# Event Reasoning with Explicit Time and Space


| 1st Ananth Rangarajan | 2nd Tanmai Mukku | 3rd Devadutt Sanka | 4th Roop Sumanth Gundu | 5th Nikhil Bodela |
|---|---|---|---|---|
| 1225285633 | 1229609006 | 1225362138 | 1226662413 | 1229975320 |
| arangar9@asu.edu | tmukku@asu.edu | dsanka@asu.edu | rgundu2@asu.edu | nbodela@asu.edu |



***Abstract*—In the realm of Natural Language Processing (NLP), the comprehension and interpretation of textual data are pivotal. One fundamental aspect of this comprehension involves the analysis of events, which are specific occurrences or incidents that transpire at distinct times and locations. These events are intimately intertwined with temporal and spatial information, where the former relates to the timing of an event, and the latter pertains to the physical space where it unfolds. For an intelligent NLP system to truly understand and reason about text, it must be equipped with the capabilities to decipher and manipulate these temporal and spatial intricacies. In this context, we propose an NLP task that merges event reasoning with the precise handling of temporal and spatial changes. This task revolves around the analysis of short stories that feature various events, each accompanied by explicit temporal and spatial details. The challenge lies in successfully answering questions that demand adept temporal and spatial reasoning, thus pushing the boundaries of natural language understanding and cognitive reasoning in artificial intelligence.**



***Index Terms*—Huggingface, Transformers, BERT, AutoModel, Tokenizer, Zero-Shot**


## I. Problem Statement

The problem we aim to address in this research pertains to the nuanced and intricate nature of event comprehension in natural language text. Events, being the building blocks of narratives, are central to understanding the dynamics of a story or description. These events are not isolated occurrences; they are intricately linked with both temporal and spatial dimensions. Temporal information encapsulates the when of an event, signifying the exact moment in time when it unfolds. Spatial information, however, encapsulates the where, signifying the specific location or space where an event takes place. The challenge arises in that, to fully grasp the context and meaning of a text, an intelligent NLP system must not only identify these events but also possess the ability to reason about their temporal and spatial attributes.

In the current NLP landscape, the capabilities of large language models (LLMs) such as GPT-4 and ChatGPT have demonstrated a significant stride in reasoning about the temporal and spatial information embedded in textual contexts. These models, boasting extensive architectures and a plethora of hyperparameters, have exhibited a remarkable ability to discern and comprehend the intricate details of events in a text. Their capacity to perform temporal and spatial reasoning is rooted in their capability to capture subtle nuances, contextual relationships, and temporal sequences, all of which are essential for a comprehensive understanding of narratives. Conversely, smaller models like the vanilla BERT and other pre-trained variations of BERT, while undoubtedly effective in various NLP tasks, often fall short in terms of their capacity to learn and reason about temporal and spatial intricacies within the text. The primary limitation lies in the sheer difference in the scale of architecture and the number of hyperparameters used during the training of these models. LLMs, like GPT-4 and ChatGPT, are designed with a more extensive focus on context modeling, enabling them to grasp the underlying temporal and spatial dynamics within events.

The primary objective of this project is to enhance the temporal and spatial reasoning capabilities of smaller language models, such as the pre-trained BERT for Multiple Choice and the Auto Model for Multiple Choice. To achieve this goal, the project focuses on the crucial step of training these models with an extensive and diverse data corpus. By exposing these smaller models to a larger and more varied set of textual contexts, we aim to empower them with the capacity to effectively reason about temporal and spatial information. This targeted training approach seeks to bridge the gap in performance between smaller models and their larger counterparts, such as GPT-4 and ChatGPT, ensuring that even the more compact NLP models can adeptly handle temporal and spatial intricacies within textual data.

## II. Approach

In this research project, we outline a systematic approach designed to empower smaller language models, specifically BERT (Pre-Trained Models), with the ability to proficiently reason about temporal and spatial information within natural language text. To begin, we employ GPT-4, a state-of-the-art large language model, to generate topics that inherently involve temporal and spatial dimensions. These topics serve as the foundation for the subsequent steps in our approach. The topics generated for further steps in our approach are as follows:

- Natural and Environmental Events
- Sports and Athletic Events
- Entertainment and Media Events
- Travel and Tourism Events
- Population and Demographic Events
- Political and Governmental Events
- Climate and Weather Events
- Medical and Health Events
- Human and Social events
- Cultural and Artistic Events
- Historical Events

- Religious and Spiritual Events

In the next phase, we prompt GPT-4 to create succinct narratives, each encompassing 5 to 6 lines, which describe events while providing explicit temporal and spatial details. Additionally, we instruct GPT-4 to craft questions, generate answer choices, provide the correct answer, and furnish a reasoning component for each generated narrative, thus creating a comprehensive dataset.

This training dataset contains 2400 data points, with each entry consisting of a story, a corresponding question, multiple answer choices, the correct answer, and reasoning. By encompassing a wide array of topics and scenarios mentioned above, this ensures the diversity and robustness required for effective training. We have collected approximately 200 MCQ pairs for each of the above topics. We have also curated close to 240 MCQ pairs which will be our testing dataset. The structure of each item of the training and validation dataset is provided in the figure 1. We have made sure that there is no data leakage so that there is no influence of the training set on the model while predicting on test set. We validate the quality and alignment of the generated testing dataset with the temporal and spatial objectives of the project, by performing manual annotation, which serves as a critical quality control measure.


```
{
    "Answer Choices": [
        "1st July",
        "10th August",
        "early October",
        "December"
    ],
    "Answer": "1st July",
    "Story": "The peaceful town of Lakewood, nestled between mountains and a river,
        was hit by a devastating earthquake at 9 am on 1st July. After struggling for a month,
        they could finally manage to get the town's basic infrastructure up and running,
        only to face a flood caused by incessant rains on 10th August. The flood swept away
        the town's newly repaired bridge, disconnecting Lakewood to the nearby cities.
        In early October, a wildfire engulfs the surrounding mountain area nicely recovering
        from the earthquake and floods. Finally, chilling winter winds in December froze
        the river, hindering the town's fishing businesses.",
    "Question": "When did the earthquake occur?",
    "Reasoning": "The story mentions that the earthquake occurred on 1st July."
}
```


Fig. 1. Structure of data

Once we ascertain the fidelity of the dataset, we proceed to train smaller language models such as BertForMultipleChoice and AutoModelForMultipleChoice using this extensive corpus, aiming to equip them with the capability to comprehend and reason about temporal and spatial intricacies within textual contexts. These models are pre-trained BERT models, which will be trained and tested on the large data corpus generated using GPT-4. Subsequently, the trained models undergo a fine-tuning process, refining their accuracy and performance in tasks associated with temporal and spatial reasoning. This iterative step is vital to ensure that the models reach a level of proficiency aligned with the objectives of the project. Once the complete prediction is completed, we perform error analysis where we try to qualitatively analyze those samples from the testing dataset where the model failed to predict the right answer.

## III. Implementation

To get started with the implementation of the above-mentioned approach, we decided to make use of Google Colab as the primary workspace due to the easy access to GPU for our model training and testing. As part of the next stage, we performed a small survey on the different huggingface models that could be used for this task and finally came to a consensus on the BertForMultipleChoice and AutoModelForMultipleChoice as the 2 pre-trained BERT models for this Multiple Choice Question Answering task. The next phase in our pipeline was to perform Zero-Shot prediction using the 2 models. In Zero-Shot prediction, we evaluate the 2 models BertForMultipleChoice and AutoModelForMultipleChoice directly on the testing dataset without training them.

As part of the actual training phase, we read the training dataset, we pre-process the data, which involves performing tokenization on the input data. Then we perform a train-val split on the data by considering 10% of the data as a validation dataset. Once the train-val dataset is ready, we then build the model by providing the necessary input parameters such as batch_size, epochs, num_labels, etc. In the next stage, we train the model and compute the loss, and accuracy for each of the epochs. Once the training phase is completed, we perform the prediction on the testing dataset. Before the prediction, we follow the same steps as done for the training dataset, such as reading the dataset, pre-processing the dataset, and tokenization of the dataset.

The implementation of our model training leveraged the BERT architecture. We initialized the training process with a predefined number of epochs set to 10, following the recommendation of the BERT authors to use 2 to 4 epochs, but experimenting to assess the impact of a higher epoch count. The training involved a total number of steps calculated as the product of the number of batches and the number of epochs. We utilized a linear scheduler for learning rate adjustments, with no warmup steps.

GPU acceleration was employed to expedite the training process, checking for the availability of a GPU and defaulting to CPU if none was available. In our case, a Tesla T4 GPU was utilized. We also defined a function, `flat_accuracy`, to calculate the accuracy of our predictions. For time tracking and formatting, the `format_time` function was implemented.

Ensuring reproducibility, we set a random seed using the Python libraries `random`, `numpy`, and `torch`. The training process involved iterating over our training data, with progress updates every 40 batches. We used gradient clipping to prevent the exploding gradient problem in our neural network. After training, the model was evaluated using a validation dataset. Statistics such as training and validation loss, accuracy, and time taken per epoch were collected and reported. This thorough process allowed us to monitor the model's performance and make necessary adjustments, leading to a comprehensive and robust training procedure.

TABLE I
RESULTS OBSERVED FOR ZERO-SHOT PREDICTION

| Task | Model | Accuracy |
|---|---|---|
| Zero Shot Prediction | Bert Model | 0.27 |
| | Auto Model | 0.23 |

TABLE II
TRAINING AND VALIDATION RESULTS

| Epoch | Train.Loss | Valid.Loss | Valid.Acc |
|---|---|---|---|
| 1 | 1.30 | 1.26 | 0.49 |
| 2 | 1.15 | 1.09 | 0.50 |
| 3 | 0.92 | 1.07 | 0.58 |
| 4 | 0.73 | 1.20 | 0.58 |
| 5 | 0.60 | 1.26 | 0.60 |
| 6 | 0.51 | 1.41 | 0.62 |
| 7 | 0.42 | 1.79 | 0.60 |
| 8 | 0.37 | 1.81 | 0.62 |
| 9 | 0.31 | 1.92 | 0.62 |
| 10 | 0.27 | 2.06 | 0.61 |

## IV. RESULTS

As part of the initial phase of this project, we performed zero-shot prediction using the BertForMultipleChoice and AutoModelForMultipleChoice. In this phase, we pass the testing dataset to the models and perform prediction tasks without any prior training. The results obtained for the zero-shot prediction are as shown in the table I. The complete evaluation metrics for both Bert and Auto models are provided in the figures 2 and 3

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.18 | 0.28 | 0.22 | 43 |
| 1 | 0.33 | 0.24 | 0.28 | 86 |
| 2 | 0.44 | 0.27 | 0.33 | 71 |
| 3 | 0.18 | 0.30 | 0.23 | 40 |
| accuracy | | | 0.27 | 240 |
| macro avg | 0.28 | 0.27 | 0.26 | 240 |
| weighted avg | 0.31 | 0.27 | 0.28 | 240 |

Fig. 2. Evaluation metric for Bert Model on Zero-Shot prediction task

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.17 | 0.23 | 0.20 | 43 |
| 1 | 0.39 | 0.24 | 0.30 | 86 |
| 2 | 0.26 | 0.21 | 0.23 | 71 |
| 3 | 0.13 | 0.23 | 0.16 | 40 |
| accuracy | | | 0.23 | 240 |
| macro avg | 0.24 | 0.23 | 0.22 | 240 |
| weighted avg | 0.27 | 0.23 | 0.24 | 240 |

Fig. 3. Evaluation metric for Auto Model on Zero-Shot prediction task

We then performed training on the training data and recorded the training and validation loss for the 10 epochs.

TABLE III
TESTING RESULTS ON TEST DATASET

| Task | Model | Accuracy |
|---|---|---|
| Evaluation Post Training | Bert | 0.52 |
| | Auto Model | 0.56 |

The training and validation accuracy obtained on the training dataset is represented in the table II. The graph showing the loss is shown in figure 4. Finally, we tested the 2 models - BertForMultipleChoice and AutoModelForMultipleChoice on the testing dataset to predict the answers from the multiple choices for the context-question pairs, and the results of the prediction are provided in the table III.

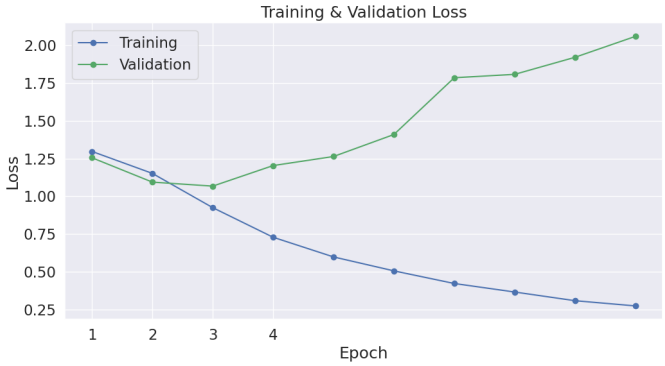


Fig. 4. Graph showing the training and validation loss

## V. ERROR ANALYSIS

We conducted a detailed qualitative analysis of 34 errors made by our model. This analysis provides insight into the specific challenges presented by event reasoning tasks. A breakdown of these errors reveals the following trends, further detailed in **??** of the supplementary materials:

- **Incorrect Date Selection:** This was the most common error type, accounting for 45% of the errors. The model often failed to correctly identify dates or times within the context, especially when multiple events were described in close proximity.
- **Misunderstanding Order of Events:** Approximately 30% of the errors were due to the model's inability to accurately track the sequence of events. These errors were particularly prevalent in narratives that involved complex temporal shifts or a series of closely occurring activities.
- **Difficulty with Indirect References:** Making up 25% of the errors, this category includes instances where the model struggled with temporal inferences that were not explicitly stated. For example, the model might incorrectly interpret "the following day" or "the day before" in relation to other given dates.

Examples of each error type are provided in **??**, illustrating how the model might misinterpret a complex event timeline or overlook subtle cues within the text. The error analysis

underscores the need for improved temporal understanding and sequence modeling in our NLP tasks.

TABLE IV
SUMMARY OF COMMON ERROR TYPES

| Error Type | Percentage |
|---|---|
| Incorrect Date Selection | 43% |
| Misunderstanding Order of Events | 25% |
| Difficulty with Indirect References | 21% |

TABLE V
EXAMPLES OF ERRORS FOR EACH TYPE

| Error Type | Example |
|---|---|
| Incorrect Date Selection | *Story:* "Alice attended a conference in Paris on March 1st. She then flew to Berlin for a meeting on March 3rd." *Question:* "When did Alice go to Berlin for her meeting?" *Correct Answer:* "March 3rd" *Model's Prediction:* "March 1st" |
| Misunderstanding Order of Events | *Story:* "John visited the museum on Friday and went to the theater on Saturday." *Question:* "Which did John visit first, the museum or the theater?" *Correct Answer:* "Museum" *Model's Prediction:* "Theater" |
| Difficulty with Indirect References | *Story:* "Linda had an appointment on the day after Monday. She then had a meeting two days before Friday." *Question:* "On what day was Linda's meeting?" *Correct Answer:* "Wednesday" *Model's Prediction:* "Tuesday" |

## VI. INDIVIDUAL CONTRIBUTIONS

Ananth - contributing significantly to various aspects of the endeavor. Instrumental in generating a diverse array of topics using GPT-4, establishing the foundation for subsequent research. Generated a substantial dataset comprising 1000 data points, with each data point encompassing a comprehensive narrative, questions, answer choices, answers, and reasoning components. This extensive dataset spanned a wide spectrum of topics, including Natural and Environmental Events, Sports and Athletic Events, Entertainment and Media Events, Travel and Tourism Events, and Population and Demographic Events, with 200 data points dedicated to each category. Researched the most suitable smaller models for this research, ultimately recommending the utilization of BertForMultipleChoice [6] or AutoModelForMultipleChoice [7] models for the task at hand. Performed essential preprocessing techniques on the dataset, aligning it with the suitable format required for training the transformer models. Carried the Zero-Shot prediction on the testing dataset using the 2 models - BertForMultipleChoic and AutoModelForMultipleChoice. Built the complete pipeline starting from reading the data, preprocessing the data, training using the small language models, and finally evaluating these models on the testing data.

Tanmai - In this project, my contributions have been multifaceted and critical to the advancement of our event reasoning objectives. **I generated a comprehensive dataset** using **OpenAI's GPT-4**, which includes a diverse array of stories from domains such as history, social movements, culture, religion, and arts, ensuring our models are exposed to a wide spectrum of event-driven narratives. This dataset was meticulously cleaned and preprocessed, subsequently formatted into a JSON structure for seamless future usage. **I conducted a thorough validation process** to maintain data integrity by excluding irrelevant samples. Concurrently, **I engaged in an extensive literature review**, exploring seminal works from prominent NLP conferences like ACL, NAACL, and EMNLP to inform our approach and align with state-of-the-art practices. Further, **I conducted a detailed error analysis** to identify and understand the model's limitations, providing a clear direction for subsequent improvements. My efforts also extended to **implementing the training code for the BERT model**, which involved fine-tuning to sharpen its event reasoning based on spatiotemporal contexts. This included hyperparameter optimization to enhance the model's temporal relationship understanding. An essential part of my role involved conducting a **detailed error analysis**, which allowed us to pinpoint the model's shortcomings and areas for potential enhancements. These findings have laid the groundwork for future extensions and improvements that I plan to explore, such as postprocessing and the integration of regular error feedback mechanisms in addition to data augmentation to better capture temporal relationships in textual data.

Devadutt Sanka - I used the OpenAI API to generate a dataset comprising nearly 1000 data points, incorporating around 20 diverse event types such as Health and Medical, Educational and Academic, Arts and Entertainment, Cultural and Heritage events, and many more. This was achieved through the utilization of varied prompts for questions, answers, and reasoning. I ensured the data's uniqueness, extracted the necessary components, and formatted everything in JSON. My responsibilities also included closely working with my colleagues, Ananth and Tanmai, to seamlessly integrate our individual contributions and create a synergistic work environment. I also took extra steps to check and improve my part of the data. I helped create a special set of data to test our project and made sure everything was correct and followed our rules. My collaboration extended to working closely with Ananth and Tanmai in the integration and application of the Bert model. I contributed significantly to the zero-shot prediction on the test data using the Bert model, a critical phase in our project. Additionally, my involvement in evaluating the test results and conducting error analysis was instrumental in identifying and rectifying potential issues, thereby enhancing the overall quality and effectiveness of our project.

Roop Sumanth - I mostly supported and followed through

with my team, Ananth, Devadutt and Tanmai. I didn't lead much but ensured their work smoothly came together. Specifically, I helped in integrating, ensuring everything flowed well in our project.

## REFERENCES


[1] Zhou, Ben, Richardson, Kyle, Ning, Qiang, Khot, Tushar, Sabharwal, Ashish, and Roth, Dan. Temporal Reasoning on Implicit Events from Distant Supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1361–1371, Online, June 2021. Association for Computational Linguistics. https://aclanthology.org/2021.naacl-main.107. DOI: 10.18653/v1/2021.naacl-main.107.

[2] Han, Rujun, Hsu, I-Hung, Sun, Jiao, Baylon, Julia, Ning, Qiang, Roth, Dan, and Peng, Nanyun. *ESTER*: A Machine Reading Comprehension Dataset for Reasoning about Event Semantic Relations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 7543–7559, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. https://aclanthology.org/2021.emnlp-main.597. DOI: 10.18653/v1/2021.emnlp-main.597.

[3] Spiliopoulou, Evangelia, Pagnoni, Artidoro, Bisk, Yonatan, and Hovy, Eduard. *EvEntS ReaLM*: Event Reasoning of Entity States via Language Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 1982–1997, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. https://aclanthology.org/2022.emnlp-main.129. DOI: 10.18653/v1/2022.emnlp-main.129.

[4] Mirzaee, Roshanak, Rajaby Faghihi, Hossein, Ning, Qiang, and Kordjamshidi, Parisa. *SPARTQA*: A Textual Question Answering Benchmark for Spatial Reasoning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4582–4598, Online, June 2021. Association for Computational Linguistics. https://aclanthology.org/2021.naacl-main.364. DOI: 10.18653/v1/2021.naacl-main.364.

[5] Zhang, Li, Lyu, Qing, and Callison-Burch, Chris. Reasoning about Goals, Steps, and Temporal Ordering with WikiHow. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4630–4639, Online, November 2020. Association for Computational Linguistics. https://aclanthology.org/2020.emnlp-main.374. DOI: 10.18653/v1/2020.emnlp-main.374.

[6] Huggingface Bert model for Multiple Choice question answering task

[7] Huggingface AutoModel for Multiple Choice question answering task.