Data Analyst - Take Home Assignment

I. SQL

The goal of this section is to prepare data from an SQL table for analysis.

The table is called 'edgar_contracts' and the schema is as follows:

[filing_id	INT (primary key)
, content	CHAR
, submission_date	DATE
, filing_company	VARCHAR(40)
, num_contracts	SMALLINT

]

The 'content' field contains the HTML of the document filed with the SEC. Each document can contain multiple contracts (or exhibits).
In this example, there are 15 exhibits (CTRL+F: "<TYPE>EX-").

- Write an SQL statement to create a new table called "exhibit_10_H1_2018" with filings submitted between January 1st, 2018 and June 30th, 2018, where the filed document has at least one exhibit-10.
  Such exhibits are specified using the tag "<TYPE>" in the content, followed by one of the following expressions: "EX-10", "EXHIBIT-10", "EX-10.*" or "EXHIBIT-10.*" where * is any number or character.
- Write an SQL statement to compute the median number of exhibits 10 contained in each document, broken down by month between January and June 2018.

II. Python & Regex

We want to extract effective dates from contracts. Assume we have a deep learning model that looks for such dates in the text of a contract (function get_eff() takes a string of text and returns the effective date in date format, or None if no effective date is found).
The goal of this section is to create a fallback solution to extract the effective date when it's missed by the model.
The logic to be implemented is to extract the first date in the first 500 words of a contract that is not surrounded by certain expressions.

- Write a python function that takes contract text as input and returns all dates contained in it. Dates need to be parsed in a standard python format.
  Dates can be mentioned in the text in different formats.
  Example: January 1st, 2018 | Jan 1st, 2018 | 01-01-2018 | 01-01-18 | 01/01/2018 | 01/01/18 | 1st day of January, 2018.

- Write a python function that takes contract text as input and returns the effective date. If the AI models fails to return an effective date, the output should be the first date in the first 500 words of the contract that's not preceded by one of the following expressions:
  "end date", "by:", "before", "expire", "expiration", "terminate", "termination".
- Bonus: Given that contract text can be the result of OCR on scanned documents, the text can contain spelling mistakes. We want the regular expression search to allow at most one mistake (one character edit - either deletion, insertion, or replacement), in searching for the dates as well as the expressions above.

III. Modeling

The goal of this section is to extract the contract value from a legal document. This is a dollar amount that appears in the text (e.g., "$1,200,000" or "1,200,000 USD")
A deep learning approach would require large training data. To build such data, we decide to use the following approach:

- Find all dollar ($ or 'USD') amounts in the text and extract the sentence containing the dollar amount, the one before and the one after
- Build a binary classifier that decides whether the extracted sentences indicate a contract value or another type of dollar amount (insurance cap, fee, etc.)

Implement the steps above. This is an open question in terms of feature engineering and model choice.