On the Robustness of Reading Comprehension Models to Entity Renaming

Anonymous ACL submission

Abstract

We study the robustness of machine reading comprehension (MRC) models to entity renaming—do models make more wrong predictions when answer entities have different names? Such failures imply that models overly rely on entity information to answer questions, and thus may generalize poorly when facts about the world change or questions are asked about novel entities. To systematically audit this issue, we present a general and scalable pipeline to replace entity names with names from a variety of sources, ranging from common English names to names from other cultures to arbitrary strings. Across five datasets and three pretrained model architectures, MRC models consistently perform worse when entities are renamed, with particularly large accuracy drops on datasets constructed via distant supervision. We also find large differences between models: SpanBERT, which is pretrained with span-level masking, is more robust than RoBERTa, despite having similar accuracy on unperturbed test data. We further experiment with different masking strategies as the continual pretraining objective and find that entity-based masking can improve the robustness of MRC models.[1]

1 Introduction

The task of machine reading comprehension (MRC) measures machines' understanding and reasoning abilities. Recent research advances (Devlin et al., 2019; Yang et al., 2019; Khashabi et al., 2020) have driven MRC models to reach or even exceed human performance on several *MRC benchmark datasets*. However, their actual ability to solve the general *MRC task* is still questionable (Kaushik and Lipton, 2018; Sen and Saffari, 2020; Sugawara et al., 2020; Lai et al., 2021). While humans show robust generalization on reading comprehension, existing works have revealed that MRC models generalize poorly to out-of-domain data distributions (Fisch et al., 2019) and are brittle under test-time perturbations (Pruthi et al., 2019; Jia et al., 2019; Jia and Liang, 2017). All these issues could naturally happen to MRC systems deployed in the wild, hindering them to make reliable predictions on user inputs with great flexibility.

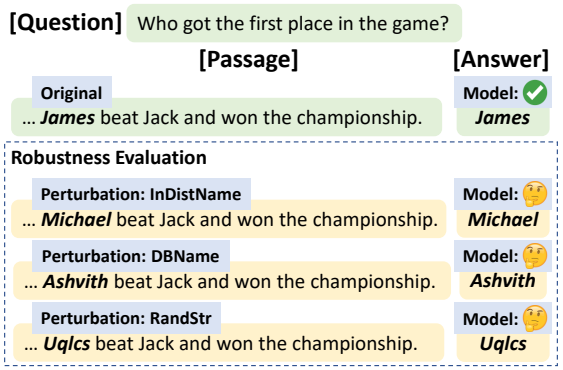

Figure 1: An illustrative example of the robustness to entity renaming and our proposed perturbations for robustness evaluation. "Michael" is from the answer of another test instance. "Ashvith" is a person name from an external database. "Uqlcs" is a random string with the same length as the original name.

In this work, we focus on an important but understudied type of test-time distribution shift caused by novel entity (e.g., people and companies) names. For a MRC model, besides the evidence provided by the surrounding context, it also has the capacity to leverage the entity information to make predictions. The information associated with the entity name covers both world knowledge that can change over time and dataset shortcuts that are unlikely to generalize. While contributing to performance on certain benchmarks, the over-reliance on specific entity names leads to an overestimation of model's actual ability to read and comprehend the provided passage (Peñas et al., 2011). It also hinders the model to generalize to novel entity names, which itself is challenging due to the large space of valid

[1] Data has been uploaded and will be published with code.

entity names induced by the flexibility of entity naming. For example, person names can be chosen from a large vocabulary depending on the culture, while companies can be named in an even more creative way, not to mention new names that are being invented every day. As illustrated in Figure 1, keeping the reasoning context unchanged, a robust MRC model is supposed to correctly locate the same span of a named entity as the answer, even after it gets renamed.

To audit model robustness, we use entity renaming as test time perturbation to mimic the situation where a deployed MRC model encounters questions asking for novel entity names in the emerging data. We design a general pipeline to generate natural perturbations of MRC instances by swapping the answer entity name with another valid name throughout the passage. We design perturbation rules and collect resources for three types of entities with large name space, including Person, Organization, and Geopolitical Entity.

With the proposed analysis framework, we conduct extensive experiments on five datasets and three pretrained language models. Data-wise, we find that distantly supervised MRC datasets lead to less robustness. Entity-wise, we find that Geopolitical Entity poses the greatest challenge than Person and Organization when renamed. Model-wise, we find that SpanBERT is more robust than BERT and RoBERTa, mainly due to less sensitivity to domain shift on names, which is likely to be the benefit of span-focused pre-training objectives. Inspired by this, we investigate several objectives via continual pretraining and find that an entity-based masking strategy can further enhance the robustness.

2 Analysis Setup

2.1 Extractive MRC

The task of MRC tests a machine' understanding and reasoning abilities by asking it to answer the question based on the provided passage. We focus on extractive MRC, where the answer is a span in the passage. Formally, given a **question** Q and a **passage** P of n tokens $P = \{x_1 \ldots, x_n\}$, an extractive MRC model is expected to predict an **answer** span $a = \{x_i, \ldots, x_{i+k}\} (1 \leq i \leq i+k \leq n)$ in the passage P as a response to the question Q. We use exact match (**EM**) as the metric for MRC evaluation, which is the percentage of test instances that the model exactly predicts one of the gold answers.

In both real-world scenarios and MRC datasets, a large portion of questions ask about entities like people, organizations and locations. While unmentioned background knowledge about the entities might be helpful for solving the questions, overly relying on it makes the model hard to adapt to updated facts provided by the passage and generalize to novel entities. Especially, we contrast MRC with closed-book QA, which requires a model to directly answer questions without access to any document passage. Closed-book QA tests a model's ability to pack knowledge into its parameters and retrieve knowledge from parameters to answer the question. On the contrary, we expect a MRC model to reason based on the provided passage.

2.2 Evaluation Protocol

We study the robustness of MRC models via test-time perturbation. Given an original test set D_{test} and a perturbation function f_{perturb} (detailed in §3) as inputs, we construct N perturbed test sets with N different random seeds. We evaluate the model on the N perturbed test sets. By averaging the results, we get the **average-case EM score** as the final metric, which measures the average impact on the model performance caused by the names from a certain perturbation. We set $N = 5$ in experiments.

2.3 Datasets

We choose five datasets with different characteristics from the MRQA 2019 shared task (Fisch et al., 2019): **SQuAD** (Rajpurkar et al., 2016), **Natural Questions (NQ)** (Kwiatkowski et al., 2019), **HotpotQA** (Yang et al., 2018), **SearchQA** (Dunn et al., 2017), and **TriviaQA** (Joshi et al., 2017) (statistics in Appendix §A). As a major difference in data collection, SQuAD, NQ, and HotpotQA employ crowdworkers to annotate the answer span in the passage, while SearchQA and TriviaQA use distant supervision to match the passage with the question. Distant supervision provides no guarantee that the passage contains enough evidence to derive the answer. The context where the entity span shows up may not even be related to the question.

2.4 MRC Models

We experiment with three pretrained language models that have demonstrated strong performance on popular MRC benchmarks. **BERT** (Devlin et al., 2019) is trained on English Wikipedia plus BookCorpus with masked language modeling (MLM) and next sentence prediction (NSP) as

self-supervised objectives. **RoBERTa** (Liu et al., 2019) improves over BERT mainly by dropping the NSP objective and increasing the pretraining time and the size of pretraining data. **SpanBERT** (Joshi et al., 2020) masks random contiguous spans to implement MLM and replaces NSP with a span-boundary objective (SBO). All pretrained language models in the main experiments are case-sensitive and in their `BASE` sizes. The finetuning details are shown in Appendix §B.

3 Entity Name Substitution

In this section, we introduce our method for perturbing a MRC test set with substitution entity names, i.e., the instantiation of f_{perturb}. Generating substitution names is at the core of our evaluation as different kinds of names measure a model's behavior in different situations with different robustness implications. We propose three categories of perturbations on three entity types and collect the corresponding name resources, aiming to audit a model's robustness from different perspectives.

3.1 Perturbation Pipeline

As illustrated in Figure 2, our perturbation pipeline consists of four steps, which are introduced below.

Step 1: Answer Entity Recognition. As we focus on the effect of answer entity renaming, we first identify entities in the answers by performing named entity recognition (NER) with spaCy (Honnibal et al., 2020) on the passage and extract the results on the answer spans. We identify three types of named entities: Person (**PER**), Organization (**ORG**), and Geopolitical Entity (**GPE**). All of them frequently appear as answers and have large space of valid names, making it important and challenging for models to robustly handle.

Step 2: Perturbable Span Identification. To facilitate name substitution, we assign metadata to detected entity names by identifying *perturbable spans* within the entity name. For each type of entity names, we define the applicable *span types* in Table 1. For PER, we only consider names with one or two words. A one-word name is considered as a first name, while a two-word name is considered as a full name, with the first word being the first name and the second word being the last name. We infer the gender of the detected name to be male, female, or neutral with `gender-guesser`.[2] For

Entity Type		Applicable Types of Perturbable Spans
PER (4)		<First Name-Male> (e.g., Richard, Morton) <First Name-Female> (e.g., Lauren, Jennifer) <First Name-Neutral> (e.g., Shine, Frankie) <Last Name> (e.g., Marx, Winfrey)
ORG (5)		<NNP> (e.g., Celtic, Tiffany) <Rare> (e.g., Hufflepuff, Pokemon)
	GPE (3)	<GPE-Country> (e.g., Iceland, Algeria) <GPE-State> (e.g., New Brunswick, Ohio) <GPE-City> (e.g., Boston, Sonsonate)

Table 1: Applicable metadata for each entity type in the perturbation pipeline.

GPE, we detect its contained country names, state names, and city names by string matching with the *Countries States Cities Database*[3]. For ORG, besides mentions of GPE names, we include two additional types of perturbable words identified using Penn Treebank (PTB) (Marcus et al., 1993). Words that are annotated as NNP(S) for more than 90% of the time in PTB are considered as proper nouns (dented as <NNP>), which are usually specialized for naming an entity. Words outside PTB are considered as rare words (denoted as <Rare>), which are likely to be invented by people to name an entity. These two kinds of words are weakly related to the characteristics of the entity and thus can be flexible. Note that given one or more entity types of interest, in this step we filter the test data to only keep a subset of instances with non-empty metadata for the corresponding entity types, which are instances that are ready to be perturbed. Sizes of the perturbable subsets for different entity types and their union (**MIX**) are shown in Appendix §A.

Step 3: Candidate Name Sampling. For each perturbable span, we gets its substitution name by querying an external dictionary with the span type. The substitution name is randomly sampled from a pool of names in the external dictionary that is labeled with the same span type. We collect multiple dictionaries with names of different characteristics serving for different analysis purpose, which are detailed in §3.2.

Step 4: Name Substitution. Once we have a candidate name for each perturbable span, we perform string mapping on the passage, question, and the gold answer, to finish the entity renaming in MRC instances. The name substitution changes all mentions of the answer entity in the passage while keeping the other reasoning context.

[2] https://pypi.org/project/gender-guesser/

[3] https://countrystatecity.in/

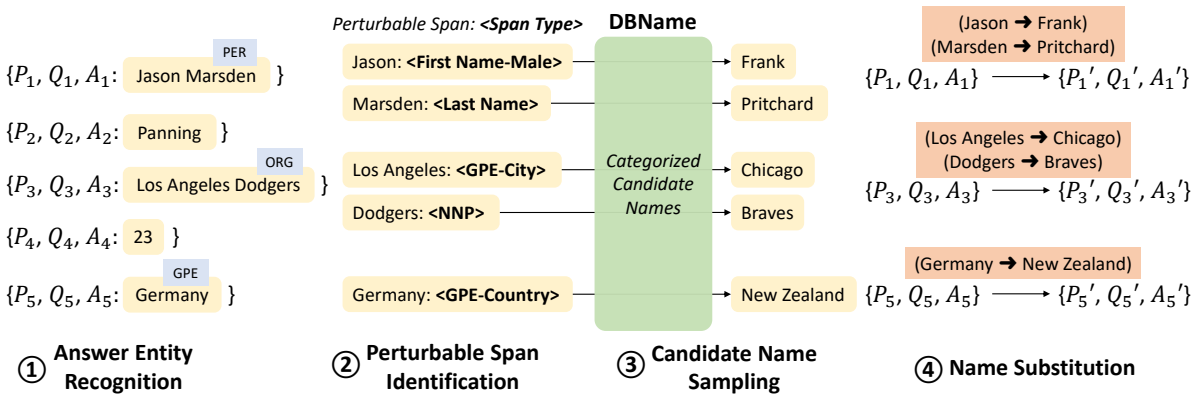

Figure 2: The perturbation pipeline for performing entity name substitution on MRC instances.

3.2 Candidate Name Collection

We consider three types of candidate names for perturbations in our main experiment to simulate the domain shift of entity names during test time.

In-Distribution Name (InDistName). The set of candidate names along with their span types is the same as the perturbable spans along with their types identified from the gold answers in the test set. This ensures that the new names follow the same distribution as the original names.

Database Name (DBName). We collect names in the real world by referring to relevant databases. For PER, we collect first names[4] (with gender frequency) and last names[5] from the official statistics of person names in the U.S.. (We experiment with names from other cultures in §4.2.) We regard a first name as a male/female name if its male/female frequency is two times larger than its frequency of the opposite gender. The remaining names are considered as neutral. Following the practice for identifying perturbable spans, we get the list of country/state/city names using *Countries States Cities Database* and the NNP list using PTB. Rare words constitute an open vocabulary so they will not be substituted under the DBName perturbation.

Random String (RandStr). The RandStr perturbation is different from the other two as it neglects the query span type when preparing the candidates. We generate a random alphabetical string of the same length and casing as the original perturbable span. Names from low-resource languages can look quite irregular to the pretrained language models. Random string as an extreme case provides an estimation of the performance in this scenario.

[4] https://www.ssa.gov/oact/babynames/limits.html

[5] https://www.census.gov/topics/population/genealogy/data/2010_surnames.html

Accuracy (%)	PER	ORG	GPE
Perturbable Span Identification	93.3	86.7	93.3
Name Substitution	86.7	96.7	96.7

Table 2: Validity of the two key steps in the perturbation pipeline on 30 randomly sampled TriviaQA instances for each entity type.

3.3 Perturbation Quality

The validity of the perturbed instances depends on the quality of the perturbation pipeline (§3.1). We manually check the accuracy of the perturbation steps on TriviaQA, which demonstrates the largest performance drop as we will show. Out of the four steps in the pipeline (Figure 2), we evaluate the accuracy of step 2 ("Perturbable Span Identification") and step 4 ("Name Substituion"). The accuracy of step 2 is evaluated based on whether the perturbable spans and their corresponding span types are all correct for an instance, which also implies the quality of step 1 ("Answer Entity Recognition") as different entity types have different applicable span types. The accuracy of step 4 is evaluated based on whether the string mapping function successfully locates all mentions of the perturbable spans in the passage to perform string mapping. The quality of step 3 can be inferred from the accuracy of step 2 for InDistName perturbation. For DBName, we assume the database is of acceptable quality in the sense that all names it provides belongs to the correct span type, which is guaranteed by the source of the data — PTB is annotated by human experts, U.S. names come from official statistics, and GPE names are actively maintained by its creator and the community for more than 3 years. RandStr is proposed to stimulate the extreme

BERT@MIX	SQuAD	NQ	HotpotQA	SearchQA	TriviaQA
Original	$81.2_{\pm 0.3}$	$64.4_{\pm 1.0}$	$60.0_{\pm 0.2}$	$69.5_{\pm 1.1}$	$73.4_{\pm 0.8}$
InDistName	$78.7_{\pm 0.6}$	$62.0_{\pm 1.2}$	$\mathbf{56.8}_{\pm 0.4}$	$\mathbf{53.6}_{\pm 1.3}$	$\mathbf{59.0}_{\pm 1.4}$
DBName	$\mathbf{78.8}_{\pm 0.9}$	$\mathbf{62.1}_{\pm 1.3}$	$54.9_{\pm 0.3}$	$50.2_{\pm 1.8}$	$50.4_{\pm 1.6}$
RandStr	$76.9_{\pm 1.0}$	$59.0_{\pm 1.7}$	$49.5_{\pm 0.8}$	$23.6_{\pm 1.2}$	$25.4_{\pm 1.4}$

Table 3: **Comparison of different datasets.** EM scores of BERT on the original and perturbed test sets of the MIX entity type.

BERT	SQuAD	NQ	HotpotQA	SearchQA	TriviaQA
PER-Original	$82.8_{\pm 0.4}$	$69.3_{\pm 0.9}$	$63.1_{\pm 0.1}$	$69.7_{\pm 1.4}$	$73.6_{\pm 0.7}$
PER-DBName	$81.7_{\pm 0.8}$	$68.0_{\pm 1.0}$	$60.8_{\pm 0.2}$	$54.6_{\pm 2.3}$	$54.6_{\pm 1.6}$
PER-Δ	1.1	1.3	2.3	15.1	19.0
ORG-Original	$79.7_{\pm 0.6}$	$52.1_{\pm 1.2}$	$58.0_{\pm 0.3}$	$66.7_{\pm 1.5}$	$73.8_{\pm 1.5}$
ORG-DBName	$77.5_{\pm 1.4}$	$50.8_{\pm 0.8}$	$55.0_{\pm 0.6}$	$54.2_{\pm 1.6}$	$57.0_{\pm 1.5}$
ORG-Δ	2.2	1.3	3.0	12.5	16.8
GPE-Original	$79.1_{\pm 1.0}$	$54.5_{\pm 1.7}$	$55.8_{\pm 0.6}$	$74.4_{\pm 0.8}$	$76.4_{\pm 0.6}$
GPE-DBName	$73.7_{\pm 1.1}$	$49.5_{\pm 2.7}$	$43.9_{\pm 1.3}$	$40.1_{\pm 0.8}$	$40.1_{\pm 1.3}$
GPE-Δ	**5.4**	**5.0**	**11.9**	**34.3**	**36.3**

Table 4: **Comparison of different entity types.** EM scores of BERT on the Original and DBName test sets.

MIX	BERT	RoBERTa	SpanBERT
Original	$69.5_{\pm 1.1}$/$73.4_{\pm 0.8}$	$\mathbf{74.1}_{\pm 0.2}$/$78.6_{\pm 0.4}$	$73.2_{\pm 0.7}$/$\mathbf{79.1}_{\pm 0.1}$
InDistName	$53.6_{\pm 1.3}$/$59.0_{\pm 1.4}$	$\mathbf{60.7}_{\pm 0.4}$/$67.8_{\pm 1.1}$	$60.3_{\pm 1.4}$/$\mathbf{68.3}_{\pm 0.8}$
DBName	$50.2_{\pm 1.8}$/$50.4_{\pm 1.6}$	$54.0_{\pm 1.0}$/$60.5_{\pm 0.9}$	$\mathbf{57.9}_{\pm 1.0}$/$\mathbf{63.1}_{\pm 0.8}$
RandStr	$23.6_{\pm 1.2}$/$25.4_{\pm 1.4}$	$21.0_{\pm 4.8}$/$35.6_{\pm 0.2}$	$\mathbf{41.5}_{\pm 3.2}$/$\mathbf{51.9}_{\pm 2.3}$

Table 5: **Comparison of different models.** EM scores on the original and perturbed test sets of the MIX entity type on SearchQA/TriviaQA.

case, and we therefore do not evaluate its quality. As shown in Table 2, our method gets acceptable accuracy on the three entity types, confirming the quality of the perturbation pipeline.

4 Results and Analysis

4.1 Main Results

The EM scores of the three models on the original and perturbed test sets are presented in Figure 3. We analyze the results from several angles by aggregating across certain dimensions.

Training on MRC datasets created with distant supervision leads to less robustness. In Table 3, we show the results of BERT on the original and perturbed test sets, while results of RoBERTa and SpanBERT show similar patterns. The perturbations on all 3 entity types are combined (shown as "MIX"). We find that models trained on SQuAD, NQ, and HotpotQA (with at most 6% performance drop under the DBName perturbation) are significantly more robust than models trained on SearchQA and TriviaQA (with about 20% performance drop under the DBName perturbation). While the first group of datasets are human-labeled, the later group of datasets are constructed using distant supervision. Such correlation indicates that training noise due to mismatched questions and passages harms model's robustness. We hypothesize the reason to be that, the passage in the human-annotated datasets usually provides enough evidence to derive the answer, so a model is able to learn the actual task of "reading comprehension" from the data. On the contrary, SearchQA and TriviaQA use web snippets as the source of passages. The labeling process of distant supervision assumes that "the presence of the answer string implies the document *does* answer the question" (Joshi et al., 2017), while the document may or may not contain all facts needed to support the answer. In this case, because the actual reading comprehension task is difficult to learn due to lack of evidence, the model could be prone to use entity-specific background knowledge (e.g. assuming that "Jack Higgins" is a British author regardless of the context) or learn dataset-specific shortcuts associated with certain names via memorization (e.g., choosing "Jack Higgins" whenever it's mentioned in the passage and the question asks for an author), which causes the robustness issue.

GPE renaming poses the greatest robustness challenge. The renaming of PER and ORG are similarly less challenging. In Table 4, we present the performance drop caused by the DBName perturbation for each entity type. GPE renaming shows the largest performance drop. The comparison of PER and ORG differs across datasets, but their corresponding performance drops are generally similar. The reason is likely to be that the model is only exposed to a small number of distinct GPE names during finetuning compared to PER and ORG. In the training set of TriviaQA, there are 40k ORG names and 54k PER names, but only 12k GPE names. The lack of seen names makes it hard to learn the generalization ability.

On distantly supervised datasets, SpanBERT is more robust than RoBERTa, which is more robust than BERT. In Table 5, we show the performances of the three models under perturbations of the MIX entity type on SearchQA and TriviaQA. While RoBERTa and SpanBERT show comparable performances on the original and InDistName test sets, SpanBERT's improvement over RoBERTa becomes larger as the substitution names become

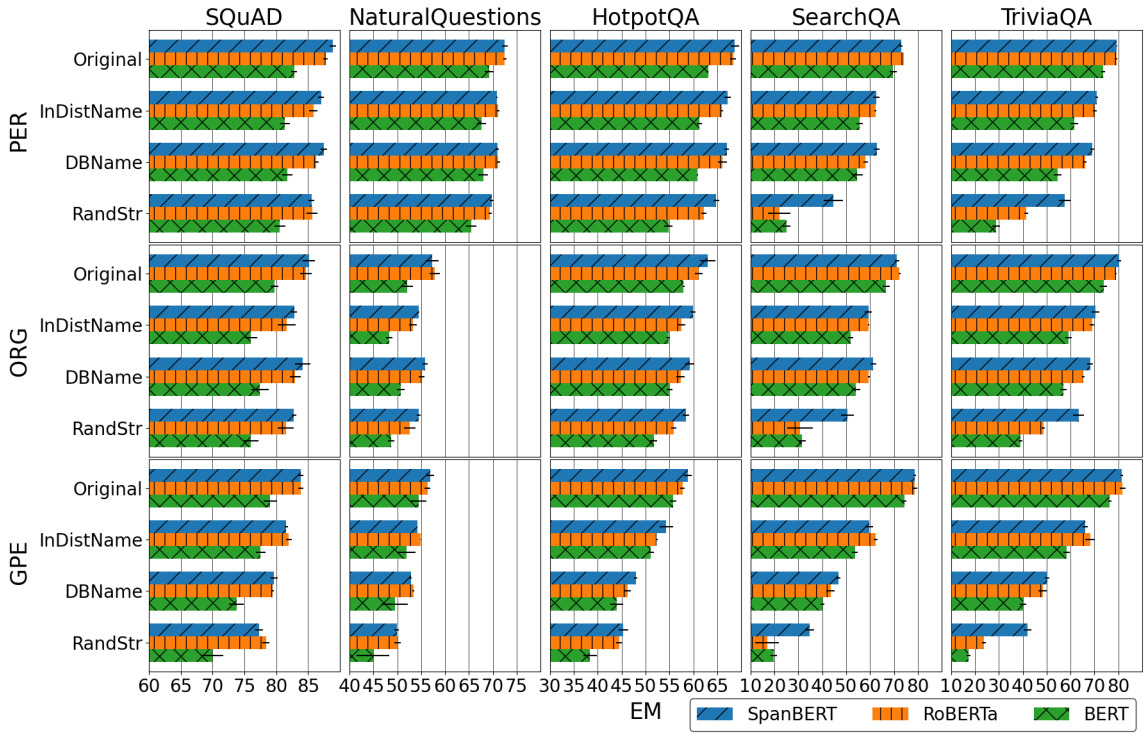

Figure 3: **Main results.** EM scores for MRC models evaluated on datasets under different perturbations.

more out-of-distribution. Meanwhile, BERT shows even larger performance decreases than RoBERTa. The models' performance differences are mainly attributed to their different pretraining strategies. RoBERTa's improvement over BERT indicates that a better pretraining configuration (as measured by the performance on the in-domain original test set) is also beneficial to the performance on the perturbed test sets, suggesting better generalization ability to the out-of-domain data. This correlation is consistent with the findings in Miller et al. (2021). SpanBERT's particular advantage on the perturbed test sets indicates its span-focused pretraining objective (span-based MLM and span prediction based on boundary tokens) is especially helpful for the span-related robustness, which is desired for the MRC task.

Both loss of entity knowledge and domain shift on names happen during renaming. SpanBERT's superior robustness over RoBERTa is mainly from handling domain shift. The information associated with the entity name that can be leveraged by the model includes both entity knowledge and name clues. **Entity knowledge** refers to the *world knowledge* about with *the referred entity*, like "Michelle Obama is the wife of Barack Obama," while **name clues** refer to *statistical clues* associated with the name's *surface form*, like "Barack Obama is likely to be a male name", "Barack Obama as an in-distribution name is likely to be the answer for this dataset". While all perturbations break the entity knowledge about the original entity, InDistName doesn't introduce domain shift on names and largely preserve the name clues. Going from InDistName to other perturbations, the substituted names can be more and more out of the dataset distribution. This performance drop can be attributed to the model's sensitivity to name-related domain shift. From SearchQA and TriviaQA results in Table 5, we find that RoBERTa and SpanBERT rely similarly on the entity knowledge (~13% performance drop from Original to InDistName on SearchQA and ~11% on TriviaQA). SpanBERT's advantage over RoBERTa is mainly on its good robustness to domain shift on names, shown by the perfromance drop from IndistName to other perturbations. BERT relies slightly more on entity knowledge but much more sensitive to domain shift on names.

4.2 Bias Exhibited by Person Names

Cultural Bias. As the DBName perturbation uses person names in the U.S., it cannot fully reflect the model's robustness behavior when encountering real-world names from diverse cultural backgrounds. Therefore, we additional collect names

Country/ Language	BERT	RoBERTa	SpanBERT
U.S.	$54.6_{\pm 2.3}$/$54.6_{\pm 1.6}$	$58.1_{\pm 0.9}$/$\mathbf{66.1}_{\pm 0.6}$	$63.0_{\pm 1.1}$/$\mathbf{69.1}_{\pm 0.7}$
French	$\mathbf{55.5}_{\pm 2.2}$/$\mathbf{56.1}_{\pm 1.7}$	$\mathbf{58.2}_{\pm 1.1}$/$66.0_{\pm 0.5}$	$63.0_{\pm 1.2}$/$68.8_{\pm 0.9}$
India	$53.5_{\pm 2.5}$/$51.9_{\pm 2.7}$	$56.5_{\pm 1.9}$/$63.9_{\pm 0.8}$	$\mathbf{63.0}_{\pm 1.1}$/$68.0_{\pm 0.4}$
Arabic	$53.3_{\pm 3.1}$/$48.8_{\pm 3.2}$	$56.3_{\pm 2.1}$/$61.8_{\pm 0.9}$	$62.8_{\pm 1.0}$/$66.2_{\pm 0.8}$
China	$46.2_{\pm 2.5}$/$44.8_{\pm 3.6}$	$54.0_{\pm 0.8}$/$63.0_{\pm 1.4}$	$59.3_{\pm 2.0}$/$65.2_{\pm 0.4}$
RandStr	$25.0_{\pm 1.6}$/$28.9_{\pm 1.6}$	$22.0_{\pm 4.7}$/$41.3_{\pm 0.8}$	$44.6_{\pm 4.0}$/$57.4_{\pm 2.4}$

Table 6: **Performance comparison of person names from different cultures.** EM scores on the original and perturbed test sets of the PER entity type on SearchQA/TriviaQA.

(sources listed in Appendix §D) from more countries (India, China) and languages (French, Arabic) to study the cultural bias in MRC models. We use the romanized form of names. Table 6 shows the performance comparison of models when evaluated with the person names from different cultures on SearchQA and TriviaQA. Names form the U.S. and French-speaking countries generally achieve the highest EM scores. Names from China get the lowest performance for the most of time, with significant EM drop (8.4% on SearchQA and 9.8% on TriviaQA for BERT) from U.S. names. The performance gap between different cultures becomes smaller with more robust models.

Other Factors. We also consider other factors of a name that could be related to biased model performance. We limit our scope to the U.S. first names and sample 1500 names from the database. We consider two features for each name. *Gender polarity* is defined as $\texttt{max}(\frac{f_m}{f_f}, \frac{f_f}{f_m})$, where f_m, f_f are the male frequency and female frequency of a name provided by the database. It measures the gender ambiguity of the name. *Popularity* is defined as $f_m + f_f$. We calculate the EM score for a name by evaluating on a test set where all answer first names get replaced with this name. For what we have tried, we didn't find evidence to support a correlation between each factor and the EM score. For example, with SpanBERT on TriviaQA, names with top 20% gender polarity gets 72.7% EM; while the bottom 10% names gets 72.8% EM. The numbers are 73.0% vs 72.7% for popularity. We leave exploring factors that correlate with the difficulty of a name as future work.

4.3 Improving Robustness with Continual Pretraining

SpanBERT's advantage over BERT suggests that some variants of MLM could be helpful for model robustness. To further robustify SpanBERT, we adopt a training paradigm with an inserted continual pretraining stage and compare MLM with different masking strategies as the objectives.

Training Paradigm Existing works mainly seek to robustify the model during finetuning with strategies like data augmentation (Ribeiro et al., 2019; Min et al., 2020), but they usually increase finetuning time and requires additional data. Some recent works (Gururangan et al., 2020; Ye et al., 2021) have explored improving a pretrained language model with "continual pretraining" — continuing to train a pretrained model for more steps with the some objective. This can generate a checkpoint that can be used as for finetuning on any dataset in the *standard* way with no additional cost.

Experimental Setup The masking policy in MLM plays an important role in instructing model learning, which can be potentially used to robustify the model. Inspired by previous works, we experiment with four heuristic masking policies as baselines to implement the MLM objective: **MLM (vanilla)**, **MLM (whole word)**, **MLM (span)**, and **MLM (entity)**. They perform masking at token, whole-word, span, and entity level repestively. Starting from SpanBERT (`-BASE`), we run continual pretraining with the above objectives for 8,000 steps. More details are described in Appendix §C.

Results The results for models finetuned from SpanBERT and different continually pretrained models are shown in Table 7. On SQuAD, all masking policies slightly downgrade the performance. With no much room for robustness improvement, running continual pretraining is probably at the cost of slightly sacrificing the performance due to the inconsistent objective and discontinuous learning rate that are applied when starting the continual pretraining. On SearchQA and TriviaQA, out of the four masking policies, the entity-based masking policy shows clear improvement over SpanBERT. As analyzed in §4.1, name-related domain shift is a major challenge for the model to handle. By predicting the masked entity, the model is exposed to the diverse entities in the pertraining corpus in a more explicit way, and gain a better sense of the entity boundaries. All these are helpful for the model to robustly handle novel entities.

Model / Perturbation (MIX)	SQuAD			SearchQA			TriviaQA		
	Original	DBName	RandStr	Original	DBName	RandStr	Original	DBName	RandStr
SpanBERT	**86.8**$_{\pm 0.5}$	**84.9**$_{\pm 0.4}$	**83.0**$_{\pm 0.1}$	73.2$_{\pm 0.7}$	57.9$_{\pm 1.0}$	41.5$_{\pm 3.2}$	79.1$_{\pm 0.1}$	63.1$_{\pm 0.8}$	51.9$_{\pm 2.3}$
SpanBERT w/ continual pretraining									
+ MLM (vanilla)	85.6$_{\pm 0.5}$	83.9$_{\pm 0.2}$	81.8$_{\pm 0.2}$	72.3$_{\pm 0.8}$	57.0$_{\pm 0.3}$	34.8$_{\pm 2.9}$	78.9$_{\pm 0.8}$	64.1$_{\pm 0.2}$	48.1$_{\pm 2.9}$
+ MLM (whole word)	86.0$_{\pm 0.7}$	84.5$_{\pm 0.3}$	82.7$_{\pm 0.4}$	72.9$_{\pm 0.4}$	58.0$_{\pm 0.2}$	41.6$_{\pm 3.3}$	79.1$_{\pm 0.5}$	64.2$_{\pm 0.4}$	50.1$_{\pm 0.9}$
+ MLM (span)	85.7$_{\pm 0.2}$	84.1$_{\pm 0.1}$	82.6$_{\pm 0.1}$	73.3$_{\pm 0.5}$	57.9$_{\pm 0.8}$	39.5$_{\pm 2.4}$	79.4$_{\pm 0.8}$	64.3$_{\pm 0.4}$	54.1$_{\pm 1.5}$
+ MLM (entity)	86.0$_{\pm 0.4}$	84.3$_{\pm 0.3}$	82.7$_{\pm 0.1}$	**73.4**$_{\pm 0.4}$	**59.3**$_{\pm 1.1}$	**48.1**$_{\pm 4.6}$	**79.6**$_{\pm 0.6}$	**65.9**$_{\pm 1.1}$	**55.5**$_{\pm 2.7}$

Table 7: EM scores on the original and perturbed test sets of TriviaQA for different continually pretrained models.

5 Related Work

Robustness of MRC Models. The robustness of MRC models are usually evaluated against test-time perturbations and out-of-domain data. Research on test-time perturbation proposes perturbation methods at different levels as attacks (Si et al., 2021), such as word replacement with neighbors in the vector space (Rychalska et al., 2018; Jia et al., 2019), question paraphrasing (Gan and Ng, 2019; Ribeiro et al., 2018), sentence distractor injection (Jia and Liang, 2017; Zhou et al., 2020). Another line of research (Fisch et al., 2019; Sen and Saffari, 2020) tests a model on data with out-of-domain passage or question distributions, usually from different datasets. Our work mainly falls into the category of test-time perturbation. We distinguish from previous work by focusing on the effect of entity replacement, with the motivation that entities can have flexible and diverse names in the real life.

Model Robustness to Entity Substitution. It is non-trivial for NLP models to be able to properly handle the large space of named entities. With different types of candidate names, previous works audit NLP models' sensitivity to entity substitution with different implications. Agarwal et al. (2020) replace original entities in the NER datasets with entities of different national origin to study NER model's robustness and fairness. Lin et al. (2021) study NER model's reliance on name and context with entity substitution under the same fine-grained class. For other tasks, Balasubramanian et al. (2020) investigate the robustness of models trained on several NLP benchmarks with person name replacement. However, they only experiment on SQuAD for MRC and conclude it to be quite robust, failing to unveil the actual challenge. The contemporaneous work of Longpre et al. (2021) mainly analyzes the memorization behavior of *generative* open-domain QA models using *knowledge conflicts*. The use entity substitution to create test passages that contain facts contradicting to what the model has learned during training time. In contrast, we analyze *extractive* MRC model's robustness when encountering new entities, by evaluating on *modified test sets* without intentionally introduced knowledge conflicts. The extractive task formulation also makes the model unable to output its memorized knowledge as generative models, leading to different analysis questions and methods.

6 Conclusion

In this paper, we systematically study the robustness of MRC models to entity name substitution. Specifically, we propose a substitution framework along with candidate names of different implications. We experiment with three pretrained language models on five MRC datasets. We find that models trained on distantly-supervised datasets are susceptible to entity name substitution, while models trained on human-annotated datasets are relatively robust, with GPE renaming harder than PER and ORG renaming. The lack of robustness can be further attributed to model's overreliance on entity knowledge and name clues. We also find that SpanBERT, which is pretrained using span-level objectives, shows better robustness than BERT and RoBERTa. Leveraging these insights, we study defense approaches based on continual pretraining and demonstrate that entity-based masking policies are beneficial to model's robustness.

References

Oshin Agarwal, Yinfei Yang, Byron C. Wallace, and A. Nenkova. 2020. Entity-switched datasets: An approach to auditing the in-domain robustness of named entity recognition models. *ArXiv preprint*, abs/2004.04123.

Sriram Balasubramanian, Naman Jain, Gaurav Jindal, Abhijeet Awasthi, and Sunita Sarawagi. 2020. What's in a name? are BERT named entity representations just as good for any other name? In *Proceed-*

ings of the 5th Workshop on Representation Learning for NLP, pages 205–214, Online. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *ArXiv preprint*, abs/1704.05179.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13, Hong Kong, China. Association for Computational Linguistics.

Wee Chung Gan and Hwee Tou Ng. 2019. Improving the robustness of question answering systems to question paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6065–6075, Florence, Italy. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *ArXiv preprint*, abs/2002.08909.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. Certified robustness to adversarial word substitutions. 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)*, pages 4129–4142, Hong Kong, China. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Mandar Joshi, Eunsol Choi, Daniel 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 (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Divyansh Kaushik and Zachary C. Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015, Brussels, Belgium. Association for Computational Linguistics.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur 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.

Yuxuan Lai, Chen Zhang, Yansong Feng, Quzhe Huang, and Dongyan Zhao. 2021. Why machine reading comprehension models learn shortcuts? In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 989–1002, Online. Association for Computational Linguistics.

Bill Yuchen Lin, Wenyang Gao, Jun Yan, Ryan Moreno, and Xiang Ren. 2021. Rockner: A simple method to create adversarial examples for evaluating the robustness of named entity recognition models. *ArXiv preprint*, abs/2109.05620.

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. *ArXiv preprint*, abs/1907.11692.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. *ArXiv preprint*, abs/2109.05052.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank.

John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. 2021. Accuracy on the line: On the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning*, pages 7721–7735. PMLR.

Junghyun Min, R. Thomas McCoy, Dipanjan Das, Emily Pitler, and Tal Linzen. 2020. Syntactic data augmentation increases robustness to inference heuristics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2339–2352, Online. Association for Computational Linguistics.

Anselmo Peñas, Eduard H Hovy, Pamela Forner, Álvaro Rodrigo, and Richard FE Sutcliffe. 2011. Overview of qa4mre at clef 2011: Question answering for machine reading evaluation. Citeseer.

Danish Pruthi, Bhuwan Dhingra, and Zachary C. Lipton. 2019. Combating adversarial misspellings with robust word recognition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5582–5591, Florence, Italy. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Marco Tulio Ribeiro, Carlos Guestrin, and Sameer Singh. 2019. Are red roses red? evaluating consistency of question-answering models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6174–6184, Florence, Italy. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia. Association for Computational Linguistics.

Barbara Rychalska, Dominika Basaj, and Przemyslaw Biecek. 2018. Are you tough enough? framework for robustness validation of machine comprehension systems. *ArXiv preprint*, abs/1812.02205.

Priyanka Sen and Amir Saffari. 2020. What do models learn from question answering datasets? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2429–2438, Online. Association for Computational Linguistics.

Chenglei Si, Ziqing Yang, Yiming Cui, Wentao Ma, Ting Liu, and Shijin Wang. 2021. Benchmarking robustness of machine reading comprehension models. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 634–644, Online. Association for Computational Linguistics.

Saku Sugawara, Pontus Stenetorp, Kentaro Inui, and Akiko Aizawa. 2020. Assessing the benchmarking capacity of machine reading comprehension datasets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8918–8927.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Qinyuan Ye, Belinda Z Li, Sinong Wang, Benjamin Bolte, Hao Ma, Wen-tau Yih, Xiang Ren, and Madian Khabsa. 2021. On the influence of masking policies in intermediate pre-training. *ArXiv preprint*, abs/2104.08840.

Xiaorui Zhou, Senlin Luo, and Yunfang Wu. 2020. Co-attention hierarchical network: Generating coherent long distractors for reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9725–9732.

A Statistics of Evaluation Datasets

As the official test sets of the MRQA datasets are hidden, we use the development set as the in-house test set, and hold out 10% of the training data as the in-house development set. Their statistics are shown in Table 8.

Dataset	# Train	# Dev	# Test	DS?
SQuAD	77,929	8,659	10,507	✗
NQ	84,577	9,367	12,836	✗
HotpotQA	65,636	7,292	5,901	✗
SearchQA	105,646	11,738	16,980	✓
TriviaQA	42,569	4,696	7,785	✓

Table 8: Evaluation datasets. "DS?" indicates whether distant supervision is used for data collection.

We show the sizes of the perturbable subsets for different entity types and their union[6] (**MIX**) in Table 9.

Dataset	# PER	# ORG	# GPE	# MIX
SQuAD	1,170	1,095	602	2,613
NQ	3,257	1,207	1,414	5,150
HotpotQA	1,351	824	788	2,614
SearchQA	5,707	2,450	2,248	8,688
TriviaQA	2,747	1,276	1,270	4,351

Table 9: Statistics of the perturbable subsets for different entity types and their union ("MIX").

B MRC Model Training

The pretrained language models are finetuned on the MRC dataset to predict the start and end tokens of the answer span based on the concatenated question and passage. We train using mixed precision, with batch size of 16 sequences for 4 epochs. The maximum sequence length is set to 256 tokens. We use the AdamW optimizer (Loshchilov and Hutter, 2019) with an initial learning rate of 2e-5 that is linearly decayed to 0 during finetuning.

C Details for Continual Pretraining

MLM (vanilla) refers to the masking strategy used by BERT (Devlin et al., 2019), where the masked tokens are randomly sampled. **MLM (whole word)** always masks all tokens corresponding to a word at once. **MLM (span)** uses the masking strategy proposed by Joshi et al. (2020), which masks random spans rather than individual whole words or tokens. **MLM (entity)** masks a random entity for 50% of the time, and uses MLM (span) for the other 50% of the time. The idea is inspired by salient span masking proposed in Guu et al. (2020).

To eliminate domain shift during continual pretraining as a possible explanation for any improvements, we keep the corpus for continual pretraining consistent with the pretraining corpus used by SpanBERT, which is the concatenation of BookCorpus and English Wikipedia. We train using mixed precision, with effective batch size of 2,048 sequences for 8,000 steps, with 256 tokens per sequence. We use the AdamW optimizer with a constant learning rate of 1e-4.

D Sources for Person Names from More Cultures

Spanish

https://data.world/axtscz/spanish-first-name/workspace/file?filename=ESGivenMale.json
https://data.world/axtscz/spanish-first-name/workspace/file?filename=ESGivenFemale.json
https://www.kaggle.com/migalpha/spanish-names?select=male_names.csv

India

https://gist.github.com/mbejda/7f86ca901fe41bc14a63
https://gist.github.com/mbejda/9b93c7545c9dd93060bd
https://github.com/merishnaSuwal/indian_surnames_data/blob/master/indian_caste_data.csv

Arabic

https://github.com/zakahmad/ArabicNameGenderFinder
https://parenting.firstcry.com/baby-names/unisex/religion/muslim/
https://en.wikipedia.org/wiki/Category:Arabic-language_surnames

French

https://www.kaggle.com/haezer/french-baby-names
https://en.wikipedia.org/wiki/Category:French-language_surnames

China

http://www.cjki.org/samples/chin100mil.htm
https://github.com/psychbruce/ChineseNames

[6] Some answer spans contain multiple entities of different types. Some entities are recognized as different types for their different mentions in the passage.