REALTIME QA: What's the Answer Right Now?

Anonymous ACL submission

Abstract

We introduce REALTIME QA, a dynamic question answering (QA) platform that announces questions and evaluates systems on a regular basis (weekly in this version). REALTIME QA inquires about the *current* world, and QA systems need to answer questions about novel events or information. It therefore challenges static, conventional assumptions in open-domain QA datasets and pursues instantaneous applications. We build strong baseline models upon large pretrained language models, including GPT-3 and T5. Our benchmark is an ongoing effort, and this paper presents real-time evaluation results over the past two months. Our experimental results show that GPT-3 can often properly update its generation results, based on newly-retrieved documents, highlighting the importance of up-to-date information retrieval. Nonetheless, we find that GPT-3 tends to return *outdated* answers when retrieved documents do not provide sufficient information to find an answer. This suggests an important avenue for future research: can an open-domain QA system identify such unanswerable cases and communicate with the user or even the retrieval module to modify the retrieval results? We hope that REALTIME QA will spur progress in instantaneous applications of question answering and beyond. Website anonymized.

1 Introduction

How many home runs has Shohei Ohtani[1] *hit so far this season?* A user of a question answering (QA) system might ask such time-sensitive questions and seek out answers in *real time* (Fig. 1). Widely-used evaluation benchmarks of QA systems, however, implicitly assume that answers are static regardless of the time of inquiry. Several recent works (Jia et al., 2018; Chen et al., 2021; Zhang and Choi, 2021; Liška et al., 2022) challenged this assumption and proposed QA datasets

[1] https://en.wikipedia.org/wiki/Shohei_Ohtani.

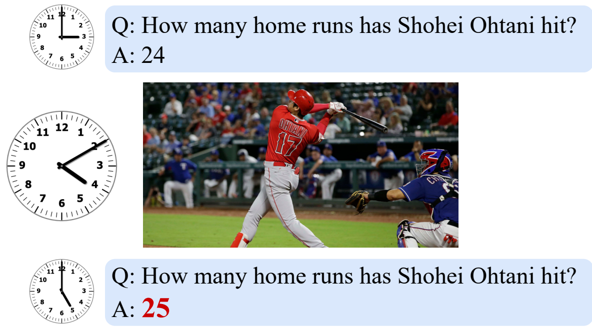

Figure 1: REALTIME QA establishes a framework to benchmark question answering at the present time: answers (e.g., the number of Shohei Ohtani's home runs) change in real time. Source: https://thecomeback.com/mlb/shohei-ohtani-home-runs-tommy-john.html.

that specify the temporal context (e.g., *who was the President of the U.S. in 1940?*). We extend these recent efforts on time-sensitive QA to fulfill real-time, instantaneous information needs from users: we establish a dynamic benchmark based on newly-published news articles—REALTIME QA—and provide a regularly-updated (weekly in the current version) evaluation platform for the research community.

We develop an annotation framework (§2) and a benchmarking timeline for real-time QA system submissions. Every week, REALTIME QA retrieves news articles and ~30 human-written, multiple-choice questions from news websites (CNN, THE WEEK, and USA Today), covering a wide range of topics, including politics, business, sports, and entertainment. We upload these data, as well as our baseline results, to our website, and any model submission can be evaluated until the next set of questions is posted. This dynamic scheme contrasts with the well-established QA annotations (Chen et al., 2017; Chen and Yih, 2020) that are performed only *once* with information available at the time. Such annotations are effective for factoid (Berant et al., 2013; Hermann et al., 2015; Rajpurkar et al., 2016; Joshi et al., 2017) or commonsense questions (Zellers et al., 2018, 2019; Talmor et al., 2019; Sakaguchi et al., 2020), but not the real-time information needs that are our target.

We present two classes of real-time baseline systems that are built on strong, recent models (GPT-3: Brown et al., 2020; T5: Raffel et al., 2020; BART: Lewis et al., 2020a): open-book and closed-book QA models. We present a prompting method to use GPT-3 for open-domain QA. The former class uses an external knowledge source, such as Wikipedia (Min et al., 2019; Guu et al., 2020; Lewis et al., 2020b; Izacard and Grave, 2021) or news articles. The latter class of closed-book models directly outputs an answer to each question. By design, these closed-book baselines have no access to information more recent than the time of pretraining or finetuning, thereby helping us understand the degree to which real-time information is truly necessary. Notably, some of the questions in REALTIME QA do not strictly require recent information; for example, Shohei Ohtani hits a home run today, leading one to ask where he was born. The closed-book baselines thus outperform random selection among multiple choices.

We evaluate six baselines both in multiple-choice and generation settings *in real time* and report the results over the period of June 17 through August 26, 2022. These evaluation data resulted in a total of 329 QA pairs. Further, we provide 2,886 QA pairs that are collected in the same way but preceded our real-time evaluations. These can be used in later work for model development (e.g., finetuning). Our results show that an open-book GPT-3 model augmented with up-to-date text retrieval substantially outperforms closed-book baselines, as well as open-book models with retrieval from a past Wikipedia dump (Lewis et al., 2020b). This result illustrates that large language models can adjust their knowledge, based on the retrieved passages (§3). Nonetheless, we find that they still struggle, especially when the multiple choices include uncertainty (e.g., "none of the above"). Most of the errors originate from retrieval, rather than reading comprehension. The REALTIME QA benchmark, therefore, highlights the importance of fast, up-to-date text retrieval (Seo et al., 2019) to better serve instantaneous information needs. We share all data and code to reproduce our baselines so that follow-up work can build upon our first attempts to tackle this task: `anonymized`.

REALTIME QA can also serve as an important step toward much-needed, broader, real-time applications of NLP. For example, a QA system with timely updates can improve emergency management of natural disasters (Imran et al., 2013, 2015, 2016; Nguyen et al., 2016) or pandemics (e.g., COVID-19; Wang et al., 2020; Lee et al., 2020; Möller et al., 2020; Alzubi et al., 2021).[2] With the advent of online news, prior work developed automated systems that regularly retrieve and summarize news articles from the Internet (Allan et al., 2001; Radev et al., 2001; McKeown et al., 2002, 2003; Evans et al., 2004). Models developed for the REALTIME QA task can be further enhanced with such retrieval/summarization systems. We hope that our REALTIME QA interface and baseline models will serve as a useful platform for research and real-world applications.

2 REALTIME QA Framework

REALTIME QA is a dynamic platform that announces questions every week, based on news articles published within the past week. Here we establish the workflow (§2.1) and the framework for annotations (§2.2) and evaluations (§2.3). We then discuss our built-in baselines (§2.4) that are continually evaluated every week.

2.1 Workflow

Fig. 2 depicts the REALTIME QA workflow for each week. We announce ~30 multiple-choice questions at 3 am GMT every Saturday. We internally run API search (Google custom search, GCS) for these questions and share a set of documents (mostly news articles) with the URLs that are available at that time. Participants run their model on these questions, optionally using the documents from our API search as a knowledge source (indicated as dashed lines in Fig. 2). While we provide our document set to lower barriers to submission, **participants are also allowed to create and use knowledge sources by themselves** (e.g., custom retrieval models or other external APIs such as Twitter API). System submissions are shared on our website with their performance and submission time. The submission window closes when the new set of questions is announced the next week.

Note that fair, *retroactive* comparisons of systems are also possible, as long as they use data available when the submission window was still

[2]A more detailed discussion of NLP applications for emergency response can be found in §4.

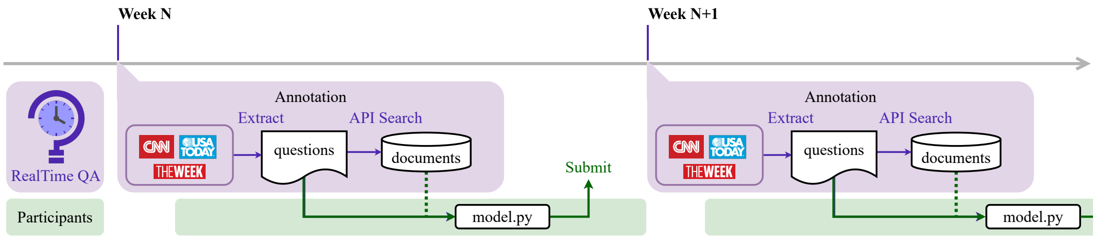

Figure 2: REALTIME QA annotation framework and submission workflow. At 3 am GMT on every Saturday, we extract questions from news websites and post them on our website. We immediately run API search for these questions (Google custom search) and share the results as a document pool. The use of this document pool is optional (indicated by a dashed line); participants are allowed to retrieve evidence documents by themselves. All evaluations are done on our website, and the submission window closes when the next set of questions is announced.

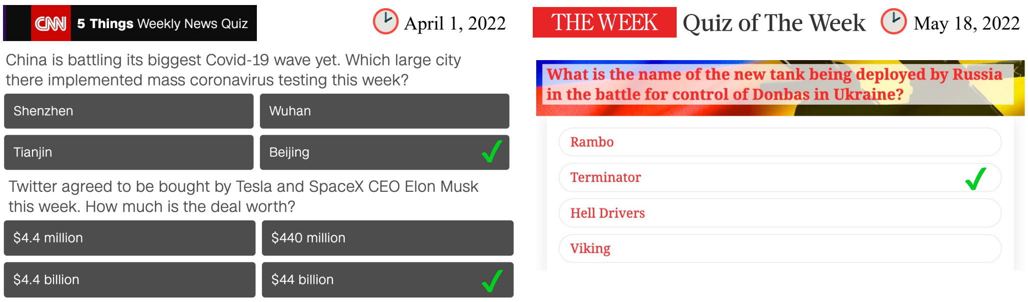

Figure 3: Examples of weekly quizzes from CNN and THE WEEK that are extracted during annotations of REALTIME QA. They span diverse genres, including politics, business, and entertainment.

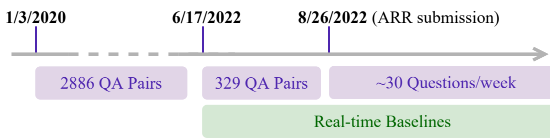

Figure 4: REALTIME QA data statistics as of August 26, 2022. We started our real-time baselines on June 26, 2022 (§2.4). We also provide 2,886 QA pairs that we extracted; they can be used by model developers (e.g., finetuning).

open. For instance, participants might be interested in evaluating their model against a past submission on the Week N questions. In this case, they can do so by ensuring that their system only relies on data up to Week N and simulating how their system *would have performed* at that time. Our platform still focuses on real-time evaluations and encourages every participant to submit real-time results to better reflect real-world applications.

2.2 Annotation

Question Extraction The authors of this paper perform weekly annotations in a human-in-the-loop way. We first find web pages for "weekly quizzes" from three news websites: CNN (US-based), USA Today, and The WEEK (UK-based).[3] Shown in Fig. 3 are examples that span politics and business genres. We then execute our extraction script to collect multiple-choice questions. Each of these three websites posts ~10 questions per week, resulting in ~30 weekly questions in total. Weekly quizzes are also available from the New York Times and ABC Australia, but they are not included in the current version, due to issues with automatic extraction or a paid subscription system.

API Search Using each of these ~30 questions as a retrieval query, we run Google custom search[4] to collect the top-10 documents from the web. The retrieval target is all articles from CNN, USA Today, and THE WEEK. We then parse every document using the `newspaper3k` package[5] and store the text as well as metadata, such as the publication date and author name. In some rare cases, articles from the search get taken down, in which case we disregard them. This indeed illustrates a unique challenge of real-time applications with constantly-changing, dynamic information.

2.3 Evaluation

Multiple Choice Since REALTIME QA is a multiple-choice question dataset, we can simply

[3]Fair use is allowed under Section 107 of the Copyright Act in the U.S.: https://www.copyright.gov/title17/92chap1.html#107.

[4]https://programmablesearchengine.google.com/.

[5]https://github.com/codelucas/newspaper.

measure performance by accuracy. We also explored a NOTA (none of the above) setting: one of the original choices is randomly replaced with "none of the above," thereby helping prevent models from exploiting heuristics (Rajpurkar et al., 2018). As expected, the NOTA setting resulted in performance degradation across the board (§3.1). NOTA choices can be found in other multiple-choice QA or reading comprehension datasets, such as MCTest (Richardson et al., 2013) and RACE (Lai et al., 2017).

Generation We also experiment with a generation setting where no choices are given, to better reflect real-world applications. Under this setting, we evaluate performance with exact matching and token-based F1 scores, following the standard practice in question answering (Rajpurkar et al., 2016; Asai et al., 2021).

Human Performance Many QA datasets estimate human performance as a reference point for automatic QA systems (e.g., Rajpurkar et al., 2016; Yang et al., 2018; Clark et al., 2020). For the sustainability of the dynamic benchmark, we do not provide an estimate of human performance. However, we note that most questions in REALTIME QA, if not all, are straightforward (e.g., single-hop questions) and a human with Internet access can easily answer them. In fact, USA Today has a record of human top scorers every week, and they all get perfect scores.[6] We can thus assume that the human accuracy would be close to 100% in REALTIME QA.

2.4 Real-time Baselines

REALTIME QA executes six baselines in real time that are based on strong pretrained models: four open-book and two closed-book models. These six models are evaluated and made publicly available when weekly questions are announced. Any submission to REALTIME QA is compared against them. Participants can also build their model upon our baselines. See Appendix §A for more detail.

2.4.1 Open-book QA Models

Open-book QA models follow a two-step pipeline: **document retrieval** that finds evidence documents from an external knowledge source (e.g., Wikipedia) and **answer prediction** (or reading comprehension) that outputs an answer conditioned on the question and evidence documents. For either step, we experiment with two variants, resulting in a total of four configurations. Open-book systems have the advantage of being capable of updating the external knowledge source at test time (Lewis et al., 2020b). This property is particularly crucial for questions in REALTIME QA that inquire about information at the present time.

Document Retrieval For the retrieval step, we experiment with two configurations: top-5 Wikipedia documents from dense passage retrieval (**DPR**; Karpukhin et al., 2020) and top-5 news articles from **GCS** (§2.2). In DPR, English Wikipedia articles from the December 2018 dump are segmented into 100-word documents (Wang et al., 2019). DPR encodes the question and every document into 768-dimensional vectors; it then computes the inner product to obtain a matching score and selects documents with top-5 matching scores. We use the BERT-based model (Devlin et al., 2019), finetuned on the Natural Questions dataset (Kwiatkowski et al., 2019) from the Hugging Face Transformers library (Wolf et al., 2020). GCS uses an external API, and we found that it sometimes returned fewer than five documents (~10% of the time); in this case, we add top documents from DPR to create a top-5 document set.

Answer Prediction We explore two methods for answer prediction, conditioned on the question and the corresponding retrieved text: retrieval-augmented generation (**RAG**; Lewis et al., 2020b) and a prompting method with **GPT-3** (text-davinci-002; Brown et al., 2020). In the multiple-choice setting, we compute the log probability of every choice and normalize it by the generation sequence length. We then select the choice with the best score. For the generation setting, we simply perform text decoding.

For the **RAG** baseline, we use the BART-based (Lewis et al., 2020a) RAG-sequence model, again finetuned on Natural Questions from the Transformers library. This model predicts the answer sequence $\mathbf{y}$ autoregressively from left to right while marginalizing over the set of top-5 retrieved documents ($\mathcal{Z}$):

$$P(\mathbf{y}) = \sum_{z \in \mathcal{Z}} P(z) \prod_{t=1}^{|\mathbf{y}|} P(y_t | z, \mathbf{y}_{<t})$$

Here $P(z)$ is given by the matching score from the retrieval step.[7] In the equation, the conditioned-upon question is suppressed for brevity.

[6] E.g., https://www.usatoday.com/storytelling/quiz/news-quiz/2022-07-01/.

[7] Unlike DPR, GCS does not provide matching scores. We treat top-5 documents with equal probabilities.

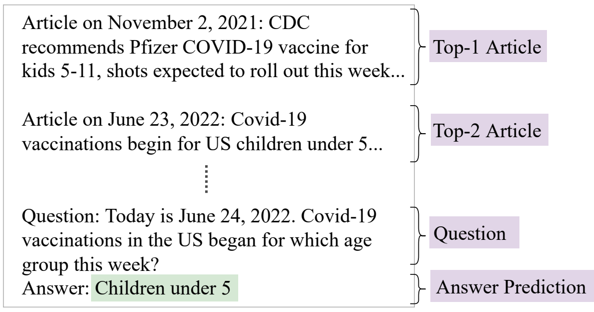

Figure 5: Example prompt for answer generation with the open-book GPT-3 baseline. For the closed-book GPT-3 baseline, the top-5 articles are not given. We perform ablation studies on the date information (§3.2).

We propose a straightforward **GPT-3** prompting method with temporal contexts (Fig. 5).[8] We prepend to every question the title and the first two paragraphs of the top-5 articles from the document retrieval step.[9] The publication date is inserted, using the metadata of each retrieved article (e.g., "Article on November 2, 2021" in Fig. 5). For Wikipedia passages retrieved by DPR, we prepend "December 31, 2018," based on the Wikipedia dump date (Karpukhin et al., 2020). Our ablation studies on date insertion will show that the open-book GPT-3 system benefits from specifying the dates of the question and the retrieved articles to some extent (§3.2).

2.4.2 Closed-book QA Models

Closed-book QA models directly answer questions without access to external knowledge. They have proven competitive with open-book models on some QA datasets (Roberts et al., 2020; Guu et al., 2020). Since these models are trained/finetuned on the data available at that time, they cannot address questions about new events or updated information. Nonetheless, some of the real-time information needs do not necessarily require up-to-date information. Indeed, REALTIME QA contains a small portion of such questions. For instance, *Microsoft retired its Internet Explorer browser this week. What year did it debut?* Such questions are triggered by a new event but inquire about facts in the past that have not changed recently. Closed-book baselines thus quantify the degree to which up-to-date information is necessary to answer questions in REALTIME QA. We use the following two strong methods for closed-book QA.

[8] See Lazaridou et al. (2022) for other prompt templates.

[9] This substantially reduces the inference computations. They contain most of the key information in each article.

Finetuning Method We use the T5 model (T5-11B; Raffel et al., 2020) finetuned on the Natural Questions data, again from the Transformer library. Following the open-book baseline, we select the choice with the maximum average log score in the multiple-choice setting.

Prompting Method Similar to the open-book baselines (§2.4.1), we apply a prompting method to GPT-3 (text-davinci-002). We use the same prompt as Fig. 5, except that no articles are inserted before the question. Again following the open-book baselines, we select the choice with the maximum average log score in the multiple-choice setting.

3 Experiments and Analysis

We started our real-time experiments on June 17 2022, spanning 11 weeks as of August 26 (329 questions in total). We will continue our weekly annotations, but here we report our experimental and analysis results so far and give guidance to future participants.

3.1 Results

Seen in Table 1 are the results from the 11 weeks. In all three settings (original/NOTA multiple choice and generation), GPT-3 with Google custom search (GCS) retrieval achieves the best performance. In particular, GPT-3 with GCS substantially outperforms both closed-book GPT-3 and GPT-3 with DPR (from a December 2018 Wikipedia dump): e.g., 31.2 vs. 11.9/12.2 in generation exact matching. This suggests that GPT-3 is able to answer questions based on the given prompt, rather than relying on past information from pretraining. Nevertheless, we still see a large performance drop of all baselines from the original multiple-choice setting to NOTA ("none of the above"): e.g., 62.6 vs. 70.5 for GPT-3 with GCS retrieval. Future work can further improve GPT-3's ability of reading comprehension, especially regarding answer uncertainty. We also note that the best baseline uses the black-box Google custom search API; we encourage participants to develop a competitive open-source system.

3.2 Analysis and Ablations

Performance vs. Submission Time Fig. 6 plots the performance of the open-book GPT-3 baseline with Google custom search (GCS) over varying submission (i.e., GCS retrieval) time. All results are averaged over 179 questions between June 17

Real-time Baselines			Multi-Choice		Generation	
	Retrieve	Predict	Orig.	NOTA	EM	F1
Open	DPR	RAG	30.4	25.5	2.8	4.9
	DPR	GPT-3	46.5	36.2	12.2	19.1
	GCS	RAG	52.9	42.9	24.5	30.7
	GCS	GPT-3	**70.5**	**62.6**	**31.2**	**43.8**
Closed	—	T5	37.1	35.6	8.3	12.9
	—	GPT-3	43.2	42.1	11.9	20.0

Table 1: Results from the past two months (from June 17 through August 26, 2022) over a total of 329 questions. GCS: Google custom search; DPR: dense passage retrieval (Karpukhin et al., 2020); RAG: retrieval-augmented generation (Lewis et al., 2020b).

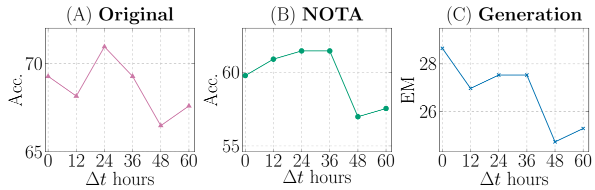

Figure 6: Performance vs. submission time (hours after the announcement of questions, 3 am GMT on Saturday) over the three evaluation settings (A: original multiple choice; B: none of the above; C: generation). All results are from open-book GPT-3 with Google custom search (GCS) and averaged over 179 questions from June 17, 2022 through July 22, 2022. $\Delta t = 0$ for all of our six real-time baselines by default.

and July 22, 2022. We see a consistent pattern: the performance remains high (or improves) up to around 24 hours after the announcement but substantially degrades later. While the performance can improve when GCS starts to retrieve more recent articles, it eventually suffers from temporal gaps. Our website provides the submission time of every system as well as its performance.

Date Insertion			Multi-Choice		Generation	
	Articles	Question	Orig.	NOTA	EM	F1
Open	✓	✓	**69.3**	59.8	**28.7**	**39.4**
	✓	✗	66.5	**62.6**	24.7	36.3
	✗	✓	67.0	57.5	28.1	38.2
	✗	✗	65.9	61.5	**28.7**	38.3
Closed	—	✓	39.7	31.3	7.3	15.2
	—	✗	45.8	38.5	9.0	15.9

Table 2: Ablation studies on date insertion in the prompt for the open-book (Google custom search; GCS) and close-book GPT-3 baselines. All results are averaged over the 179 questions from the first six weeks: June 17 through July 22, 2022.

Date Insertion for Prompting Our prompt for the GPT-3 baselines prepends date information both to the articles and question (Fig. 5). Table 2 shows results from ablation studies on date insertion for the open-book (GPT-3 with Google custom search) and closed-book GPT-3 models. Temporal specification almost always helps the open-book GPT-3 model. Interestingly, it hurts the performance of the closed-book model, perhaps because the specified date is generally unseen during pretraining and the prompt becomes "out-of-domain."

Error Breakdown We conducted a manual error analysis of the results so far. In particular, we categorized answers from the best generation model (open-book GPT-3 with GCS) into three categories: correct, retrieval error, and reading comprehension error. For the 179 questions from the first six weeks, the breakdown was the following: correct (52%), retrieval error (34%), and reading comprehension error (13%). This suggests that the key to instantaneous applications of question answering is **accurate, up-to-date information retrieval**.

Examples Table 3 shows some examples that compare the closed-book and open-book GPT-3 models. The first three examples illustrate that GPT-3 can correctly update its answer based on the retrieved documents across diverse genres: natural disasters, the COVID-19 pandemic, and entertainment. The last three cases, on the other hand, demonstrate a critical limitation of current large language models in temporal understanding: **the retrieved documents do not suffice to answer the questions due to a temporal gap, and GPT-3 still generates an outdated answer**. Ideally, GPT-3 should inform the user or even the retrieval module that it does not have enough evidence to answer the question. This way, the retrieval module can expand its search, or the user can consult other resources.

Note that it is possible to limit the retrieval target to recent articles,[10] but there are potential failure modes. Firstly, some questions in REALTIME QA inquire about the past, and models can benefit from older articles when answering such questions. Further, the appropriate date range for retrieval varies from question to question in real-world applications; some questions inquire about this year, while others about this week. We thus do not implement such filtering for the current real-time baselines.

4 Related Work

REALTIME QA has time sensitivity, which several prior works addressed on various NLP tasks.

[10] Indeed, Google custom search has a *paid* version with a date range feature that filters retrieval results by date.

Question	**Retrieved Documents (Top-5)**
Historic rainfall led to flooding, mudslides and visitor evacuations at which national park? **Date**: June 17, 2022 **Answer**: Yellowstone National Park **Closed GPT-3**: Yosemite National Park **Open GPT-3**: Yellowstone National Park	**June 14, 2022** Yellowstone National Park flooding 'still raging'... **June 13, 2022** Yellowstone National Park closes entrances, evacuates visitors amid 'unprecedented' rainfall... **June 15, 2022** Dozens evacuated as unprecedented flooding forces Yellowstone National Park to close... **June 15, 2022** Yellowstone still closed as flooding recedes and thousands evacuate... **June 14, 2022** Home swept away as Yellowstone National Park is hit by major floods and mudslides...
Covid-19 vaccinations in the US began for which age group this week? **Date**: June 24, 2022 **Answer**: Children under 5 **Closed GPT-3**: 18 and up **Open GPT-3**: Children under 5	**November 2, 2021** CDC recommends Pfizer COVID-19 vaccine for kids 5-11, shots expected to roll out this week... **June 23, 2022** Covid-19 vaccinations begin for US children under 5... **July 22, 2021** Biden says kids under 12 could be eligible for COVID vaccines in weeks... **November 10, 2021** COVID-19 cases on the rise again in Iowa... **November 1, 2021** Everything to know about COVID-19 vaccine and children...
Which wildly popular show was recently green lit for a new season? **Date**: June 17, 2022 **Answer**: Squid Game **Closed GPT-3**: The show "Game of Thrones" was recently green lit for a new **Open GPT-3**: Squid Game	**January 13, 2022** Netflix green lights 'Squid Game' season 2... **June 17, 2022** 5 things to know for June 17... **June 4, 2019** 'Looking for Alaska' details revealed for Hulu limited series... **February 4, 2022** The Busch Light Clash goes green this weekend... **September 26, 2018** Dip into 4 new mysteries for fall, including Kate Atkinson's spy novel 'Transcription'...
The IRS announced it will do what this week? **Date**: June 24, 2022 **Answer**: Finish processing the backlog of 2021 tax returns **Closed GPT-3**: The IRS announced it will begin processing tax returns this week. **Open GPT-3**: The IRS announced it will begin processing 2021 tax returns as soon as Jan. 24	**January 10, 2022** IRS 2022 tax season set to begin 2 weeks early on Jan. 24... **March 12, 2021** When will I get my third stimulus check?... **March 22, 2021** IRS says more stimulus checks on the way... **February 11, 2022** Don't panic if you got a scary IRS notice... **January 10, 2022** IRS will begin processing 2021 tax returns as soon as Jan. 24
Which country is now "bankrupt," according to a statement this week from its administration? **Date**: July 8, 2022 **Answer**: Sri Lanka **Closed GPT-3**: Greece **Open GPT-3**: Venezuela	**March 2, 2022** Gun manufacturers are not entirely exempt from being sued... the now-bankrupt gun manufacturer... **March 12, 2021** Mitch McConnell seeks to end Democrat's 'crazy policy' of beefed-up unemployment benefits... let states go bankrupt... **March 20, 2022** Half of US hotels could close amid coronavirus crisis... hotels around the country go bankrupt... **September 26, 2013** Colo. farmers arrested... the now-bankrupt Jensen Farms... **January 10, 2022** Trump administration restrictions on asylum are cruel... Immigration policy is morally bankrupt...
Which head of state announced his resignation this week? **Date**: July 8, 2022 **Answer**: UK Prime Minister Boris Johnson **Closed GPT-3**: Japanese Prime Minister Shinzo Abe announced his resignation this week. **Open GPT-3**: Andrew Cuomo	**August 11, 2021** NY Gov. Andrew Cuomo will resign in two weeks... **September 21, 2021** Maricopa County Supervisor Steve Chucri to resign... **January 25, 2016** Ball State president Ferguson resigns... **March 23, 2021** Oregon State University President F. King Alexander resigns... **August 10, 2021** NY Gov. Andrew Cuomo to resign amid scandal...

Table 3: Examples that compare closed-book and open-book GPT-3 answers with top-5 articles from Google custom search (GCS) retrieval. As in the first three examples, GPT-3 can adjust its answer based on newly-retrieved documents. When the retrieved documents are *outdated* or unrelated, however, GPT-3 ignores the temporal gap and yields an outdated answer.

Here we discuss its relation to long-standing summarization and text retrieval tasks, as well as recent work on temporal misalignment between training and evaluation. We then discuss its connections to broad frameworks of dynamic evaluations and open-domain QA.

Summarization/Retrieval over Time Temporal (or timeline) summarization is a task that retrieves documents from the web and provides their summary *over time* (Catizone et al., 2006; Aslam et al., 2013, 2014, 2015; Martschat and Markert, 2017, 2018). Update summarization (Witte et al., 2007; Dang and Owczarzak, 2008) and new event detection/track (Allan et al., 1998; Li et al., 2005) are tasks that monitor and track newly-added information. Prior work created datasets and systems for these tasks (Tran et al., 2013, 2015; Wang et al., 2015; Chen et al., 2019; Gholipour Ghalandari and Ifrim, 2020). Their evaluations are usually executed *statically*, with information available at the time of data collection.

In contrast, the TREC real-time summarization track evaluates systems in real time during a 1–2 week evaluation period (Lin et al., 2016, 2017; Sequiera et al., 2018). Several other works and initiatives focused particularly on financial news summarization (Filippova et al., 2009; Passali et al., 2021) or emergency management technology (Temnikova et al., 2014; Ghosh et al., 2017; McCreadie et al., 2019), including the COVID-19 pandemic (Buntain et al., 2020; Pasquali et al., 2021). This work regularly evaluates question answering systems over diverse topics in real time, but we share the goal of dealing with novel and evolving information over time; retrieval or summarization methods from these tasks (e.g., Yan et al., 2011a,b, 2012; Shou et al., 2013) can be combined with models in REALTIME QA to serve various time-sensitive information needs from users. REALTIME QA can also be used to evaluate time-sensitive retrieval systems by the downstream QA performance.

Temporal Misalignment and Degradation While not particularly motivated by instantaneous information needs like REALTIME QA, prior work also explored temporal aspects of a variety of NLP tasks. A flurry of recent work analyzed performance degradation from temporal misalignment between (pre)training and evaluation/deployment on many NLP tasks (Lazaridou et al., 2021; Röttger and Pierrehumbert, 2021; Luu et al., 2022; Onoe et al., 2022) and proposed mitigation methods for temporal adaptation (Huang and Paul, 2018, 2019; Dhingra et al., 2022; Jang et al., 2022a,b; Lee et al., 2022). An open-book QA model conditions answer generation upon newly-retrieved documents (Lewis et al., 2020b), but the extent to which answer generation can be updated based on the retrieved documents is limited (Longpre et al., 2021b). Temporal degradation is, therefore, one of the challenges that models in REALTIME QA need to address.

Dynamic Benchmarks Unlike the majority of datasets in natural language processing, REALTIME QA evaluates systems *dynamically* and its evaluations change over time. Several other prior works update challenge test sets (Kiela et al., 2021; Potts et al., 2021; Ma et al., 2021), evaluation tasks (Thrush et al., 2022), or text generation metrics (Gehrmann et al., 2021, 2022; Mishra and Arunkumar, 2021; Kasai et al., 2022). REALTIME QA hosts a similar online platform and adopts a dynamic scheme specifically to pursue instantaneous applications.

Open-Domain QA Much prior work proposed datasets for open-domain (or open-retrieval) QA for English and beyond (Clark et al., 2020; Asai et al., 2021, 2022; Longpre et al., 2021a; Zhang et al., 2021). Several recent works challenged the conventional problem setups (Chen and Yih, 2020) where correct answers can be found from a fixed, external knowledge source, such as Wikipedia. Similar to REALTIME QA, Zhang and Choi (2021); Liška et al. (2022) focused on temporal or geographical contexts that can change the answer to the same question. Min et al. (2020) pointed out inherent ambiguity in questions from human users and proposed a benchmark that provides multiple answers to every question. Consistent with these prior efforts, REALTIME QA aims toward broader applications of question answering beyond the conventional framework.

5 Conclusion and Future Work

We introduce REALTIME QA, a dynamic, open-domain QA benchmark that asks questions at the present time. Our platform announces ~30 questions every week and continually evaluates six real-time baselines. Our experiments from the first 11 weeks suggest that accurate, up-to-date information retrieval is particularly important to serve speedy information needs. We hope that REALTIME QA encourages research efforts toward fast, accurate applications of natural language processing.

Limitations and Ethical Considerations

This work aims to develop a QA benchmark for addressing instantaneous information needs, including emergency management. The current version of REALTIME QA has two major limitations due to our annotation framework (§2.2): 1) question/answer pairs are all written in English, and the covered topics tend to be English-centric (US and UK); 2) questions are announced on a weekly basis, rather than a truly instantaneous basis. Nevertheless, our benchmark departs from many static datasets from prior work and provides an important step towards the research goal. We hope to develop future versions of REALTIME QA that mitigate these limitations.

References

James Allan, Rahul Gupta, and Vikas Khandelwal. 2001. Temporal summaries of new topics. In *Proc. of SIGIR*.

James Allan, Ron Papka, and Victor Lavrenko. 1998. On-line new event detection and tracking. In *Proc. of SIGIR*.

Jafar Ahmad Abed Alzubi, Rachna Jain, Anubhav Singh, Pritee Parwekar, and Meenu Gupta. 2021. COBERT: COVID-19 question answering system using BERT. *Arabian Journal for Science and Engineering*.

Akari Asai, Jungo Kasai, Jonathan H Clark, Kenton Lee, Eunsol Choi, and Hannaneh Hajishirzi. 2021. XOR QA: Cross-lingual open-retrieval question answering. In *Proc. of NAACL*.

Akari Asai, Shayne Longpre, Jungo Kasai, Chia-Hsuan Lee, Rui Zhang, Junjie Hu, Ikuya Yamada, Jonathan H. Clark, and Eunsol Choi. 2022. MIA 2022 shared task: Evaluating cross-lingual open-retrieval question answering for 16 diverse languages. In *Proc. of MIA*.

Javed A. Aslam, Fernando Diaz, Matthew Ekstrand-Abueg, Richard McCreadie, Virgil Pavlu, and Tetsuya Sakai. 2015. TREC 2015 temporal summarization track overview. In *Proc. of TREC*.

Javed A. Aslam, Matthew Ekstrand-Abueg, Virgil Pavlu, Fernando Diaz, Richard McCreadie, and Tetsuya Sakai. 2014. TREC 2014 temporal summarization track overview. In *Proc. of TREC*.

Javed A. Aslam, Matthew Ekstrand-Abueg, Virgil Pavlu, Fernando Diaz, and Tetsuya Sakai. 2013. TREC 2013 temporal summarization. In *Proc. of TREC*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proc. of EMNLP*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proc. of NeurIPS*.

Cody Buntain, Richard McCreadie, and Ian Soboroff. 2020. Incident streams 2020: TRECIS in the time of COVID-19. In *Proc. of TREC*.

Roberta Catizone, Angelo Dalli, and Yorick Wilks. 2006. Evaluating automatically generated timelines from the web. In *Proc. of LREC*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proc. of ACL*.

Danqi Chen and Wen-tau Yih. 2020. Open-domain question answering. In *Proc. of ACL: Tutorial Abstracts*.

Wenhu Chen, Xinyi Wang, and William Yang Wang. 2021. A dataset for answering time-sensitive questions. In *Proc. of NeurIPS Datasets and Benchmarks*.

Xiuying Chen, Zhangming Chan, Shen Gao, Meng-Hsuan Yu, Dongyan Zhao, and Rui Yan. 2019. Learning towards abstractive timeline summarization. In *Proc. of IJCAI*.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *TACL*.

Hoa Trang Dang and Karolina Owczarzak. 2008. Overview of the TAC 2008 update summarization task. In *Proc. of TAC*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *TACL*.

David Kirk Evans, Judith L. Klavans, and Kathleen R. McKeown. 2004. Columbia Newsblaster: Multilingual news summarization on the web. In *Proc. of NAACL: Demonstrations*.

Katja Filippova, Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2009. Company-oriented extractive summarization of financial news. In *Proc. of EACL*.

Sebastian Gehrmann, Tosin P. Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Aremu Anuoluwapo, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh D. Dhole, Wanyu Du, Esin Durmus, Ondrej Dusek, Chris Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Rubungo Andre Niyongabo, Salomey Osei, Ankur P. Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In *Proc. of GEM*.

Sebastian Gehrmann, Abhik Bhattacharjee, Abinaya Mahendiran, Alex Wang, Alexandros Papangelis, Aman Madaan, Angelina McMillan-Major, Anna Shvets, Ashish Upadhyay, Bingsheng Yao, Bryan Wilie, Chandra Bhagavatula, Chaobin You, Craig Thomson, Cristina Garbacea, Dakuo Wang, Daniel Deutsch, Deyi Xiong, Di Jin, Dimitra Gkatzia, Dragomir Radev, Elizabeth Clark, Esin Durmus, Faisal Ladhak, Filip Ginter, Genta Indra Winata, Hendrik Strobelt, Hiroaki Hayashi, Jekaterina Novikova, Jenna Kanerva, Jenny Chim, Jiawei Zhou, Jordan Clive, Joshua Maynez, João Sedoc, Juraj Juraska, Kaustubh D. Dhole, Khyathi Raghavi Chandu, Laura Perez-Beltrachini, Leonardo F. R. Ribeiro, Lewis Tunstall, Li Zhang, Mahima Pushkarna, Mathias Creutz, Michael White, Mihir Sanjay Kale, Moussa Kamal Eddine, Nico Daheim, Nishant Subramani, Ondrej Dusek, Paul Pu Liang, Pawan Sasanka Ammanamanchi, Qi Zhu, Ratish Puduppully, Reno Kriz, Rifat Shahriyar, Ronald Cardenas, Saad Mahamood, Salomey Osei, Samuel Cahyawijaya, Sanja Stajner, Sébastien Montella, Shailza Jolly, Simon Mille, Tahmid Hasan, Tianhao Shen, Tosin P. AMahidewumi, Vikas Raunak, Vipul Raheja, Vitaly Nikolaev, Vivian Tsai, Yacine Jernite, Ying Xu, Yisi Sang, Yixin Liu, and Yufang Hou. 2022. GEMv2: Multilingual NLG benchmarking in a single line of code.

Demian Gholipour Ghalandari and Georgiana Ifrim. 2020. Examining the state-of-the-art in news timeline summarization. In *Proc. of ACL*.

Saptarshi Ghosh, Kripabandhu Ghosh, Debasis Ganguly, Tanmoy Chakraborty, Gareth J.F. Jones, and Marie-Francine Moens. 2017. ECIR 2017 workshop on exploitation of social media for emergency relief and preparedness (SMERP 2017). *SIGIR Forum*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. REALM: Retrieval-augmented language model pre-training. In *Proc. of ICML*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proc. of NeurIPS*.

Xiaolei Huang and Michael J. Paul. 2018. Examining temporality in document classification. In *Proc. of ACL*.

Xiaolei Huang and Michael J. Paul. 2019. Neural temporality adaptation for document classification: Diachronic word embeddings and domain adaptation models. In *Proc. of ACL*.

Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. *ACM Computing Surveys*.

Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, and Patrick Meier. 2013. Practical extraction of disaster-relevant information from social media. In *Proc. of WWW*.

Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a lifeline: Human-annotated Twitter corpora for NLP of crisis-related messages. In *Proc. of LREC*.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proc. of EACL*.

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. 2022a. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models.

Joel Jang, Seonghyeon Ye, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, Stanley Jungkyu Choi, and Minjoon Seo. 2022b. Towards continual knowledge learning of language models. In *Proc. of ICLR*.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018. TempQuestions: A benchmark for temporal question answering. In *Companion of the WWW*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proc. of ACL*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proc. of EMNLP*.

Jungo Kasai, Keisuke Sakaguchi, Ronan Le Bras, Lavinia Dunagan, Jacob Morrison, Alexander R. Fabbri, Yejin Choi, and Noah A. Smith. 2022. Bidimensional leaderboards: Generate and evaluate language hand in hand. In *Proc. of NAACL*.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In *Proc. of NAACL*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *TACL*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proc. of EMNLP*.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomás Kociský, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. Mind the gap: Assessing temporal generalization in neural language models. In *Proc. of NeurIPS*.

Jinhyuk Lee, Sean S. Yi, Minbyul Jeong, Mujeen Sung, WonJin Yoon, Yonghwa Choi, Miyoung Ko, and Jaewoo Kang. 2020. Answering questions on COVID-19 in real-time. In *Proc. of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*.

Kyungjae Lee, Wookje Han, Seung-won Hwang, Hwaran Lee, Joonsuk Park, and Sang-Woo Lee. 2022. Plug-and-play adaptation for continuously-updated QA. In *Findings of the ACL: ACL 2022*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proc. of ACL*.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020b. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proc. of NeurIPS*.

Zhiwei Li, Bin Wang, Mingjing Li, and Wei-Ying Ma. 2005. A probabilistic model for retrospective news event detection. In *Proc. of SIGIR*.

Jimmy Lin, Salman Mohammed, Royal Sequiera, Luchen Tan, Nimesh Ghelani, Mustafa Abualsaud, Richard McCreadie, Dmitrijs Milajevs, and Ellen M. Voorhees. 2017. Overview of the TREC 2017 real-time summarization track. In *Proc. of TREC*.

Jimmy Lin, Adam Roegiest, Luchen Tan, Richard McCreadie, Ellen M. Voorhees, and Fernando Diaz. 2016. Overview of the TREC 2016 real-time summarization track. In *Proc. of TREC*.

Adam Liška, Tomáš Kočiský, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, Cyprien de Masson d'Autume, Tim Scholtes, Manzil Zaheer, Susannah Young, Ellen Gilsenan-McMahon Sophia Austin, Phil Blunsom, and Angeliki Lazaridou. 2022. StreamingQA: A benchmark for adaptation to new knowledge over time in question answering models.

Shayne Longpre, Yi Lu, and Joachim Daiber. 2021a. MKQA: A linguistically diverse benchmark for multilingual open domain question answering. *TACL*.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021b. Entity-based knowledge conflicts in question answering. In *Proc. of EMNLP*.

Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A. Smith. 2022. Time waits for no one! analysis and challenges of temporal misalignment. In *Proc. of NAACL*.

Zhiyi Ma, Kawin Ethayarajh, Tristan Thrush, Somya Jain, Ledell Wu, Robin Jia, Christopher Potts, Adina Williams, and Douwe Kiela. 2021. Dynaboard: An evaluation-as-a-service platform for holistic next-generation benchmarking. In *Proc. of NeurIPS*.

Sebastian Martschat and Katja Markert. 2017. Improving ROUGE for timeline summarization. In *Proc. of EACL*.

Sebastian Martschat and Katja Markert. 2018. A temporally sensitive submodularity framework for timeline summarization. In *Proc. of CoNLL*.

Richard McCreadie, Cody Buntain, and Ian Soboroff. 2019. TREC incident streams: Finding actionable information on social media. In *Proc. of ISCRAM*.

Kathleen McKeown, Regina Barzilay, John Chen, David Elson, David Evans, Judith Klavans, Ani Nenkova, Barry Schiffman, and Sergey Sigelman. 2003. Columbia's newsblaster: New features and future directions. In *Proc. of NAACL: Demonstrations*.

Kathleen R. McKeown, Regina Barzilay, David Evans, Vasileios Hatzivassiloglou, Judith L. Klavans, Ani Nenkova, Carl Sable, Barry Schiffman, and Sergey Sigelman. 2002. Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proc. of HLT*.

Sewon Min, Danqi Chen, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. Knowledge guided text retrieval and reading for open domain question answering.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proc. of EMNLP*.

Swaroop Mishra and Anjana Arunkumar. 2021. How robust are model rankings : A leaderboard customization approach for equitable evaluation. In *Proc. of AAAI*.

Timo Möller, Anthony Reina, Raghavan Jayakumar, and Malte Pietsch. 2020. COVID-QA: A question answering dataset for COVID-19. In *Proc. of the 1st Workshop on NLP for COVID-19 at ACL 2020*.

Dat Tien Nguyen, Kamla Al-Mannai, Shafiq R. Joty, Hassan Sajjad, Muhammad Imran, and Prasenjit Mitra. 2016. Rapid classification of crisis-related data on social networks using convolutional neural networks. In *Proc. of ICWSM*.

Yasumasa Onoe, Michael J. Q. Zhang, Eunsol Choi, and Greg Durrett. 2022. Entity cloze by date: What LMs know about unseen entities. In *Findings of the ACL: NAACL 2022*.

Arian Pasquali, Ricardo Campos, Alexandre Ribeiro, Brenda Santana, Alípio Jorge, and Adam Jatowt. 2021. TLS-Covid19: A new annotated corpus for timeline summarization. In *Advances in Information Retrieval*.

Tatiana Passali, Alexios Gidiotis, Efstathios Chatzikyriakidis, and Grigorios Tsoumakas. 2021. Towards human-centered summarization: A case study on financial news. In *Proc. of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*.

Christopher Potts, Zhengxuan Wu, Atticus Geiger, and Douwe Kiela. 2021. DynaSent: A dynamic benchmark for sentiment analysis. In *Proc. of ACL*.

Dragomir R. Radev, Sasha Blair-Goldensohn, Zhu Zhang, and Revathi Sundara Raghavan. 2001. NewsInEssence: A system for domain-independent, real-time news clustering and multi-document summarization. In *Proc. of HLT*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proc. of ACL*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. of EMNLP*.

Matthew Richardson, Christopher J.C. Burges, and Erin Renshaw. 2013. MCTest: A challenge dataset for the open-domain machine comprehension of text. In *Proc. of EMNLP*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proc. of EMNLP*.

Paul Röttger and Janet Pierrehumbert. 2021. Temporal adaptation of BERT and performance on downstream document classification: Insights from social media. In *Findings of the ACL: EMNLP 2021*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WinoGrande: An adversarial Winograd schema challenge at scale. In *Proc. of AAAI*.

Minjoon Seo, Jinhyuk Lee, Tom Kwiatkowski, Ankur Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2019. Real-time open-domain question answering with dense-sparse phrase index. In *Proc. of ACL*.

Royal Sequiera, Luchen Tan, and Jimmy Lin. 2018. Overview of the TREC 2018 real-time summarization track. In *Proc. of TREC*.

Lidan Shou, Zhenhua Wang, Ke Chen, and Gang Chen. 2013. Sumblr: Continuous summarization of evolving tweet streams. In *Proc. of SIGIR*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proc. of NAACL*.

Irina Temnikova, Andrea Varga, and Dogan Biyikli. 2014. Building a crisis management term resource for social media: The case of floods and protests. In *Proc. of LREC*.

Tristan Thrush, Kushal Tirumala, Anmol Gupta, Max Bartolo, Pedro Rodriguez, Tariq Kane, William Gaviria Rojas, Peter Mattson, Adina Williams, and Douwe Kiela. 2022. Dynatask: A framework for creating dynamic AI benchmark tasks. In *Proc. of ACL: System Demonstrations*.

Giang Tran, Mohammad Alrifai, and Eelco Herder. 2015. Timeline summarization from relevant headlines. In *Advances in Information Retrieval*.

Giang Binh Tran, Mohammad Alrifai, and Dat Quoc Nguyen. 2013. Predicting relevant news events for timeline summaries. In *Proc. of WWW Companion*.

Lu Wang, Claire Cardie, and Galen Marchetti. 2015. Socially-informed timeline generation for complex events. In *Proc. of NAACL*.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Michael Kinney, Yunyao Li, Ziyang Liu, William Merrill, Paul Mooney, Dewey A. Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex D. Wade, Kuansan Wang, Nancy Xin Ru Wang, Christopher Wilhelm, Boya Xie, Douglas M. Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. 2020. CORD-19: The COVID-19 open research dataset. In *Proc. of the 1st Workshop on NLP for COVID-19 at ACL 2020*.

Zhiguo Wang, Patrick Ng, Xiaofei Ma, Ramesh Nallapati, and Bing Xiang. 2019. Multi-passage BERT: A globally normalized BERT model for open-domain question answering. In *Proc. of EMNLP*.

René Witte, Ralf Krestel, and Sabine Bergler. 2007. Generating update summaries for DUC 2007. In *Proc. of DUC*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proc. of EMNLP: System Demonstrations*.

Rui Yan, Liang Kong, Congrui Huang, Xiaojun Wan, Xiaoming Li, and Yan Zhang. 2011a. Timeline generation through evolutionary trans-temporal summarization. In *Proc. of EMNLP*.

Rui Yan, Xiaojun Wan, Mirella Lapata, Wayne Xin Zhao, Pu-Jen Cheng, and Xiaoming Li. 2012. Visualizing timelines: Evolutionary summarization via iterative reinforcement between text and image streams. In *Proc. of CIKM*.

Rui Yan, Xiaojun Wan, Jahna Otterbacher, Liang Kong, Xiaoming Li, and Yan Zhang. 2011b. Evolutionary timeline summarization: A balanced optimization framework via iterative substitution. In *Proc. of SIGIR*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proc. of EMNLP*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proc. of EMNLP*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proc. of ACL*.

Michael J.Q. Zhang and Eunsol Choi. 2021. SituatedQA: Incorporating extra-linguistic contexts into QA. In *Proc. of EMNLP*.

Xinyu Zhang, Xueguang Ma, Peng Shi, and Jimmy Lin. 2021. Mr. TyDi: A multi-lingual benchmark for dense retrieval. In *Proc. of MRL*.

Appendices

A Baseline Configurations

We provide the configurations for our real-time baselines (§2.4). We use the open-source, Transformers library and ensure easy replication of our results. Table 4 lists the configurations for dense passage retrieval (Karpukhin et al., 2020) and retrieval-augmented generation (Lewis et al., 2020b). We generally follow the default settings from the Transformers library. Seen in Table 5 is the configuration for the closed-book T5 baseline. We again generally follow the default setting.

Option	Value
n_docs	5
max_combined_length	300
retrieval_vector_size	768
retrieval_batch_size	8
is_encoder_decoder	True
prefix	None
bos_token_id	None
pad_token_id	None
eos_token_id	None
decoder_start_token_id	None
title_sep	’/’
doc_sep	’//’
dataset	’wiki_dpr’
dataset_split	’train’
index_name	’compressed’
index_path	None
passages_path	None
use_dummy_dataset	False
reduce_loss	False
label_smoothing	0.0
do_deduplication	True
exclude_bos_score	False
do_marginalize	False
output_retrieved	False
use_cache	True
forced_eos_token_id	None

Table 4: Configurations for dense passage retrieval (Karpukhin et al., 2020) and retrieval-augmented generation (Lewis et al., 2020b) from the Transformers library (Wolf et al., 2020).

Option	Value
_name_or_path	/home/patrick/t5/t5-11b-ssm-nq
architectures	["T5ForConditionalGeneration"]
d_f	65536
d_kv	128
d_model	1024
decoder_start_token_id	0
dropout_rate	0.1
eos_token_id	1
feed_forward_proj"	relu
initializer_factor	1.0
is_encoder_decoder	True
layer_norm_epsilon	1e-06
model_type	t5
num_decoder_layers	24
num_heads	128
num_layers	24
output_past	True
pad_token_id	0
relative_attention_num_buckets	32
tokenizer_class	T5Tokenizer
vocab_size	32128

Table 5: Configuration for the closed-book T5 baseline (Raffel et al., 2020) from the Transformer library.