X-FACTR: Multilingual Factual Knowledge Retrieval from Pretrained Language Models

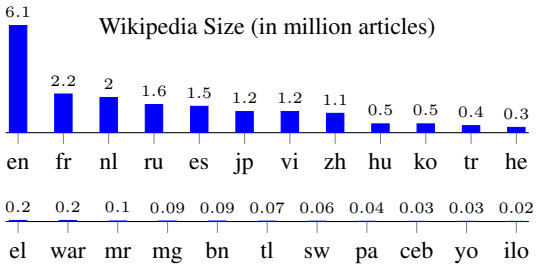
Zhengbao Jiang[†], **Antonios Anastasopoulos**[♣,*], **Jun Araki**[‡], **Haibo Ding**[‡], **Graham Neubig**[†]
[†]Languages Technologies Institute, Carnegie Mellon University
[♣]Department of Computer Science, George Mason University
[‡]Bosch Research

Abstract

Language models (LMs) have proven surprisingly successful at capturing factual knowledge by completing cloze-style fill-in-the-blank questions such as "Punta Cana is located in _." However, while knowledge is both written and queried in many languages, studies on LMs' factual representation ability have almost invariably been performed on English. To assess factual knowledge retrieval in LMs in different languages, we create a multilingual benchmark of cloze-style probes for 23 typologically diverse languages. To properly handle language variations, we expand probing methods from single- to multi-word entities, and develop several decoding algorithms to generate multi-token predictions. Extensive experimental results provide insights about how well (or poorly) current state-of-the-art LMs perform at this task in languages with more or fewer available resources. We further propose a code-switching-based method to improve the ability of multilingual LMs to access knowledge, and verify its effectiveness on several benchmark languages. Benchmark data and code have be released at https://x-factr.github.io.

1 Introduction

Language models (LMs; (Church, 1988; Kneser and Ney, 1995; Bengio et al., 2003)) learn to model the probability distribution of text, and in doing so capture information about various aspects of the syntax or semantics of the language at hand. Recent works have presented intriguing results demonstrating that modern large-scale LMs also capture a significant amount of *factual knowledge* (Petroni et al., 2019; Jiang et al., 2020; Poerner et al., 2019). This knowledge is generally probed by having the LM fill in the blanks of cloze-style prompts such as "Obama is a _ by profession.", where these prompts are invariably written in English. However, it goes without saying that there are many languages of the world other than English, and it is quite conceivable that (1) users may want to query this factual knowledge in other languages, and (2) some facts will be written in non-English languages and thus multilingually trained LMs (hereinafter, M-LMs) may be more equipped to recall these facts in the languages of the original data. In this paper, we study the intersection of *multilinguality and the factual knowledge included in LMs*.



fact	⟨Bloomberg L.P., founded_in, New York⟩
en prompt	[X] was founded in [Y].
es prompt	[X] fue [fundar.Gerund;X] en [Y].
es sentence	Bloomberg L.P. fue fundada en ⟨mask⟩ $\times 1 \sim 5$.

es outputs	prediction	#tokens	confidence
	2012	1	-1.90
	Nueva York	2	-0.61
	EE. UU	3	-1.82
	Chicago, Estados Unidos	4	-3.58
	2012 Bloomberg L.P	5	-3.06

Figure 1: X-FACTR contains 23 languages, for which the data availability varies dramatically. Prompts get instantiated to produce grammatical sentences with different numbers of mask tokens and are used to obtain predictions for [Y]. In this Spanish example, the verb gerund "fundar" *to found* is rendered as "fundada" to agree in gender and number with the subject "Bloomberg L.P.". The final prediction is in bold.

We create a new multilingual benchmark for probing factual knowledge in LMs – the Cross-lingual FACTual Retrieval benchmark (X-FACTR).

∗: Work done at Carnegie Mellon University. The first two authors contributed equally.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 5943–5959,
November 16–20, 2020.

X-FACTR shares a similar formulation as the LAMA benchmark of Petroni et al. (2019), which assesses whether LMs have memorized a fact (i.e., a subject-relation-object triple) by having LMs predict the blank (i.e. object) in a cloze-style prompt for each relation after filling in the subject. We manually create such prompts for 23 languages spanning different language families and different levels of data availability (§ 3.1). Because many languages that we handle are morphologically rich, we design a morphology-sensitive annotation schema (see example in Fig. 1) that can properly instantiate prompts using entity metadata (e.g. gender) and a morphological inflection model (§ 3.3).

In addition, while previous works (Petroni et al., 2019; Jiang et al., 2020; Poerner et al., 2019) have limited examination to single-token entities (e.g. "France"), we expand our setting to include multi-token entities (e.g. "United States"), which comprise more than 75% of facts included in our underlying database (Wikidata; § 3.2). We propose several decoding algorithms for prediction of these multi-token entities using masked LMs (§ 4). We discuss the related work in depth in § 7.

We perform experiments on X-FACTR (§ 5), comparing and contrasting across languages and LMs to answer the following research questions: (1) How and why does performance vary across different languages and models? (2) Can multilingual pre-training increase the amount of factual knowledge in LMs over monolingual pre-training? (3) How much does knowledge captured in different languages overlap? We find that the factual knowledge retrieval of M-LMs in high-resource languages is easier than in low-resource languages, but the overall performance is relatively low, indicating that this is a challenging task. We analyze the types of failure cases, shedding light on future directions to improve factual knowledge in M-LMs. In addition, multilingual pre-training does not necessarily lead to a higher recall of facts compared to language-specific monolingual pre-training. The knowledge memorized by M-LMs in fact is largely distinct across languages, with almost 50% of facts being recalled in only one language.

Inspired by the above observations, we propose a code-switching-based objective function to improve the ability of M-LMs to access knowledge using queries from a variety of languages. We replace entities in a sentence from the original language with counterparts in another language, and further fine-tune the LM on these code-switched data (§ 6). We perform experiments on three languages (French, Russian, and Greek, code-switched with English). Results demonstrate that this code-switching-based learning can successfully improve the knowledge retrieval ability with low-resource language prompts.

2 Retrieving Facts from LMs

In this paper we follow the protocol of Petroni et al. (2019)'s English-language LAMA benchmark, which targets factual knowledge expressed in the form of subject-relation-object triples from Wikidata[1] curated in the T-REx dataset (ElSahar et al., 2018). The cloze-style prompts used therein are manually created and consist of a sequence of tokens, where [X] and [Y] are placeholders for subjects and objects (e.g. "[X] is a [Y] by profession."). To assess the existence of a certain fact, [X] is replaced with the actual subject (e.g. "Obama is a ⟨mask⟩ by profession.") and the model predicts the object in the blank $\hat{y}_i = \text{argmax}_{y_i} p(y_i|\boldsymbol{s}_{i:i})$, where $\boldsymbol{s}_{i:i}$ is the sentence with the i-th token masked out. Finally, the predicted fact is compared to the ground truth. In the next section, we extend this setting to more languages and predict multiple tokens instead of a single one.

3 Multilingual Multi-token Factual Retrieval Benchmark

3.1 Languages

In sampling the languages to create our multilingual benchmark, we attempted to create a subset as diverse as possible with regards to data availability, typology, and script – within the constraints of requiring inclusion in Wikidata and standard pre-trained M-LMs. To this end, we created prompts in 23 languages: English, French, Dutch, Spanish, Russian, Japanese, Chinese, Hungarian, Hebrew, Turkish, Korean, Vietnamese, Greek, Cebuano, Marathi, Bengali, Waray, Tagalog, Swahili, Punjabi, Malagasy, Yoruba, and Ilokano.

Our subset includes languages from 11 families (the Indo-European ones include members of the Germanic, Romance, Greek, Slavic, and Indic genera), using 10 different scripts. Our languages display high variance with respect to Wikipedia presence, a proxy for overall data availability, ranging from very large to very small (see Fig. 1).[2]

[1] https://www.wikidata.org/

[2] We excluded bot-made pages for Cebuano and Waray.

	en	fr	nl	es	ru	ja	zh	hu	he	tr	ko	vi	el	bn	ceb	mr	war	tl	sw	pa	mg	yo	ilo
#all	45.7	40.2	38.3	37.1	26.3	25.1	23.1	20.4	17.1	16.1	16.1	13.6	13.0	9.4	8.2	7.9	7.3	7.1	6.8	5.5	4.9	4.6	4.1
#single-token	18.9	13.9	12.8	13.5	3.4	1.3	0.2	6.2	1.1	2.5	2.0	3.9	0.7	0.1	3.3	0.2	3.0	3.2	2.8	0.1	1.7	0.9	2.1
#multi-token	26.8	26.4	25.5	23.6	22.9	23.8	22.9	14.2	16.0	13.6	14.1	9.7	12.3	9.3	4.9	7.7	4.4	3.9	4.0	5.4	3.2	3.7	2.0

Table 1: X-FACTR benchmark statistics (in thousands). More details in the Appendix (Tab. 5 and Fig. 6).

3.2 Facts

While Petroni et al. (2019) and follow-up works focus on entities that can be represented by a single token, since many popular entities consist of multiple tokens (e.g. "United States"), we argue that it is crucial to include multi-token entities in the benchmark to make the evaluation unbiased. Similar to Petroni et al. (2019), we use the T-REx dataset to collect facts for our benchmark. Since T-REx aligns facts from Wikidata with sentences in abstract sections from DBpedia, we can estimate the commonality of each fact based on its frequency of being grounded to a sentence in these abstracts.

For each of the 46 relations in T-REx, we sample 1000 subject-object pairs with probability proportional to their frequency. Frequency-proportional sampling makes the distribution of the facts in our benchmark close to real usage and covers facts of different popularity. To keep the benchmark unbiased, we did not constrain the facts with any language-related criteria (e.g., require the entities to have translations in all languages we considered). As a result, some entities (either subjects or objects) might not have translations in all languages. The number of facts in different languages in our multilingual multi-token X-FACTR benchmark is shown in Tab. 1. Because many modern pre-trained M-LMs almost invariably use some variety of subword tokenization, the number of tokens an entity contains will depend on the tokenization method used in the LM. We report the statistics based on the WordPiece tokenization used in multilingual BERT (Devlin et al., 2019). The tokenization scheme statistics for the other M-LMs are similar.

3.3 Prompts

Some languages we include in the benchmark require additional handling of the prompts to account for their grammar or morphology. For example, (some) named entities inflect for case in languages like Greek, Russian, Hebrew, or Marathi. In some languages syntactic subjects and objects need to be in particular cases. Similarly, languages often require that the verb or other parts of the sentence agree with the subject or the object on some morphological features like person, gender, or number.

Our prompts provide the necessary information in order to generate grammatical sentences, given the gender and number of the entities. For example, the Russian prompt for "[X] was born in [Y]" is:

$$\left[\text{X.Nom}\right] \left[\text{родился;X=MASC} \mid \text{родилась;X=FEM} \mid \text{родилось;X=NEUT}\right] \text{в} \left[\text{Y.Ess}\right].$$

The prompt denotes that the subject ([X]) needs to be in the nominative (`Nom`) case and the object ([Y]) needs to be inflected in the essive case (`Ess`). The prompt also accounts for the variation of the gender of [X] providing options (separated by `|`) for the subject being masculine, feminine, or neuter (`MASC`, `FEM`, `NEUT` respectively).

Everything within square brackets gets concretely instantiated given the subject and object. Grammatical gender is assigned through a combination of Wikidata information and language-specific heuristics, constructed based on feedback from native speakers of each language. When the entity corresponds to a person, we retrieve their "sex_or_gender" properties from Wikidata. In addition, for languages like Greek or French, the gender of an entity can be inferred with fairly high certainty given the form of the word (e.g. looking at the ending). Last, some categories of entities (such as cities, countries, organizations, etc, which can be obtained using the "instance_of" Wikidata property) often get assigned a general grammatical case based on the category.

Once all the morphological features have been specified as detailed above, we use the `unimorph_inflect` package (Anastasopoulos and Neubig, 2019) to generate the appropriately inflected surface form of the bracketed words.[3] We note that the target entity ([Y]) might also need to be inflected, as in the above Russian example, in which case we require the model's predictions to match the inflected target forms.

To verify the quality of the prompts we performed user studies with native speakers, finding that 88% on average were judged as natural and grammatically correct. Details are shown in Appendix B, but it is worth noting that the majority

[3] https://github.com/antonisa/unimorph_inflect

of errors are due to prompts being awkward or incorrect for some senses captured by the relation, and not due to our gender heuristics or automatic inflection. This issue is also present in the LAMA English prompts (Jiang et al., 2020).

3.4 Evaluation

As noted in Petroni et al. (2019), because some subject-relation pairs might have multiple correct objects (e.g., America maintains diplomatic relations with multiple countries), we collect all valid objects and judge a prediction as correct if it can match any object (e.g., both France and Canada are correct). Since an entity might have multiple aliases (e.g., "America" and "the US"), we collect all aliases for each entity from Wikidata, and the prediction is marked as correct if it can match any one of them after lowercasing.

4 Multi-token Decoding

As Tab. 1 shows, many facts involve multi-token entities and thus a LM would need to predict these entities in multiple steps. Generating multiple predictions is straightforward for traditional left-to-right LMs (Sundermeyer et al., 2015; Radford et al., 2019), where we can autoregressively decode the next token conditioned on previous tokens. However, many pre-trained LMs such as BERT (Devlin et al., 2019) are *masked* LMs that predict individual words given left and right contexts, and decoding from such masked LMs remains an open problem (Lawrence et al., 2019; Salazar et al., 2020; Ghazvininejad et al., 2019; Wang and Cho, 2019; Cho, 2019). We systematically examined different multi-token decoding algorithms from three orthogonal perspectives: (1) how the initial predictions are produced, (2) how to refine the predictions, and (3) other commonly used components in neural text generation systems. We assume that the following conditional probability distribution is defined by the masked LM for a sentence with n tokens:

$$p(x_k|x'_1, ..., x'_{k-1}, \langle\text{mask}\rangle_k, x'_{k+1}, ..., x'_n), \quad (1)$$

where the subscript of $\langle\text{mask}\rangle$ indicates its position, and the surrounding token x' can either be an actual word $x.$ or $\langle\text{mask}\rangle$. We aim to handle sentences containing multiple mask tokens conditioning on the surrounding actual words:

$$\boldsymbol{s}_{i:j} = x_1, ..., x_{i-1}, \langle\text{mask}\rangle_i, ..., \langle\text{mask}\rangle_j, x_{j+1}, ..., x_n, \quad (2)$$

where $\boldsymbol{s}_{i:j}$ indicates a sentence with the i-th to j-th tokens masked out.[4]

[4]We assume that the mask tokens are consecutive for notation simplicity, although all following methods/equations can be easily adapted to non-consecutive cases.

(a) Independent: Barack Obama is a United$_1$ of$_1$ president$_1$ by profession

(b) Order: Barack Obama is a United$_1$ State$_2$ President$_3$ by profession

(c) Confidence: Barack Obama is a minister$_2$ of$_3$ cabinet$_1$ by profession

Figure 2: Illustration of three initial prediction and refinement methods. Green boxes are mask tokens to be filled, and subscripts indicate the prediction order.

4.1 Initial Prediction and Refinement

Given a sentence with multiple mask tokens, e.g., Eq. 2, we can either generate outputs in parallel independently or one at a time conditioned on the previously generated tokens. These methods are similar to the prediction problems that BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019b) perform in their pre-training stages respectively. We define $\boldsymbol{c} \in \mathbb{R}^n$ as the probability of each prediction, with details varying by prediction methods.

After all mask tokens are replaced with the initial predictions, i.e., $\hat{\boldsymbol{s}}_{i:j} = x_1, ..., \hat{y}_i, ..., \hat{y}_j, ..., x_n$, we can further refine the predictions by iteratively modifying one token at a time until convergence or until the maximum number of iterations is reached. Here we outline the algorithms with high-level descriptions, and provide concrete details in Appendix C.

Independent. For independent initial prediction (Fig. 2a), the mask tokens are all predicted in parallel (at once). We also consider two autoregressive methods for initial prediction or refinement.

Order-based. Mask tokens are predicted from left to right, in each step conditioning also on the previously generated tokens (Fig. 2b). In the refinement stage, we modify predictions also from left to right, and convergence is reached when there are no changes in a left-to-right scan.

Confidence-based. In each step, we choose the prediction with the highest probability, so the order of predictions can be arbitrary (Fig. 2c). In the refinement stage, we choose from all predicted tokens the one with the lowest confidence (i.e., the lowest probability) and re-predict it similarly to Ghazvininejad et al. (2019). Convergence is reached when the re-predicted token is the same as the original token.

4.2 Final Prediction

Because we do not know the number of tokens of the ground truth in advance, we enumerate from 1 to M mask tokens and choose the final prediction based on the confidence. Given the prompt in Eq. 2, the simplest way to compute the confidence is pseudo log likelihood, which is the sum

of log probabilities of each predicted token conditioned on the other tokens (Salazar et al., 2020): $v(j-i+1) = \sum_{k=i}^{j} \log c_k$, where c_k is the confidence (probability) of the k-th predicted token, and $v(m)$ is the overall prediction confidence with m initial mask tokens. Among M predictions, we choose the one with the highest confidence.

4.3 Additional Components

We also investigate additional components commonly used in neural generation systems. Specifically, we consider **length normalization** in computing the final confidence (i.e., divide $v(m)$ by the number of mask tokens m) because a simple sum might favor short predictions. In addition, the confidence value $\boldsymbol{c}$ in previous methods contains probabilities when the predictions are first generated, which will become stale once the surrounding tokens change (Ghazvininejad et al., 2019). We consider **re-computing confidence $\boldsymbol{c}$** whenever a change happens. Last, we attempted **beam search** to keep track of the most plausible B predictions at each step. Details of these components can be found in Appendix C, along with a general schema of the overall decoding algorithm in Alg. 1.

5 X-FACTR Benchmark Performance

Implementation Details. We use the implementations of different multilingual/monolingual pre-trained LMs in the Transformers library (Wolf et al., 2019). We examine 3 multilingual pre-trained LMs, M-BERT, XLM, XLM-R (Devlin et al., 2019; Conneau and Lample, 2019; Conneau et al., 2019),[5] and 8 monolingual pre-trained LMs, BERT (en), CamemBERT (fr), BERTje (nl), BETO (es), RuBERT (ru), Chinese BERT (zh), BERTurk (tr), and GreekBERT (el) (Martin et al., 2020; de Vries et al., 2019; Cañete et al., 2020; Kuratov and Arkhipov, 2019; Schweter, 2020). Details of these models can be found in Appendix D.

We set the maximal number of mask tokens to $M=5$ for English, French, Dutch, and Spanish. In these languages more than 90% of the entities are split into $\leq$5 tokens. For all other languages we use $M=10$. This is expected because the vocabulary of M-LMs based on WordPiece tokenization is dominated by frequent words and low-resource-language words tend to split into more pieces (Ács, 2019). We set the maximal number of iterations to $T=2M$, so that we can approximately refine all the predicted tokens once for a sentence with M mask tokens (the initial prediction takes exactly M iterations). In our main results, we report results with two decoding algorithms: the simplest independent generation method and the confidence-based method for both initial and refinement predictions. The latter performs better than order-based methods, as we will show in Tab. 3. To save computation time, we only use confidence re-computation for $M=5$. We discuss computation complexity in Appendix C.

Evaluation Metrics. We follow Petroni et al. (2019), computing the accuracy of predicted objects for each relation and macro-average them as final scores. For fine-grained analysis of different decoding methods, pre-trained LMs, and languages, we report results on **all** facts as well as on subsets consisting only of single-token objects (**single**) and multi-token objects (denoted as **multi**).

5.1 Experimental Results

We run both the independent and confidence-based decoding methods with 3 M-LMs, and when available 8 monolingual LMs, across 23 languages,[6] with results shown in Fig. 3. Overall, even in the most favorable settings, the performance of state-of-that-art M-LMs at retrieving factual knowledge in the X-FACTR benchmark is relatively low, achieving less than 15% on high-resource languages (e.g., English and Spanish) and less than 5% for some low-resource languages (e.g., Marathi and Yoruba). This may initially come as a surprise, given the favorable performance reported in previous papers (Petroni et al., 2019; Jiang et al., 2020), which achieved accuracies over 30% on English. We justify this discrepancy in our following analysis. We note that, although we provide baseline results in almost all languages, we perform our extensive analysis on a representative subset, consisting of 13 languages.

Performance on Different Languages. Performance on high-resource languages is usually better than performance on middle- or low-resource languages regardless of the (M-)LMs. This is probably due to high-resource languages having more data in the pre-training stage. It is also possible that even if the fact of low-resource languages is written in the available data for these languages, it is not appropriately memorized due to lack of model capacity or forgetting (Kirkpatrick et al., 2017). It

[5] Yoruba is not in the training data of XLM and XLM-R.

[6] Check https://x-factr.github.io for latest results.

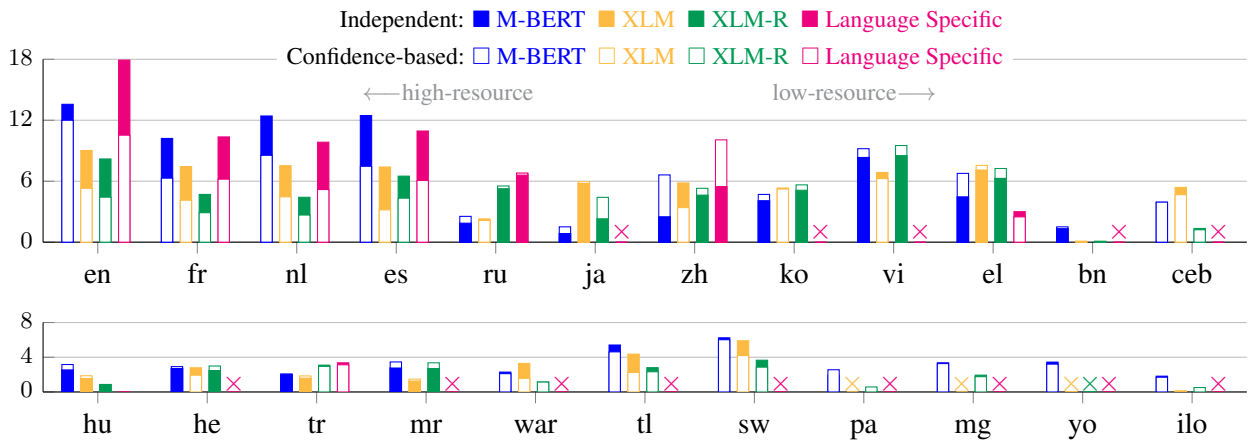


Figure 3: Accuracy on different languages using different LMs (%). Independent prediction (solid bars) outperforms confidence-based prediction (no-fill bars) on high-resource languages but not on low-resource languages. Different models are color-coded, with missing/unsupported models marked with ×. Languages are ranked by the total number of facts in our benchmark. Details in Appendix Tab. 10.

is worth noting that the best results are in Indo-European languages which not only have the most data, but also share the same (Latin) script which could further facilitate cross-lingual learning.

Performance of Different LMs. Comparing the performance of different M-LMs, we found that M-BERT outperforms XLM and XLM-R on high-resource languages, while on low-resource languages performance is similar. This is contradictory to the conclusion on other cross-lingual tasks, such as natural language inference and syntactic prediction, as reported in Hu et al. (2020). Our conjecture is that because factual knowledge probing requires retrieving the identity and relations of individual entities, it is more fine-grained than more coarse-grained understanding of syntactic and semantic classes that are required to solve other tasks. We posit that pre-training methods that show superior performance on inference and syntactic prediction tasks (i.e., XLM-R) might achieve good syntactic/semantic abstraction at the cost of making less concrete lexical distinctions.

Comparing M-BERT with language-specific LMs, we find M-BERT outperforms the monolingual BERT on Dutch, Spanish, and Greek, while underperforming on English, Russian, Chinese, and Turkish. Since most of the LMs follow the architecture and pre-training settings of BERT (Devlin et al., 2019) or RoBERTa (Liu et al., 2019), we hypothesize that training corpus is the major contributor to the final performance, and summarize those corpora in Tab. 8 in the Appendix. Another potential explanation is that model capacity limitations preclude M-BERT from effectively memorizing entity names/relations in all of the languages.

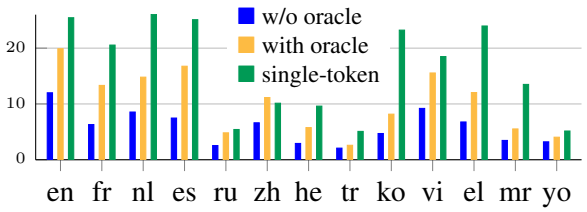


Figure 4: Accuracy of the confidence-based decoding algorithm on different languages using M-BERT w/ and w/o oracle length (%).

Single-token vs Multi-token. Since we choose among M candidate predictions with different numbers of mask tokens based on confidence, it is possible that the prediction with the correct number of mask tokens has lower confidence than the other predictions. To investigate the errors introduced by this step, we conduct an ablation experiment that assumes we know the ground-truth number of mask tokens. As shown in Fig. 4, performance improves significantly by 75% on average across all languages using the oracle mask number, indicating that pre-trained LMs have difficulties in choosing the correct number of mask tokens. The performance on single-token facts (i.e., the setting of previous works that only predicts a single token) is even higher, demonstrating the difficulty of multi-token prediction.[7]

[7]The 31.1% accuracy of BERT in Petroni et al. (2019) is over a different set of facts in English, constrained to be in the intersection of vocabularies of several LMs. We have no such constraint, which may explain the slightly lower 25.5% accuracy on the English single-token performance in Fig. 4.

Type	Prompt	Prediction	Gold	en	es	el
Correct	Macintosh 128K is produced by _.	Apple	Apple	*19.89*	*16.68*	*12.02*
Repeating subjects	Malin Reuterwall plays with _.	the Reuterwall team	Sweden's Womens Football	22.21	24.62	25.06
Wrong entities	Austria maintains diplomatic relations with _.	the United States	Italy, Russia, ...	16.66	29.07	18.74
Non-informativeness	Switzerland is named after _.	him	Canton of Schwyz	18.24	9.81	26.78
Type errors	Nin9 2 5ive was written in _.	the 1880s	Cantonese	7.93	6.11	0.00
Related concepts	Christof Lauer used to work in _.	Germany	Melsungen	7.14	1.67	1.91
Unk	Randy Newman plays _.	D.D	piano	5.55	8.33	11.67
False Negative	Switzerland maintains diplomatic relations with _.	the Federal Republic of Germany	Germany	2.38	3.52	3.06
Inflection	-	-	-	0.00	0.19	0.77

Table 2: Error cases of M-BERT in English and ratio of different error types in English, Spanish, and Greek (%). Error cases in Spanish and Greek can be found in Tab. 9 in the Appendix.

Error Analysis. Even with access to an oracle for the number of target tokens, though, the performance is still lower than 20%. To understand the types of errors made by the LMs, we sample over 400 error cases in English, Spanish, and Greek, and classify them. The error type distributions along with English examples are outlined in Tab. 2.

The most prominent error type, about one-fourth of mistakes for all LMs, was **repeating subjects**, whereby the prediction repeats either the full or partial subject. Predicting the **wrong entities** is also fairly common, especially in Spanish (29%). Interestingly, we find that wrong predictions are often a language-specific "common" entity such as 'Αθήνα' (Athens, the capital of Greece) in Greek location prompts, while the Spanish model insisted most musicians play 'flauta' (flute). Another error type, particularly common in Greek (27%), is producing **non-informative** output, where the predictions are function words that could never be an entity. **Type errors** when the semantic type of the prediction is different than expected (e.g. predicting dates instead of locations) are fairly common (English: 8%, Spanish 6%), as are **related concepts** predictions (English: 7%), where the model predicts relevant, possibly factually correct entities (e.g. predicting a country or a state instead of a city). Worryingly, in a fair amount of cases (English: 5%, Spanish: 8%, Greek: 11%) the models output non-existent words (**unk**). Errors of the last 4 types could potentially be avoided by limiting the allowed outputs of the model to specific entity classes; we leave this for future work. Last, we identified around 3% of **false negatives**, where the prediction is actually correct but is not part of our aliases list and less than 1% of **inflection** errors where the prediction is the correct entity but improperly inflected.

Performance of Different Decoding Methods. Overall, the confidence-based decoding method improves the accuracy in middle- and low- resource languages, while it hurts the performance on high-resource languages. To better understand the effect of different components on the final performance, we conduct a comprehensive comparison on English and Chinese. We compare the three initial prediction methods and the three refinement options (including not performing refinement), for a total of nine decoding methods (§ 4.1). We further apply additional improvements (§ 4.3) on the confidence-based decoding method.

		English			Chinese		
Init.	Refine	All	Single	Multi	All	Single	Multi
Indep.	-	13.57	**22.40**	5.57	2.50	**9.61**	2.22
	Order	**13.91**	21.71	6.71	4.26	8.80	4.01
	Conf.	13.38	21.49	5.82	4.04	9.33	3.80
Order	-	13.54	20.37	6.60	5.06	8.57	4.85
	Order	13.30	19.75	6.57	5.79	8.29	5.61
	Conf.	13.36	19.86	6.56	5.68	8.29	5.50
Conf.	-	13.64	19.53	7.38	6.55	5.34	6.41
	Order	13.73	19.48	7.57	**6.79**	4.63	**6.67**
	Conf.	13.72	19.44	7.48	6.62	5.21	6.40
+Len. norm		8.60	9.43	6.18	3.96	2.27	3.93
+Re-comp.		12.00	12.91	10.08	5.89	2.71	5.84
+Beam		10.84	9.29	**11.06**	6.34	2.38	6.30

Table 3: Accuracy of different decoding methods using M-BERT on English and Chinese (%).

By comparing the performance in Tab. 3, we first see advanced decoding methods improve performance on multi-token objects, but hurt performance on single-token ones. The best-performing decoding method on English improves the multi-token accuracy from 5.57% to 11.06%, indicating that advanced decoding methods have a better chance to elicit multi-token facts from M-BERT. Some examples are shown in Tab. 7 in the Appendix. The lower performance on single-token objects is probably caused by the fact that advanced decoding methods discover multi-token predictions that have higher confidence than single-token ones (§ 4.2). For example, the single-token prediction for "Enrique Iglesias used to communicate in _." is "Spanish", while the best decoding method outputs "his own words" with higher confidence. Second, initial prediction methods have a greater effect on the final performance than refinement methods. We

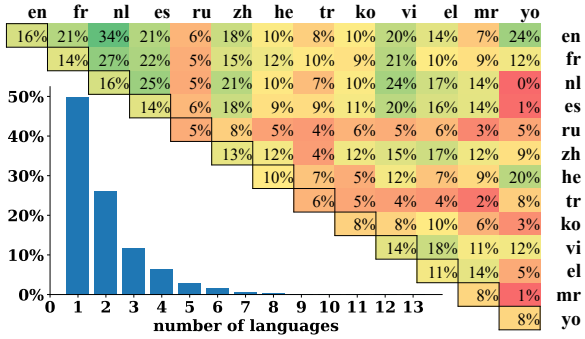


Figure 5: Bottom-left: the ratio of facts with respect to the number of languages in which the facts could be successfully retrieved. Top-right: overlap ratio of correct predictions between two languages. The values on the diagonal are the average overlap ratio of the corresponding language with the other languages.

hypothesize that this is because the greedy decoding process heavily depends on previous predictions, and refinement cannot recover from unsatisfactory initial predictions. Third, length normalization was not found useful in either case.

There are also observations not consistent across the two languages. First, since Chinese has a larger portion of multi-token objects than English (as shown in Tab. 1), the overall performance on Chinese increases while it decreases on English, which is consistent with the observation in Fig. 3. Second, confidence re-computation and beam search are not as effective on Chinese, which we conjecture is because that the distribution over English sentences exhibits more multimodality than the distribution over Chinese sentences due to more training data.

6 Improving Multilingual LM Retrieval

As the performance of M-LMs is relatively low, especially on low-resource languages, an obvious endeavor is to refine the model to improve fact retrieval performance in various languages. We analyze how similarly M-BERT performs on queries in different languages. We collect correctly predicted facts across all languages, and count in how many languages each fact was retrieved correctly. As shown in the bottom-left histogram of Fig. 5, half of the correctly predicted facts were correct in a single language, indicating little overlap across languages (Lin et al., 2018). Only 3% of facts were correct in more than 5 languages, and objects in those facts are usually sub-strings of subjects, making them easy to retrieve regardless of the language. This observation is also confirmed by the overlap between pairs of languages in the top-right chart of Fig. 5; even the most similar languages (i.e., English and Dutch) only have 34% of correct predictions in common.

We find that facts retrievable only in a single language tend to be knowledge that is mainly mentioned in a certain language. For example, M-BERT mistakenly predicts "QQ" in the English sentence "Tencent QQ is developed by _.", while the prediction "腾讯" (Tencent) in the corresponding Chinese sentence "腾讯QQ是由_开发的。" is correct. This is probably because Tencent, a Chinese company, is more frequently mentioned in the Chinese training corpus.

6.1 Methods

Inspired by these observations, we propose to use *code-switching* to create data to fine-tune pre-trained LMs, replacing entity mentions in one language (e.g., English/Greek) with their counterparts in another language (e.g., Greek/English). Through this bi-directional code-switching, entity mentions serve as pivots, enabling knowledge that was originally learned in one language to be shared with others. Given a pair of languages, we first identify Wikipedia sentences that mention entities from our benchmark using SLING (Ringgaard et al., 2017). The M-LM is then finetuned on these sentences. Following Wu et al. (2020), with 30% of probability we switch all the entity mentions (can be one or multiple) from the original language to their counterparts in the other language, ending up with sentences like "Ομπάμα later reflected on his years ...", where we substituted "Obama" with a Greek mention of the entity, and vice-versa for Greek-to-English. 70% of the sentences remain the same. If there are multiple mention texts for an entity, we sample proportionally to their frequencies, which we found in our preliminary experiments performed better than using a fixed translation. We fine-tune M-BERT using the masked LM objective on this data, with 15% of non-mention words and 50% of mention words masked out.[8]

6.2 Experimental Results

We choose three languages with different data availability, namely French, Russian, and Greek, and pair them with English, producing 560k, 396k, and 129k code-switched sentences respectively. We compare M-BERT after code-switched fine-tuning (denoted as **cs**) with both the original M-BERT and with fine-tuning only on raw text (**raw**). We vary the evaluation settings to illustrate the effect of code-switching: on top of matching predictions to

[8]The larger ratio on entities encourages the model to focus on predicting entities, as in the downstream task.

Lang.	Method	Single-eval All	Single-eval Single	Single-eval Multi	Double-eval All	Double-eval Single	Double-eval Multi
French	M-BERT	10.21	19.07	3.92	10.67	19.24	4.55
	+raw	**15.06**	**26.81**	**7.40**	15.69	26.92	8.27
	+cs	13.15	24.37	6.34	**16.90**	**26.98**	**10.29**
Russian	M-BERT	1.87	4.58	0.96	3.04	7.72	2.28
	+raw	**7.92**	**24.37**	**3.59**	8.77	**26.28**	4.57
	+cs	7.64	22.41	3.55	**11.69**	25.31	**7.85**
Greek	M-BERT	4.49	20.75	2.19	4.97	20.87	2.83
	+raw	**11.49**	**35.27**	**7.65**	12.65	**35.27**	9.27
	+cs	9.30	26.31	5.73	**18.41**	30.93	**15.30**

Table 4: Accuracy of M-BERT after fine-tuning on raw and code-switched text (%).

ground truth aliases in the prompt language (**single-eval**), we evaluate with targets in both languages (**double-eval**; English and prompt).

As shown in Tab. 4, continued fine-tuning on raw text outperforms the original M-BERT, likely due to our fine-tuning on a subset of sentences with mentions of entities from our benchmark. Results on code-switched text are slightly worse when only matching entities in the original target language, but significantly better if we allow matching in both the original language and English. This indicates that code-switched fine-tuning allows M-BERT to retrieve facts, albeit in English rather than in the prompt language. Encouragingly, the increase is larger for low-resource (Greek) and typologically distant-to-English (Russian) languages. For example, the prediction for the Greek prompt "η Θεωρία κατηγοριών είναι μέρος των _." ("Category theory is part of _.") is "mathematics" (in English!), while the prediction without code-switching is the non-informative "οποίων" ("which"). Considering that we have more raw than code-switched sentences in the dataset, this seems to indicate that English entities are easier to predict than their prompt-language counterparts, which might be because facts expressed in English are better learned in the pre-trained model due to training data abundance.

7 Related Work

Factual Knowledge Retrieval from LMs Several works have focused on probing factual knowledge solely from pre-trained LMs without access to external knowledge. They do so by either using prompts and letting the LM fill in the blanks, which assumes that the LM is a static knowledge source (Petroni et al., 2019; Jiang et al., 2020; Poerner et al., 2019; Bouraoui et al., 2020), or fine-tuning the LM on a set of question-answer pairs to directly generate answers, which dynamically adapts the LM to this particular task (Roberts et al., 2020). Impressive results demonstrated by these works indicate that large-scale LMs contain a significant amount of knowledge, in some cases even outperforming competitive question answering systems relying on external resources (Roberts et al., 2020). Petroni et al. (2020) further shows that LMs can generate even more factual knowledge when augmented with retrieved sentences. Our work builds on these works by expanding to multilingual and multi-token evaluation, and also demonstrates the significant challenges posed by this setting.

Multilingual Benchmarks Many multilingual benchmarks have been created to evaluate the performance of multilingual systems on different natural language processing tasks, including question answering (Artetxe et al., 2020; Lewis et al., 2019; Clark et al., 2020), natural language understanding (Conneau et al., 2018; Yang et al., 2019a; Zweigenbaum et al., 2018; Artetxe and Schwenk, 2019), syntactic prediction (Nivre et al., 2018; Pan et al., 2017), and comprehensive benchmarks covering multiple tasks (Hu et al., 2020; Liang et al., 2020). We focus on multilingual factual knowledge retrieval from LMs, which to our knowledge has not been covered by any previous work.

8 Conclusion

We examine the intersection of multilinguality and the factual knowledge included in LMs by creating a multilingual and multi-token benchmark X-FACTR, and performing experiments comparing and contrasting across languages and LMs. The results demonstrate the difficulty of this task, and that knowledge contained in LMs varies across languages. Future directions include other pre-training or fine-tuning methods to improve retrieval performance and methods that encourage the LM to predict entities of the right types.

Acknowledgements

This work was supported by a gift from Bosch Research. The authors are thankful to the reviewers for the thorough and insightful comments. They are also particularly grateful for everyone who helped create, check, or evaluate the templates and the outputs of our models: Aman Madaan, Aditi Chaudhary, Paul Michel, Sergio Franco, Maria Ryskina, Chan Young Park, Hiroaki Hayashi, Toan Nguyen, David Ifeoluwa Adelani, Bonaventure Dossou, Emre Yolcu, Happy Buzaaba, and Fahim Faisal.

References

Judit Ács. 2019. Exploring bert's vocabulary. Accessed May 2020.

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. 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 984–996, Hong Kong, China. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the Cross-lingual Transferability of Monolingual Representations. In *Proceedings of ACL 2020*.

Mikel Artetxe and Holger Schwenk. 2019. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the ACL 2019*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Zied Bouraoui, Jose Camacho-Collados, and Steven Schockaert. 2020. Inducing relational knowledge from BERT. In *Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, New York, USA.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*.

Kyunghyun Cho. 2019. Bert has a mouth and must speak, but it is not an mrf. Accessed May 2020.

Kenneth Ward Church. 1988. A stochastic parts program and noun phrase parser for unrestricted text. In *Second Conference on Applied Natural Language Processing*, pages 136–143, Austin, Texas, USA. Association for Computational Linguistics.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. In *Transactions of the Association of Computational Linguistics*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 7057–7067.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. 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.

Hady ElSahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon S. Hare, Frédérique Laforest, and Elena Simperl. 2018. T-rex: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA).

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019. Mask-predict: Parallel decoding of conditional masked language models. 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 6112–6121, Hong Kong, China. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080.

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 (TACL)*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 181–184. IEEE.

Yuri Kuratov and Mikhail Arkhipov. 2019. Adaptation of deep bidirectional multilingual transformers for russian language. *CoRR*, abs/1905.07213.

Carolin Lawrence, Bhushan Kotnis, and Mathias Niepert. 2019. Attending to future tokens for bidirectional sequence generation. 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 1–10, Hong Kong, China. Association for Computational Linguistics.

Patrick Lewis, Barlas Oğuz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2019. MLQA: Evaluating Cross-lingual Extractive Question Answering. *arXiv preprint arXiv:1910.07475*.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Bruce Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation. *CoRR*, abs/2004.01401.

Bill Y. Lin, Frank F. Xu, Kenny Q. Zhu, and Seung-won Hwang. 2018. Mining cross-cultural differences and similarities in social media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 709–719. 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.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. Camembert: a tasty french language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Joakim Nivre, Mitchell Abrams, Željko Agić, Lars Ahrenberg, Lene Antonsen, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Mohammed Attia, et al. 2018. Universal dependencies 2.2.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

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

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

Nina Poerner, Ulli Waltinger, and Hinrich Schütze. 2019. E-bert: Efficient-yet-effective entity embeddings for bert. *CoRR*, abs/1911.03681.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Michael Ringgaard, Rahul Gupta, and Fernando C. N. Pereira. 2017. SLING: A framework for frame semantic parsing. *CoRR*, abs/1710.07032.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? *CoRR*, abs/2002.08910.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of ACL 2020*.

Stefan Schweter. 2020. Berturk - bert models for turkish.

Martin Sundermeyer, Hermann Ney, and Ralf Schlüter. 2015. From feedforward to recurrent LSTM neural networks for language modeling. *IEEE ACM Trans. Audio Speech Lang. Process.*, 23(3):517–529.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. Bertje: A dutch BERT model. *CoRR*, abs/1912.09582.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a markov random field language model. *CoRR*, abs/1902.04094.

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

Shijie Wu, Alexis Conneau, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Emerging cross-lingual structure in pretrained language models. In *Proceedings of ACL 2020*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019a. PAWS-x: A cross-lingual adversarial dataset for paraphrase identification. 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 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019b. 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, 8-14 December 2019, Vancouver, BC, Canada*, pages 5754–5764.

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2018. Overview of the third bucc shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of 11th Workshop on Building and Using Comparable Corpora*, pages 39–42.

A Benchmark Details

Tab. 5 shows the detailed number of facts in each language in our X-FACTR benchmark. Fig. 6 demonstrates the ratio of facts with respect to the number of tokens of the object in different languages, where high-resource languages (e.g., English, French, Dutch, and Spanish) have more portion of single-token facts than low-resource languages.

B Benchmark Prompt Quality

The prompts generated in different languages may not be perfectly natural. This could be due to awkwardness of attempting to express relational phrases that were originally devised for English in languages where the semantic distinctions of the underlying words may differ, or due to our errors in our automated approach to grammatical attribute inference and subsequent inflection. To this end, we evaluated our prompts on a sample of languages, providing native speakers with 10 sentences per prompt with the missing slots filled by our inflection models. Our approach produces sentences that are annotated as correct 97.9% of the cases in Spanish, 90.5% in Yoruba, 86.7% in Greek, 82.3% in Marathi, and 81.9% in Russian.

We present an analysis of the annotations on the erroneous prompts in Table 6. The error types differ drastically across languages. Russian and Marathi have comparatively large percentages of inflection-related errors, but for different reasons: the prediction of non-human entity grammatical gender in Russian is difficult and this results in mistakes in the inflection. In Marathi, this issue is also exacerbated by the inflection model, which is of slightly lower quality due to the scarcity of training data availability.

Despite these two outliers, we consider the rest of our prompts to be of high quality. Even if small inflection or grammatical gender assignment mistakes occur (e.g. in Greek) this should not render the prompt unintelligible to native speakers – the burden is on the model to be robust to such slight variations, just as humans are. We point out that the prompts can be awkward or incorrect for some senses captured by the relation, an issue unrelated to our gender heuristics or automatic inflection. This issue, though, is also present in the LAMA English prompts (Petroni et al., 2019; Jiang et al., 2020) and is the result of the original Wikidata annotation.

C Multi-Token Decoding

We outline here the exact concrete formulation of our multi-token decoding algorithms. Given a sentence with multiple mask tokens, e.g., Eq. 2, we can either generate outputs in parallel independently or one at a time conditioned on the previously generated tokens. These methods are similar to the prediction problems that BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019b) perform in their pre-training stages respectively. We define $\boldsymbol{c} \in \mathbb{R}^n$ as the confidence of each prediction, with details varying by prediction method.

C.1 Initial Prediction and Refinement

Independent For independent initial prediction, the mask tokens are all predicted in parallel:

$$\hat{y}_k = \underset{y_k}{\operatorname{argmax}}\, p(y_k|\boldsymbol{s}_{i:j}), c_k = p(\hat{y}_k|\boldsymbol{s}_{i:j}),$$
$$\forall k \in \{i, ..., j\}.$$

We also consider two autoregressive methods for initial prediction or refinement.

Order-based Mask tokens are predicted from left to right, conditioned on previously generated tokens in each step:

$$\hat{y}_i = \underset{y_i}{\operatorname{argmax}}\, p(y_i|\boldsymbol{s}_{i:j}), c_i = p(\hat{y}_i|\boldsymbol{s}_{i:j}).$$

In the refinement stage, we modify the predicted tokens from left to right by replacing the token with a ⟨mask⟩ and re-predicting it:

$$\hat{y}_i = \underset{y_i}{\operatorname{argmax}}\, p(y_i|\hat{\boldsymbol{s}}_{i:j} \setminus i), c_i = p(\hat{y}_i|\hat{\boldsymbol{s}}_{i:j} \setminus i),$$

where $\boldsymbol{s} \setminus i$ means that the i-th token in $\boldsymbol{s}$ is replaced with ⟨mask⟩. Convergence is reached when there are no changes in a left-to-right scan.

Confidence-based Among all the predictions for masked positions, we choose the one with the highest confidence (i.e., the highest probability), so the actual order of predictions can be arbitrary, as shown in Fig. 2:

$$\hat{y}_k = \underset{i \le k \le j, y_k}{\operatorname{argmax}}\, p(y_k|\boldsymbol{s}_{i:j}), c_k = p(\hat{y}_k|\boldsymbol{s}_{i:j}).$$

In the refinement stage, we choose from all predicted tokens the one with the lowest confidence (i.e., the lowest probability) and re-predict it (Ghazvininejad et al., 2019):

$$\hat{y}_k = \underset{y_k}{\operatorname{argmax}}\, p(y_k|\hat{\boldsymbol{s}}_{i:j} \setminus k), c_k = p(\hat{y}_k|\hat{\boldsymbol{s}}_{i:j} \setminus k),$$
$$k = \underset{i \le k \le j}{\operatorname{argmin}}\, c_k.$$

Convergence is reached when the re-predicted token is the same as the original token.

	en	fr	nl	es	ru	ja	zh	hu	he	tr	ko	vi
#facts	45684	40240	38291	37065	26265	25144	23142	20438	17050	16104	16098	13642
#single-word facts	18903	13886	12812	13463	3391	1312	210	6241	1057	2506	1964	3909
#multi-word facts	26781	26354	25479	23602	22874	23832	22932	14197	15993	13598	14134	9733

	el	bn	ceb	mr	war	tl	sw	pa	mg	yo	ilo
#facts	13034	9383	8160	7877	7342	7116	6834	5455	4945	4609	4053
#single-word facts	742	53	3257	199	2981	3208	2840	67	1748	930	2099
#multi-word facts	12292	9330	4903	7678	4361	3908	3994	5388	3197	3679	1954

Table 5: Detailed X-FACTR Benchmark statistics. Languages are ranked by the total number of facts.

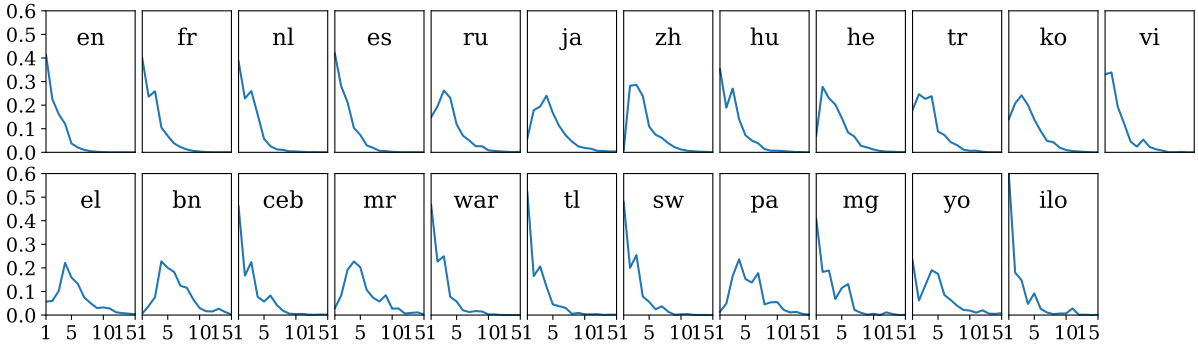


Figure 6: Ratio of facts with respect to the number of tokens of the object in different languages.

		% Errors				
Language	% Correct	Inflection	Gender	Number	Awkward	Wrong Sense
Greek	86.7	5.4	7.4	0.5	5.0	5.0
Spanish	97.9	–	1.6	0.8	1.9	0
Marathi	82.3	15.1	–	0.2	0	4
Russian	81.9	16.1*	–	–	18.1*	6.7
Yoruba	90.5	–	–	–	4.1	0

Table 6: Error analysis on the prompts after instantiating with actual examples. We note that the error categories are not mutually exclusive. *: The Russian inflection percentage includes gender and number errors, unlike the other languages; the Russian annotator also marked all erroneous sentences as “awkward", skewing the results.

C.2 Additional Decoding Components

Length Normalization Since the sum used in § 4.2 might favor short predictions, we consider normalizing it by the number of the mask tokens:

$$v(j-i+1) = \frac{1}{j-i+1}\sum_{k=i}^{j}\log c_k,$$

Confidence Re-computation Note that the confidence of each predicted token $\boldsymbol{c}$ in previous equations is the probability when the token is predicted. However, the probability will become stale once the surrounding tokens change because of the bidirectional conditional distributions, and this is also noted in (Ghazvininejad et al., 2019). To make the confidence up-to-date, given the prompt in Eq. 2, when a new token is predicted (in the initial stage) or a token is modified (in the refinement stage), we re-compute c_i to c_j. This makes the time complexity quadratic to the number of mask tokens, because every time we make a modification, we have to re-compute the confidence values of all predictions. As a result, the final confidence becomes:

$$c_k = p(\hat{y}_k|\hat{\boldsymbol{s}}_{i:j} \setminus k),$$

where $\hat{\boldsymbol{s}}_{i:j} = x_1, ..., \hat{y}_i, ..., \hat{y}_j, ..., x_n$ contains the final predictions.

Beam Search All of the previous methods use the most plausible prediction at each masked position. We also consider performing beam search that keeps track of the most plausible B predictions. Our beam search algorithm is very similar to the case of conventional left-to-right decoding, except that the decoding order might be arbitrary if we use confidence-based initial or refinement prediction methods. As a result, extending different samples in the beam might lead to the same results so we need an additional deduplication step. The time complexity with all the above components is $\mathcal{O}(M^2BT)$, where M is the maximal number of mask tokens, and T is the maximal number of iteration. Alg. 1 outlines the overall multi-token decoding algorithm. The confidence-based decoding method takes 20 minutes to 2 hours on a Nvidia Geforce RTX 2080 Ti GPU depending on the number of facts of each language.

Algorithm 1: Multi-token decoding.

Result: The final sentence $\hat{\boldsymbol{s}}$.
max number of mask tokens M, beam size B, max number of iteration T, an initial sentence $\boldsymbol{s}^{(0)}$;
for *number of mask tokens* $m = 1, ..., M$ **do**
  $\boldsymbol{s}_m^{(0)} \leftarrow$ insert m ⟨mask⟩ tokens in $\boldsymbol{s}^{(0)}$;
  $\mathbf{S} \leftarrow \{\boldsymbol{s}_m^{(0)}\}$;
  for *iteration* $t = 1, ..., T$ **do**
    $\mathbf{S}' \leftarrow \phi$;
    for *each sentence* $\boldsymbol{s}_m^{(t-1)} \in \mathbf{S}$ **do**
      $\{\boldsymbol{s}_m^{(t,b)}\}_{b=1}^{B} \leftarrow$ top B predictions after an initial or refinement step;
      $\mathbf{S}' \leftarrow \mathbf{S}' \cup \{\boldsymbol{s}_m^{(t,b)}\}_{b=1}^{B}$
    end
    $\mathbf{S} \leftarrow$ deduplicate and get the top B from $\mathbf{S}'$;
  end
end
$\hat{\boldsymbol{s}} \leftarrow$ top one from $\mathbf{S}$;

Prompts	Ind.	Best
The capital of India is _.	Rajasthan	New Delhi
The capital of Auvergne is _.	Lyon	Clermont-Ferrand
American League is part of _.	the League	Major League Baseball
First Epistle to Timothy is part of _.	Christianity	the New Testament
KGB is a legal term in _.	KGB	the Soviet Union
Centers for Disease Control and Prevention is a legal term in _.	CDC	the United States

Table 7: Prediction results of M-BERT where the best-performing decoding method makes correct predictions while the independent prediction method does not.

D Details of Pre-trained LMs

LMs examined in this paper share similar architecture and pre-training setting as BERT (Devlin et al., 2019) or RoBERTa (Liu et al., 2019), but are trained on different corpora. We provide the shortcut name of each LM in the HuggingFace's Transformer library (https://huggingface.co/transformers/pretrained_models.html) and their training corpora in Tab. 8, from which you can find more information.

E Detailed Experimental Results

Detailed performance across LMs and languages and error cases in Spanish and Greek are shown in Tab. 10 and Tab. 9 respectively.

Model	Shortcut	Corpus
	multilingual LMs	
M-BERT	bert-base-multilingual-cased	Wikipedia
XLM	xlm-mlm-100-1280	Wikipedia
XLM-R	xlm-roberta-base	CommonCrawl
	monolingual LMs	
BERT (en)	bert-base-cased	BooksCorpus, English Wikipedia
CamemBERT (fr)	camembert-base	French OSCAR$^{\circ}$
BERTje (nl)	bert-base-dutch-cased	Dutch Wikipedia, Books, TwNC*, SoNaR-500†, Web news
BETO (es)	dccuchile/bert-base-spanish-wwm-cased	Spanish Wikipedia, Spanish OPUS$^{\triangleleft}$
RuBERT (ru)	DeepPavlov/rubert-base-cased	Russian Wikipedia, news data
Chinese BERT (zh)	bert-base-chinese	Chinese Wikipedia
BERTurk (tr)	dbmdz/bert-base-turkish-cased	Turkish Wikipedia, Turkish OSCAR, Turkish OPUS, etc
GreekBERT (el)	nlpaueb/bert-base-greek-uncased-v1	Greek Wikipedia, Greek Europarl$^{\diamond}$, Greek OSCAR

Table 8: Shortcut name of each multilingual/monolingual LM in HuggingFace's Transformers library, and their training copora. $^{\circ}$ The OSCAR corpus is extracted from the CommonCrawl corpus. * TwNC is a multifaceted Dutch News Corpus. † SoNaR-500 is a multi-genre Dutch reference corpus. $^{\triangleleft}$ OPUS is a translated text corpus from the web. $^{\diamond}$ Europarl is a corpus of parallel text.

Type	**Prompt**	**Prediction**	**Gold**	**Ratio**
Correct	Vilna y _ son ciudades gemelas.	Minsk	Minsk	*16.68*
Repeating subjects	La capital de Bali es _.	Bali	Denpasar	24.62
Wrong entities	John Goldschmidt es un _ de profesiòn.	comerciant	director de cine	29.07
Non-informativeness	Lionel Heald fue educado en la Universidad de _.	la Universidad	Charterhouse School	9.81
Type errors	Jänta å ja fue creada en _.	2005	Suecia	6.11
Related concepts	Bas Heijne nació en _.	el Reino de Holanda	Nimega	1.67
Unk	Tanaj consiste de _.	:1.2	Torá	8.33
False Negative	BMW S1000RR es producido por _.	BMW	BMW Motorrad	3.52
Inflection	proteína de membrana es una subclase de _.	proteínas	proteína	0.19
Correct	το Καμερούν βρίσκεται στην _.	Αφρική	Αφρική	*12.02*
Repeating subjects	η Λάσα ντε Σέλα δούλευε στην _.	Λάσα ντε Σέλα	Μόντρεαλ	25.06
Wrong entities	η Χένσελ ιδρύθηκε στην _.	Ιταλία	Κάσσελ	18.74
Non-informativeness	ο Πωλ Καρνό δουλεύει στο _.	χωριό	Πανεπιστήμιο ραρισιού	26.78
Related concepts	οι The Kooks ιδρύθηκαν στην _.	Αγγλία	Μπράιτον	1.91
Unk	ο Ραβί Σανκάρ παίζει _.	π	Σιτάρ	11.67
False Negative	το Disneyland ανήκει στο _.	Walt Disney	the Walt Disney Company	3.06
Inflection	ο Χριστός είναι μέρος του _.	Χριστός	Χριστού	0.77

Table 9: Error cases of M-BERT in Spanish and Greek (%).

Model	Decoding	Part	en	fr	nl	es	ru	zh	he	tr	ko	vi	el	mr	yo
M-BERT	Ind.	all	13.57	10.21	**12.42**	**14.30**	1.87	2.50	2.70	2.00	4.08	8.34	4.46	2.76	**3.44**
		single	22.40	19.07	**25.21**	**24.25**	4.58	9.61	7.43	4.50	**21.14**	**17.69**	21.11	12.11	**5.15**
		multi	5.57	3.92	4.42	4.90	0.96	2.22	2.56	1.03	1.61	2.91	2.16	2.18	**3.29**
	Conf.	all	12.00	6.30	8.55	7.47	2.54	6.62	2.92	2.08	4.70	9.20	6.77	**3.46**	3.21
		single	12.91	7.77	12.20	9.13	3.65	5.21	4.33	4.34	16.15	14.60	13.69	8.99	3.87
		multi	**10.08**	4.78	**5.22**	5.11	1.86	6.49	**2.90**	1.19	2.88	5.22	5.72	**3.07**	3.06
XLM	Ind.	all	9.03	7.44	7.53	7.40	2.29	5.83	2.79	1.59	5.33	6.86	7.10	1.26	-
		single	20.74	16.58	18.38	16.44	7.62	**17.12**	11.58	5.53	13.28	12.12	18.03	12.62	-
		multi	4.75	4.03	3.00	3.40	1.40	2.57	1.82	0.50	3.24	3.93	5.16	0.10	-
	Conf.	all	5.30	4.13	4.46	3.18	2.14	3.40	1.93	1.85	5.23	6.26	**7.56**	1.48	-
		single	8.79	6.14	6.48	4.18	3.61	10.44	5.97	4.89	11.15	8.98	13.86	9.76	-
		multi	5.63	3.56	4.06	3.09	2.01	1.38	1.71	1.06	**3.82**	4.38	**6.50**	0.42	-
XLM-R	Ind.	all	8.19	4.70	4.42	6.50	5.26	4.63	2.47	3.09	5.11	8.52	6.28	2.71	-
		single	15.21	11.29	10.95	13.37	**14.41**	11.85	**12.34**	4.04	16.71	14.22	**27.33**	**19.47**	-
		multi	3.32	2.34	2.58	3.29	3.77	4.49	2.18	**2.49**	2.61	5.12	2.94	1.07	-
	Conf.	all	4.43	2.90	2.67	4.33	5.53	5.30	**2.99**	2.95	**5.64**	**9.51**	7.25	3.36	-
		single	5.19	4.38	3.57	4.93	14.15	11.79	11.42	3.93	15.88	12.56	25.60	18.85	-
		multi	3.86	2.33	2.70	4.17	4.12	5.17	2.73	2.43	3.44	**6.97**	4.29	1.97	-
Specific	Ind.	all	**17.92**	**10.36**	9.84	10.94	6.77	5.47	-	**3.36**	-	-	3.00	-	-
		single	**31.21**	**20.30**	19.22	19.07	9.64	3.55	-	**5.88**	-	-	5.53	-	-
		multi	5.88	**4.88**	3.40	6.10	5.50	5.18	-	2.29	-	-	0.92	-	-
	Conf.	all	10.53	6.20	5.18	6.07	**6.80**	**10.07**	-	3.13	-	-	2.49	-	-
		single	19.01	15.50	8.21	5.22	9.22	3.04	-	5.56	-	-	4.08	-	-
		multi	3.44	3.09	3.06	**6.40**	**5.59**	**9.80**	-	2.15	-	-	1.35	-	-

Model	Decoding	Part	ja	hu	bn	ceb	war	tl	sw	pa	mg	ilo
M-BERT	Ind.	all	0.85	2.54	1.33	3.93	2.29	**5.41**	**6.24**	1.91	**3.36**	**1.82**
		single	7.13	**8.31**	**2.39**	**7.13**	4.42	**10.12**	**10.00**	4.35	**4.36**	**3.06**
		multi	0.48	0.62	1.12	0.23	0.42	0.64	2.25	1.48	3.27	0.19
	Conf.	all	1.51	**3.16**	**1.51**	3.94	2.11	4.62	6.02	**2.56**	3.27	1.70
		single	6.50	7.85	1.52	6.30	3.73	7.80	8.42	3.80	3.40	2.41
		multi	1.21	**1.68**	**1.34**	0.64	0.69	1.25	**3.60**	**2.30**	**3.52**	0.24
XLM	Ind.	all	5.77	1.56	0.10	**5.39**	**3.29**	4.36	5.90	-	-	0.13
		single	**24.95**	6.71	1.13	6.98	**5.35**	7.35	8.60	-	-	0.43
		multi	3.04	0.60	0.00	**2.15**	**1.83**	1.36	2.18	-	-	0.00
	Conf.	all	**5.95**	1.87	0.06	4.67	1.57	2.25	4.19	-	-	0.04
		single	18.60	5.49	0.81	4.88	2.17	3.53	5.90	-	-	0.07
		multi	**4.24**	1.34	0.00	2.11	1.08	1.11	2.28	-	-	0.00
XLM-R	Ind.	all	2.30	0.86	0.07	1.35	1.15	2.80	3.66	0.23	1.94	0.11
		single	9.23	2.22	0.00	1.73	1.32	5.05	5.57	5.75	3.70	0.39
		multi	2.07	0.24	0.07	1.03	1.08	**1.42**	1.91	0.00	1.61	0.02
	Conf.	all	4.41	0.86	0.09	1.22	1.14	2.33	2.86	0.58	1.76	0.51
		single	8.82	2.02	0.00	1.39	1.29	4.25	4.34	**5.75**	3.49	0.39
		multi	4.21	0.31	0.10	0.99	1.07	1.28	1.85	0.36	1.45	**0.52**

Table 10: Accuracy on different languages using different LMs (%). We use $M = 5$ mask tokens for en, fr, nl es, vi (on the left) and $M = 10$ mask tokens for the other languages on the right. Best results for each language-part combination are in bold. "-" denotes missing/unsupported models.