Q: What wild animal will be found at the Outer banks of North Carolina in September 2019? **Choices: Horses** (*answer*), Cows, Turtles, Donkeys.
Article: *Tillis Introduces Legislation to Protect Corolla Wild Horses Washington: Office of the Senator Thom Tillis has issued the following news release:* (1/29/19) U.S. Senator Thom Tillis (R-NC) introduced the Corolla Wild Horses Protection Act, legislation that would provide responsible management of the wild horse population around Corolla, North Carolina and the Outer Banks. Representative Walter Jones (R-NC) introduced companion legislation in the House of Representatives in previous Congresses and has been a long time champion of protecting the Corolla wild horse population.
Reasoning Process: The Corolla Wild Horses Protection Act will make people to protect the wild horses (**forecasting skills - causal relations**). If people start to protect the wild horses from January, the wild horses will be found in September (**forecasting skills - inferring based on past events** - *we can find the answer from this part*). Horse is an animal (**commonsense - world knowledge**). The Outer banks of North Carolina = North Carolina and the Outer Banks (**language understanding - paraphrase**).

Table 7: Detailed example to show how to solve a question.

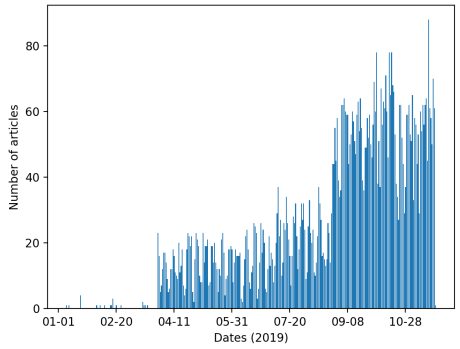

Figure 10: Date distribution of gold articles for questions. Each question is made from gold articles. The dates denote release dates of news articles and they range from 01-01-2019 to 11-31-2019.

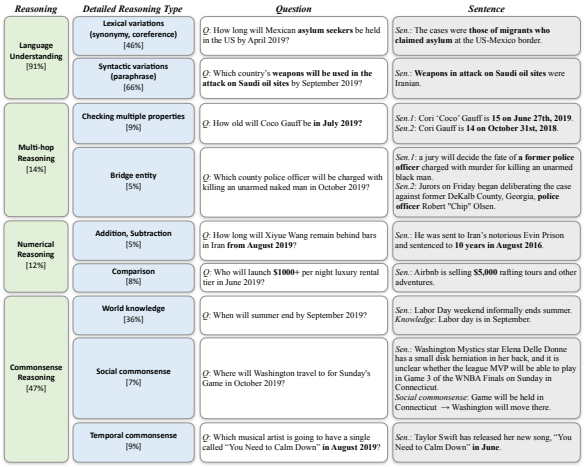

Figure 11: Examples of each type of reasoning in FORECASTQA. Words relevant to the corresponding reasoning type are bolded. Also, [%] represents the percentage of questions that requires the reasoning type.

tion does not have a temporal phrase, then we filter out the question.

B Example of Reasoning

Table 7 shows an example of reasoning process to solve a question.

Measurement	Value
Average question length (tokens)	13.85
Average answer length (tokens)	2.46
# of distinct words in questions	17,521
# of distinct words in choices	5,187
# of distinct time stamps associated w. questions	218
Average gold article length (# tokens)	720.21
Maximum question time stamp	2019-11-22
Minimum question time stamp	2019-01-01

Table 8: Statistics of FORECASTQA.

C Additional Reasoning Types

Figure 11 shows additional reasoning types.

Language Understanding. We introduce lexical variations and syntactic variations following Rajpurkar et al. (2016, 2018). Lexical variations represent synonyms or coreferences between the question and the evidence sentence. When the question is paraphrased into another syntactic form and the evidence sentence is matched to the form, we call it syntactic variation. We find that many questions require language understanding; lexical variations account for 46% and syntactic variations do for 66%.

Multi-hop Reasoning. Some questions require multi-hop reasoning (Yang et al., 2018), such as checking multiple properties (9%) and bridge entities (5%) . The former one requires finding multiple properties from an article to find an answer. The latter one works as a bridge between two entities, where one must identify a bridge entity, and find the answer in the second hop.

Numerical Reasoning. To answer our questions, one needs numerical reasoning (Dua et al., 2019). The answer is found by adding or subtracting two numbers (5%), or comparing two numbers (8%) in the given articles.

Commonsense Reasoning. The questions require world knowledge (Talmor et al., 2019), social commonsense (Sap et al., 2019), and temporal commonsense (Zhou et al., 2019). To solve these questions, an AI agent must leverage assumed common knowledge in addition to what it finds in the news corpus. We find that 36% questions need world knowledge and 7% questions require social commonsense. The other type of commonsense reasoning is temporal commonsense, which is related to temporal knowledge (Zhou et al., 2019). 9% questions are related to temporal commonsense.

D Statistics

Tables 8 and 9 show the statistics and answer types in FORECASTQA.

Answer Type	%	Examples
Yes/No	56.8%	-
Person	8.1%	Boris Johnson, Mark Zuckerberg
Group/Org	5.8%	BBC, United Nations, EU
Location	8.0%	Canada, Iran, U.S.
Date/Time	1.6%	January, July
Number	6.7%	530, Thirty eight
Other Entity	1.1%	Boeing 737
Common Noun Phrase	5.8%	A hurricane, Asteroid dust
Verb Phrase	3.1%	Defend his innocence
Adjective Phrase	1.4%	Cruel and Misguided, Due to the bad weather
Sentence	1.6%	Liverpool will become the first English team to play their 400th international game.

Table 9: Types of answers in FORECASTQA.

E Experiments

E.1 Details on a Text Encoder

We use Huggingface's codes[11]. We chose the best learning rate among $\{3e-5, 1e-5, 5e-6\}$ and the number of epochs is 3. We set the max sequence length to 512.

E.2 Details on IR methods

We index the English news articles with Elasticsearch (Gormley and Tong, 2015). We followed the setups in Qi et al. (2019). We use Elasticsearch's simple analyzer which performs basic tokenization and lowercasing for the title. We use the standard analyzer which allows for removal of punctuation and stop words from the body of articles. At retrieval time, we use a `multi_match` query in the Elasticsearch against all fields with the same query, which performs a full-text query employing the BM25 ranking function (Robertson et al., 1995) on all fields, and returns the score of the best field for ranking. To promote documents whose title matches the search query, we boost the search score of any result whose title matches the search query by 1.25, which results in a better recall for entities with common names.

E.3 Details on Baselines.

We consider following baselines: (1) **Event-based approaches**: We test event-based approach, BERT with event triples (two entities and a relation between them) and BERT based on SAM-Net (Lv et al., 2019) for our setup. It is non-trivial to apply the event-based approaches to our setup. Thus, we preprocess the retrieved news articles into event triples (subject, relation, object) using Liu et al. (2019a). We simply regard them as text, we concatenate the triples, and feed them into BERT and call it **BERT with event triples**. In addition, we apply a script learning approach (SAM-Net (Lv et al., 2019)) to our setup. A question and choices are not used in their original method; thus we encode them using BERT and concatenate the encodings with the approach's final representation. This representation is fed into a linear layer and the linear layer predicts whether the choice is correct or not. We used $\text{BERT}_{\text{LARGE}}$ for the former one and $\text{BERT}_{\text{BASE}}$ for the latter one. (2) **ESIM** (Chen et al., 2017b). An NLI model, where we change their output layer so that the model outputs probabilities for each answer choice with a softmax layer. We use ELMo (Peters et al., 2018) for word embeddings. (3) **BIDAF++** (Clark and Gardner, 2018). The model requires context, and thus we use a top-1 article by an IR method. We augment it with a self-attention layer and ELMo representations. To adapt to the multiple-choice setting, we choose the answer with the highest probability. The input to ESIM is a question and a set of choices (Q, C), while that of BIDAF++'s is a question, a set of choices, and retrieved articles $(Q, C, \overline{A})$.[12]

E.4 Time Modeling

We conduct an ablation study on modeling time information of the retrieved articles. We test the following models: a) pre-pending date string as BERT input $[\texttt{[CLS]}\,Q\,\texttt{[SEP]}\,C\,\texttt{[SEP]}\,\textit{Date}\,\texttt{[SEP]}\,A_i]$, where the date format is "YYYY-MM-DD", b) using binary encodings of dates: we first encode the time into a binary encoding using "Temporenc[13]" and concatenate the encoding with an article encoding before aggregation, c) using char-RNN (Goyal and Durrett, 2019) for encoding date string before aggregation.

E.5 Experiments with Recent LMs.

As mentioned in Sec 5, we did not report more recent pre-trained language models (e.g., RoBERTa (Liu et al., 2019b), ALBERT (Lan et al., 2020)) because they are trained using text data published after the earliest timestamp in our dataset

[11] https://github.com/huggingface/transformers

[12] We did not include existing event forecasting methods since they are designed for modeling structured event data (Fawaz et al., 2019) and thus are not directly applicable to FORECASTQA which requires modeling of unstructured text.

[13] https://temporenc.org

Methods / Metrics	Accuracy		
	yes/no	*multi*	*all*
$\text{BERT}_{\text{BASE}}$, AGG (GRU)	67.6	41.5	55.4
$\text{RoBERTa}_{\text{BASE}}$, AGG (GRU)	69.3	44.8	57.9
$\text{ALBERT}_{\text{BASE}}$, AGG (GRU)	67.4	23.4	46.9
$\text{BERT}_{\text{LARGE}}$, AGG (GRU)	69.2	47.5	59.1
$\text{RoBERTa}_{\text{LARGE}}$, AGG (GRU)	70.1	51.3	61.3
$\text{ALBERT}_{\text{LARGE}}$, AGG (GRU)	68.4	30.2	50.6
Human performance	81.3	77.4	79.4

Table 10: Results on different pre-trained language models, BERT, RoBERTa, ALBERT).

Methods / Metrics	Accuracy (%, ↑)			Brier score (↓)		
	yes/no	*multi*	*all*	*yes/no*	*multi*	*all*
$\text{BERT}_{\text{LARGE}}$, AGG (GRU)	69.2	47.5	59.1	0.483	0.655	0.563
$\text{BERT}_{\text{LARGE}}$, GRU(A), QC	67.8	42.5	56.0	0.583	0.758	0.665

Table 11: Performance of baseline models on FORECASTQA test set.

Methods / Metrics	Accuracy (%)			Brier score		
	yes/no	*multi*	*all*	*yes/no*	*multi*	*all*
$\text{BERT}_{\text{BASE}}$						
– Question	65.6	43.7	55.4	0.506	0.698	0.596
– Article	78.1	84.8	81.2	0.351	0.210	0.285
– Evidence sentence	81.4	90.5	85.6	0.324	0.147	0.241

Table 12: Results on gold articles on the dev set. We give different inputs to the BERT to find out which part is important for the questions.

(2019-01-01). We are worried that these models in theory would have access to information that was published after the associated timestamp of a question.

As a reference, we show the results of RoBERTa and ALBERT in Table 10. Even though these two models may violate our forecasting scenario, they still struggle when compared to human performance, suggesting that our task is still challenging.

E.6 Experiments with different GRU architectures.

We investigate GRU modeling for the input. $\text{BERT}_{\text{LARGE}}$ GRU(A), QC refers to a model that encodes each article with a text encoder, these encodings are fed into GRU, and concatenate the last hidden representation of GRU and Q,C (question and choice) encoding from the text encoder. Table 11 shows comparison between the two architectures. Separating the articles with the question and choice leads to the worse performance.

E.7 Error Analysis

We randomly select 50 errors made by the best baseline method from the test set and identify 4 phenomena:

Retrieving Wrong Articles. 28% of the errors are from the retrieval of irrelevant articles. The baseline approach relies on information retrieval methods such as BM25. Retrieved articles might not be relevant or contain facts that can confuse the model, thus causing incorrect predictions. For example, consider the first question in Fig. 12, the model has retrieved an irrelevant article and conflated Ms. Merkel's health with policy decisions. This results in the model incorrectly choosing Health Care as the appropriate answer.

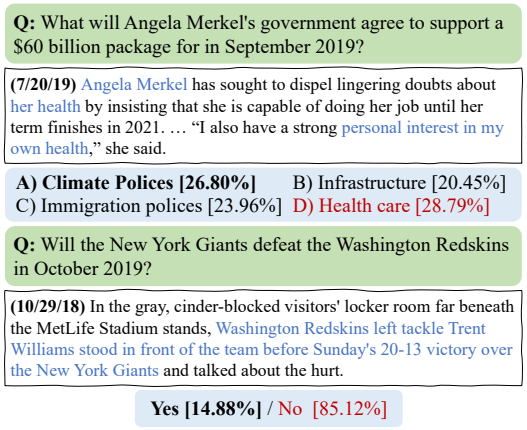

Figure 12: Examples of erroneous model predictions. Bold choices are actual answers and red choices are model predictions.

Incorrect Use of Relevant Evidence. 24% of the errors are (partially) caused by incorrect usage of relevant evidence. Even though useful articles are retrieved, the model incorrectly reasons over the evidence. Take the second question in Fig. 12, where the model incorrectly predicts *No*. The model may depend on a relevant, but outdated fact from 2018 (one year before the event in question) to answer the question, and failed to incorporate more recent information.

Lacking Human Common Sense. 32% of the errors are from the model's lack of common sense or world knowledge. An example question is, *"Who will host 2020 Olympics by July 2019?,"* where the answer is Japan, but the model predicts Hong Kong. To answer this question, a model must know the cities of each country, as without this knowledge the model does not know that "Tokyo is in Japan," and thus the model predicts the wrong answer.

Numerical Questions. 8% of the errors are from numerical questions. Numerical questions ask about numbers such as a person's age. For example, *"What will be Roger Federer's age by August 2019."* The model must know his birth month and age and know how to increment on one's birthday.