Reading auditor Report

Apporach -> First we convert the pdf into images using pymupdf then extract the text from all the images one by one using google’s py-tesseract and then search for

1. Independent Auditor
2. Annexure A
3. Annexure B

If these strings are found the data of each of these is extracted in a json with keys as “independent\_auditor”, “annexure\_a”, “annexure\_b”

This data is being extracted under the function getting\_heading() this function is the parent function of getting the data, while reading the data, all the images of annexure a are converted back into the pdf so that any table there can be read by aws textract and then saved into s3. then the script calls another function named get\_comment() which is used to get the auditors comment. For independent auditors we need to check for Qualified opinion only and if it is there then we are required to change the language into the third person, including the heading of basis of qualified opinion and emphasis of matter. Now for other matters and legal of regulatory we need to check for the disqualification of the directors.

Now for annexure a we look for the adverse comment which follows the caro headers rule mentioned in the HLD. For annexure a we first get the data from the whole raw data using llm and then classifiy it using llm that whether the point is adverse or not. And check for tables as well if there is any table then it is read by aws textract and then the languages for adverse comments classified by the llm into third person. And for last part annexure b, we only look for any negative point in the opinion, if it is there then we capture it.

This process runs in document process step

Ratings

Apporach -> We first get the data (the data which is needed to get the rationale/ ratings) based on the frameworks provided by the Vishal sir from the data to generate word doc, and priority and then pass them to llm for generating rationale for products, profitability, cutomers, foreign exchanges and other rationale are created by comparing the values, and for sensitive we generate all the weaknes into sensitive, with help of llm.

This process run after the adverse check as it requires the EPFO delays.

Gstn

From gstn we first extract all the data using fitz and then we pass all the data to the llm and then we only read gstin numbers and all the addresses present in the gstn are extracted with key as place of business and then save them back to the s3

Old Report

Added reading key executive, revenue terms and export terms and length of relationship with customers and later realized that the current method is not working fine hence I have invented a method of utilizing the table of content page and then the page number present in table of content is extracted for particular key and the relative data extracted from those page numbers are send to llm to get the necessary details, hence this way never fails and also performs well in both accuracy and also take less time in full execution. Also added this way in try except, that if by any chance this method got fail then we go with old method which is already present.

In old report we also read previous auditor data using the above method only and then pass it to llm to get the table and text out of it in the order as it was before in the pdf. The data from auditors from old report is first saved to the database and then saved into s3.

This process runs in document process step

Finetune

The script is made to regularly finetune the model for fin and auditor writeups on a regular basis, to improve the accuracy of the writeups. In auditors we only use the finetune model for annexure a as it is only trained on annexure a only. As for other points of auditors we directly copy paste the data with just changing them into third person.

The script donwloads the data from s3 from writeuptraining data and then maintains the data, cleans it and then process it and then make them in a final json which has “assistant”, “system”, “user” data

And then we check for errors present in it, if yes then stop the finetuning. If everything is good then finetuning is started this will take approximately 20 to 30minutes depending the traffic on the server of openai. After fintunning there is a function which automatically gives us the finctuned model id when the status becomes completed and then we save/store the finetuned model into environment

Finetune\_model\_id\_0 is for Fine writeup creation

Finetune\_model\_id\_1 is for auditor writeup creation

This id ->0 represent the fin writeup section and id->1 represent auditors section

Priority changes

For revenue not reading I made change In customer section added float to it instead of int

Added priority for mca and gst as well, we read address from the gst which is given the first priority if it is there.

MCa

Added other directorships from the mca data . We extract the data for the top director

Solving capta for it is giving us error hence we click anywhere and then we extract all the other directorships.

Writeup creation for location and markets and clientel and ceo and group

We first extract the necessary data from the data to generate word doc and priority and then filter the necessary data to prevent the statements generated from llm like -> “The information is not provided”

And the pass the filtered data to the llm with prompts to generate the writeup

Then the writeup is passed to answer checker , following the old report method, that check if any wrong statement is there or not.

This process is runned after the adverse check are made.