Can Language Models Serve as Temporal Knowledge Bases?

Ruilin Zhao[1,2], **Feng Zhao**[1*], **Guandong Xu**[2], **Sixiao Zhang**[2], **Hai Jin**[1]
[1]National Engineering Research Center for Big Data Technology and System,
Services Computing Technology and System Lab, Cluster and Grid Computing Lab,
School of Computer Science and Technology, Huazhong University of Science and Technology, China
[2]Data Science and Machine Intelligence Lab, University of Technology Sydney, Sydney, Australia
{ruilinzhao,zhaof,hjin}@hust.edu.cn, guandong.xu@uts.edu.au, zsx57575@gmail.com

Abstract

Recent progress regarding the use of *language models* (LMs) as *knowledge bases* (KBs) has shown that language models can act as structured knowledge bases for storing relational facts. However, most existing works only considered the LM-as-KB paradigm in a static setting, which ignores the analysis of temporal dynamics of world knowledge. Furthermore, a basic function of KBs, i.e., the ability to store conflicting information (i.e., 1-N, N-1 and N-M relations), is underexplored. In this paper, we formulate two practical requirements for treating LMs as temporal KBs: (i) the capacity to store temporally-scoped knowledge that contains conflicting information and (ii) the ability to use stored knowledge for temporally-scoped knowledge queries. We introduce a new dataset called LAMA-TK which is aimed at probing temporally-scoped knowledge, and investigate the two above requirements to explore the LM-as-KB paradigm in the temporal domain. On the one hand, experiments show that LMs can memorize millions of temporally-scoped facts with relatively high accuracy and transfer stored knowledge to temporal knowledge queries, thereby expanding the LM-as-KB paradigm to the temporal domain. On the other hand, we show that memorizing conflicting information, which has been neglected by previous works, is still challenging for LMs and hinders the memorization of other unrelated one-to-one relationships.

1 Introduction

Recently, *language models* (LMs) such as BERT (Devlin et al., 2019) and T5 (Raffel et al., 2020) have been suggested as an alternative to world *knowledge bases* (KBs) (Petroni et al., 2019). The parameters of these models appear to store extensive real-world knowledge during training and stored knowledge can be recalled by filling cloze statements (e.g. "Dani Alves plays with [MASK]. –> Barcelona"). As a result, recent works have considered LMs for tasks such as closed-book question answering (Roberts et al., 2020), automated fact-checking (Guo et al., 2021), and knowledge-grounded dialogue systems (Liu et al., 2022).

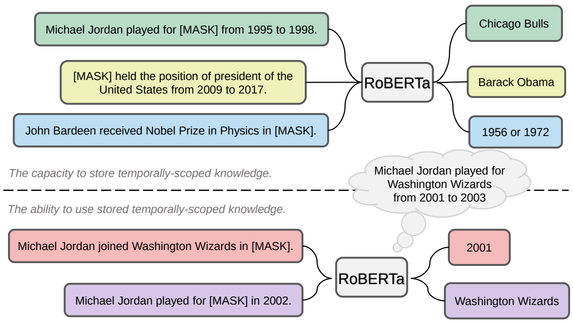
Michael Jordan played for [MASK] from 1995 to 1998.
[MASK] held the position of president of the United States from 2009 to 2017.
John Bardeen received Nobel Prize in Physics in [MASK].
RoBERTa
Chicago Bulls
Barack Obama
1956 or 1972
The capacity to store temporally-scoped knowledge.
The ability to use stored temporally-scoped knowledge.
Michael Jordan played for Washington Wizards from 2001 to 2003
Michael Jordan joined Washington Wizards in [MASK].
Michael Jordan played for [MASK] in 2002.
RoBERTa
2001
Washington Wizards

Figure 1: Expansion of the LM-as-KB paradigm to the temporal domain. We introduce two requirements to further explore the capability of LMs. (i) The capability to store temporal knowledge and (ii) the ability to use stored knowledge for temporal knowledge queries.

Relational facts in world knowledge often change with time. For example, "Michael Jordan played for Washington Wizards." is true only from 2001 to 2003. However, most existing works only considered the LM-as-KB paradigm in a "static" setting, ignoring the temporal dynamics of world knowledge. However, this temporally-scoped knowledge raises several potential challenges for the LM-as-KB paradigm.

Conflicting Information While training on the large textual corpus, the model will inevitably encounter conflicting information (i.e., 1-N, N-1, N-M relations), e.g., "Giannis Antetokounmpo played for Filathlitikos / Milwaukee Bucks". Dhingra et al. (2022) limits the time period of facts to *reduce* the amount of conflicting information. However, conflicting information still exists, from the players who played for a team to the politician who held multiple positions. These conflicting facts will hinder the memorizing process and

[*]Corresponding author

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 2024–2037
December 7-11, 2022 ©2022 Association for Computational Linguistics

cause the model to have difficulty memorizing all correct answers.

Correlation Between Temporal Scopes Temporal facts usually contain temporal scopes (e.g., a start time and an end time), and a strong correlation is present between these timestamps. For example, "Shinzo Abe was the prime minister of Japan *from 2006 to 2007*." and "Shinzo Abe was the prime minister of Japan *from 2012 to 2020*." are two temporally-scoped facts. These facts have the same subject, object, and predicate but different temporal scopes. As temporal knowledge bases, LMs need to memorize not only the timestamps associated with the facts but also the matching relationships between temporal scopes.

Implicit Temporal Knowledge Temporally-scoped facts usually contain implicit facts. For example, "François Hollande served as president of the French Republic from 2012 to 2017" contains the following facts: "François Hollande served as president of the French Republic in *2015*" and "François Hollande *was elected* president of the French Republic in 2012". These implicit facts are not directly mentioned in temporally-scoped facts.

Temporally-scoped knowledge widely exists in real-world knowledge bases like Wikidata. However, existing QA datasets such as LAMA (Petroni et al., 2019), Natural Questions (Kwiatkowski et al., 2019) focus on probing static knowledge, ignoring the temporal dynamics of world knowledge. The temporal dataset TEMPLAMA (Dhingra et al., 2022) focuses on querying factual objects with single timestamps, ignoring the temporally-scoped information such as the start and end times. Moreover, temporal facts often contain extensive conflicting information, but previous works did not pay enough attention to these conflicts. They explored LM-as-KB within 1-1 relations (e.g. born in) or discarded facts with multiple objects. Therefore, we propose LAMA-TK (short for ***LA**nguage **M**odel **A**nalysis for **T**emporal **K**nowledge*), a new dataset for probing LMs for temporally-scoped knowledge. LAMA-TK queries temporal knowledge including entity names and special timestamps, and reserves all correct answers for each query.

Based on LAMA-TK, we introduce two practical questions for LMs as temporal KBs **to explore the LM-as-KB paradigm in the temporal domain**. We examine LMs on two basic functions of KBs: the storage capacity and the use of stored temporal knowledge, and identify the challenges mentioned above during the experiments (Section 4).

Table 1: Examples from LAMA, TEMPLAMA, and our proposed LAMA-TK. LAMA-TK is a novel dataset of temporal knowledge statements, which takes into account entities, temporal scopes and multiple answers.

Input	**Target(s)**
LAMA	
Dante was born in [MASK]. Bailey Peninsula is located in [MASK].	Florence Antarctica
TEMPLAMA	
year: 2013 text: Marina Silva is a member of the _X_. year: 2018 text: Marina Silva is a member of the _X_.	Brazilian Socialist Party Sustainability Network
LAMA-TK	
Michael Jordan played for [MASK] from 1995 to 1998. Michael Jordan played for [MASK] in 2002. Michael Jordan received NBA Most Valuable Player Award in [MASK].	Chicago Bulls Washington Wizard 1988, 1991, 1992, 1996, 1998

First question: What is the storage capacity of LMs for storing temporal knowledge? Here, we ask the models to memorize millions of temporal facts and record the storage performance of these LMs (Section 4.1). Results show that LMs can memorize millions of temporal facts with relatively high accuracy. However, we also show that conflicting information poses a great challenge to the storage capacity of LMs and hinders the memorizing process of other unrelated facts.

Second question: Can LMs use stored temporal knowledge for temporally-scoped knowledge queries? Here, we design targeted queries to recall stored temporal facts (Section 4.2) and further explore the ability of LMs to recall implicit temporal facts (Section 4.3). Results show that pretrained LMs can transfer stored temporally-scoped knowledge to new queries with similar semantics even if the query templates are not observed during training. Moreover, we show that with prompts like "*from* ST *to* ET", LMs can understand the difference and continuity of temporal scopes. These results show that LMs can efficiently handle temporal knowledge.

Contributions: (1) We introduce three challenges and two practical requirements for treating LMs as temporal KBs, which expands the LM-as-KB paradigm (Petroni et al., 2019) to the temporal domain. (2) We offer LAMA-TK, a new dataset for probing LMs for temporally-scoped knowledge. (3) We propose a prompt-based temporal scope modeling method to jointly model temporal scopes and facts for adapting LMs to temporally-scoped facts. (4) We conduct experiments to evaluate the capacity of LMs to store temporal facts and examine the ability of LMs to use stored knowledge for temporal queries. (5) We show the negative impact of conflicting information on the storage capacity of LMs, which was neglected by previous works.

2 The LAMA-TK Probe

In this section, we detail the construction of LAMA-TK[1], our new temporally-scoped knowledge probing dataset, including its data sources and a set of natural language queries for probing temporal knowledge, as well as the evaluation metric we use.

2.1 Knowledge Sources

CronQuestions CronQuestions (Saxena et al., 2021) is a KGQA[2] dataset, including a *knowledge graph* (KG) with associated timestamps and 350K temporal questions. There are 323k facts, 125k entities, and 203 relations in its KG. We selected the top-5 most frequent temporally rich relations, resulting in a KG with 226K facts, 96k entities, and 1322 timestamps.

Wikidata Wikidata[3] is a public KB that stores a massive amount of structured data. We use the dump of the January 3rd, 2022 version and retrieve facts that have both start and end dates using SPARQL queries. Following Dhingra et al. (2022), we identify the factual knowledge that has more than one object at the different time periods and select 6 relations with the most such objects. This results in a KG with 497K facts, 260k entities, and 1132 timestamps.

2.2 Temporal Knowledge Queries

According to the above knowledge sources, we finally construct a KG with 639k facts, 316k entities, 1539 timestamps, and 7 relations. Following Jiang et al. (2020); Dhingra et al. (2022), we write templates for these relations and convert temporal knowledge to natural language statements. For example, the temporal knowledge <Giannis Antetokounmpo, Member of Sports Team, Filathlitikos B.C., 2011, 2013> was converted into a natural language statement "Giannis Antetokounmpo played for Filathlitikos B.C. from 2011 to 2013". Based on these statements, we design targeted cloze-style queries and reserve the **masked entity** as the training target. Templates, example queries, and data pre-processing details have been shown in Appendix A.

Real-world knowledge contains extensive conflicting information, from the players who played for a team to a politician who held multiple positions. However, most previous works tend to explore the LM-as-KB paradigm within one-to-one relationships (e.g. "born in") or only reserve one of the correct answers as the target, without taking into account whether LMs have similar confidences in other correct answers. Therefore, in LAMA-TK, we do not discard conflicting information and reserve all correct answers of each masked statement as the **answer list** (see Table 9 for the different between *masked entity* and *answer list*). Among the 2.48M masked factual statements, there are 379K statements (15%) with multiple answers.

2.3 Evaluation Metric

As many queries have multiple answers, we use the top-K accuracy (Acc@K) to measure how well the model performs on these queries. The Acc@k is "query-oriented". For each query, the top-K accuracy is 1 if any of the correct answer is in the top k predictions, and is 0 otherwise. In this work, we use both Acc@1 and Acc@5.

However, Acc@K can only measure whether the model can recall at least one correct answer, but it ignores the memorization performance of other correct answers of a multi-answer query[4]. Therefore, we use Hit at top k (Hit@K) to take into account all correct answers to each query. The Hit@K is "answer-oriented". For each correct answer to the query, if the correct answer is in the top k predictions, Hit@K is 1, otherwise is 0. In this work, we use Hit@5 and Hit@10.

3 Models

Following Heinzerling and Inui (2021), we use RoBERTa (Liu et al., 2019), the bidirectional LM, as the knowledge base. Moreover, we adopt several approaches to adapt the original RoBERTa to temporally-scoped knowledge and prepare three different RoBERTa models for examination.

[1] The LAMA-TK dataset is available at `https://github.com/CGCL-codes/LAMA-TK`

[2] Question Answering over Knowledge Graph.

[3] `www.wikidata.org`

[4] See Appendix C for more details.

3.1 Adaptations

Prompt-based Temporal Scope Modeling To jointly model temporal scopes and texts, we manually write *prompt templates* for temporal facts and directly encode temporal scopes in training process. Given a factual sequence of tokens $X = [x_1, x_2, .., x_n]$ and its associating temporal scope <ST, ET> . We use prompt template "*from* ST *to* ET" to convert temporal scope to natural language text and incorporate this text into the factual sequence. In this case, the final factual sequence $X' = [x_1, x_2..., x_n, from, ST, to, ET]$. See Appendix B for further analysis.

Symbolic Representation However, the pretrained masked language model can only handle entities whose names are in its vocabulary (e.g., entities like "English" and "Florida"). This results in its inability to predict entities with multiple tokens (e.g., entities like "Barack Obama"). In this work, we follow Heinzerling and Inui (2021) to store entities by symbolic representation, i.e., augmenting the vocabulary of LM and representing all the entities as entries in the vocabulary. The LM will project the final hidden state of the [MASK] token onto the vocabulary and take a softmax over all entities (Heinzerling and Inui, 2021). Although symbolic representation is computationally expensive, it can memorize entities with high accuracy and will not be affected by the length of the entity name.

Memorizing Facts via MLM In this work, we train the model to memorize factual knowledge via *Masked Language Modeling* (MLM) (Devlin et al., 2019). We use an entity-level MLM to allow LMs to memorize entities mentioned in factual statements. For example, given an input sequence of tokens $X = [x_1, x_2, ..., x_i, x_{i+1}, ..., x_n]$ and a two-token entity $e = [x_i, x_{i+1}]$. We convert the whole tokens of the entity to *one* mask token. In this case, the masked sequence of tokens $X' = [x_1, x_2, ..., x_{i-1}, [MASK], x_{i+2}, ..., x_n]$. Since we use symbolic representation, the *masked entity* is in the vocabulary of the LM.

3.2 Employed Models

RoBERTa(12L) We prepare a RoBERTa model with 12 layers as the temporal knowledge base. The RoBERTa(12L) is initialized from RoBERTa-base (Liu et al., 2019).

RoBERTa(6L) We prepare a 6-layer RoBERTa model, initialized from DistilRoBERTa-base (Sanh et al., 2019), to investigate how knowledge base capability scales with model size[5].

RoBERTa-randinit(12L) Heinzerling and Inui (2021) shows that LMs without pre-training can memorize more factual statements than pretrained models. However, it only focuses on memorizing static and one-to-one relationships. In this work, we also prepare a 12-layer RoBERTa with randomly initialized parameters to further explore the effect of pre-training in a more practical condition.

4 Experiments

Storage and the use of stored knowledge are two basic functions of KBs. To explore the LM-as-KB paradigm in the temporal domain, we design several experiments to answer the two questions of LMs as temporal KBs (Section 1).

First, we conduct a reciting experiment to evaluate the **storage capacity** of LMs for storing temporal facts and explore the impact of conflicting information (Section 4.1). Next, we construct targeted queries to recall stored temporal knowledge in terms of temporal boundaries (Section 4.2) and implicit temporal knowledge (Section 4.3) to explore the ability of LMs to **use stored knowledge**.

4.1 Storage Capacity

Storage is the foundation of KB applications. Here, we conduct a *reciting* experiment to investigate **how much temporal knowledge LMs can memorize (the first question)**. Firstly, we train prepared RoBERTa models to memorize temporal knowledge in LAMA-TK. For each fact in LAMA-TK, we mask the subject, object, start time, and end time, and generate four masked statements. These masked statements then serve as the training data for LMs to memorize. For example, given the masked statement "Michael Jordan played for [MASK] from 1995 to 1998.", the model should predict the masked entity "Chicago Bulls". We call this process as **Feeding Temporal Knowledge into LMs**. Then, we test the models to evaluate how many factual statements in training data have been memorized and record the Acc@K and Hit@K (see Section 2.3). We call this process as **Reciting**.

[5] DistilRoBERTa-base is the distilled version of RoBERTa-base, with 6 layers. Details of the models are in Appendix D.

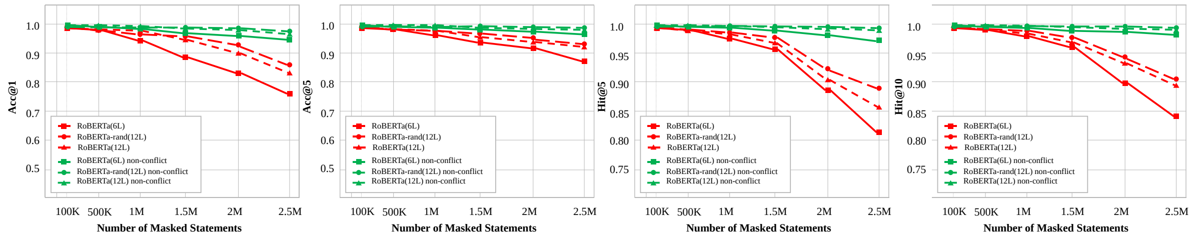

Figure 2: Overall results of statement memorization. We report Acc@1, Acc@5, Hit@5, and Hit@10 of each model. The green lines show the performances of models trained on LAMA-TK without conflicting information, while the red lines show the performances of models trained on LAMA-TK with conflicting information.

Table 2: Results of single-answer and multi-answer statement memorization of RoBERTa(12L) trained on 2.48M masked statements with conflicting information.

#Answer	Acc@1	Acc@5	Hit@5	Hit@10
single	0.8441	0.9550	0.9550	0.9623
multiple	0.7876	0.9358	0.4748	0.5527

Table 3: single-answer statement (1-1 relation) memorization results of RoBERTa(12L) trained on 2.48M statements with and without conflicting information.

Train Data	1-1 relations			
	Acc@1	Acc@5	Hit@5	Hit@10
non-conflict	0.9700	0.9910	0.9910	0.9930
conflict	0.8441	0.9550	0.9550	0.9623

Different from previous works, we examine the capacity of LMs to memorize temporally-scoped facts which often change with time. For example, "Michael Jordan played for Birmingham Barons" is only true from 1994 to 1995. In 1995, Michael Jordan left Birmingham Barons and joined Chicago Bulls. Therefore, to correctly recall the sports team Michael Jordan played for, LMs should additionally take into account the temporal scopes of facts.

Moreover, we reserve 1-N, N-1, and N-M relations in LAMA-TK. During training, the model will see **conflicting information (the first challenge)**, such as the politician who held *multiple positions* at once and the scientist who received *multiple prizes* in a year. We call these statements as **multi-answer statements**. Previous works discarded facts with multiple objects and considered the LM-as-KB paradigm within 1-1 relations, which made this task lightweight, but less practical. However, storing conflicting information is a basic function that a KB should have.

Result The red lines in Fig 2 show the accuracies of statements memorization accuracies achieved with different RoBERTa models. The randomly initialized RoBERTa model has the highest recall accuracy for storing temporal knowledge, correctly answering 86 percent of 2.48 million masked statements; RoBERTa(6L) has the lowest recall accuracy, with 0.76 Acc@1. As the amount of training data increases, the storage accuracy of all the models gradually decreases. Compared to the RoBERTa(12L), RoBERTa(6L) has more difficulty storing millions of masked statements. This result indicates that LMs with more parameters show better storage capacity. Moreover, we show that the randomly initialized LM exhibits better storage capacity than the pretrained LM. This result is the same as previous work (Heinzerling and Inui, 2021). The knowledge stored during pretraining affects the memorization of new knowledge.

Table 2 shows the memorization results of statements with single and multiple answers. Results show that RoBERTa(12L) can memorize single-answer statements with high Acc@K and Hit@K. However, memorizing multi-answer statements is still challenging, which shows high Acc@K but low Hit@K. This result shows that LMs can only memorize one of the correct answers, but do not have similar confidence in other correct answers.

Influence of Conflicting Information To explore the influence of conflicting information on the storage capacity of LMs, we compare models trained on LAMA-TK with and without conflicting information. For the version of LAMA-TK without conflicting information (non-conflict), we remove all masked statements with multiple answers. Then, we examine RoBERTa(6L) and RoBERTa(12L) on LAMA-TK without conflicting information.

The green lines in Fig 2 show the statement memorization performances achieved without conflicting information. All models can memorize 2.48M statements with over 0.95 Acc@1, which is much better than memorizing statements with conflicting information. The accuracy drop indi-

Table 4: Performances of RoBERTa models with and without dynamic time masking on 200k time queries in zero-shot settings. The models above the midline use original masking, while the ones below the midline use dynamic time masking. The green numbers in brackets show the improvement dynamic time masking brings over RoBERTa(12L) with original masking. The highest and second-highest scores among all models are **boldfaced** and underlined respectively. Scores with asterisks are the highest among the models with original masking.

Model	**Acc@1**	**Acc@5**	**Hit@5**	**Hit@10**
RoBERTa(6L)	0.1890*	0.4510*	0.3849*	0.4944*
RoBERTa-rand(12L)	0.1280	0.3260	0.2614	0.3590
RoBERTa(12L)	0.1226	0.3240	0.2689	0.3596
RoBERTa(12L) dynamic mask 10%	0.3774(+0.2658)	0.7042(+0.3802)	0.6628(+0.3939)	0.7740(+0.4144)
RoBERTa(12L) dynamic mask 100%	**0.4879**(+0.3653)	**0.8367**(+0.5127)	**0.7611**(+0.4922)	**0.8838**(+0.5242)

cates that the storage capacity of LMs is greatly affected by conflicting information. The accuracy drop yielded by RoBERTa(6L) is greater than RoBERTa(12L), showing that models with fewer parameters are more susceptible to conflicting information.

Moreover, Table 3 shows the influence of conflicting information on memorizing other one-to-one relationships. The performance drops indicate that conflicting information hinders the memorizing process of other unrelated 1-1 relations.

4.2 Temporal Boundary Query

In the first experiment, we observe that it is possible for LMs to recite millions of temporal knowledge. We now turn to investigate **whether LMs can use stored knowledge for temporal knowledge queries or merely recite facts learned by rote (the second question)**. Firstly, we test whether LMs can differentiate between stored timestamps. For example, if an LM has memorized "Barack Obama held the position of president of United States from 2009 to 2017", the model should recall the start time "2009" with the query "Barack Obama was elected president of the United States in [MASK]" or recall the end time "2017" with the query "Barack Obama resigned from president of the United States in [MASK]".

To ensure that the LMs can memorize all required knowledge, we first sample 100k fact statements with the predicate "position held" from LAMA-TK and mask their start and end times. This results in 200k masked factual statements. Then we train the RoBERTa models to fully memorize all these statements, with 0.99Acc@1.

Next, we write cloze-style templates to query the start and end times mentioned in stored facts, such as "S was elected O in [MASK]" and "S resigned from O in [MASK]". We use these queries to test the capability of the model to understand the difference between temporal scopes. We conduct this experiment in a zero-shot setting; i.e., the target query templates are not observed during training. The zero-shot setting can better show the capability of LMs to understand natural language queries and transfer knowledge to targeted queries.

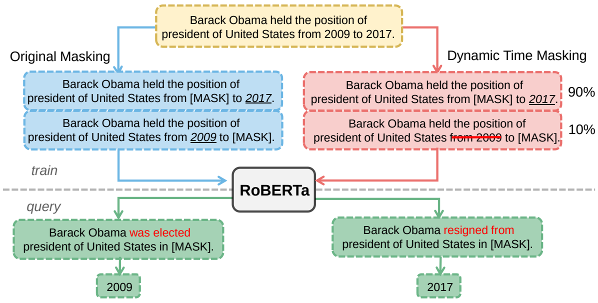
Barack Obama held the position of president of United States from 2009 to 2017.
Original Masking
Dynamic Time Masking
Barack Obama held the position of president of United States from [MASK] to 2017.
Barack Obama held the position of president of United States from 2009 to [MASK].
Barack Obama held the position of president of United States from [MASK] to 2017.
90%
Barack Obama held the position of president of United States ~~from 2009~~ to [MASK].
10%
train
RoBERTa
query
Barack Obama was elected president of United States in [MASK].
Barack Obama resigned from president of United States in [MASK].
2009
2017

Figure 3: Examples of two types of masking and the process used by LMs for temporal boundary queries. The remaining timestamps are underlined. The predicates written in red are new query templates.

Result The results are shown in the first three rows of Table 4. In the case where the model has fully memorized all required temporal knowledge, the model with fewer parameters performs better. The performance of RoBERTa(12L) is similar to that of RoBERTa-randinit(12L), but both are lower than that of RoBERTa(6L).

Dynamic Time Masking Through the above experiment, we find that the model's capability to query temporal boundaries is not satisfactory (low Acc@1). We speculate that this result may be due to the strong **correlation between temporal scopes (the second challenge)**. Original masking makes the model relies too much on the remaining timestamp and makes it difficult to query the masked timestamp separately. For example, we use "Barack Obama held the position of president of United State from [MASK] to 2017" to train the LMs, which makes the prediction for the masked timestamp "2009" excessively rely on the

Table 5: Results on 20k queries with original query templates and new query templates (original query templates: "S held the position of O in T", new query templates: "S served as O in T"). We report Acc@1/Acc@5 and Hit@5/Hit@10 of each model on two template types.

Model	Parameters	Acc@1 / Acc@5		Hit@5 / Hit@10	
		Template Type		Template Type	
		Original	New	Original	New
RoBERTa(6L)	82M	0.4114 / 0.6521	0.2242 / 0.4115	0.6192 / 0.6993	0.3798 / 0.4540
RoBERTa-rand(12L)	125M	**0.4147 / 0.6868**	0.0131 / 0.0562	**0.6457 / 0.7215**	0.0757 / 0.0518
RoBERTa(12L)	125M	0.3440 / 0.5666	**0.3113 / 0.5020**	0.5281 / 0.6028	**0.4698 / 0.5480**

remaining timestamp "2017". This makes it difficult for LMs to transfer stored timestamps to temporal boundary queries and results in answering these queries with low accuracy.

Inspired by the dynamic masking of RoBERTa (Liu et al., 2019), we design a dynamic time masking method to verify this conjecture. As shown in Figure 3, while constructing masked factual statements, we only mask the specific timestamp *1-k%* of time, and for the other *k%* of time we mask the specific timestamp and delete the other time information. To avoid using the same time mask in every epoch, we duplicate the training data 10 times so that each statement is masked in 10 different ways over 50 epochs of training. Therefore, the model will see 10 variations of each statement.

Dynamic time masking reduces the strong correlation between temporal scopes by adding perturbation to the other temporal information during training. In this experiment, we evaluate RoBERTa(12L) with 10% and 100% dynamic time masking. Table 4 shows the performance of these models. By adding 10% perturbation, the accuracy of RoBERTa(12L) significantly increases to 0.3774 Acc@1, 0.7042 Acc@5. The Hit@K of RoBERTa(12L) also increases significantly. Moreover, we evaluate RoBERTa with 100% dynamic time masking which completely ignores the correlation between the start and end times. RoBERTa with 100% dynamic time masking performs the best (both Acc@K and Hit@K). However, 100% dynamic time masking causes the model to be unable to associate the start time and the end time and to handle facts such as a politician who held one position *several times*. These results show that dynamic time masking efficiently reduces the strong correlation between temporal scopes and helps LMs recall the stored timestamps.

4.3 Implicit Temporal Knowledge Query

Prompt-based temporal scope modeling allows LMs to memorize temporal facts with their associated temporal scopes. (e.g., "Michael Jordan played for Chicago Bulls *from* 1995 *to* 1998"). Compared with jointly modeling text and a single timestamp (e.g., "Michael Jordan played for Chicago Bulls in 1995/1996/1997/1998."), prompt-based temporal scope modeling introduces fewer factual statements and less conflicting information, but more **implicit temporal knowledge (the third challenge)**. However, **can LMs use stored knowledge for implicit temporal knowledge queries (the second question)?** For example, if an LM has memorized "François Hollande held the position of president of the French Republic from 2012 to 2017", can the LM understand that François Hollande was the president of the French Republic for each year between 2012 and 2017? Moreover, can LMs answer the query "François Hollande *served as* [MASK] in 2015" even if the template "S served as O in T" is not seen during training?

A controlled experiment is designed for these questions. We choose one predicate "position held" and sample all statements generated by the template "S held the position of O from ST to ET". Inspired by Heinzerling and Inui (2021), we add distractors to recognize whether LMs answer these queries by using stored knowledge or simple by generic association. For a fact <S, P, O, ST, ET>, we add its distractor <S, P, O', ST', ET'> which involves the same subject S and predicate P, but a different Object O'. Moreover, we add its distractor <S, P', O', ST", ET"> which involves the same subject S but a different predicate P' and object O'. For example, distractors for <Barack Obama, Position Held, President of United States, 2009, 2017> are <Barack Obama, Position Held, United States senator, 2007, 2008> and <Barack Obama, award received, Nobel Peace Prize, 2009, 2009>. To correctly answer the query "Barack Obama held the position of [MASK] in 2012.", the model needs to consider both the predicate and the temporal scopes since there are three distinct entities associated with "Barack Obama". Every fact has at least one distractor. This results in 20k statements.

Next, we train the RoBERTa models to memorize all these fact statements and construct elaborate queries. For each fact, we randomly select one year between the start and end years as the timestamp of the associated query. We do not consider the start and end years because these boundary timestamps can bring prompts to the query. Then we use two types of templates to generate queries. First, we use the *Original Template* "S held the position of O in T" which is also used to generate fact statements for training. Then, we use a *New Template* "S served as O in T". This template has similar semantics to the original template, but it is not seen during training. We use the *New Template* to evaluate the robustness of LMs to distinct templates.

Result Results are shown in Table 5. For *Original Template*, RoBERTa-Randinit(12L) has the highest Acc@K and Hit@K. Compared with RoBERTa(12L), RoBERTa(6L) with fewer parameters performs slightly better. This result is similar to that of the previous experiment, which shows that LMs with fewer parameters have a better capability to use *stored* temporal knowledge.

However, the performance achieved on the *New Template* shows a distinct result. In the case where the query template is not observed during training, the performance of RoBERTa-randinit(12L) significantly declines, with only 0.0131Acc@1 and 0.0518Hit@10. Conversely, the performance of pretrained RoBERTa(12L) drops slightly and remains at a high level. This result shows that pretrained LMs contain natural language knowledge and have strong robustness to new templates. Compared to RoBERTa(12L), RoBERTa(6L) has lower performance with a more severe drop, showing that LM with more parameters is less affected by unseen templates and shows stronger robustness.

5 Related Work

Recent research has shown that *pretrained language models* (PLMs) such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), GPT (Radford et al., 2018), and T5 (Raffel et al., 2020) can learn extensive world knowledge during pretraining and store these relational facts in their parameters. Petroni et al. (2019) constructs LAMA, a set of cloze-style queries (e.g., "Marcello Abbado was born in [MASK]. –> Milan"), to recall the factual knowledge contained in pretrained LMs such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). Their results show that a PLM contains relational knowledge and can recall stored facts without fine-tuning. Talmor et al. (2020) proposes eight cloze-style reasoning tasks to test different types of knowledge in BERT and RoBERTa. Heinzerling and Inui (2021) conduct experiments on RoBERTa to evaluate its ability to store millions of facts involving millions of entities and its ability to query stored facts. Its results provide a proof-of-concept for LM-as-KB. While these works focus on probing LM in the general domain, Sung et al. (2021) constructs the BioLAMA, a biomedical factual knowledge dataset for probing biomedical LMs, and further explores the capability of LM to act as a specific-domain KB. Moreover, Wang et al. (2019a); Zhou et al. (2020) examine PLMs on commonsense reasoning tasks, showing that PLM contains commonsense knowledge. To improve the performance on knowledge intensive tasks, Wang et al. (2019b) uses a Transformer encoder to obtain contextualized entity and relation embeddings. Yao et al. (2019) treats relational knowledge as textual sequences and finetunes BERT to model these knowledge. To improve the performance of recalling knowledge, Petroni et al. (2020) augments PLM with retrieved relevant context and improved the performance of cloze-style QA. Jiang et al. (2020) proposes mining-based and paraphrasing-based methods to generate high-quality prompts, which significantly improves the performance achieved on LAMA.

Within the current paradigm of the use of masked LMs as KBs, research has focused more on using generative LMs as KBs. As generative LMs can generate text sequences of any length, they are not limited by the length of the given knowledge. Roberts et al. (2020) fine-tunes the pretrained T5 model to three QA datasets WebQuestions (Berant et al., 2013), TriviaQA (Joshi et al., 2017), and NaturalQuestions (Kwiatkowski et al., 2019) without any access to external knowledge to test how much knowledge contained in the LM. The results show that fine-tuned T5 performs competitively with retrieval-based systems, and indicate that large pretrained LMs contain vast world knowledge. Lewis et al. (2021) argues that LMs can complete the closed-book QA tasks well, mostly due to high test-train overlaps. Wang et al. (2021) designs a knowledge memory task and a question-answering task on datasets with low test-train overlaps to evaluate the capability of BART (Lewis et al., 2020) to serve as a KB for closed-book QA. The results show that closed-book QA is still challenging for BART, both in terms of memorizing the knowledge and answering the questions. Dhingra et al. (2022) proposes a time-aware T5 model which jointly models the text with its timestamp, and constructs a new dataset called TEMPLAMA for probing LMs for temporal facts. Apart from closed-book QA, Dai et al. (2022) examines cloze tasks for BERT to identify the neurons that store specific facts. The results demonstrate the provenance of specific knowledge in the parameters of an LM. Zhu et al. (2020); Cao et al. (2021) focus on editing stored knowledge without affecting other facts. These works further explore the ability of LMs and expand their functions as KBs.

6 Conclusion

Temporal knowledge widely exists in real-world KBs. In this work, we extend the LM-as-KB paradigm to the temporal domain and argue that pretrained LMs have fairly good capability to serve as temporal knowledge bases in terms of their capacity to store temporal knowledge and their ability to use stored temporal knowledge. However, our analysis also shows that conflicting information poses great challenges to the LM-as-KB paradigm, such as the drop in storage accuracy and the difficulty in memorizing multiple answers.

Limitations

Our proposed dataset (LAMA-TK) takes the temporal scopes of temporal facts and *N-M* relations into account. However, LAMA-TK does not contain questions that require complex temporal reasoning, such as "*First-Last*: [MASK] was the first president of the United States." and "*Before-After*: [MASK] was the president of United States after Barack Obama.". (Saxena et al., 2021) evaluated BERT, RoBERTa, KnowBERT, and T5 on CronQuestions, which contains 232k such complex questions, but the results showed that these large pretrained language models perform very poorly (lower than 0.01 Hit@1 values).

In this work, we propose utilizing the masked LM RoBERTa as a temporal KB. Compared to T5-cbqa (Roberts et al., 2020) (737 million parameters), RoBERTa with 12 layers only has 120 million parameters. This makes our experiments lightweight. Moreover, we train RoBERTa to memorize temporal facts via MLM (Devlin et al., 2019). It is possible that incorporating factual knowledge into PLMs (Sun et al., 2019, 2020) or augmented LMs with memory banks (Févry et al., 2020; Verga et al., 2020) would allow these LMs to memorize factual knowledge more efficiently.

Finally, to explore the capability of an LM to memorize conflicting information (*1-N*, *N-1*, *N-M* relations), we additionally use Hit@K as the evaluation metric to evaluate how many correct answers are contained in the top k predictions. However, we do not consider how to distinguish correct answers from the predictions of LMs and how many correct answers should be recalled for a query.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China under Grants No.62072203 and Australian Research Council Under Grants DP22010371, LE220100078.

References

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA*, pages 1533–1544. ACL.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6491–6506. Association for Computational Linguistics.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 8493–8502. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN,* *USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

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.

Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. 2020. Entities as experts: Sparse memory access with entity supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4937–4951. Association for Computational Linguistics.

Zhijiang Guo, Michael Sejr Schlichtkrull, and Andreas Vlachos. 2021. A survey on automated fact-checking. *CoRR*, abs/2108.11896.

Benjamin Heinzerling and Kentaro Inui. 2021. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 1772–1791. Association for Computational Linguistics.

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

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1601–1611. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Patrick S. H. Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 1000–1008. 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. *CoRR*, abs/1907.11692.

Zihan Liu, Mostofa Patwary, Ryan Prenger, Shrimai Prabhumoye, Wei Ping, Mohammad Shoeybi, and Bryan Catanzaro. 2022. Multi-stage prompting for knowledgeable dialogue generation. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1317–1337. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237. 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. In *Procedings of the Conference on Automated Knowledge Base Construction, AKBC 2020, Virtual, June 22-24, 2020*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2463–2473. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

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. *Journal of Machine Learning Research*, 21:140:1–140:67.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5418–5426. Association for Computational Linguistics.

Guy D. Rosin, Ido Guy, and Kira Radinsky. 2022. Time masking for temporal language models. In *Proceddings of the Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, pages 833–841. ACM.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Apoorv Saxena, Soumen Chakrabarti, and Partha P. Talukdar. 2021. Question answering over temporal knowledge graphs. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6663–6676. Association for Computational Linguistics.

Tianxiang Sun, Yunfan Shao, Xipeng Qiu, Qipeng Guo, Yaru Hu, Xuanjing Huang, and Zheng Zhang. 2020. Colake: Contextualized language and knowledge embedding. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 3660–3670. International Committee on Computational Linguistics.

Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. ERNIE: enhanced representation through knowledge integration. *CoRR*, abs/1904.09223.

Mujeen Sung, Jinhyuk Lee, Sean S. 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, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 4723–4734. 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*, 8:743–758.

Pat Verga, Haitian Sun, Livio Baldini Soares, and William W. Cohen. 2020. Facts as experts: Adaptable and interpretable neural memory over symbolic knowledge. *CoRR*, abs/2007.00849.

Cunxiang Wang, Shuailong Liang, Yue Zhang, Xiaonan Li, and Tian Gao. 2019a. Does it make sense? and why? A pilot study for sense making and explanation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4020–4026. Association for Computational Linguistics.

Cunxiang Wang, Pai Liu, and Yue Zhang. 2021. Can generative pre-trained language models serve as knowledge bases for closed-book qa? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3241–3251. Association for Computational Linguistics.

Quan Wang, Pingping Huang, Haifeng Wang, Songtai Dai, Wenbin Jiang, Jing Liu, Yajuan Lyu, Yong Zhu, and Hua Wu. 2019b. Coke: Contextualized knowledge graph embedding. *CoRR*, abs/1911.02168.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. KG-BERT: BERT for knowledge graph completion. *CoRR*, abs/1909.03193.

Xuhui Zhou, Yue Zhang, Leyang Cui, and Dandan Huang. 2020. Evaluating commonsense in pre-trained language models. In *Proceedings of the AAAI Conference of the Artificial Intelligence*, volume 34, pages 9733–9740.

Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix X. Yu, and Sanjiv Kumar. 2020. Modifying memories in transformer models. *CoRR*, abs/2012.00363.

A Details of LAMA-TK

Knowledge Sources The knowledge sources of LAMA-TK are from CronQuestions(Saxena et al., 2021) and Wikidata. We extracted all facts with both a start date and an end date. Following (Dhingra et al., 2022), we reserved the top 7 most frequent temporally rich relations from our collected temporal knowledge, namely

- *position held*
- *member of sports team*
- *employer*
- *award received*
- *country of citizenship*
- *spouse*

This results in 639k temporally-scoped facts, with 316k entities. Statistics of entities in LAMA-TK have been shown in Table 6.

Fact to Text For each fact, we write a template to convert it to natural language statements. For example, for the fact <Giannis Antetokounmpo, Member of Sports Team, Filathlitikos B.C., 2011, 2013>, we use the template "[X] held the position of [Y]" to convert the fact triple <Giannis Antetokounmpo, Member of Sports Team, Filathlitikos B.C.> to text "Giannis Antetokounmpo played for Filathlitikos B.C.". Templates are shown in Table 8.

For most relations, we use the *prompt template* "*from* ST *to* ET" to convert temporal scopes to natural language texts. However, "award received" is an exception. It is not a durative relation, the start time of the facts is always equal to the end time. Therefore, we use a new *prompt template* "*in* T" to convert these temporal scopes to texts.

Finally, we concatenate the factual text and temporal text, and we get the factual statements.

Constructing Queries Table 9 shows the masking process. For each factual statement, we mask the subject, object, start time, and end time, resulting in four masked statements. We reserve the *masked entity* and collect all correct answers to the masked statements as the *answer list*. We train the LMs to predict the *masked entity*, and use both the *masked entity* and the *answer list* for evaluation purposes. If the masked entity is in the top k answers of the model, Hit@K is 1. If any of the top k answers of the model is in the answer list, Acc@K is 1.

Note that previous works tended to mask the object of each factual statement only because masking the subject would introduce multi-answer statements. For example, "[MASK] played for Chicago Bulls from 1995 to 1998" has more correct answers than "Michael Jordan played for [MASK] from 1995 to 1998".

Table 6: Number of entities in LAMA-TK across different types. Please refer to Appendix A.

Person	**Position**	**Sport Team**	**Company**
248463	42006	10953	6768
Prize	**Institution**	**Country**	**Time**
3399	3696	152	1539

Table 7: Examples of Time-aware T5, TempoBERT and our proposed prompt-based temporal scope modeling. Our proposed pompt-based temporal scope modeling jointly model text and temporal scopes, which is more suitable for handling temporally-scoped knowledge.

Time-aware T5 (Dhingra et al., 2022)
year 1995: Michael Jordan plays for Chicago Bulls.
TempoBERT (Rosin et al., 2022)
<1995> Michael Jordan plays for Chicago Bulls.
Prompt-based Temporal Scope Modeling(ours)
Michael Jordan played for Chicago Bulls *from 1995 to 1998.*

B Further Analysis on Prompt-based Temporal Scope Modeling

Some works have focused on jointly modeling time and text. Time-aware T5(Dhingra et al., 2022) adds a time prefix to each text to jointly model time and text (e.g., "year:2016 Eden Hazard plays for Chelsea F.C"). TempoBERT(Rosin et al., 2022) adds a time token to the top of the input sequence and designs time masking to encode time into the models (e.g., "<2022> Joe Biden serves as the president of the United States"). Examples have been shown in Table 7.

These approaches focus on modeling text with a single timestamp. However, the temporal knowledge stored in knowledge bases usually contains temporal scopes (start and end times). Although we can split temporal scopes into years and jointly model the years and texts, this splitting process will lead to a massive increase in factual statements that the model needs to memorize, and introduce a large amount of conflicting information (e.g., "Michael Jordan played for Chicago Bulls from 1995/1996/1997/1998"). Section 4.1 has shown that conflicting information can lead to a decrease in the storage capacity of an LM. There-

Table 8: Templates used for converting temporally-scoped knowledge to natural language statements.

Wikidata ID	Relation Name	# Temporal Knowledge	Template
P54	member of sport team	276633	[X] played for [Y] from [T] to [T]
P39	position held	227487	[X] held the position of [Y] from [T] to [T]
P108	employer	25154	[X] worked for [Y] from [T] to [T]
P166	award received	75027	[X] received [Y] in [T]
P69	educated at	17842	[X] studied at [Y] from [T] to [T]
P26	spouse	14645	[X] and [Y] were spouses from [T] to [T]
P27	country of citizenship	2145	[X] was a citizen of [Y] from [T] to [T]

Table 9: Example queries for different relations from LAMA-TK. Different from previous work, we mask not only the object, but also the subject and timestamps. Moreover, we reserve all correct answers for each query. [X], [Y], [T] refers to the masked subject, object, timestamp respectively. The underlined entities are unmasked entities.

Relation Name	Example Query	Masked Entity	Answer List
educated at	[X] studied at University of Freiburg from 1928 to 1929	Philip Showalter Hench	Philip Showalter Hench, Bernhard Neumann
position held	Murray Hill held the position of [Y] from 1968 to 1970	Minister for Transport	Minister for Transport, Minister of Roads
employer	Emiliano Aguirre worked for University of Granada from [T] to 1974	1971	1971
member of sport team	Michael Jordan played for Chicago Bulls from 1984 to [T]	1993	1993
award received	John Bardeen received Nobel Prize in Physics in [T]	1956	1956, 1972
spouse	[X] and Rita Gam were spouses from 1949 to 1955	Sidney Lumet	Sidney Lumet
country of citizenship	Pasquale Brignoli was a citizen of [Y] from 1861 to 1884	Kingdom of Italy	Kingdom of Italy

fore, we need to find a joint modeling method that can preserve the semantic information of temporal scopes and reduce the introduction of conflicting information.

To this end, we design a prompt-based temporal scope modeling method. We use *prompt templates* such as "*from* ST *to* ET" and "*in* T" to jointly model the temporal scopes and factual texts. These prepositions in the prompt templates augment the semantic information of timestamps. Section 4.2 shows RoBERTa with prompt-based temporal scope modeling method preserves the temporal boundary of factual knowledge, and Section 4.3 shows that with prompt-based temporal scope modeling method, RoBERTa can understand the continuity of temporal scopes without finetuning. These results provide a proof of concept that prompt-based template scope modeling can indeed model temporally-scoped knowledge well.

C Limitations of Top-K Accuracy for LM-as-KB Tasks

The top-K accuracy metric indicates whether the top k predictions contain correct answers. For example, for the query "John Bardeen received Nobel Prize in Physics in [MASK]", we assume that the model recalls one correct answer "1956" in the top 1 and recalls another answer "1972" in the top 100. Even if the model cannot effectively recall the correct answer "1972", the Acc@1 and Acc@5 to this query are still 1. Therefore, for LM-as-KB tasks, Acc@K can only indicate whether LMs can correctly answer a query but cannot indicate whether LMs have memorized all correct answers to the query.

In this paper, we use Hit at top k (Hit@K) to evaluate whether LMs have high confidence in all correct answers. For the above example query, the model recalls one correct answer "1956" at the top 1 so that Hit@10 for the query "John Bardeenn received Nobel Prize in Physics in [MASK]. –> *1956*" is 1. However, the model recalls another correct answer "1972" at the top 100 so that Hit@10 for the query "John Bardeen received Nobel Prize in Physics in [MASK]. –> *1972*" is 0. Hit@K provides a more comprehensive result for queries with multiple answers.

D Details of Models

RoBERTa(12L) RoBERTa(12L) has 12 layers, 768 dimensions, 12 heads, and 125M parameters. Its parameters are initialized from huggingface RoBERTa-base[6].

RoBERTa(6L) RoBERTa(6L) has 6 layers, 768 dimensions, 12 heads, and 89M parameters. However, Liu et al. (2019) only provides a 12-layer pretrained RoBERTa model (RoBERTa-base) and a 24-layer pretrained RoBERTa model (RoBERTa-large). Therefore, we initialize RoBERTa(6L)

[6] https://huggingface.co/roberta-base

with huggingface DistilRoBERTa-base[7], the distilled version of RoBERTa-base. Although RoBERTa(6L) is initialized from the distilled version of RoBERTa-base, we do not focus on factual knowledge acquired during pre-training. Following (Heinzerling and Inui, 2021), we further train LMs on LAMA-TK and only take into account temporal knowledge which is contained in training data.

RoBERTa-randinit(12L) RoBERTa-randinit(12L) is a randomly initialized 12-layer Transformers model, with the same architecture as RoBERTa(12L).

E Reasons for Not Masking Predicate

In LAMA-TK, we do not mask the predicate because, for most temporal facts, there is a close association between the predicate and the object. For example, given the object "Nobel Prize in Literature", the model will directly predict the masked relation to be "award received", since the prediction for these relations is hardly affected by entities other than the object.

[7] https://huggingface.co/distilroberta-base