Dynamic Benchmarking of Masked Language Models on Temporal Concept Drift with Multiple Views

Katerina Margatina[◇*] **Shuai Wang**[†] **Yogarshi Vyas**[†]
Neha Anna John[†] **Yassine Benajiba**[†] **Miguel Ballesteros**[†]
[◇]University of Sheffield [†]AWS AI Labs
k.margatina@sheffield.ac.uk,
{wshui,yogarshi,nehajohn,benajiy,ballemig}@amazon.com

Abstract

Temporal concept drift refers to the problem of data changing over time. In NLP, that would entail that *language* (e.g. new expressions, meaning shifts) and *factual knowledge* (e.g. new concepts, updated facts) evolve over time. Focusing on the latter, we benchmark 11 pretrained masked language models (MLMs) on a series of tests designed to evaluate the effect of temporal concept drift, as it is crucial that widely used language models remain up-to-date with the ever-evolving factual updates of the real world. Specifically, we provide a holistic framework that (1) *dynamically* creates temporal test sets of any time granularity (e.g. month, quarter, year) of factual data from Wikidata, (2) constructs fine-grained splits of tests (e.g. updated, new, unchanged facts) to ensure *comprehensive* analysis, and (3) evaluates MLMs in three distinct ways (single-token probing, multi-token generation, MLM scoring). In contrast to prior work, our framework aims to unveil how *robust* an MLM is over time and thus to provide a signal in case it has become *outdated*, by leveraging *multiple views of evaluation*.

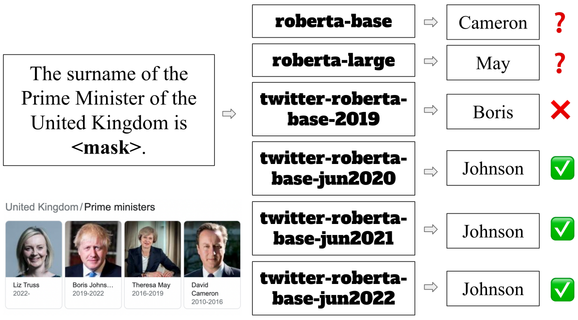

The surname of the Prime Minister of the United Kingdom is **<mask>**.
roberta-base
Cameron
?
roberta-large
May
?
twitter-roberta-base-2019
Boris
✗
twitter-roberta-base-jun2020
Johnson
✓
twitter-roberta-base-jun2021
Johnson
✓
twitter-roberta-base-jun2022
Johnson
✓
United Kingdom / Prime ministers
Liz Truss
2022-
Boris Johns...
2019-2022
Theresa May
2016-2019
David Cameron
2010-2016

Figure 1: Querying pretrained MLMs on their knowledge about the Prime Minister of the United Kingdom.

1 Introduction

In the real world, what people talk about and how they tend to speak and write changes constantly over time. In Natural Language Processing (NLP), this entails a challenging shift of the textual data distribution that is commonly referred to as *temporal concept drift*. Prior work has identified that pretrained language models (PLMs) tend to become outdated soon after new topics and concepts are emerging (Lazaridou et al., 2021; Dhingra et al., 2022; Agarwal and Nenkova, 2022; Luu et al., 2022), limiting their capability to be robust to newly generated data.

We consider the desiderata of language models' robustness to temporal drift to be twofold. First, LMs should be well adapted to the dynamic use of language, from the *linguistic* perspective. Language changes over time, pronunciations evolve, new words and expressions are borrowed or invented, the meaning of old words drifts, and morphology develops or decays (Blank, 1999; Traugott and Dasher, 2001; Kulkarni et al., 2015). Second, LMs should be aware of the ever-changing reality of the world, from a *factual* perspective. Models' factual knowledge should be up-to-date with new facts and concepts (e.g. Covid-19) to be of use continuously. In this work, we focus on the latter; *the temporal robustness of LMs to facts that change over time*.

In an ideal scenario, we would like to know exactly when the factual knowledge of a model is "expired" so that we could adapt it to the new (or updated) set of facts. In reality, this is a challenging task. A large body of work has focused on the part of (continually) *adapting* an "outdated" model to the new data distribution (Guu et al., 2020; Yogatama et al., 2021; Sun et al., 2020; Biesialska et al., 2020; Jang et al., 2022b; Jin et al., 2022; Chakrabarty et al., 2022). This line of work is parallel to ours, as we focus on the crucial step before adaptation, the evaluation of the model on temporal concept drift: *How can we know if a language model is outdated or not?*

Let us consider the case where we desire a lan-

[*] Work done during an internship at AWS AI Labs.

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2881–2898
May 2-6, 2023 ©2023 Association for Computational Linguistics

guage model to be up-to-date with the Prime Minister of the United Kingdom (Figure 1).[1] A plausible way to evaluate this is to use the LAMA-probe paradigm (Petroni et al., 2019) and query the LM as a knowledge base (KB). This would mean that we could form the query as "`The surname of the Prime Minister of the United Kingdom is <mask>.`", give it as an input to a (masked) LM and inspect the output token distribution for the `<mask>` token. Figure 1 shows the top prediction for a series of RoBERTa models.[2] We first observe that the most widely used RoBERTa `base` and `large` models are both outdated in terms of factual knowledge, as they predict the names of PMs that served from 2010 until 2019. Next, while the last three models (2020-2022) answer correctly, the 2019 model answers the (correct) *first name* of the PM (`Boris`), not the *surname* (`Johnson`) which is asked for.

This is a handy illustration of the many *challenges* in evaluating MLMs for temporal robustness in the LMs-as-KBs framework. First, this 2019 model would be considered to have made a mistake (as the prediction is different than the gold label and the metric is accuracy), even though the factual knowledge was correct (the name of the PM of the UK). Second, notice that we designed the query to ask for the surname (instead of the name of the PM), as this results in a single mask. The LAMA-probe and related frameworks do not handle *multi-token* queries for MLMs (e.g., `Boris Johnson`). Finally, we mark with a ? the answers of the first two RoBERTa models, because even though their answers are out-of-date for our current evaluation (October 2022), their answers could have been correct in an evaluation setting in the time of the training data (2019). This illustrates the obscurity of the temporal window in which the model is expected to be correct, if the model is not trained with a temporally-aware design (Lazaridou et al., 2021; Dhingra et al., 2022; Loureiro et al., 2022; Jang et al., 2022a).

In this work, we aim to address such limitations and provide a holistic framework for dynamic benchmarking of masked language models on temporal concept drift, with a focus on facts that change over time. Following the propositions of Kiela et al. (2021) and Søgaard et al. (2021) that advocate for a focus on *dynamic* (i.e., test sets should not become saturated) and *targeted* (i.e., use of multiple, independent test sets for realistic performance estimates) benchmarking respectively, and building on prior work (Jiang et al., 2020b; Dhingra et al., 2022; Jang et al., 2022a), we create a large open-source test set that can be dynamically updated over time, containing temporal fine-grained subsets of examples that can be used to query masked language models and evaluate their factual knowledge over time.

Contributions **(1)** We release DynamicTempLAMA, an improved version of the static TempLAMA (Dhingra et al., 2022) test set consisting of Wikidata relations, that is used to evaluate temporal robustness of MLMs. We provide data and code to dynamically keep DynamicTempLAMA up-to-date over time.[3] **(2)** We propose a novel evaluation framework to first create temporal splits of test sets of any granularity (month, quarter, year) and then to further create fine-grained splits of facts that are *unchanged*, *updated*, *new* or *deleted*, aiming to improve comprehensiveness (§3.1). **(3)** We introduce three distinct evaluation views with multiple metrics (§3.3) to ensure comprehensive results and provide analysis of benchmarking a large set open-source temporal RoBERTa models (§3.2).

2 Related Work

Temporal Concept Drift Evaluation of the robustness of language models on temporal concept drift has seen a rising interest in the recent years. Previous work has focused on methods to continually adapt models over time (Hombaiah et al., 2021; Rosin et al., 2022; Lazaridou et al., 2022). Another area of research is evaluation of temporal robustness which has been explored both in the upstream LM pretraining task (Jiang et al., 2020b; Lazaridou et al., 2021; Dhingra et al., 2022; Jang et al., 2022a; Loureiro et al., 2022) and in downstream tasks such as sentiment analysis (Lukes and Søgaard, 2018; Agarwal and Nenkova, 2022), named entity recognition (Rijhwani and Preotiuc-Pietro, 2020; Onoe et al., 2022), question answering (Mavromatis et al., 2021; Liška et al., 2022), and rumor detection (Mu et al., 2023). It has also been studied for model explanations (Zhao et al., 2022) and for text classification in legal, biomedical (Chalkidis and Søgaard,

[1] The time of writing of this paper is September 2022.

[2] Except for the RoBERTa `base` and `large` models, we also show the predictions of models trained with Twitter data until 2019, 2020, 2021, and 2022, respectively (Loureiro et al., 2022).

[3] https://github.com/amazon-science/temporal-robustness

2022), and social media (Röttger and Pierrehumbert, 2021) domains.

Luu et al. (2022) explore the setting of temporal misalignment (i.e., training and test data drawn from different periods of time) for both upstream and downstream tasks and find that temporal adaptation should not be seen as a substitute for finding temporally aligned labeled data for fine-tuning.

The closest work to ours is TEMPLAMA (Dhingra et al., 2022). However, we differ across four axes: (i) TEMPLAMA is static, while we provide code to dynamically download facts in a fine-grained fashion from any periods of time (not only yearly), (ii) we evaluate the *same* models over time focusing on the evaluation of robustness over time, we do not explore the best adaptation technique to address the problem, (iii) we do not fine-tune the models to adapt them to the domain/format of the test data, and (iv) we address benchmarking of masked LMs (not auto-regressive) including more evaluation techniques. Finally, similar to our motivation, Jang et al. (2022a) recently explored lifelong adaptation and evaluation of temporal concept drift in LMs and introduced TEMPORALWIKI for continual adaptation and TWIKI-PROBES for evaluation. The major difference is that the authors focus on providing corpora to adapt an LM over time, while in our paper we focus on evaluating temporal robustness of LMs. DYNAMICTEMPLAMA is a holistic evaluation framework, while "TWIKI-PROBES are not natural sentences; they are factual phrases synthetically generated from a naive concatenation of Subject, Relation, and Object".

Language Models as Knowledge Bases The cloze-style LM evaluation framework for factual knowledge, LAMA Petroni et al. (2019), follows the setting depicted in Figure 1. A knowledge base relation is transformed into natural language text with a manually created template and then passed as an input to an LM. The framework is based on treating the output distribution for the `mask` token as the retrieved answers to the query (AlKhamissi et al., 2022). The LAMA probe has since been extensively used to evaluate factual knowledge in LMs (Petroni et al., 2020; Talmor et al., 2020; Kassner et al., 2021; Sung et al., 2021; Dhingra et al., 2022; Fierro and Søgaard, 2022), while other works have been exploring its limitations and ways to improve it (Kassner and Schütze, 2020; Haviv et al., 2021; Elazar et al., 2021; Zhong et al., 2021; Qin and Eisner, 2021). A particular challenge in our experimental setting, is the text compatibility between the model (i.e., its pre-training data) and the format of test examples, named as "language mismatch" by Talmor et al. (2020). Dhingra et al. (2022) opts to fine-tune the model under evaluation with part of the test set to adapt it to the format of the task. We argue that this process suffers from many caveats; it is inefficient and impractical to fine-tune a model whose capabilities are under evaluation, it risks optimization stability and overfitting issues due to the small training dataset, and enforces extra biases and errors, especially in the case of temporal robustness evaluation.

3 Dynamic Benchmarking of Temporal Concept Drift

In this section we describe in detail the steps to (re)create DYNAMICTEMPLAMA, our dynamically updated test set with facts from Wikidata (§3.1). We then present the open-source temporal ROBERTA models (TIMELMS) (Loureiro et al., 2022) that we use for benchmarking (§3.2). Finally, we introduce the evaluation framework under which we investigate how well the TimeLMs perform in terms of temporal robustness (§3.3).

The research question that we try to address with our work is: *How can we measure temporal drift robustness of PLMs with an evaluation framework that is*: *unsupervised* (no labeled downstream data), *efficient* (quality test set of facts—no need to run inference on a large corpus to compute perplexity for every token), *dynamic* (test set easily generated per request—can be used to dynamically evaluate new concepts over time), *general* (option to create test sets of any time granularity), and *comprehensive* (battery of targeted test sets that evaluate different LM capabilities and multiple views of evaluation).

3.1 DYNAMIC-TEMPLAMA

We base our implementation on the TEMPLAMA (Dhingra et al., 2022) code, while we make several changes in terms of accessibility (i.e. option to dynamically update the test set), flexibility (i.e. option to adjust the granularity of the temporal splits) and comprehensiveness (i.e. fine-grained splits and multiple evaluation views). We provide a high-level overview of the process to create DYNAMICTEMPLAMA in Figure 2.

Data Collection We start the process by selecting a set of *relations* collected from the Wikidata KB

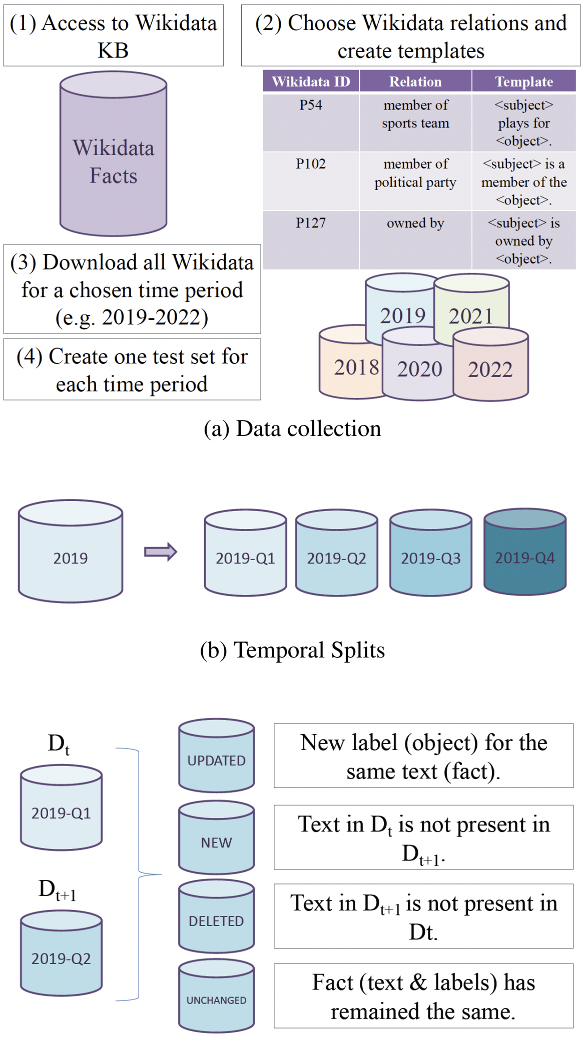
WIKIDATA ID	RELATION	TEMPLATE	#FACTS	#EXAMPLES	POSSIBLE SPLIT(S)
P54	member of sports team	`<subject> plays for <object>.`	3772	50558	$\mathcal{D}^{\text{UPDATED}}$
P69	educated at	`<subject> attended <object>.`	232	2420	$\mathcal{D}^{\text{UPDATED}}$, $\mathcal{D}^{\text{UNCHANGED}}$
P6	head of government	`<object> is the head of the government of <subject>.`	578	7815	$\mathcal{D}^{\text{UPDATED}}$
P279	subclass of	`<subject> is a subclass of <object>.`	5	70	$\mathcal{D}^{\text{NEW}}$, $\mathcal{D}^{\text{UPDATED}}$

Table 1: Examples of relations and their corresponding templates that we include in DYNAMICTEMPLAMA. #FACTS denote the unique number of facts for each relation, while #EXAMPLES denotes the total number of example we have collected for each relation in the time range between `2019-Q1` and `2022-Q2`. POSSIBLE SPLIT(S) indicate the type of fine-grained split that each relation would potentially belong to.

(Figure 2a).[4] Specifically, we use the 9 relations used in the TEMPLAMA dataset, followed by 7 more that we also decided to collect. We collect all relations from Wikidata in the span of 2019 – 2022. We then manually craft a cloze style query, i.e template, for each relation.Table 1 shows a few examples of relations and templates, along with dataset statistics.[5] We explain the data collection process in detail in Appendix A.1.

Temporal Splits In this stage, we have a very large collection of facts for which we have temporal information (i.e., that the fact is true) in the time range we investigate (2019 – 2022). In the TEMPLAMA dataset, the facts are divided yearly. However, we would ideally like to benchmark temporal models of any time granularity. Specifically, since we benchmark temporal models that are trained quarterly (§3.2), a yearly split would not be useful to evaluate temporal concept drift of the four models trained on each quarter of a year. Consequently, we divide the large set of collected facts per quarter (Figure 2b), while adding the functionality to our implementation to split the facts in any time granularity (monthly, quarterly, yearly).

Fine-grained Splits For a given time range, from timestep t to $t+1$ (e.g. `2019-Q1`→`2019-Q2`), we further create comprehensive test sets that contain examples with *unchanged*, *updated*, *new* or *deleted* facts, denoted by $\mathcal{D}^{\text{UNCHANGED}}_{t+1}$, $\mathcal{D}^{\text{UPDATED}}_{t+1}$, $\mathcal{D}^{\text{NEW}}_{t+1}$ and $\mathcal{D}^{\text{DELETED}}_{t+1}$ respectively (Figure 2c). We create these splits to be able to measure different capabilities of the MLM in terms of robustness to temporal concept drift. The motivation for this stems from limitations of prior work (Dhingra et al., 2022) to shed light into what kind of data each temporal test set contains. For instance, we pose questions like *How many facts were updated from timestep $t \rightarrow t+1$? How many facts remained unchanged? What was the change? The object or the subject? Are there new facts in timestep $t+1$ that were not present before?* We argue that it is


(1) Access to Wikidata KB
(2) Choose Wikidata relations and create templates
Wikidata Facts
(3) Download all Wikidata for a chosen time period (e.g. 2019-2022)
(4) Create one test set for each time period
2019
2021
2018
2020
2022

Wikidata ID	Relation	Template
P54	member of sports team	<subject> plays for <object>.
P102	member of political party	<subject> is a member of the <object>.
P127	owned by	<subject> is owned by <object>.

(a) Data collection

2019
2019-Q1
2019-Q2
2019-Q3
2019-Q4

(b) Temporal Splits

D_t
2019-Q1
D_t+1
2019-Q2
UPDATED
NEW
DELETED
UNCHANGED

New label (object) for the same text (fact).

Text in D_t is not present in D_t+1.

Text in D_t+1 is not present in Dt.

Fact (text & labels) has remained the same.

(c) Fine-grained Splits

Figure 2: The process for creating DYNAMICTEMPLAMA. We first collect data from Wikidata (a), we then divide it to quarterly temporal splits (b) and finally we create more targeted fine-grained sets (c).

[4] All possible relations from Wikidata can be found here https://www.wikidata.org/wiki/Wikidata:List_of_properties.

[5] Details on all relations and templates of DYNAMICTEMPLAMA can be found in Tables 6 & 7 in the Appendix A.1.

essential to distinguish between these sub-tests, so that each split can target specific capabilities of the LM. First, we can use $\mathcal{D}_{t+1}^{\text{UNCHANGED}}$ to evaluate knowledge preservation (i.e. how well a model can preserve knowledge over time). Second, we can use $\mathcal{D}_{t+1}^{\text{UPDATED}}$, $\mathcal{D}_{t+1}^{\text{NEW}}$ and $\mathcal{D}_{t+1}^{\text{DELETED}}$ to measure adaptation (i.e. how well a model adapts to new information/facts). Finally, we can measure overall temporal robustness by evaluating a temporal model from timestep t on $\mathcal{D}_{t+1}^{\text{UPDATED}}$ and $\mathcal{D}_{t+1}^{\text{NEW}}$ in timesteps for $t \in [t+1, t+2, ...)$. We believe that this framework is particularly useful for insightful evaluation of methods that aim to adapt language models over time (Guu et al., 2020; Yogatama et al., 2021; Sun et al., 2020; Biesialska et al., 2020; Jang et al., 2022b; Jin et al., 2022; Chakrabarty et al., 2022).

3.2 Temporal Models

In contrast with prior work that uses private, in-house models for temporal robustness evaluation that are not accessible by the community (Lazaridou et al., 2021; Dhingra et al., 2022), we instead benchmark a series of open-source temporal models. Despite our aim for transparency, energy efficiency (Strubell et al., 2019) and reproducibility, we also believe that the *dynamic* nature of the task at hand requires *accessibility* to past, present and future models, to ensure that the findings of evaluation studies in temporal concept drift are meaningful, trustworthy and serve their purpose in evaluating models in a ever-evolving world. Under this assumption, we believe that studies on temporal robustness should ideally build on each other, so that we can have a holistic view as to how these models truly evolve over time.

To this end, we use the Diachronic Language Models (TIMELMS) (Loureiro et al., 2022) that are publicly available in the HuggingFace hub (Wolf et al., 2019).[6] TIMELMS are ROBERTAmodels (Liu et al., 2019) trained *quarterly* on Twitter data. All models are initialised from the original `roberta-base` model checkpoint and are later trained using data from the previous quarters and the new temporal data from the new time period. For instance, the first model (`2019-Q4`) was trained with data sampled from Twitter until December 2019, while the second model (`2020-Q1`) was trained on the concatenation of all the data used to train `2019-Q4` and temporally-aligned data sampled from the first quarter of 2020. There are 11 TIMELMS in total, from `2019-Q4` until `2022-Q2`.

Finally, we would like to draw attention to two specific points. First, all TIMELMS are trained using the same ROBERTA (base) tokenizer and thus have the same vocabulary. This is crucial when evaluating models in a Cloze-style format, like the LAMA-probe, in order to evaluate fair comparison among the models. Second, Loureiro et al. (2022) aim to continue training and releasing TIMELMS every quarter, which is a very important and promising initiative to help with the *dynamic* evaluation of LMs in temporal concept drift in the future.

3.3 Temporal Concept Drift Evaluation

Single-token probing Our first evaluation type is single-token probing, which was introduced in the seminal LAMA-probe work of Petroni et al. (2019). The idea is simple and follows the fill-in-the-blank format. Specifically, we convert each relation using its template to natural language text (see Figure 2(a)) replacing the `<object>` with the mask token (i.e., `<mask>` for ROBERTA). Then, as shown in Figure 1, we give the prompt as an input to the MLM and obtain a probability distribution over the vocabulary for the `<mask>` token. We use the metrics from Petroni et al. (2019), that are `Accuracy`, Mean Reciprocal Rank (`MRR`) and Precision at k (`P@k`).[7] Note that a crucial limitation of this approach is that it considers only facts with single-token objects. This results in trimming down the test sets by 95%, while limiting the actual value of the test (as most facts and concepts contain multiple words).

Multi-token generation We aim to address this limitation and include multi-token objects to our evaluation framework. It is important to note that we are benchmarking *masked* language models instead of autoregressive left-to-right language models like Dhingra et al. (2022). This is crucial because the latter, decoder-based family of models, can be used off-the-shelf to generate multiple tokens. In contrast, MLMs are trained with 15% of their inputs masked and optimized to predict *only the masked tokens*. We therefore use the formulation introduced by Wang and Cho (2019), that is essentially a decoding-based strategy for MLMs based on Gibbs sampling. Specifically, we consider

[6] `https://huggingface.co/cardiffnlp`

[7] `P@k`$=1$, if the gold label is in the top-k predictions of the model, therefore `P@1` corresponds to `Accuracy`.

the setting that we do not know a priori the correct number of masks for each label. Instead, we enumerate from a single mask up to M masks, i.e., $m = 1, ..., M$. Following Jiang et al. (2020a), we choose $M = 5$, as all our facts are in the English language. When $m > 1$, we add m consecutive masks to the input and we pass the input to the model m times, when each time we sequentially sample each mask from left to right. At each iteration we replace the mask with the corresponding token prediction of the previous iteration. This way, we can extend the LAMA probe to include multi-token labels in our test set. The setting is entirely different than the single-token approach, as here we have m predictions from the model with an increasing number of tokens, while the correct label can consist of any number of tokens in the range of $1, ..., M$. Another difference here is the evaluation metrics. Because we converted the task to text generation, we borrow generation metrics such as ROUGE (Lin, 2004), while also including standard metrics like $\texttt{F}_1\texttt{-macro}$. Finally, we also include as a metric BERT-score (Zhang* et al., 2020) as an additional informative metric from the perspective of contextual semantics. In effect, we evaluate factual knowledge over time of MLMs, where facts include *multiple correct answers* and each answer consists of *multiple tokens*. We consider a prediction correct if the model correctly predicts *any* of the acceptable answers.

MLM scoring Finally, as a third lens of evaluation we use the MLM scoring framework of Salazar et al. (2020). Contrary to the previous approaches, MLM scoring aims to *measure* the probability of the correct answer (i.e., of the masks), instead of *generating* the most probable answer. More specifically, we evaluate MLMs out of the box via their *pseudo-log-likelihood scores* (PLLs), which are computed by masking tokens one by one. PLLs have been widely used to measure the equivalence of perplexity (of autoregressive language models) for MLMs in unlabelled data (Lazaridou et al., 2021). Still, computing PLLs for large corpora is a very costly process in terms of time and resources (Loureiro et al., 2022). Instead, we propose to combine the LAMA and MLM scoring frameworks to create an efficient and targeted evaluation framework for temporal factual knowledge.

3.4 Dataset Analysis

We consider different subsets of the DYNAMICTEMPLAMA test sets for the three different evaluation settings (§3.3). For the multi-token and MLM scoring settings, we keep the full dataset, for single-token we first tokenize the labels and keep only the test examples that have at least one label with a single token. This results in a very aggressive filtering of the dataset. Specifically, each quarterly temporal split consists of $~8500$ test examples on average for the multi-token setting, but for the single-token this results in only $~450$ examples, marking a loss of 95% of the data.[8] Additionally, the distribution of the fine-grained splits is of great interest, as it will shape the interpretation of the results and the general challenges of the evaluation framework. $\mathcal{D}^{\text{UPDATED}}$ and $\mathcal{D}^{\text{UNCHANGED}}$ (i.e., the splits of the most interest) constitute around 96% and 0.3%, respectively, of the total examples for the single-token evaluation, and 95% and 1.8% for the multi-token. This is arguably a very skewed distribution, showing the importance of our work in diving the temporal splits into further fine-grained splits. This is essential, because we would have different expectations for a model trained on timestep t while tested on data from both t and $t - 1$; for *unchanged* facts it would be desirable to keep equal performance in both sets (i.e., knowledge preservation §4.2), while for updated facts we would like to see improved performance in timestep t (i.e., adaptation §4.3).

4 Results

4.1 Temporal robustness

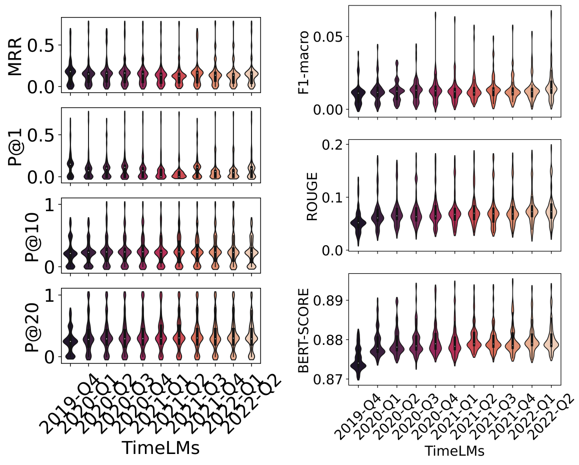
We first evaluate *temporal robustness* of the 11 TIMELMS, defined as the *overall* performance over time (§3.1). Figure 3 shows the average performance in *all* temporal and fine-grained splits in the time range from 2019-Q4 to 2022-Q2 for two types of evaluation, single-token probing and multi-token generation. For the former evaluation type, (Fig. 3a), all models perform similarly for all metrics. However, when we evaluate multi-token generation the models gradually improve over time. (Fig. 3b). This difference shows the importance of considering *multiple views* and evaluations for the same LM capability (i.e., temporal robustness).

We attribute the similar single-token performance to the fact that these temporal datasets con-

[8] Table 5 in the Appendix shows all the statistics in detail.

Models	**Temporal Splits**												
	2019-Q2	2019-Q3	2019-Q4	2020-Q1	2020-Q2	2020-Q3	2020-Q4	2021-Q1	2021-Q2	2021-Q3	2021-Q4	2022-Q1	2022-Q2
2019-Q4	34.88	33.96	34.44	34.93	34.76	34.73	34.02	34.18	34.70	34.34	34.92	35.46	35.31
2020-Q1	24.47	24.01	24.45	24.67	24.59	24.44	23.98	23.94	24.25	23.96	24.20	24.5	24.42
2020-Q2	22.94	22.29	22.92	23.24	23.23	23.12	22.57	22.55	22.90	22.59	22.91	23.23	23.11
2020-Q3	22.39	21.87	22.22	22.60	22.52	22.42	21.99	22.00	22.29	21.92	22.18	22.42	22.30
2020-Q4	25.56	25.28	25.68	25.96	25.89	25.79	25.51	25.44	25.71	25.50	25.69	25.97	25.72
2021-Q1	25.76	25.28	25.91	26.18	26.14	26.18	25.75	25.63	25.99	25.77	26.01	26.32	26.02
2021-Q2	23.75	23.47	23.94	24.10	24.10	24.12	23.63	23.60	24.05	23.75	24.12	24.37	24.16
2021-Q3	22.95	22.61	23.00	23.14	23.12	23.16	22.84	22.77	23.00	22.82	23.03	23.30	23.06
2021-Q4	23.37	23.01	23.41	23.59	23.55	23.68	23.37	23.27	23.60	23.40	23.58	23.76	23.61
2022-Q1	24.25	23.83	24.42	24.56	24.57	24.68	24.40	24.26	24.52	24.35	24.51	24.71	24.58
2022-Q2	**21.48**	**20.95**	**21.42**	**21.59**	**21.57**	**21.61**	**21.25**	**21.12**	**21.44**	**21.13**	**21.31**	**21.49**	**21.39**

Table 2: MLM scoring (median pseudo-log-likelihood scores) averaged for each temporal split.



(a) Single-token

(b) Multi-token

Figure 3: *Overall* performance over time (2019 – 2022) for both single and multi-token evaluation. X-axis corresponds to the TimeLMs and the Y-axis to different metrics depending on the type of the evaluation.

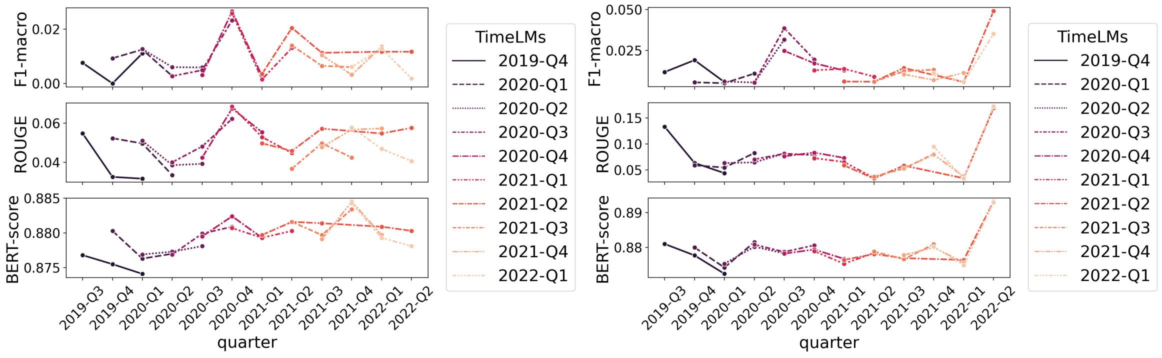
tain almost exclusively *unchanged* facts (§3.4). It is therefore a positive outcome to observe that TimeLMs can preserve acquired knowledge (§4.2). The findings for overall multi-token evaluation corroborate the intuition that more recent models, that are trained with temporal data of the entire range, should perform better than "past" (e.g. 2020) models that have not seen "future" data (e.g. 2022) during training. We also provide the overall results with MLM scoring in Table 2. We also observe that the last model performs best across all temporal splits, showing the effectiveness of adaptation with more recent unlabelled data (§3.2). Even though we observe that this pattern holds for most temporal splits (i.e., scores improving for each column ↓), the 2020-Q4 and 2021-Q1 TimeLMs produce worse PLL scores than their previous or later versions. This is more evident in the overall density plot in Figure 5. This finding entails that either the distribution shift in these quarters was a lot stronger than the other temporal periods, or the training of these particular models was not as successful as it would have been expected.

4.2 Knowledge preservation

We use the $\mathcal{D}^{\text{UNCHANGED}}$ split to evaluate the capability of MLMs to preserve knowledge over time. Figure 6 shows that for both single and multi-token evaluation all TimeLMs demonstrate similar performance over time, showing strong knowledge preserving skills. Surprisingly, different metrics show different patterns among the models for a single split. While in general we should not compare the performance of the single model over time (as the test sets are different), the comparision is valid in this case because the splits contain unchanged facts, and hence most temporal test sets are almost identical. All plots are shown in Figure 7 in the Appendix.

4.3 Adaptation to emerging & evolving concepts

Finally, we use the $\mathcal{D}^{\text{NEW}}$ and $\mathcal{D}^{\text{UPDATED}}$ splits for evaluation of emerging and evolving concepts, respectively. Here to ensure fair comparison, we evaluate the TimeLMs for a specific time window; for each model trained on timestep t, we keep the test sets from $t-1$, t and $t+1$. We observe in Figure 4 that in these cases the results vary among the models. There is not a very clear pattern as before, so case-by-case examination would be required. Still, a common pattern for the Updated split is that the middle set tends to have the highest performance ($\wedge$ shape). This means that models manage to effectively adapt to the updated facts of that timestep (t), but on the next timestep ($t+1$)



(a) UPDATED split.

(b) NEW split.

Figure 4: Multi-token evaluation for evolving and emerging facts.

EXAMPLE INPUT	GROUND TRUTH LABELS	#TOKENS	#ANSWERS	SPLIT
1 Alex Morgan plays for _X_.	United States women's national soccer team Orlando Pride	7 2	2	2021-Q4
2 Cristiano Ronaldo plays for _X_.	Juventus F.C.	5	1	2021-Q2
	Juventus F.C., Manchester United F.C.	5, 6	2	2021-Q3
	Manchester United F.C.	6	1	2021-Q4
2 _X_ is the head of the government of Italy.	Giuseppe Conte	5	1	2020-Q4
	Giuseppe Conte, Mario Draghi	5, 3	2	2021-Q1
	Mario Draghi	3	1	2021-Q2

Table 3: Qualitative analysis of certain examples in DYNAMICTEMPLAMA.

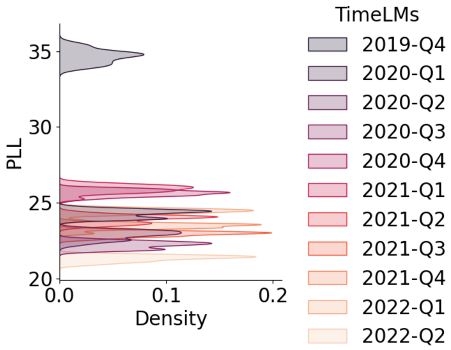


Figure 5: *Overall* PLL distributions for TIMELMS.

they underpeform as they are unaware of the factual changes, thus requiring adaptation. We provide all plots in the Appendix, including the DELETED split, which is more difficult to interpret intuitively (i.e., why are some facts deleted from Wikidata after a certain point?).

5 Qualitative Analysis

Table 3 provides some examples from the DYNAMICTEMPLAMA test set that can help us further interpret our results and inspect existing challenges. We first observe that all examples have multi-token labels (i.e., objects from the `Subject-relation-object` format) and are in effect discarded in the single-token evaluation setup, making the inclusion of multiple views essential for this task.

More specifically, in 1, we observe that one label (`United States women's national soccer team`) has more than $M = 5$ tokens. It is therefore excluded even from the multi-token the test set, leaving MLM scoring to be the only method that could evaluate it. Interestingly, we manually tested the `2021-Q4` temporal model and found that it produces 1.6 and 307.3 average PLL scores for the two options respectively, making the disregarded label a far more confident prediction.

In the second and third example, we observe how the correct answer for the query changes over time, making the granularity of the evaluation (i.e., yearly, quarterly, monthly) an important factor in the correct assessment of the model's temporal factual knowledge. For instance, for the example 3, we can carefully inspect how the predictions of the models change for facts that change over time (Table 4). However, even though PLL scores can follow intuitive temporal patterns (i.e., the PLL value can increase or decrease according to the point in time that the fact has changed), comparison between scores is not always helpful (i.e., word frequency can obscure factual knowledge) leaving

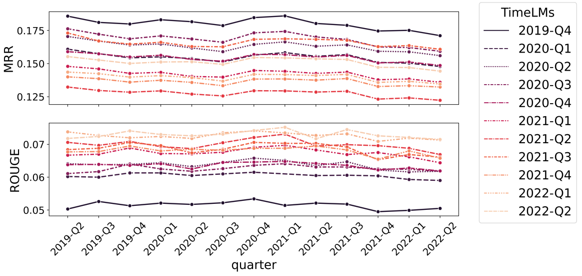


Figure 6: Single and multi-token evaluation for the UNCHANGED split.

TIMELMS	Guiseppe Conte	Mario Draghi
2020-Q4	3.8	33.3
2021-Q1	3.5	22.7
2021-Q2	3.5	25.7
2021-Q3	3.8	23.8

Table 4: PLL scores for Example 2 from Table 3.

room for improving the LAMA formulation.

6 Conclusion & Future Work

We addressed MLMs' robustness on temporal concept drift and introduced DYNAMICTEMPLAMA: a dataset for dynamic benchmarking of factual knowledge in temporal, fine-grained splits, from `2019-Q4` to `2022-Q2` that contain facts over time. We release our codebase to *dynamically* update the current test set over time and the option to extend it with custom (i) templates, (ii) relations from Wikidata, (iii) any period of time (years) and (iv) granularity of time (month/quarter/year). We include *multiple views of evaluation*, showing that it is essential in order to properly interpret the results of our benchmarking study of 11 temporal ROBERTA models. We consider experimentation with improving MLM decoding and addressing "domain mismatch" as open areas of research for future work. Our code can be found at https://github.com/amazon-science/temporal-robustness.

Acknowledgements

We would like to thank the anonymous reviewers and the AWS AI team for their valuable feedback on the paper. We also thank Robert Vacareanu and Karthikeyan K for their help with running experiments.

Limitations

Lower bound estimate A very common issue with the LAMA probe evaluation framework (Petroni et al., 2019) is that it constitutes a lower bound estimate for its performance on factual knowledge retrieval. Specifically, if a model performs well, one can infer that it has the tested reasoning skill. However, failure does not entail that the reasoning skill is missing, as it is possible that there is a problem with the lexical-syntactic construction we picked (Talmor et al., 2020). Any given prompt only provides a lower bound estimate of the knowledge contained in an LM (Jiang et al., 2020b).

Domain mismatch Despite the advantages of zero-shot evaluation, performance of a model might be adversely affected by mismatches between the language the pre-trained LM was trained on and the language of the examples in our tasks (Jiang et al., 2020b). It is quite possible that a fact that the LM does know cannot be retrieved due to the prompts not being effective queries for the fact (Jiang et al., 2020b). Prior work proposes to fine-tune the model with a small set of examples taken from the test set (and removed of course) in order to address the incompatibility problem or 'language mismatch' (Talmor et al., 2020; Dhingra et al., 2022). We argue that this process suffers for multiple limitations, such as that it not practical for a fast evaluation of the capabilities of a PLM at hand and it faces optimization stability issues due to the small training dataset, inter alia. The major limitation, however, is that such fine-tuning enforces extra biases and errors, especially in the case of temporal robustness evaluation.

MLM decoding (multi-token labels) In this work we tried to address the problem of *decoding from masked language models*, by incorporating two distinct approaches to the evaluation framework; multi-token generation with MLMs (Wang and Cho, 2019) and MLM scoring (Salazar et al., 2020). Still, we observe that both methods provide results that are hard to interpret (§5), leaving the problems of (i) decoding or generating multiple tokens from MLMs and (ii) evaluation of factual knowledge in LMs as open areas of research.

Manual Templates For LAMA-style probing (Petroni et al., 2019), prior work creates the templates *manually*. This is a limitation both in terms of scale (i.e., generalization to many different kinds of inputs) and consistency (i.e., how do models perform with minimal changes to their inputs?). LMs do not reason in an abstract manner and are context-dependent (Talmor et al., 2020). It is therefore essential to address this problem and include functionalities to incorporate a set of diverse templates for each evaluation setup.

English Twitter MLMs Finally, our dataset, DYNAMICTEMPLAMA, following prior work (Dhingra et al., 2022), collects and evaluates facts from the Wikidata in the *English* language alone, and benchmarks RoBERTa language models trained in English Twitter data. We understand that this is a limitation and further data collection and experimentation in more languages would be strongly encouraged.

References

Oshin Agarwal and Ani Nenkova. 2022. Temporal effects on pre-trained models for language processing tasks. *Transactions of the Association for Computational Linguistics*, 10:904–921.

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases.

Magdalena Biesialska, Katarzyna Biesialska, and Marta R. Costa-jussà. 2020. Continual lifelong learning in natural language processing: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6523–6541, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Andreas Blank. 1999. Why do new meanings occur? a cognitive typology of the motivations for lexical semantic change.

Tuhin Chakrabarty, Thomas Scialom, and Smaranda Muresan. 2022. Fine-tuned language models can be continual learners. In *Challenges & Perspectives in Creating Large Language Models*.

Ilias Chalkidis and Anders Søgaard. 2022. Improved multi-label classification under temporal concept drift: Rethinking group-robust algorithms in a label-wise setting. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland.

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. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031.

Constanza Fierro and Anders Søgaard. 2022. Factual consistency of multilingual pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3046–3052, Dublin, Ireland. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Adi Haviv, Jonathan Berant, and Amir Globerson. 2021. BERTese: Learning to speak to BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3618–3623, Online. Association for Computational Linguistics.

Spurthi Amba Hombaiah, Tao Chen, Mingyang Zhang, Michael Bendersky, and Marc-Alexander Najork. 2021. Dynamic language models for continuously evolving content. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*.

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 *International Conference on Learning Representations*.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020a. X-FACTR: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5943–5959, Online. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020b. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2022. Lifelong pretraining: Continually adapting language models to emerging corpora. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 1–16, virtual+Dublin. Association for Computational Linguistics.

Nora Kassner, Philipp Dufter, and Hinrich Schütze. 2021. Multilingual LAMA: Investigating knowledge in multilingual pretrained language models. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

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 *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics.

Vivek Kulkarni, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2015. Statistically significant detection of linguistic change. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, page 625–635, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

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, Tomáš Kočiský, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. Mind the gap: Assessing temporal generalization in neural language models. In *Advances in Neural Information Processing Systems*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

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.

Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-collados. 2022. TimeLMs: Diachronic language models from Twitter. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 251–260, Dublin, Ireland. Association for Computational Linguistics.

Jan Lukes and Anders Søgaard. 2018. Sentiment analysis under temporal shift. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 65–71.

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 *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5944–5958, Seattle, United States. Association for Computational Linguistics.

Costas Mavromatis, Prasanna Lakkur Subramanyam, Vassilis N. Ioannidis, Soji Adeshina, Phillip R. Howard, Tetiana Grinberg, Nagib Hakim, and George Karypis. 2021. Tempoqr: Temporal question reasoning over knowledge graphs.

Yida Mu, Kalina Bontcheva, and Nikolaos Aletras. 2023. It's about time: Rethinking evaluation on rumor detection benchmarks using chronological splits. In *Findings of the Conference of the European Chapter of the Association for Computational Linguistics*.

Yasumasa Onoe, Michael Zhang, Eunsol Choi, and Greg Durrett. 2022. Entity cloze by date: What LMs know about unseen entities. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 693–702, Seattle, United States. Association for Computational Linguistics.

Fabio Petroni, Patrick S. H. Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2020. How context affects language models' factual predictions. *CoRR*, abs/2005.04611.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing*.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5203–5212, Online. Association for Computational Linguistics.

Shruti Rijhwani and Daniel Preotiuc-Pietro. 2020. Temporally-informed analysis of named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7605–7617, Online. Association for Computational Linguistics.

Guy D. Rosin, Ido Guy, and Kira Radinsky. 2022. Time masking for temporal language models. *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*.

Paul Röttger and Janet Pierrehumbert. 2021. Temporal adaptation of BERT and performance on downstream document classification: Insights from social media. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2400–2412, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. Association for Computational Linguistics.

Anders Søgaard, Sebastian Ebert, Jasmijn Bastings, and Katja Filippova. 2021. We need to talk about random splits. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1823–1832, Online. Association for Computational Linguistics.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. *CoRR*, abs/1906.02243.

Fan-Keng Sun, Cheng-Hao Ho, and Hung-Yi Lee. 2020. {LAMAL}: {LA}nguage modeling is all you need for lifelong language learning. In *International Conference on Learning Representations*.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4723–4734, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*.

Elizabeth Closs Traugott and Richard B. Dasher. 2001. *Regularity in Semantic Change*. Cambridge Studies in Linguistics. Cambridge University Press.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36, Minneapolis, Minnesota. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Dani Yogatama, Cyprien de Masson d'Autume, and Lingpeng Kong. 2021. Adaptive semiparametric language models. *Transactions of the Association for Computational Linguistics*, 9:362–373.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Zhixue Zhao, George Chrysostomou, Kalina Bontcheva, and Nikolaos Aletras. 2022. On the impact of temporal concept drift on model explanations. In *Findings of the Conference on Empirical Methods in Natural Language Processing*.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [MASK]: Learning vs. learning to recall. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5017–5033, Online. Association for Computational Linguistics.

TEMPORAL SPLIT	UNCHANGED	UPDATED	DELETED	NEW	TOTAL	%UNCHANGED	%UPDATED	%LOST
2019-Q2	479|8523	1|165	7|124	9|121	496|8933	96.6|95.4%	0.2|1.8%	94.4%
2019-Q3	451|8154	3|248	36|430	5|205	495|9037	91.1|90.2%	0.6|2.7%	94.5%
2019-Q4	454|8271	0|151	3|140	12|120	469|8682	96.8|95.3%	0.0|1.7%	94.6%
2020-Q1	456|8243	3|296	9|126	15|273	483|8938	94.4|92.2%	0.6|3.3%	94.6%
2020-Q2	470|8451	0|92	2|95.	2|59	474|8697	99.2|97.2%	0.0|1.1%	94.5%
2020-Q3	446|8254	2|179	26|238	10|133	484|8804	92.1|93.8%	0.4|2.0%	94.5%
2020-Q4	452|8298	2|124	4|111	5|97	463|8630	97.6|96.2%	0.4|1.4%	94.6%
2021-Q1	453|8238	1|269	4|131	14|215	472|8853	96.0|93.1%	0.2|3.0%	94.7%
2021-Q2	460|8344	2|90	7|128	5|76	474|8638	97.0|96.6%	0.4|1.0%	94.5%
2021-Q3	445|8164	2|164	19|220	2|99	468|8647	95.1|94.4%	0.4|1.9%	94.6%
2021-Q4	443|8213	1|128	4|82	5|90	453|8513	97.8|96.5%	0.2|1.5%	94.7%
2022-Q1	442|8189	1|111	7|117	6|126	456|8543	96.9|95.9%	0.2|1.3%	94.7%
2022-Q2	446|8287	0|56	2|40	2|34	450|8417	99.1|98.5%	0.0|0.7%	94.7%

Table 5: Total number of examples for each temporal and fine-grained split in DYNAMICTEMPLAMA. We show both the *single-token* and the *multi-token* datasets (up to $M = 5$ tokens). Cell scheme to be read *single* | *multi*. %UNCHANGED and %UPDATED show the percentage of the total examples that are part of the UNCHANGED and UPDATED set respectively. %LOST shows the percentage of examples we lose when we filter out the dataset for the *single-token* evaluation setting.

A Appendix

A.1 Data Collection for DYNAMICTEMPLAMA

Following Dhingra et al. (2022), we identify all facts in the Wikidata snapshot, which have either a start or an end date after 2010 and whose subjects and objects are both entities with Wikipedia pages.1 Among these 482K facts, we identify subject and relation pairs which have multiple objects at different times and select 16 relations with the most such subjects. Then, for these relations we manually write template cloze queries (i.e., templates) and populate them with the 1000 most frequent subjects per relation. For each subject and each relation we gather all the objects with their associated time interval and construct a separate query for each year in that interval. When intervals for the object entities overlap, we add all of them to the list of correct answers. The query and the corresponding year form the input texts and the temporal information t, while the object entity is the target that we want to predict (i.e., gold label). In contrast to Dhingra et al. (2022), we do extra temporal divisions. Specifically, we get each yearly split and divide it further in quarterly splits (§3.1, Figure 2b), following the same algorithm.

A.2 Full Results

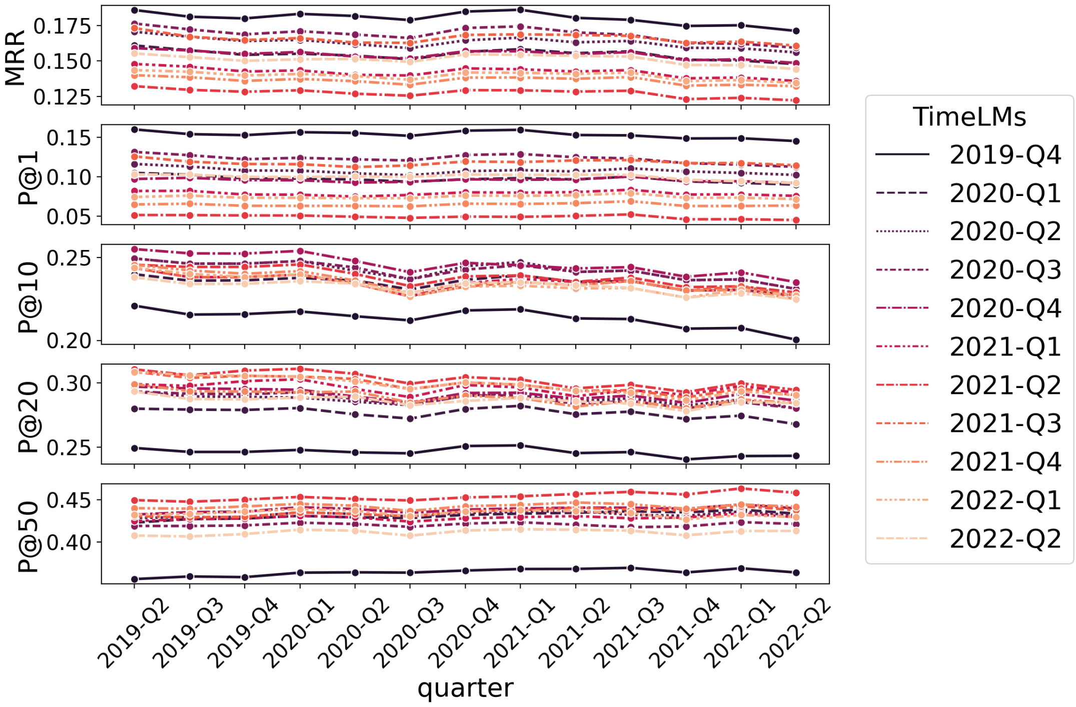
We provide the full results with all metrics for the UNCHANGED split in Figure 7, and the UPDATED, NEW and DELETED splits for multi-token generation in Figure 9.

WIKIDATA ID	RELATION	TEMPLATE	#FACTS	#EXAMPLES	POSSIBLE SPLIT(S)
P54	member of sports team	`<subject> plays for <object>.`	3772	50558	$\mathcal{D}^{\text{UPDATED}}$
P39	position held	`<subject> holds the position of <object>.`	2961	34835	$\mathcal{D}^{\text{UPDATED}}$
P108	employer	`<subject> works for <object>.`	1544	20531	$\mathcal{D}^{\text{UPDATED}}$
P102	political party	`<subject> is a member of the <object>.`	1068	14232	$\mathcal{D}^{\text{UPDATED}}$
P286	head coach	`<object> is the head coach of <subject>.`	987	11935	$\mathcal{D}^{\text{UPDATED}}$
P69	educated at	`<subject> attended <object>.`	232	2420	$\mathcal{D}^{\text{UPDATED}}$, $\mathcal{D}^{\text{UNCHANGED}}$
P488	chairperson	`<object> is the chair of <subject>.`	629	8468	$\mathcal{D}^{\text{UPDATED}}$
P6	head of government	`<object> is the head of the government of <subject>.`	578	7815	$\mathcal{D}^{\text{UPDATED}}$
P279	subclass of	`<subject> is a subclass of <object>.`	5	70	$\mathcal{D}^{\text{NEW}}$, $\mathcal{D}^{\text{UPDATED}}$
P127	owned by	`<subject> is owned by <object>.`	394	5326	$\mathcal{D}^{\text{UPDATED}}$, $\mathcal{D}^{\text{UNCHANGED}}$
P1001	legal term	`<subject> is a legal term in <object>.`	37	423	$\mathcal{D}^{\text{UNCHANGED}}$
P106	profession	`<subject> is a <object> by profession.`	83	1090	$\mathcal{D}^{\text{UPDATED}}$, $\mathcal{D}^{\text{NEW}}$, $\mathcal{D}^{\text{UNCHANGED}}$
P27	citizen	`<subject> is <object> citizen.`	147	1983	$\mathcal{D}^{\text{NEW}}$, $\mathcal{D}^{\text{UNCHANGED}}$
P176	produced by	`<subject> is produced by <object>.`	24	276	$\mathcal{D}^{\text{NEW}}$, $\mathcal{D}^{\text{UNCHANGED}}$
P138	named after	`<subject> is named after <object>.`	73	1009	$\mathcal{D}^{\text{NEW}}$, $\mathcal{D}^{\text{UNCHANGED}}$
P937	work location	`<subject> used to work in <object>.`	38	507	$\mathcal{D}^{\text{NEW}}$, $\mathcal{D}^{\text{UNCHANGED}}$

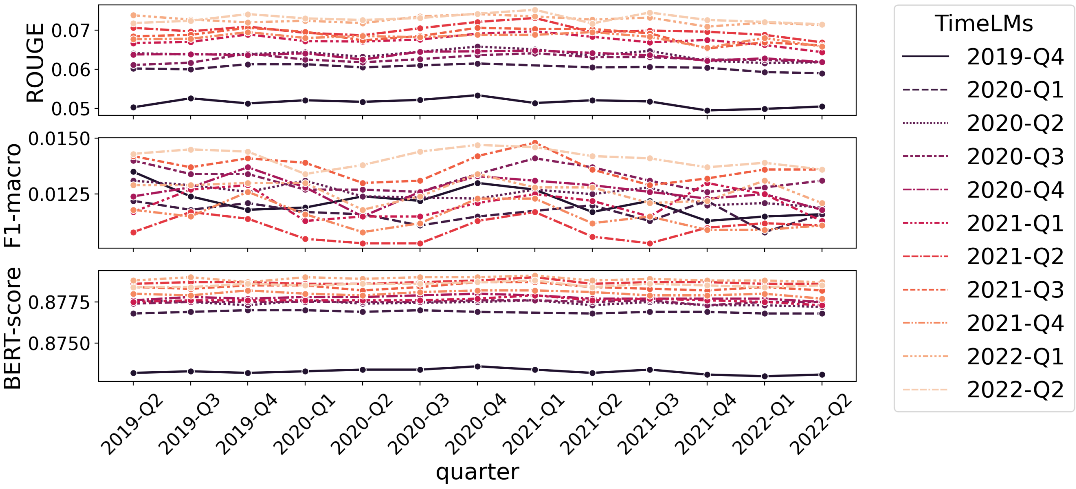
Table 6: The list of templates we used for each relation in the DYNAMICTEMPLAMA dataset.

Wikidata ID	Relation	Input	Labels	Split
P54	member of sports team	Cristiano Ronaldo plays for _X_.	Juventus F.C., Manchester United F.C.	2021-Q3
P39	position held	Martina Anderson holds the position of _X_.	member of the European Parliament	2019-Q4
P108	employer	George van Kooten works for _X_.	University of Cambridge	2022-Q2
P102	political party	Elena Kountoura is a member of the _X_.	Independent Greeks, SYRIZA	2019-Q2
P286	head coach	_X_ is the head coach of New York Red Bulls.	Gerhard Struber	2020-Q4
P69	educated at	Sarafina Nance attended _X_.	Tufts University, University of California, Berkeley	2020-Q2
P488	chairperson	_X_ is the chair of Lloyds Banking Group.	Lord Blackwell	2022-Q2
P6	head of government	_X_ is the head of the government of United Kingdom.	Theresa May, Boris Johnson	2019-Q3
P279	subclass of	Mercedes-Benz A-Class is a subclass of _X_.	compact car	2022-Q2
P127	owned by	DeepMind is owned by _X_.	Alphabet Inc.	2021-Q4
P1001	legal term	Commonwealth of Independent States Free Trade Area is a legal term in _X_.	’Ukraine’, ’Russia’, ’Belarus’, ’Armenia’, ’Kazakhstan’, ’Moldova’, ’Kyrgyzstan’, ’Uzbekistan’, ’Tajikistan’	2022-Q2
P106	profession	Penny James is a _X_ by profession.	chief executive officer	2019-Q3
P27	citizen	Yulia Putintseva is _X_ citizen.	Kazakhstan 2022-Q1	
P176	produced by	Land Rover Discovery series is produced by _X_.	Jaguar Land Rover	2022-Q2
P138	named after	Bayes Business School is named after _X_.	Thomas Bayes	2021-Q3
P937	work location	Eliza Vozemberg used to work in _X_.	Strasbourg, City of Brussels	2022-Q2

Table 7: Examples of DynamicTempLAMA for each relation.

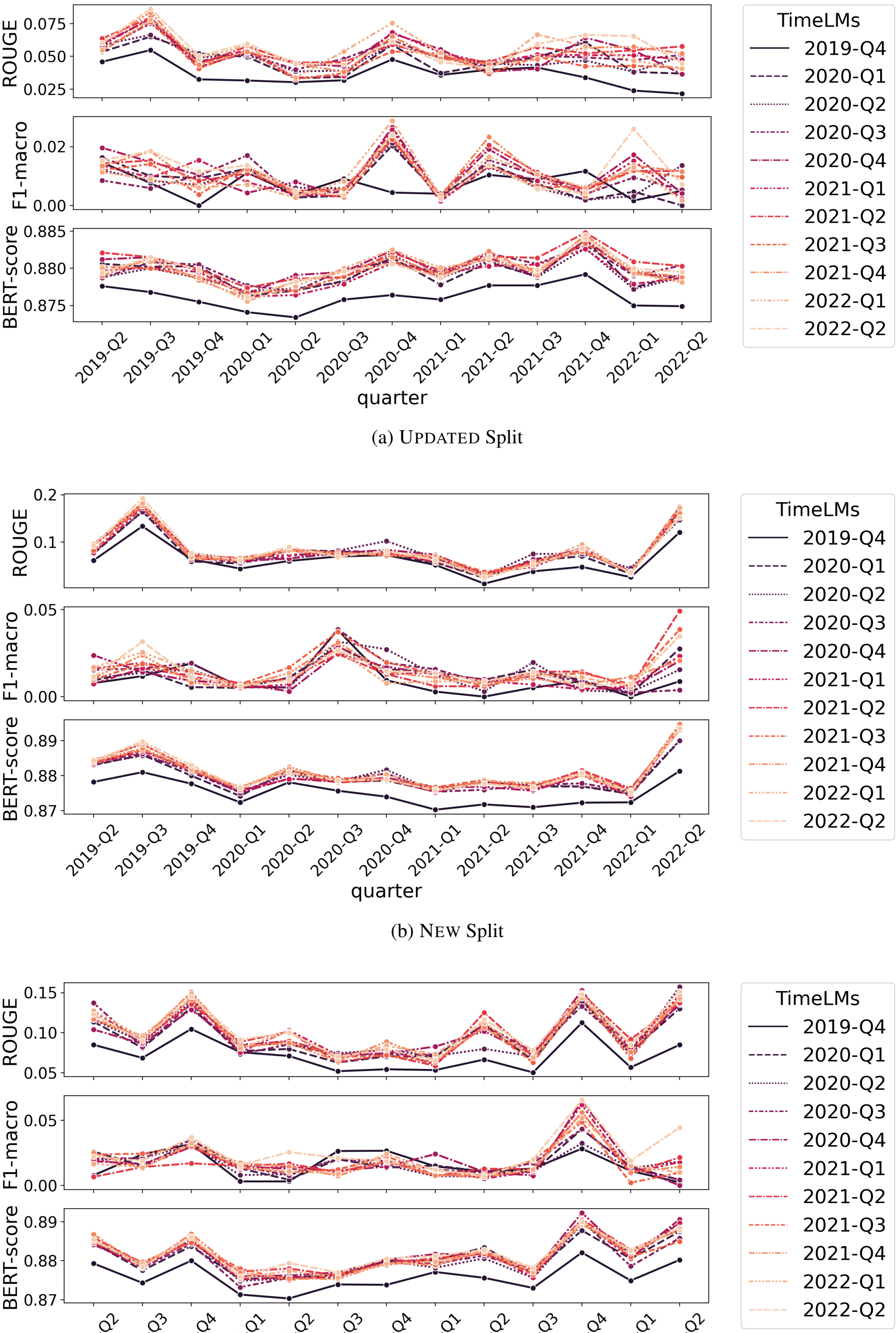


(a) Single Token



(b) Multi-token

Figure 7: Single-token probing and multi-token generation for the UNCHANGED split.

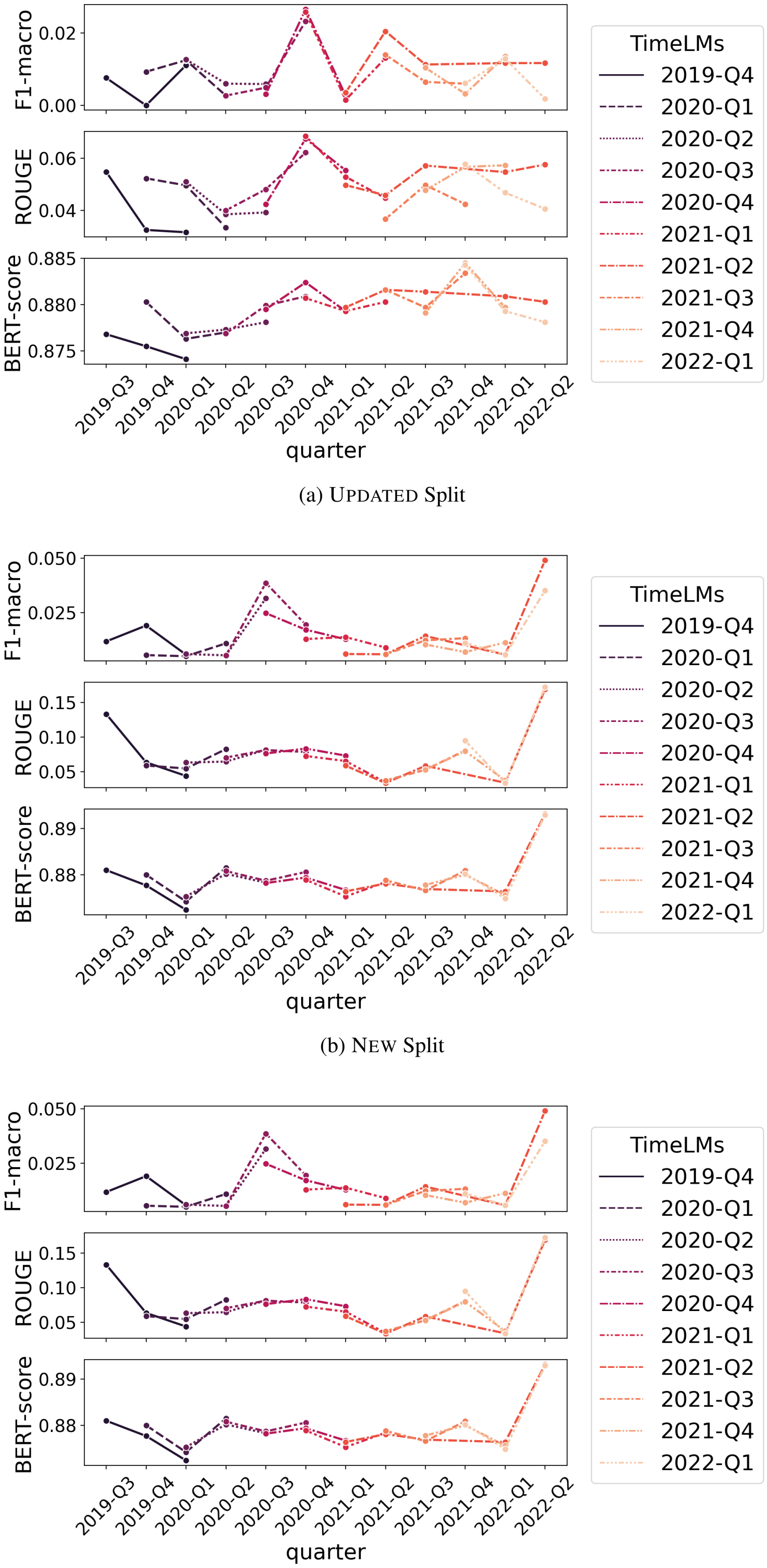


(a) UPDATED Split

(b) NEW Split

(c) DELETED Split

Figure 8: Multi-token generation results for various fine-grained splits.



(a) UPDATED Split

(b) NEW Split

(c) DELETED Split

Figure 9: Multi-token generation results for various fine-grained splits. Here for each model trained on timestep t, we keep the test sets from $t-1$, t and $t+1$.