1. SELECT MODELING TECHNIQUE

As the first step in modeling, select the actual initial modeling technique. If multiple techniques are to be applied, perform this task separately for each technique.
Remember that not all tools and techniques are applicable to each and every task. For certain problems, only some techniques are appropriate. It may be that only one tool or technique is available to solve the problem at hand—and that the tool may not be absolutely the best, from a technical standpoint.

1.1. Modeling technique

Record the actual modeling technique that is used.

- Decide on appropriate technique for exercise, bearing in mind the tool selected.

In aligning our modeling approach with the project's focus on temporal reasoning within the TORQUE dataset, we are using a strategic combination of Large Language Models (LLMs) to evaluate their temporal reasoning performance.

We will utilize the following algorithms, selected for their proven strengths and potential in handling the complexities of temporal data:

1) Google FLAN-T5: This model is a refined version of the T5, which has been pre-trained on a wide range of language understanding tasks using a 'fill-in-the-blank' strategy. FLAN-T5's adeptness at contextually rich tasks makes it a prime candidate for parsing and understanding the TORQUE dataset's intricate temporal nuances.

2) BERT: Leveraging BERT's breakthrough approach in bidirectional context understanding, we aim to dissect the TORQUE dataset's rich temporal narratives. BERT's architecture is fundamentally designed to grasp the subtleties of context, which is critical in interpreting the before, during, and after of temporal events as presented in TORQUE. A pre-trained BERT model was fine tuned for a masked language modeling task on the TORQUE dataset.

3) Falcon 7B: This model is integrated to evaluate its performance on high-speed inference tasks. Given that the TORQUE dataset demands rapid understanding of complex temporal relationships, Falcon 7B's efficiency will be rigorously assessed for real-time temporal analysis.

4) Graph Generation (GPT-2-based): We used another individual's work for this task (Madaan, Yang 2021), which involved using a fine tuned version of gpt 2 for generating temporal graphs from text. By transforming textual temporal data into structured graphs, we expect to unveil patterns and relationships that are not immediately apparent through standard language modeling techniques. No training was required for this model, as we will just employ it during inference time.

The selection of these models is based on their complementary strengths in handling different aspects of temporal reasoning:

- Google FLAN-T5's generalist approach to language tasks, including those requiring an understanding of 'when'.
- BERT's deep bidirectional context analysis, crucial for the nuanced time-event relationships in TORQUE.
- Falcon 7B's speed and scalability, which are essential for processing the dataset efficiently.
- Graph Generation's ability to visualize and analyze connections within the data, offering an alternative perspective on temporal reasoning.

The overarching goal is to explore each model's ability to interpret the TORQUE dataset accurately and to generate well reasoned responses to temporal questions. This exploration will include the following:

- Fine-tuning each model with a subset of TORQUE, adjusted to their learning paradigms.
- Conducting iterative evaluations to identify and reduce temporal reasoning errors.
- Customizing training procedures to better align the models with the complex temporal queries present in the TORQUE dataset.

Through this multi-pronged approach, we anticipate a deeper understanding of the strengths and weaknesses of each model, informing our strategy for enhancing temporal reasoning across LLMs. Documentation of each step will ensure a thorough and reproducible record of our methodology and Modeling assumptions.

1.2. Modeling technique

Many modeling techniques make specific assumptions about the data.

- Define any built-in assumptions made by the technique about the data (e.g., quality, format, distribution)
- Compare these assumptions with those in the Section 2 (Describe Data) of the Data Understanding report
- Make sure that these assumptions hold and go back to the Data Preparation report, if necessary

Modeling assumptions are important as they set the foundation for the predictive accuracy of the algorithms and influence the understanding of the results we produce. Our modeling approach is based upon several key assumptions that align with the investigations conducted in the earlier phases of this project, as documented in Reports 1 through 3.

Based on the previous data preparation phase and a comprehensive understanding of the data (Report 2), we outline the following modeling assumptions for our chosen algorithms:

- Numeric and Categorical Data Handling: As established in Report 3, our data is predominantly categorical, represented by strings within the context, questions, and answers. Models such as BERT and T5-FLAN are designed to inherently manage such data types through embeddings and tokenization. Falcon 7B and our Graph Generation model are expected to handle these efficiently with pre-processing adjustments.

- Data Quality, Missing Data, Noise and Outliers: As reported, the TORQUE dataset has been thoroughly pre-processed, and any missing data is minimal and non-impactful. Our models are chosen for their robustness against incomplete inputs, ensuring stability in their predictions. The data has been rigorously prepared, cleaned, and sampled. It is assumed that the information within the dataset accurately represents real-world temporal reasoning scenarios and that any noise or outliers have been minimized.

- Data Representation: The TORQUE dataset consists predominantly of textual data. We assume that our LLMs can convert this textual data into a suitable internal representation, such as token embeddings, that captures both semantic meaning and temporal relationships.

- Class Distribution: In line with the observations from Report 2, certain temporal queries are more frequent than others. We assume our models must cope with this imbalance, and we will use techniques such as class weighting and focused training on underrepresented classes to address this.

- Negative Examples: Including questions without answers in the training set will provide the models with negative examples, as real-world scenarios often include unanswerable queries (Report 3).

- Temporal Reasoning Limitations: Our project recognizes the existing limitations in LLMs regarding temporal reasoning, as highlighted in Report 1. This acknowledgment drives the exploration of new techniques and training strategies to enhance the models' abilities.

- Algorithmic Learning: Each model's learning capabilities have been assumed to be suited for the TORQUE dataset. T5-FLAN and BERT are presumed to leverage their vast pre-trained knowledge bases effectively, Falcon 7B is expected to process the data swiftly, and the Graph Generation model is assumed to construct interpretable and informative structures from the data.

- Generalization: The models are assumed to generalize well from the training data to unseen data, maintaining high performance on temporal reasoning tasks across diverse contexts within the TORQUE dataset.

These assumptions will guide our experimental design and help frame our evaluation criteria. They are critical for the next phases of the project, as they will guide the fine-tuning, evaluation, and iteration processes. Any deviations from these assumptions will be monitored and addressed to enhance the reliability of our findings.

2. GENERATE TEST DESIGN

Prior to building a model, it is necessary to define a procedure to test the model's quality and validity. For example, in supervised data mining tasks such as classification, it is common to use error rates as quality measures for data mining models. Therefore, the test design specifies that the dataset should be separated into training and test sets. The model is built on the training set and its quality estimated on the test set. Describe the intended plan for training, testing, and evaluating the models. A primary component of the plan is to decide how to divide the available dataset into training data, test data, and validation test sets.

- Check existing test designs for each data mining goal separately
- Decide on necessary steps (number of iterations, number of folds, etc.)
- Prepare data required for test

There are no existing test designs provided by the source data provider (AllenAI) but we follow a fairly standard model fine-tuning setup using tools provided by the transformers library and Pytorch.

The following steps are taken for model testing and training:

1) The dataset is split into 3 parts: training set (70%), validation set (20%), testing set (10%). We also use the Pytorch dataloader to batch our data for more efficient training.

2) We additionally use checkpoints (created every 20 steps) to ensure that if training is disrupted, we can continue training based on a saved version of the model.

3) The test dataset will be used to test and evaluate our fine-tuned model by calculating accuracy and BLEU (further described in the evaluation section).

4) BERT Masked Language Modeling design: The correct keywords are replaced with a [MASK] token and the model is trained to fill in that masked token with an accurate predicting

5) Question/Answering design: A language model is prompted with the question and context and does a next word prediction task.

6) Graph generation for further usage in the inference/evaluation aspect of this project: We try to generate temporal graphs from the context and will attempt to pass them into question answering models to see if better responses are generated as opposed to just a text representation of the context.

The specific number of epochs and initialization parameters are further discussed in section 3, under each model that was trained.

3. BUILD MODEL

Run the modeling tool on the prepared dataset to create one or more models.

3.1. Parameter settings

With any modeling tool, there are often a large number of parameters that can be adjusted. List the parameters and their chosen values, along with the rationale for the choice.

- Set initial parameters
- Document reasons for choosing those values

1) BERT: The learning rate for this model was set to 5e-7. This model was trained for 10 epochs utilizing an adamW optimizer and a linear learning rate scheduler. BERT's default loss function, cross entropy loss, was used for training. Multiple learning rates were experimented with, and initializing the learning rate with 5e-7 was found to be the best result, but generally a lower learning rate is preferred for fine tuning tasks, to encourage more subtle weight updates. Furthermore, employing a linear learning rate scheduler is generally used in research for fine tuning BERT.

2) FLAN-T5: The learning rate for this model was set to 1e-4, a common value used for fine-tuning pre-trained models like T5. The model was trained on 3 epochs due to hardware constraints, with a batch size of 10 which provides a good balance between computational efficiency and model convergence.. These parameters were determined to work on a 2021 M1 Pro chip with 16GB of unified memory. The AdamW optimizer was also used as it is well-suited for various natural language processing tasks. Gradient clipping was also applied to stabilize the training process.

3) Falcon-7b-Instruct: This model was provided by the transformers library. It has 7 billion parameters (hence the 7b) and was fine-tuned for question-answering. This pretrained model was used as fine-tuning falcon-7b was not feasible given our current hardware and will be explored going into the the next few weeks (looking into Google Colab).

3.2. Models

Run the modeling tool on the prepared dataset to create one or more models.

- Run the selected technique on the input dataset to produce the model
- Post-process data mining results (e.g., edit rules, display trees)

- BERT fine tuning and inference
    - The following scripts were created:
        - bert.py
        - inference.py
    - bert.py saves the model weights and configuration setting in a .json file
    - inference.py loads the fine tuned model from the .json files and runs inference on a few examples from the dataset
    - Inference post processing:
        - Since BERT was trained a masked language modeling task during inference its required to specify the indices of text that represent the masked token, you wish to predict.
- GPT 2 graph generation
    - The following scripts are run:
        - run_generation.py
    - run_generation.py takes document text as input (in the form of .jsonl files) and uses a fine tuned version of gpt 2 to generate temporal graphs representing document text. A textual representation of a graph is written to a user specified output file.
- FLAN-T5
    - The following scripts were created:
        - t5.py
    - t5.py does the following:
        - trains flan-t5-small on the temporal dataset
        - saves the model weights and tokenizers
        - evaluates the performance of the model
        - runs inference on an example question-answer pair

- Falcon 7b
    - The following notebook was created:
        - falcon7b_inference.ipynb
    - falcon7b_inference.ipynb accomplishes the following:
        - utilizes a pretrained falcon7b-instruct model (from the transformers library)
        - run inference based on the context/question/answer format created through the data preprocessing
        - determine the number of matched expected response words

3.3. Model description

Describe the resulting model and assess its expected accuracy, robustness, and possible shortcomings. Report on the interpretation of the models and any difficulties encountered.

- Describe any characteristics of the current model that may be useful for the future
- Record parameter settings used to produce the model
- Give a detailed description of the model and any special features
- For rule-based models, list the rules produced, plus any assessment of per-rule or overall model accuracy and coverage
- For opaque models, list any technical information about the model (such as neural network topology) and any behavioral descriptions produced by the modeling process (such as accuracy or sensitivity)
- [Optional] Describe the model's behavior and interpretation
- [Optional] State conclusions regarding patterns in the data (if any); sometimes the model reveals important facts about the data without a separate assessment process (e.g., that the output or conclusion is duplicated in one of the inputs)

1) BERT, specifically the bert base uncased version is an encoder transformer model pretrained on a large corpus of English data in a self-supervised way. This model has two objectives- masked language modeling (MLM) and next-word prediction. The masked language prediction objective was leveraged for this work. Since this model was fine-tuned on English text, English-based data is a requirement for fine-tuning this model. In the future we could try fine tuning or use a fine tuned version of this model for question answering, which is similar to the next word prediction task, but for this project we took a MLM approach as it was the most simple and applicable to the data we were using.

2) FLAN-T5

FLAN-T5, is a variant of the T5 (Text-to-Text Transfer Transformer) model that has been fine-tuned on a diverse set of tasks, including question answering, natural language inference, and sentence completion. The primary goal of FLAN-T5 is to improve the model’s performance on a wide range of NLP tasks by leveraging transfer learning. By fine-tuning the model on multiple tasks simultaneously, FLAN-T5 learns to adapt its knowledge and skills to various domains and problem types.

The “google/flan-t5-small” model is the smallest configuration at 77m parameters. It strikes a balance between performance and computational efficiency, making it an attractive choice for running NLP tasks with hardware constraints. T5ForQuestionAnswering is a specialized version of the T5 model that includes a question-answering head on top of the pre-trained architecture. This head is designed to predict the start and end positions of the answer span within the given context.

By combining the power of the “google/flan-t5-small” model with the T5FOrQuestionAnswering architecture, we create an effective question-answering system. The model’s ability to understand and reason about text, coupled with its task-specialized head, allows it to accurately locate and extract the most relevant answer from the given context.

3) Falcon 7b, the smallest of the Falcon models, is a decoder only transformer model pretrained on the RefinedWeb dataset containing various English language data from different parts of the web. This model is primarily intended for text generation in the form of either open text generation or question answering. Like the other models, this is also an English language model so the base requirement for fine-tuning and inference is English data. The specific version of the model we have used for our assessment, falcon7b-instruct, has been specifically trained for question answering however seems to be biased more toward text generation over context based answering.

4. ASSESS MODEL

The model should now be assessed to ensure that it meets the data mining success criteria and passes the desired test criteria. This is a purely technical assessment based on the outcome of the modeling tasks.

4.1. Model assessment

Summarize results of this task, list qualities of generated models (e.g., in terms of accuracy), and rank their quality in relation to each other.

- Evaluate results with respect to evaluation criteria
- Test result according to a test strategy (e.g.: Train and Test, Cross-validation, bootstrapping, etc.)
- Compare evaluation results and interpretation
- Create ranking of results with respect to success and evaluation criteria
- Select best models
- Interpret results in business terms (as far as possible at this stage)
- [Optional] Get comments on models by domain or data experts
- Check plausibility of model
- Check effect on data mining goal
- Check model against given knowledge base to see if the discovered information is novel and useful
- Check reliability of result
- Analyze potential for deployment of each result
- If there is a verbal description of the generated model (e.g., via rules), assess the rules: Are they logical, are they feasible, are there too many or too few, do they offend common sense?
- Assess results
- Get insights into why a certain modeling technique and certain parameter settings lead to good/bad results

Finetuned BERT Model with raw text input examples:
Note: top 3 responses for this model are shown

Context:
"The death of veteran ultranationalist opposition leader Alpaslan Turkes will shake up right-wing politics in Turkey,

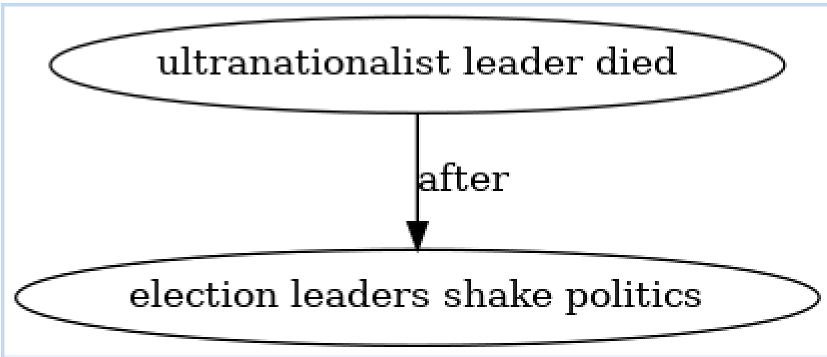
as his party is likely to disintegrate with his demise, political analysts said Saturday. Turkes, leader of the Nationalist Action Party, a small but influential political group, died here on early Saturday after a heart attack."
Question:
"What event has already finished?"
Model response:
Predicting the masked token “comments”: ['appointment', 'resignation', 'election']
Predicting the masked token “came”: ['came', 'occurred', 'was']
Predicting the masked token “arugued”: ['said', 'argued', 'stated']

Context: “At least 50,000 U.S. troops are expected to be committed to Desert Shield within weeks, including Marines, Army air assault forces, paratroopers and infantry. Iraq said it invaded Kuwait because of disputes over oil and money.”
Question: “What event has already happened?”
Model response:
Predicting the masked token “disputes”: ['disputes', 'concerns', 'disagreements']
Predicting the masked token “said”: ['claims', 'claimed', 'said']

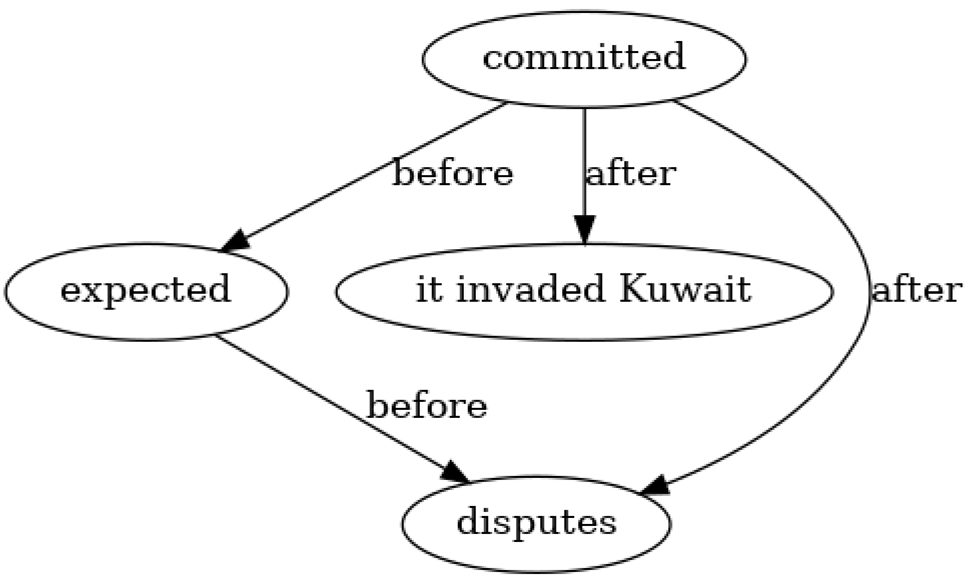
Context: “Thomas and Nuggets coach George Karl had an argument during a summer league meeting. Before the fight, Anthony had a remarkable game, demoralizing the Knicks with easy drives to the basket.”
Question: “What did Anthony do before the Knicks were demoralized?”
Predicting the masked token “dives”: ['shots', 'access', 'hits']

Note: The results of the BERT model (without fine tunning) were very similar to the above, meaning all the top 3 words were the same but occasionally the order was different. The lack of difference between the fine tuned and base model could be because bert was not fine tuned with enough data or a different approach needs to be taken to address this task.

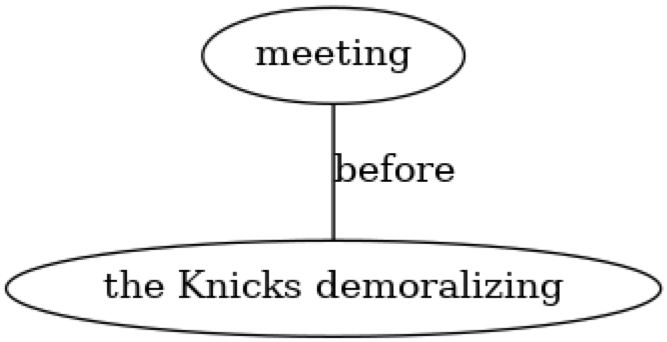
Generated graphs from finetuned GPT2:
Context:
“The death of veteran ultranationalist opposition leader Alpaslan Turkes will shake up right-wing politics in Turkey, as his party is likely to disintegrate with his demise, political analysts said Saturday. Turkes, leader of the Nationalist Action Party, a small but influential political group, died here on early Saturday after a heart attack."



context: “At least 50,000 U.S. troops are expected to be committed to Desert Shield within weeks, including Marines, Army air assault forces, paratroopers and infantry. Iraq said it invaded Kuwait because of disputes over oil and money.”



Context: “Thomas and Nuggets coach George Karl had an argument during a summer league meeting. Before the fight, Anthony had a remarkable game, demoralizing the Knicks with easy drives to the basket.”



FLAN-T5 Inference:
Context: “After developing the World Wide Web, Tim Berners-Lee founded the World Wide Web Consortium (W3C) at MIT in 1994.”
Question: “What happened after an event?”
Expected: “founded”
Untrained Model Response: “After”
Trained Model Response: “founded”
Context: “Grace Hopper reported the first computer bug in 1947 when a moth was found in the Harvard Mark II.”
Question: “What events have already finished?”
Expected: “reported”
Untrained Model Response: “pper reported the first computer”
Model Response: “reported”
Context: “Linus Torvalds released the first version of the Linux kernel on the Usenet newsgroup comp.os.minix on September 17, 1991.”
Question: “What events have already finished?”
Expected: “released”
Untrained Model Response: “”
Trained Model Response: “released”
The inference results indicate an accurate and directly relevant understanding of temporal relations given the context and question. The benefits of the training are much more apparent when compared to the untrained output, which is more or less complete nonsense. As one of the smaller models at 77m params, this performance is to be expected.

Falcon 7b Inference on TORQUE Dataset:

Context:
“A respected figure and twice a member of the Chinese People's Political Consultative Congress, Cai has used his scientific achievements to reach practical results for wetland conservation, defending the idea that 'wetlands are closely linked to human survival and development.' Cai also stressed the importance of wetlands for sustainable economic development and human well-being, especially in their function as the 'kidneys of nature.' ”

Question:
What happened before conservation?
Model Response:
"In the early stages before the 21st century, Cai had been deeply involved in the development and construction of wetlands, and his work has been a key factor in promoting China's ecological progress.
In the 1950s, Cai started research into the conservation and restoration of wetlands and lakes, which was one of China's most important scientific achievements in wetland conservation.
He is one of the most important people to be involved in the conservation of wetlands and lakes in China, and the Chinese government has also made great efforts in the protection."
Key Words found in Correct Response:
['reach', 'used']

Context:
"During the journey, Chinese and the U.S. navy held their first joint maritime exercise."
Question:
What happened during the joint maritime exercise?
Model Response:
"Chinese Navy and the US Navy held a joint maritime exercise in the western Pacific on 5 May 2010.
Chinese Navy's J-15 fighter jets took part in the exercise with the USS John C Stennis. "
Key Words found in Correct Response:
[N/A]

Context:
"Canada's largest department store operator said the rights offering will entitle holders of its ordinary shares, except residents in the U.S.and Britain, to subscribe for two additional shares for every five shares held at a price of C$31.25 a share. The record date is Nov.9"
Question:
What will happen in the future?
Model Response:
"The stock has been trading down for the last three-four days.
Will it be a short term trade? Or a long term investment?
As an investment, a lot depends on whether it is trading at a discount compared to other retailers or not.
If it is trading at a discount and you can see a long term upturn, then it may make a good long term investment if it goes down in price.
If it is trading at a discount, then it may be worth investing in this stock. "
Key Words found in Correct Response:
['entitle', 'subscribe', 'held']

Further evaluation plan:
We took a random sample of our data to run evaluations on the model results.

We are considering using more evaluation metrics to take a more thorough look at the performance of our models. Some of the methods we are in the process of implementing include simple keyword match and determining where the correct answers were positioned in the ranked outputs of the model.
We also explored using ROUGE and BLEU scores, but have decided not to move forward with implementing either method. Regarding ROUGE, we decided not to use this metric because we believe it is better suited for lengthier generated texts instead of question-and-answer outputs. This is a similar reason as to why we are not using BLEU score; we do not have multiple reference texts to compare the output to. These scores did not prove to be as informative to us as other methods we tried.

4.2. Revised parameter settings

According to the model assessment, revise parameter settings and tune them for the next run in the Build Model task. Iterate model building and assessment until you find the best model.

- Adjust parameters to produce better models.

In the next stage of the project, we plan to integrate the graphs into our language models to see if any further improvements can be made.

The only model we have where a rigorous parameter search would apply to is Falcon 7b. We did not have time to do a rigorous parameter search for that model during this phase of the project, as it is computationally expensive but we plan to do that in the future.