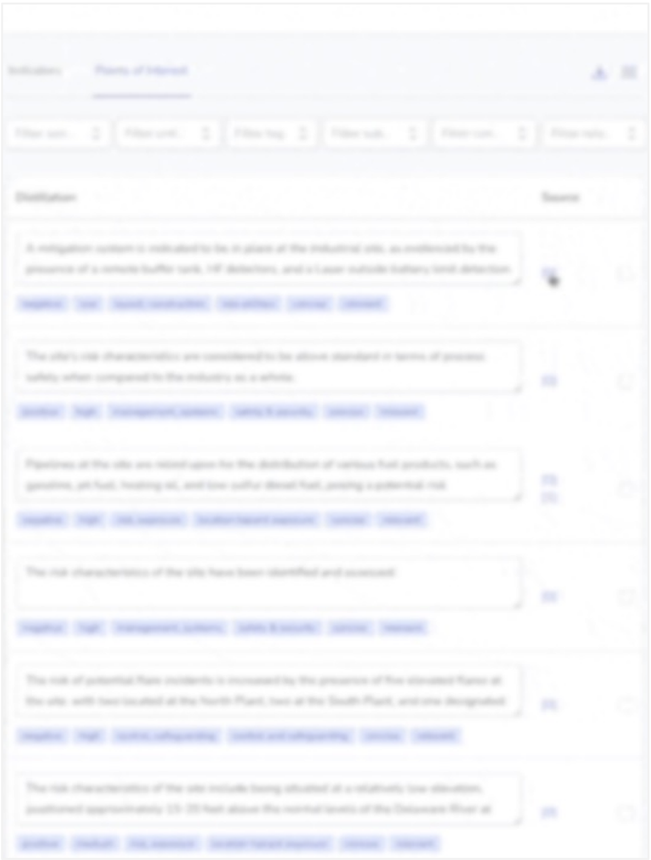
+20% relative diff. of average tokens

0.62 similarity average

0.97$ LLM cost per doc. on average

3 reports at the same time

Raw PDF → **1,5** min on average for OCR → **6** min on average for LLMs → **3,5** min on average for post proc. → Final rating



- IT **integration** details:
    - Mainly serverless asynchronous REST APIs
    - Deployed on AWS
    - User interface for experimentation

5. Conclusion
Challenges, experimentations & conclusions

- **Model**
  - Embedding may be custom trained (SetFit)
  - LLM could be adapted to the different tasks (PEFT)
  - LLM vulnerabilities may be reviewed (Giskard)
  - Better parsers for structured data extraction
  - New LLMs are available and should be tested

- Other **experimentations**:
  - **Risk score** prediction (based on points of interest)
  - **Non prior criteria** chain (in absence of prior knowledge)

- **Back**
  - Improve multi concurrence (on indicator part in particular)
  - LLM access may be accelerated
  - Supervised ML model inference may be optimized
  - Production run should be managed differently

- **Data**:
  - More (annotated) data may be helpful to fine tuned all supervised tasks, specifically regarding tags and criticality
  - A deeper paraphrase mining process on former point of interest may be relevant.

- Overall **conclusions**:

Objectivising all business expectations is complex	Having a mix of LLMs really helps	LLM implies an excellent command of computer engineering	Contribution to a better understanding of underlying risks is substantial

6. Appendix
Model references

- References related to **data processing**:

  https://www.sbert.net/
  https://huggingface.co/spaces/mteb/leaderboard
  https://huggingface.co/sentence-transformers/all-mpnet-base-v2

- References related to **LLM models**:

  https://huggingface.co/tiiuae/falcon-40b
  https://huggingface.co/OpenAssistant/llama2-13b-orca-8k-3319
  https://platform.openai.com/docs/models

- References related to **post processing**:

  https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-mnli-xnli
  https://osf.io/74b8k

Point of interest - Tag
Multi-class classification – deberta - English

Loss: 0.378
Accuracy: 0.926
Macro F1: 0.922
Micro F1: 0.926
Weighted F1: 0.926
Macro Precision: 0.925
Micro Precision: 0.926
Weighted Precision: 0.927
Macro Recall: 0.920
Micro Recall: 0.926
Weighted Recall: 0.926
CO2 Emissions (in grams): 0.1294

Point of interest - Sentiment
Binary classification – deberta - English

Loss: 0.061
Accuracy: 0.986
Precision: 0.981
Recall: 0.990
AUC: 0.998
F1: 0.986
CO2 Emissions (in grams): 1.4664