BABELEDITS: A Benchmark and a Modular Approach for Robust Cross-lingual Knowledge Editing of Large Language Models

Anonymous ACL submission

Abstract

With Large Language Models (LLMs) becoming increasingly multilingual, effective knowledge editing (KE) needs to propagate edits across languages. Evaluation of the existing methods for cross-lingual knowledge editing (CKE) is limited both w.r.t. edit *effectiveness*: benchmarks do not account for entity aliases and use faulty entity translations; as well as *robustness*: existing work fails to report on LLMs' downstream generation and task-solving abilities after editing. In this work, we aim to (i) maximize the effectiveness of CKE while at the same time (ii) minimizing the extent of downstream model collapse due to the edits. To accurately measure the effectiveness of CKE methods, we introduce BABELEDITS, a new CKE dataset covering 60 languages that combines high-quality multilingual synsets from BabelNet with marker-based translation to ensure entity translation quality. Unlike existing CKE benchmarks, BABELEDITS accounts for the rich variety of entity aliases within and across languages. We then propose BABELREFT, a modular CKE approach based on representation fine-tuning (ReFT) which learns entity-scope ReFT modules, applying them to all multilingual aliases at inference. Our experimental results show that not only is BABELREFT more effective in CKE than state-of-the-art methods, but, owing to its modular design, much more robust against downstream model collapse when subjected to many sequential edits.[1]

1 Introduction

Large Language Models (LLMs) require continuous updates to maintain factual correctness as new information emerges. Knowledge editing (KE) in LLMs aims at injecting new knowledge (i) *efficiently*, i.e., without the need for expensive retraining or continued training of a large model and (ii) *robustly*, i.e., without disrupting the model's language modeling abilities and downstream performance. As LLMs grow increasingly multilingual (Grattafiori et al., 2024; Riviere et al., 2024; Dang et al., 2024), effective multilingual knowledge editing is paramount. In particular, we need the cross-lingual transfer of added or changed knowledge (CKE) (Fierro and Søgaard, 2022; Qi et al., 2023): that a multilingual LLM reflects a fact imparted into it in one language (e.g., *"Kanye West's wife is Bianca Censori"*) in all other languages that it supports (e.g., when answering *"Tko je žena Kanye Westa?"* in Croatian).

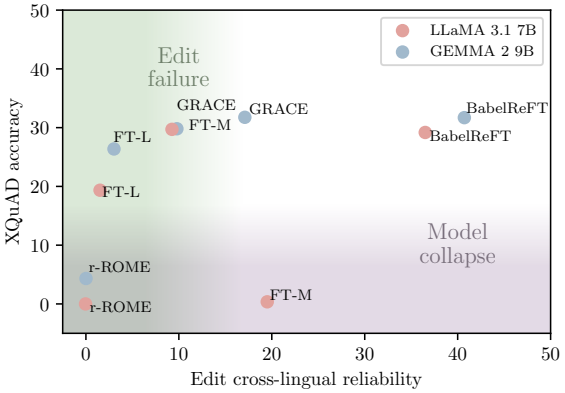

Figure 1: BABELREFT pushes the effectiveness-robustness Pareto-front in sequential CKE. *Effectiveness* refers to reliability of propagation of edits made in one language to other languages on BABELEDITS; *Robustness* denotes LLMs' downstream performance in question answering (on XQuAD) averaged over 4 languages after editing.

The existing body of (C)KE work, however, comes with prominent limitations, especially for proper evaluation of both (1) *effectiveness* of imparting new knowledge into the model and (2) model *robustness* after the edits.

The effectiveness of KE is measured via metrics such as exact match, efficacy score, magnitude (Meng et al., 2022), and rewrite score (Hase et al., 2023), all of which operate on the same formulation of the fact (i.e., precisely the same tokens) as

[1] We release our data and code upon acceptance.

English	MT (Target)	Correct	Error Type
Mortal Kombat	Combattimento Mortale (IT)	Mortal Kombat (IT)	Literal translation
Turkey	Truthahn (DE)	Türkei (DE)	Wrong entity type (animal vs country)
Mountain Dew	Whisky (FR)	Mountain Dew (FR)	Wrong entity entirely
2006	2549 (TH)	2006 (TH)	Wrong translation

Table 1: Samples of entity MT errors (Google Translate)

used for the edit itself. Such evaluation fails to reflect different possible formulations that elicit the same knowledge downstream, including, prominently, entity aliases (e.g., *"Who is Ye's wife?"*). This problem is exacerbated in CKE, where existing evaluation benchmarks are predominantly built by machine-translating English facts (Wang et al., 2024a; Nie et al., 2024; Wang et al., 2024c) and entity mentions: automatically translating entity names with little or no context is particularly error-prone, as illustrated in Table 1.

KE has been shown to harm LLMs' general performance, with even single edits sharply reducing downstream performance. This "model collapse" (Yang et al., 2024b,c) is known to worsen in real-world scenarios involving multiple sequential edits (Gupta et al., 2024a,b; Gu et al., 2024; Li et al., 2024), rendering LLMs virtually useless after editing. While the existing CKE work (Wang et al., 2024a,c) tests the effectiveness of cross-lingual transfer of the edited knowledge, no prior work investigates model collapse in CKE, i.e., how edits in one language affect the LLMs' multilingual abilities, i.e., quality of text generation in other languages as well as effectiveness in downstream tasks.

Contributions. In this work, we simultaneously tackle the aspects of *effectiveness* and *robustness* in multilingual knowledge editing. In contrast to existing work, we aim to (i) maximize the effectiveness of CKE, i.e., propagation of knowledge edits across languages while at the same time (ii) minimizing the extent of model collapse, considering all languages supported by a multilingual LLM.

1) We introduce BABELEDITS, the largest and most multilingual dataset for CKE to date, spanning 60 languages and 13,366 facts. It couples high-quality multilingual synonym sets from BabelNet (Navigli and Ponzetto, 2010; Navigli et al., 2021) with marker-based label projection (Chen et al., 2023) to ensure entity translation quality. Unlike existing CKE benchmarks, it captures the diversity of entity aliases across languages.

2) We propose BABELREFT, a modular CKE approach based on representation fine-tuning (ReFT, Wu et al., 2024) where we (i) learn small entity-scope ReFT modules during editing and (ii) apply a ReFT module of an entity to all its aliases across languages, obtained both from BabelNet and via marker-based translation. Results on two widely used LLMs show that, due to its highly modular nature, BABELREFT avoids the negative interference present in existing CKE approaches and mitigates model collapse effects, proving to be very robust in sequential editing with many edits. Figure 1 shows how BABELREFT pushes the effectiveness-robustness Pareto-front in CKE.

2 Background and Related Work

In the most common task formulation, KE aims to alter a fact provided as a subject-relation-object triple (s, r, o) by replacing o with a new object o', denoted with $(s, r, o \rightarrow o')$, e.g. (*Kanye West*, *wife*, *Kim Kardashian* $\rightarrow$ *Bianca Censori*). A prompt $\pi(s, r)$ formulated from the subject s and predicate r is typically used to impart the new knowledge and test the success of the editing. The prompt $\pi(s, r)$ is effectively asking the LLM to complete the incomplete fact $(s, r, ?)$ (e.g., *"Who is the wife of Kanye West?"*). We refer throughout the paper to the extended prompt $\pi(s, r, o')$ as the prompt $\pi(s, r)$ immediately followed by the new object o'.

We next provide an overview of KE work w.r.t. to the two dimensions of our contributions: methods for imparting new knowledge (§2.1) and KE evaluation metrics and protocols (§2.2). In §2.1, we provide more details for methods that we employ as baselines in our evaluation (see 4).

2.1 Knowledge Editing Methods

Yao et al. (2023) provide a taxonomy of KE methods, where the approaches are divided into *parameter-preserving* and *parameter-altering* methods, indicating whether the method modifies the original parameters of the LLM or not.

Parameter-Altering Methods. These approaches treat KE as any other downstream task for which a subset of the model's weights need to be updated and are further divided into *locate-then-edit* and *meta-learning* approaches.

Locate-then-edit approaches. Most approaches in this category modify only the parameters of the down-projection matrices of the MLP layers, following the observations that these components play

a central role in recalling factual knowledge (Geva et al., 2021, 2022; Dai et al., 2022; Meng et al., 2022; Geva et al., 2023; Chughtai et al., 2024). In the light of this, arguably the simplest approach is to train the down-projection matrix of a specific MLP layer via language modeling on the tokens of the new object for the prompt $\pi(s, r)$, an approach known as **FT-M** (Zhang et al., 2024b). A related variant, known as **FT-L** (Meng et al., 2022), applies an L_∞ norm (i.e., max-norm) on the weight changes and minimizes a different variant of the language modeling loss. **ROME** (Meng et al., 2022) first identifies which MLP to edit using causal mediation analysis (Vig et al., 2020) and then applies a rank-one modification to the MLP layer's down-projection to impart a new fact. MEMIT (Meng et al., 2023) extends ROME to support batch edits, i.e. modifying multiple facts in a single edit step.

Meta-learning approaches typically employ hypernetworks, auxiliary networks that generate weights for the LLM, to learn the necessary weight updates for editing of the LLMs as the main model. Key examples include Knowledge Editor (De Cao et al., 2021) and MEND (Mitchell et al., 2022a).

Model Collapse. Although editing via directly updating the model's weights is effective, prior work has shown that such methods often induce "disabling edits" (Yang et al., 2024b,c), i.e. *single* edits that cause the plummeting of downstream performance to random chance, a phenomenon named "model collapse". In sequential editing, where multiple edits are applied one at a time, several parameter-altering methods were shown to cause model collapse with a few hundred edits (Gupta et al., 2024a,b; Gu et al., 2024; Li et al., 2024).

Parameter-Preserving Methods. The motivation for parameter-preserving KE methods can be manifold: computational efficiency (Zheng et al., 2023), continuous KE (Hartvigsen et al., 2023), or the ability to edit models without having access to its weights (Mitchell et al., 2022b). More importantly and intuitively, avoiding to directly edit the LLMs' parameters reduces the risk of model collapse. Besides simple in-context learning for KE (Zheng et al., 2023), gating approaches constitute the bulk of the approaches in this category. Gating methods store the edited knowledge into separate weights which are activated based on soft routing. The gating (i.e., routing) function can be a dedicated model like a scope classifier in SERAC (Mitchell et al., 2022b) or a simple key-value similarity threshold as in GRACE (Hartvigsen et al., 2023). The additional parameters are, accordingly, a whole separate model (SERAC) or a vector that replaces the activation values at a given layer (GRACE).

With GRACE as the most competitive baseline in our evaluation (§5), we provide further details about it in Appendix A.2.

2.2 Knowledge Editing Evaluation

KE evaluation protocols encompass several dimensions (Zhang et al., 2024b; Yao et al., 2023). *Reliability* reflects whether an edit $(s, r, o \rightarrow o')$ successfully triggers the new answer o' when the model is prompted with $\pi(s, r)$. *Generality* assesses if the edit holds across paraphrases of the original prompt π. *Locality* verifies that unrelated knowledge (s'', r'', o'') remains unchanged after editing. *Portability* evaluates how well the model generalizes from edited knowledge (Cohen et al., 2024). This includes *multi-hop portability*, which tests if the model can reason with the edited fact (e.g., inferring "architect" as Kanye West's wife's profession after editing his wife to be "Bianca Censori"), and *subject-aliasing portability*, checking if the edit applies to alternative subject formulations (e.g., "Kanye West" vs. "Ye").

Cross-lingual Knowledge Editing Given the documented cross-lingual inconsistencies in LLMs' factual knowledge (Fierro and Søgaard, 2022; Qi et al., 2023), multilingual knowledge editing needs methods that enable effective cross-lingual transfer of edits (CKE). Accordingly, several CKE benchmarks have emerged: **BiZsRE** (Wang et al., 2024a), a GPT-4-translated English-Chinese version of ZsRE (Levy et al., 2017); **MzsRE** (Wang et al., 2024c), extending BiZsRE to 10 more languages; and **BMIKE-53** (Nie et al., 2024), which unifies and translates multiple datasets into 53 languages. These benchmarks have primarily been derived by machine-translating both prompts and entities. Entities are translated in isolation, which not only leads to numerous translation errors (as shown in Table 1) but also results in a single translation for each entity (Koehn and Knowles, 2017; Yan et al., 2019; Liang et al., 2024). Focusing on multi-hop portability **MLaKE** (Wei et al., 2025) takes a different approach: they mine parallel fact chains in 5 languages and use ChatGPT to generate prompts, but do not provide test sets for generality, locality, and subject aliasing.

3 Methodology

We first describe the process of creating BABELEDITS, our new benchmark for CKE spanning 60 languages in §3.1 and then our novel modular CKE approach BABELREFT in §3.2. Both leverage BabelNet (Navigli et al., 2021), a massively multilingual knowledge graph in which concepts are represented as multilingual synonym sets (*synsets*).

3.1 BABELEDITS

We utilize BabelNet's graph structure to generate edits and its multilingual synsets to collect entity aliases, which, combined with marker-based machine translation (Yang et al., 2024a), allows us to obtain high-quality fact translations. This enables a (i) more robust evaluation of edits through *subject* aliasing and (ii) acceptance of multiple correct answers thanks to *object* aliases. BABELEDITS comes with multi-parallel prompts in 60 languages, supporting reliability, generality, locality, subject-aliasing and multi-hop portability evaluation.

BABELEDITS Creation. We next describe in detail all steps of the BABELEDITS creation pipeline, which is fully reproducible from our code.

1. Language selection. We start from 50 languages of the popular NLU benchmark XTREME-R (Ruder et al., 2021) as they cover a wide range of scripts and language families. We remove Wolof (WO) as it is currently not supported by Google Translate (GT). We add 11 more languages with more than 500,000 Wikipedia articles (as of Aug '23) and supported by GT, obtaining the final set L of 60 languages, listed in Appendix A.3.

2. Subject extraction. Wikipedia page titles are (often) entity names that we use to query BabelNet to gather subjects for constructing the edit $(s, r, o \rightarrow o')$. For each language-specific Wikipedia, we first retrieve the 30,000 most viewed pages from 2021.[2] We keep only the pages that have a corresponding BabelNet synset with (multi-parallel) senses in all languages in L. We finally sample 20,000 pages (i.e., entities) from each language-specific Wikipedia, ensuring that BABELEDITS is fully balanced across languages.

3. Relation extraction. Having obtained subject synsets, we next collect all relations these synsets have in BabelNet (i.e., labels of all corresponding outgoing edges). From these, we manually select 100 prominent relations (selection criteria provided in Appendix A.4). Finally, we prompt GPT-4o (see Appendix C.4 for the exact prompt) to verbalize each relation r as a template sentence with a slot to be filled with a subject: e.g., for $r = \text{LOCATEDIN}$, we get the template *"Where is* $\langle s \rangle$ *located in?"*.

4. Edit creation. In the next step, we create the edits $(s, r, o \rightarrow o')$, following a procedure similar to that of Cohen et al. (2024). Let S and R be sets of our retrieved synsets and relations, respectively. Each $\sigma \in S$ then becomes the subject s in the edit request: we then look for a relation r from R, a ground truth object o, and a target object o' that cover all languages in L. For a synset $\sigma \in S$, we randomly select from its outgoing edges one relation r to another synset ω that also fully covers our set of languages L. A meaningful edit object o' needs to be of the same category as o. In BabelNet, this generally holds for objects of the same relation r. We thus randomly select a target object synset ω' from the set of all BabelNet edges with r as the relation, i.e. $\{(\sigma', r, \omega') \mid \sigma \neq \sigma', \omega \neq \omega'\}$: the edit is then given as $(\sigma, r, \omega \rightarrow \omega')$. For creating locality sets we sample an additional relation $r_{\text{loc}} \neq r$ and a ground truth object ω_{loc} such that it also covers all languages from L. For multi-hop portability, we start from the target object synset ω' and perform, if possible, a hop in the BabelNet graph via a new relation $r' \neq r$ to another synset ω'', so that we obtain the 2-hop chain $(\sigma, r, \omega', r', \omega'')$. In both cases, we feed the obtained tuples to GPT-4o to create portability prompts, which we detail in Table 12 in Appendix C.4. Finally, for each obtained synset, we collect senses to serve as subject and object aliases. Here, we filter only senses from more trustable sources: Wikipedia, OmegaWiki, WordNet, and OpenMultilingualWordNet (Francis and Kyonghee, 2012) and exclude those obtained via automatic translation and Wiki redirections.

5. Marker-based translation. We resort to marker-based translation with EasyProject (Chen et al., 2023) to translate templated prompts, to easily identify entity spans in the translation. Concretely, we wrap the subject s and object o (in English) of the reliability prompt in special markers, e.g.: *"Which language does* <s>*Leonardo Di Caprio* <\s>*speak?*<o>*Japanese*<\o>*"*. We then translate this markered reliability prompt with GT. The markup in the translation allows us to easily replace the content between <s> and <\s> with the aliases of s from BabelNet. We next feed these

[2] Selecting a more recent year could have potentially yielded entities unseen in pretraining by older LLMs.

aliased prompts to NLLB (Team et al., 2022) to leverage its denoising training to correct possible grammatical errors that arise from the replacement (e.g., gender, article, or case adjustments).

Quality Assessment. The above-described process results in 13,366 samples which we split into training (11,498), validation (480), and test portions (1042). The train-validation-test split is based on the relations r, i.e., there is no overlap between relation sets of any two portions. We manually assess the quality of the obtained BABELEDITS prompts in four target languages: German, Italian, French, and Croatian. We select up to 100 reliability test prompts where our marker-based BABELEDITS translation differs from separately machine-translating each component (subject, object, and prompt) of $\pi(s, r, o')$ as done in prior work. We maximally diversify the set of relations among the selected instances. We then present both translations to the annotators (native speakers) and ask them to indicate a preference between the two (instructions available in Table 8 in the Appendix). The results (detailed in Appendix B.1) show that annotators predominantly—ranging from 59.2% for Croatian to 90.0% for French—prefer our marker-based translations.

3.2 BABELREFT

Improving the effectiveness-robustness Pareto front in CKE requires a method that is (i) modular, as direct editing of model's parameters jeopadizes robustness and (ii) effective in massively multilingual settings, enabling propagation of edits across a wide range of diverse languages. In BABELREFT, we leverage Representation Finetuning (ReFT, Wu et al., 2024), a parameter-efficient finetuning method that modifies hidden representations of only *some tokens*, originally based on their position in the sequence. The standard approach, Low-rank Linear Subspace ReFT (LoReFT), projects hidden representations of selected tokens into a low-dimensional subspace using trainable matrices $\mathbf{R} \in \mathbb{R}^{m \times d}$ and $\mathbf{W} \in \mathbb{R}^{m \times d}$, where d is the dimensionality of $\mathbf{h}$ and $m \ll d$. The transformation (or, as called in ReFT, *intervention*) applied at layer ℓ and token position i is defined as follows:

$$\mathbf{h}_i^\ell \leftarrow \mathbf{h}_i^\ell + \mathbf{R}^T(\mathbf{W}\mathbf{h}_i^\ell + \mathbf{b} - \mathbf{R}\mathbf{h}_i^\ell) \quad (1)$$

LoReFT updates the parameters $\phi = \{\mathbf{R}, \mathbf{W}, \mathbf{b}\}$, while the LLM's parameters remain frozen. $\mathbf{R}$ is a low-rank matrix with orthonormal rows, while $\mathbf{W}$ and $\mathbf{b}$ define an affine transformation of $\mathbf{h}$.

BABELREFT couples ReFT with a lexical gating function: i.e., we do not select tokens that undergo a ReFT transformation based on their position, but rather based on whether the token is part of an entity mention. This allows us to train *entity-scope ReFT modules* and route tokens of an entity being edited, as well as tokens of their translations and aliases through the same ReFT module. For each entity e, we construct a vocabulary V_e that consists of all lexicalizations of the entity in the source language (i.e., the language of the edit) and all target languages: with "lexicalizations" we here refer to the union of all entity translations we obtain with marker-based MT and all senses (i.e., aliases) from BabelNet.

Prior to the forward pass, we search for mentions of entities e by string-matching (with the Aho-Corasick algorithm (Aho and Corasick, 1975)) the input text against the entries in V_s. When a match is found, all tokens of the matched mention are routed through the ReFT intervention of the respective entity e, i.e., the hidden representations of those tokens are modified as follows:

$$\mathbf{h}_i^\ell \leftarrow \mathbf{h}_i^\ell + \mathbf{W}_{2[e]}^T \left(\mathbf{W}_{1[e]}\, \mathbf{h}_i^\ell + \mathbf{b}_{[e]} - \mathbf{W}_{2[e]}\, \mathbf{h}_i^\ell \right) \quad (2)$$

with $\mathbf{W}_{1[e]}, \mathbf{W}_{2[e]} \in \mathbb{R}^{m \times d}$ and $\mathbf{b}_{[e]} \in \mathbb{R}^m$ as trainable parameters of the ReFT module of entity e. Entity-specific ReFT modules prevent negative interference between edits by design, which should provide robustness and prevent model collapse in the face of a larger number of sequential edits. Unlike LoReFT, in BABELREFT we do not use rotation matrices, assuming that enforcing orthogonality could introduce interference between edits in sequential editing. ReFT parameters $\mathbf{W}_{1[e]}$, $\mathbf{W}_{2[e]}$, and $\mathbf{b}_{[e]}$ of an entity e are trained by feeding the extended prompt $\pi(s, r, o')$, where s is a lexicalization of e, into the LLM and minimizing the language modeling loss on the tokens of o'.

4 Experimental Setup

Models. We run single and sequential editing experiments with the instruction fine-tuned variants of Llama 3.1 8B and Gemma 2 9B (cf. Appendix C.3).

Languages. Due to computational constraints, we carry out the evaluation on English and 10 other languages (out of the 60 languages in BABELEDITS): Arabic (AR), German (DE), French (FR), Croatian (HR), Italian (IT), Japanese (JA), Georgian (KA), Burmese (MY), Quechua (QU), and Chinese (ZH).

We manually selected these languages to ensure diversity across linguistic typology, scripts, and "resourcefulness" (Joshi et al., 2020).

KE Methods. We compare BABELREFT against FT-M, FT-L, r-ROME[3], and GRACE. We conduct an exhaustive search for the optimal hyperparameters that maximize average reliability across languages on our validation split of BABELEDITS, considering all combinations of models, methods, and both single and sequential editing.[4] In our experiments, we solely use the test set, as none of the methods require training an auxiliary editor network. We inject the edit using a single reliability prompt in the editing language from the test set. We subsequently evaluate the edited model on all the evaluation dimensions using the prompts for the same edit in all 11 selected languages. In sequential editing, we carry out the evaluation after injecting $n = 100, 250, 500, 1042$ (test set size) edits.

Metrics. We use the 'rewrite and rephrase' scores introduced by Hase et al. (2023) to measure reliability and generality. We adapt these scores for multi-hop and subject-aliasing portability: if an edit has multiple aliases for the same target object, we compute the metric for each and then take the best value. We follow Hoelscher-Obermaier et al. (2023) and use neighborhood KL-divergence (NKL) to evaluate locality.[5] We evaluate the zero-shot downstream multilingual performance of the edited models on two tasks: (1) multiple-choice reading comprehension using Belebele (Bandarkar et al., 2024) and the (2) extractive question answering on the XQuAD dataset (Artetxe et al., 2020). We report the results in terms of accuracy for Belebele and exact match for XQuAD.[6]

5 Results and Discussion

Sequential Editing. Table 3 presents the results of the sequential editing task, where the number of sequentially applied edits successively increases from 100 to the entire test set (1042). We first apply the edit to the model in English. We then test the edited model both on KE in all languages on the BABELEDITS test set and on downstream performance on XQuaD and Belebele.

	EN	FR	IT	JA	MY
FT-M	63.77	25.87	33.78	8.42	0.33
GRACE	99.08	0.38	0.75	0.00	0.00
BABELREFT	98.49	49.86	64.38	19.44	2.17

Table 2: Reliability of three methods with sequential editing in English on the full BABELEDITS dataset using LLaMA 3.2 7B. Results are provided for five languages due to space constraints, full results in Appendix B.2.

BABELREFT demonstrates superior performance across several editing aspects (top half of Table 3), achieving by far the highest scores on reliability, generality, and subject aliasing. GRACE performs better than FT-M on many dimensions but is still very far from achieving the effectiveness of BABELREFT. Near-zero results in other languages largely explain GRACE's low average reliability; GRACE, however, remains highly reliable in English, as shown in Table 2.

Multi-hop portability performance (Appendix B.2) is close to zero across all models, with BABELREFT performing slightly better. On the full dataset, BABELREFT gets a score of 1.27 and 1.64 for LLaMA and Gemma respectively, whereas FT-M holds the second-best score with 0.12 and 0.26. This shows that further research is needed to make models generalize from the imparted edits.

BABELREFT also shows robustness in downstream evaluation (bottom half of Table 3). It preserves the original model's performance, matching GRACE's stability while significantly outperforming other baselines, in particular r-ROME which shows large degradation with as few as 100 edits.

Our downstream evaluation also shows that the choice of downstream task is critical for detecting model collapse. For instance, after 500 sequential edits, FT-M loses only 5 points on Belebele, yet its XQuAD performance plummets to 1.58, clearly indicating a collapse of its generative abilities. This discrepancy reflects the nature of the tasks: Belebele is a multiple-choice QA task where inference simply decodes the answer letter with the highest log-probability, i.e., it does not reflect the generative ability of the models. In contrast, XQuAD requires that the model generates a response containing the actual tokens of the answer. We thus advocate for evaluating downstream performance after KE on free-form generative tasks, as these can detect early signs of model collapse.

We next test if our findings generalize beyond

[3] r-ROME is a re-implementation of ROME that mitigates model collapse (Gupta et al., 2024a).

[4] Details about the procedure and the best hyperparameter configurations are provided in Appendix C.2.

[5] We provide precise formulations for all metrics in Appendix A.1.

[6] Further details about the evaluation and the prompts used can be found in Appendix C.5

	Llama 3.1					Gemma 2				
Edits	**FT-L**	**FT-M**	**r-ROME**	**GRACE**	**BABELREFT**	**FT-L**	**FT-M**	**r-ROME**	**GRACE**	**BABELREFT**
Cross-lingual Knowledge Editing Performance										
↑ *Reliability: edit success*										
100	1.59	21.94	-0.01	9.33	**37.12**	3.16	15.58	0.12	15.51	**42.30**
500	1.59	19.98	-0.02	9.27	**37.48**	2.12	10.26	0.01	16.54	**40.90**
Full	1.52	19.51	-0.04	9.26	**36.51**	3.02	9.79	0.01	17.10	**40.72**
↑ *Generality: edit success over paraphrases*										
100	1.29	19.73	-0.02	0.72	**34.40**	2.35	10.28	0.12	8.30	**39.52**
500	1.48	17.87	-0.02	0.58	**35.15**	1.41	5.99	0.01	8.86	**38.48**
Full	1.71	17.69	-0.04	0.47	**34.25**	2.37	5.76	0.01	9.36	**38.18**
↑ *Subject-Alias: edit success over prompts with a subject alias*										
100	1.49	17.23	-0.01	2.45	**23.08**	1.72	10.30	0.01	9.86	**26.93**
500	0.83	11.05	-0.01	1.47	**28.33**	0.92	5.07	0.00	7.96	**27.95**
Full	0.87	11.93	-0.01	1.32	**27.85**	1.36	4.66	0.00	9.18	**28.81**
Downstream Performance										
↑ *Belebele (accuracy)*										
Original	73.59	73.59	73.59	73.59	73.59	84.68	84.68	84.68	84.68	84.68
100	73.42	73.99	34.89	73.59	73.50	84.48	84.46	24.14	**84.71**	84.59
500	73.79	68.06	28.64	73.56	73.50	84.48	84.38	26.07	**84.71**	84.70
Full	72.92	60.26	22.58	**73.50**	73.39	84.64	84.40	28.62	**84.71**	84.70
↑ *XQuAD (EM)*										
Original	29.60	29.60	29.60	29.60	29.60	31.79	31.79	31.79	31.79	31.79
100	18.07	20.00	0.00	**29.60**	29.60	29.64	29.98	0.13	**31.76**	31.76
500	19.37	1.58	0.00	**29.71**	29.45	29.03	28.78	2.77	**31.76**	31.64
Full	19.35	0.36	0.00	**29.71**	29.18	26.37	29.81	4.33	**31.76**	31.70

Table 3: Comprehensive comparison of cross-lingual knowledge editing (*effectiveness*) and downstream task performance (*robustness*) for Llama 3.1 8B and Gemma 2 9B models for different number of sequential edits in English. Editing metrics are averaged over all target languages and multiplied by 100 for readability. Bold numbers show the best performance for each metric/model combination. Downstream performance is averaged over the target languages: *Original* denotes the model performance before editing. Full results in Appendix B.2.

BABELEDITS by evaluating sequential KE on the MzsRE benchmark (Zhang et al., 2025). Specifically, we perform sequential editing in English across the entire test set (742 edits) and evaluate the results for languages in which MzsRE overlaps with BABELEDITS (DE, EN, FR, ZH). For BABELREFT, we use the subject s from each edit to query BabelNet and incorporate all retrieved senses into the vocabulary V_s. As shown in Table 4, MzsRE results closely mirrors our findings from BABELEDITS: BABELREFT achieves the highest average reliability (with LLama 3.1) without a decline in downstream performance, while other methods fall short either in cross-lingual reliability (GRACE) or provoke model collapse (FT-M).

Single Edits. While sequential editing represents a more realistic use case, single editing is still often used in CKE evaluations. We thus evaluate BABELREFT against the same baselines on the test set of BABELEDITS but this time by performing each edit in the dataset independently. We perform editing in each of the 11 languages in our evaluation set. Since evaluating downstream performance after each edit is computationally prohibitive, we follow Yang et al. (2024b) and use perplexity as a surrogate metric. We compute perplexity variation (Delta PPL) before and after editing on a translated version of their ME-PPL-50 dataset, comprising randomly sampled sentences from widely used corpora such as BookCorpus (Zhu et al., 2015) and ROOTS (Laurençon et al., 2022).

Methods	**avg**	de	en	fr	zh
↑ *Reliability: edit success*					
FT-M	27.53	23.09	64.76	17.52	4.75
GRACE	25.17	0.78	**99.51**	0.40	0.00
BABELREFT	**45.24**	**44.31**	97.23	**34.25**	**5.17**
↑ *XQuAD (EM)*					
Original	29.83	34.71	32.44	-	22.35
FT-M	5.97	4.29	7.56	-	6.05
GRACE	**29.94**	**34.54**	**33.03**	-	**22.27**
BABELREFT	29.75	34.45	32.52	-	**22.27**

Table 4: Sequential editing in English on the MzsRE dataset, showing reliability and XQuAD exact match. Full results in Appendix B.2.

Results in Table 5 show that BABELREFT and GRACE exhibit similarly high reliability with Llama 3.1. However, while GRACE did not show any model collapse in sequential editing, it shows a massive increase in perplexity, particularly in the case of Llama 3.1: this suggests that its gating function is often activated when not necessary, severely

Methods	avg	de	en	fr	zh
↑ *Reliability: edit success*					
FT-M	28.87	39.29	37.10	35.98	26.32
GRACE	**32.58**	**45.33**	**46.19**	**40.65**	24.42
BABELREFT	30.96	43.65	37.48	39.02	**27.64**
↓ *Delta PPL*					
FT-M	79.52	40.24	3.37	16.05	5.42
GRACE	5.46e4	5.09e4	8.65e4	3.61e4	1.02e5
BABELREFT	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**

Table 5: Reliability and variation of perplexity for single edits with Llama 3.1 8B. Each column (except 'avg') corresponds to an editing language, and the results are averaged across all the target languages. Column 'avg' averages those results. Full results in Appendix B.2

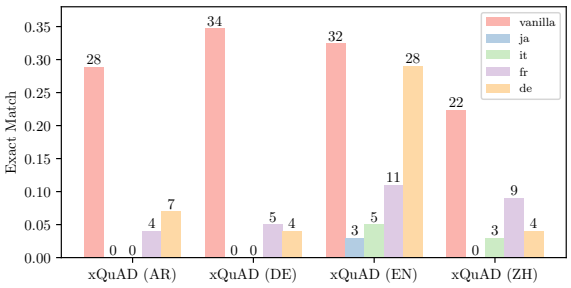

Figure 2: Average EM on XQuAD of the top-5 most destructive edits in terms of perplexity.

damaging the LLM's generative capabilities.

Altough single editing can be seen as an unrealistic (i.e., *in vitro*) use-case, BABELREFT remains competitive, still providing the best solution when considering both editing effectiveness and downstream robustness.

Gating Scope. We empirically observe that the failure of GRACE in either transferring edits across languages (sequential editing) or causing model collapse (single edits) stems from its difficulty to balance precision and recall of its gating activation. In sequential editing, the clusters get gradually smaller as edits are injected hence making the gating function seldom activate (precision over recall). This, coupled with the limited cross-lingual semantic alignment of LLMs' representations, explains the negligible edit transfer. This would also explain the higher reliability of GRACE with Gemma 2, given Gemma's better cross-lingual alignment compared to Llama 3.1 (Kargaran et al., 2025). In the case of single edits, there is only one cluster with a large fixed radius, which is promoted by the hyperparameter selection procedure that aims to maximize the edit transfer across languages (i.e., recall over precision). This, however, makes the gating function fire on almost every input, causing the model collapse and rendering the model useless for downstream tasks.

Cross-Lingually Disabling Edits. Gupta et al. (2024b) have shown that a single disabling edit can completely disrupt the model's downstream abilities. To the best of our knowledge, we are the first to observe *cross-lingually disabling edits*, i.e., that edits in one language compromise the performance across languages. To shed more light on this phenomenon, we compute for all target languages the top 5 most destructive edits, i.e. those that cause the highest increase of perplexity, across all possible editing languages. Figure 2 illustrates the effect of cross-lingually disabling edits for FT-M applied to Llama 3.1 (as FT-M performed overall comparably to GRACE but with less perplexity variation) for pairs of edit-test languages. We observe, e.g., that a single edit in Japanese disables the model for many languages, whereas a single edit in German reduces performance for English much less than for other languages. The latter is particularly insidious, as editing in some languages can collapse the model only w.r.t. some other languages, which can be difficult to detect for model users.

6 Conclusion

Knowledge Editing (KE) shows promise for maintaining LLM factual accuracy, but faces limitations, especially in cross-lingual contexts both from the evaluation (data quality) and methodological perspective (model collapse). Our benchmark, BABELEDITS, addresses the limitations of previous research by offering diverse, high-quality entity representations obtained using BabelNet synsets and marker-based translation. Our modular approach, BABELREFT, couples entity-scope ReFT modules that activate only when necessary using BabelNet synsets as "multilingual keys", achieving CKE *effectiveness* (through wide coverage) and model *robustness* (avoiding model collapse). This prevents indiscriminate gate activation or non-existing cross-lingual edit-transfer, displayed by competing methods such as GRACE in single edits and sequential editing, respectively. We find that cross-lingual multi-hop portability is challenging for all methods, including BABELREFT. Future work could further exploit multilingual knowledge graphs like BabelNet to address this limitation, extending existing monolingual approaches (Zhang et al., 2024a).

7 Limitations

Choice of baselines. Our experiments compare against four baselines: FT-L, FT-M, r-ROME, and GRACE. More baselines could have been used but we chose to keep baselines that are the most relevant to our discourse. First, we used fine-tuning baselines (FT-L and FT-M) because they are the simplest baselines we could find. Then we chose r-ROME and GRACE as competitive baselines representing parameter-altering and parameter-preserving methods respectively.

r-ROME (Gupta et al., 2024a) was selected among all the parameter-altering approaches because it was explicitly designed to avoid model collapse, while most methods in the same category are detrimental to downstream performance (Li et al., 2024), including MEMIT, PMET, MEND, and KN. Moreover, we discarded all meta-learning approaches like MEND because they require additional training data to train the hypernetwork that can then be applied to new unseen edits. While this paper provides such a training set through the BABELEDITS dataset, meta-learning methods are deemed out-of-scope for our work, since they are not directly comparable to other methods.

GRACE is chosen among other parameter-preserving approaches for similar reasons: it aims to avoid model collapse, while other methods were often proposed for different purposes. For example, SERAC was proposed for editing a model without access to its weights (Mitchell et al., 2022b), and IKE aims at compute-efficiency (Zheng et al., 2023). Moreover, we discard in-context learning approaches because it is unclear how they should be applied to downstream tasks. More importantly, while IKE is compute-efficient for a single edit, performing thousands of edits would require a larger prompt that might exceed the context window or render inference latency impractical.

Choice of models. We evaluate BABELREFT and the baselines with two relatively small models: LLaMA 3.1 7B Instruct and Gemma 9B Instruct. Models of this size were selected for practical reasons. Instruct versions of the models were chosen over base ones because they are expected to perform better on downstream evaluation. Finally, this work focuses on English-centric models, while it could have been tested on more multilingual models like Aya or Bloom. Nevertheless, English-centric models are still widely used, even in a multilingual context. While our work focuses on the editing method rather than the model itself, future work that attempts to get the most accurate edited multilingual model might need to rely on larger and explicitly multilingual models.

Choice of languages The proposed BABELEDITS dataset contains 60 languages and improves upon previous datasets which contain at most 53 languages (Nie et al., 2024). BABELEDITS includes several low-resource languages, with namely 9 languages among the class 1 from Joshi et al. (2020) (the "scrapping-bys"). In contrast, the only absent class is class 0, for obvious reasons since it contains languages with virtually no unlabelled data available.

BABELREFT and the compared baselines are not evaluated on all 60 languages, but only on a subset of 10 languages due to computational constraints. However, those 10 languages were selected before the experiments to obtain diversity in scripts, language families, and degrees of resourcefulness.

8 Ethical considerations

Like any other knowledge editing method, the proposed BABELREFT method can be used for harmful purposes. Since it injects new knowledge into an existing LLM, it can be used to propagate false information. While the KE methods still seem to be in their infancy, they might not directly threaten access to information. But if and when KE methods become production-ready, they could help make LLM more accurate just as well as inject harmful false information.

Cross-lingual knowledge editing also presents an opportunity to bridge some gaps in information access across languages. LLMs can have factual inconsistencies across languages (Fierro and Søgaard, 2022; Qi et al., 2023), and CKE could help address that. However, there is also a chance that KE techniques could uniformize information across languages to a point where cultural exception is suppressed. While this paper is still far from posing such a threat, we advocate that all researchers involved in knowledge editing keep this ethical consideration in mind.

References

Alfred V. Aho and Margaret J. Corasick. 1975. Efficient string matching: an aid to bibliographic search. *Commun. ACM*, 18(6):333–340.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.

Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sid Black, Jordan Clive, Anthony DiPofi, Julen Etxaniz, Benjamin Fattori, Jessica Zosa Forde, Charles Foster, Mimansa Jaiswal, Wilson Y. Lee, Haonan Li, Charles Lovering, Niklas Muennighoff, Ellie Pavlick, Jason Phang, Aviya Skowron, Samson Tan, Xiangru Tang, Kevin A. Wang, Genta Indra Winata, Franccois Yvon, and Andy Zou. 2024. Lessons from the trenches on reproducible evaluation of language models. *ArXiv*, abs/2405.14782.

Yang Chen, Chao Jiang, Alan Ritter, and Wei Xu. 2023. Frustratingly easy label projection for cross-lingual transfer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5775–5796, Toronto, Canada. Association for Computational Linguistics.

Bilal Chughtai, Alan Cooney, and Neel Nanda. 2024. Summing up the facts: Additive mechanisms behind factual recall in llms. *ArXiv*, abs/2402.07321.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298.

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

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, et al. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*.

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

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

Bond Francis and Paik Kyonghee. 2012. A survey of wordnets and their licenses. In *Proceedings of the 6th Global WordNet Conference, Matsue, Japan*. Tribun EU. Gebeurtenis: GWC2012.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, Singapore. Association for Computational Linguistics.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, and et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Jia-Chen Gu, Hao-Xiang Xu, Jun-Yu Ma, Pan Lu, Zhen-Hua Ling, Kai-Wei Chang, and Nanyun Peng. 2024. Model editing harms general abilities of large language models: Regularization to the rescue. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16801–16819, Miami, Florida, USA. Association for Computational Linguistics.

Akshat Gupta, Sidharth Baskaran, and Gopala Anumanchipalli. 2024a. Rebuilding ROME : Resolving model collapse during sequential model editing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21738–21744, Miami, Florida, USA. Association for Computational Linguistics.

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. 2024b. Model editing at scale leads to gradual and catastrophic forgetting. In *Findings of*

the Association for Computational Linguistics: ACL 2024, pages 15202–15232, Bangkok, Thailand. Association for Computational Linguistics.

Thomas Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. 2023. Aging with grace: Lifelong model editing with discrete key-value adaptors. In *Advances in Neural Information Processing Systems*.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. 2023. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 17643–17668. Curran Associates, Inc.

Jason Hoelscher-Obermaier, Julia Persson, Esben Kran, Ioannis Konstas, and Fazl Barez. 2023. Detecting edit failures in large language models: An improved specificity benchmark. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11548–11559, Toronto, Canada. Association for Computational Linguistics.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Amir Hossein Kargaran, Ali Modarressi, Nafiseh Nikeghbal, Jana Diesner, François Yvon, and Hinrich Schuetze. 2025. Mexa: Multilingual evaluation of english-centric LLMs via cross-lingual alignment.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, Jörg Frohberg, Mario Šaško, Quentin Lhoest, Angelina Mcmillan-Major, Gérard Dupont, Stella Biderman, Anna Rogers, Loubna Ben Allal, Francesco de Toni, Giada Pistilli, Olivier Nguyen, Somaieh Nikpoor, Maraim Masoud, Pierre Colombo, Javier de la Rosa, Paulo Villegas, Tristan Thrush, Shayne Longpre, Sebastian Nagel, Leon Weber, Manuel Romero Muñoz, Jian Zhu, Daniel van Strien, Zaid Alyafeai, Khalid Almubarak, Vu Minh Chien, Itziar Gonzalez-Dios, Aitor Soroa, Kyle Lo, Manan Dey, Pedro Ortiz Suarez, Aaron Gokaslan, Shamik Bose, David Ifeoluwa Adelani, Long Phan, Hieu Tran, Ian Yu, Suhas Pai, Jenny Chim, Violette Lepercq, Suzana Ilić, Margaret Mitchell, Sasha Luccioni, and Yacine Jernite. 2022. The BigScience ROOTS Corpus: A 1.6TB Composite Multilingual Dataset. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, New Orleans, United States.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada. Association for Computational Linguistics.

Qi Li, Xiang Liu, Zhenheng Tang, Peijie Dong, Zeyu Li, Xinglin Pan, and Xiaowen Chu. 2024. Should we really edit language models? on the evaluation of edited language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Tian Liang, Xing Wang, Mingming Yang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024. Addressing entity translation problem via translation difficulty and context diversity. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11628–11638, Bangkok, Thailand. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, volume 35, pages 17359–17372. Curran Associates, Inc.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022a. Memory-based model editing at scale. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 15817–15831. PMLR.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022b. Memory-based model editing at scale. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 15817–15831. PMLR.

Roberto Navigli, Michele Bevilacqua, Simone Conia, Dario Montagnini, Francesco Cecconi, et al. 2021. Ten years of babelnet: A survey. In *IJCAI*, pages 4559–4567. International Joint Conferences on Artificial Intelligence Organization.

Roberto Navigli and Simone Paolo Ponzetto. 2010. BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 216–225, Uppsala, Sweden. Association for Computational Linguistics.

Ercong Nie, Bo Shao, Zifeng Ding, Mingyang Wang, Helmut Schmid, and Hinrich Schütze. 2024. Bmike-53: Investigating cross-lingual knowledge editing with in-context learning. *Preprint*, arXiv:2406.17764.

Jirui Qi, Raquel Fernández, and Arianna Bisazza. 2023. Cross-lingual consistency of factual knowledge in multilingual language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10650–10666, Singapore. Association for Computational Linguistics.

Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, and et al. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10215–10245, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. 2020. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, volume 33, pages 12388–12401. Curran Associates, Inc.

Jiaan Wang, Yunlong Liang, Zengkui Sun, Yuxuan Cao, Jiarong Xu, and Fandong Meng. 2024a. Cross-lingual knowledge editing in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11676–11686, Bangkok, Thailand. Association for Computational Linguistics.

Peng Wang, Ningyu Zhang, Bozhong Tian, Zekun Xi, Yunzhi Yao, Ziwen Xu, Mengru Wang, Shengyu Mao, Xiaohan Wang, Siyuan Cheng, Kangwei Liu, Yuansheng Ni, Guozhou Zheng, and Huajun Chen. 2024b. EasyEdit: An easy-to-use knowledge editing framework for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 82–93, Bangkok, Thailand. Association for Computational Linguistics.

Weixuan Wang, Barry Haddow, and Alexandra Birch. 2024c. Retrieval-augmented multilingual knowledge editing. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 335–354, Bangkok, Thailand. Association for Computational Linguistics.

Zihao Wei, Jingcheng Deng, Liang Pang, Hanxing Ding, Huawei Shen, and Xueqi Cheng. 2025. MLaKE: Multilingual knowledge editing benchmark for large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4457–4473, Abu Dhabi, UAE. Association for Computational Linguistics.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. 2024. ReFT: Representation finetuning for language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Omry Yadan. 2019. Hydra - a framework for elegantly configuring complex applications. Github.

Jinghui Yan, Jiajun Zhang, JinAn Xu, and Chengqing Zong. 2019. The impact of named entity translation for neural machine translation. In *Machine Translation*, pages 63–73, Singapore. Springer Singapore.

Haoran Yang, Deng Cai, Huayang Li, Wei Bi, Wai Lam, and Shuming Shi. 2024a. A frustratingly simple decoding method for neural text generation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 536–557, Torino, Italia. ELRA and ICCL.

Wanli Yang, Fei Sun, Xinyu Ma, Xun Liu, Dawei Yin, and Xueqi Cheng. 2024b. The butterfly effect of model editing: Few edits can trigger large language models collapse. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5419–5437, Bangkok, Thailand. Association for Computational Linguistics.

Wanli Yang, Fei Sun, Jiajun Tan, Xinyu Ma, Du Su, Dawei Yin, and Huawei Shen. 2024c. The fall of ROME: Understanding the collapse of LLMs in model editing. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4079–4087, Miami, Florida, USA. Association for Computational Linguistics.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu

Zhang. 2023. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10222–10240, Singapore. Association for Computational Linguistics.

Mengqi Zhang, Xiaotian Ye, Qiang Liu, Pengjie Ren, Shu Wu, and Zhumin Chen. 2024a. Knowledge graph enhanced large language model editing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22647–22662, Miami, Florida, USA. Association for Computational Linguistics.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. 2024b. A comprehensive study of knowledge editing for large language models. *Preprint*, arXiv:2401.01286.

Xue Zhang, Yunlong Liang, Fandong Meng, Songming Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2025. Multilingual knowledge editing with language-agnostic factual neurons. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5775–5788, Abu Dhabi, UAE. Association for Computational Linguistics.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. 2023. Can we edit factual knowledge by in-context learning? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4862–4876, Singapore. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books . In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27, Los Alamitos, CA, USA. IEEE Computer Society.

A Additional Methodology

A.1 Evaluation Metrics

We report the formulations of rewrite score (RS), paraphrase score (PS), and portability score (PoS) used in our study.

$$RS = \frac{p_{\theta^*}(o'|\pi_{rel}(s,r)) - p_\theta(o'|\pi_{rel}(s,r))}{1 - p_\theta(o'|\pi_{rel}(s,r))} \tag{3}$$

$$PS = \frac{p_{\theta^*}(o'|\pi_{gen}(s,r)) - p_\theta(o'|\pi_{gen}(s,r))}{1 - p_\theta(o'|\pi_{gen}(s,r))} \tag{4}$$

$$PoS = \frac{p_{\theta^*}(o'|\pi_{port}(s,r)) - p_\theta(o'|\pi_{port}(s,r))}{1 - p_\theta(o'|\pi_{port}(s,r))} \tag{5}$$

where p_{θ^*} is the output distribution of the edited model and p_θ is the output distribution of the original model. For locality, we use the neighborhood KL-divergence score over a locality prompt:

$$NKL = \sum_{o'_1,\ldots,o'_m} p_\theta(o'_i|\pi_{loc}(s,r,o'_{:i})) \log \frac{p_\theta(o'_i|\pi_{loc}(s,r,o'_{:i}))}{p_{\theta^*}(o'_i|\pi_{loc}(s,r,o'_{:i}))} \tag{6}$$

where o'_i is the i-th token of the object o' and $\pi_{loc}(s,r,o'_{:i})$ is the locality prompt truncated at the i-th token.

A.2 GRACE

GRACE maintains a codebook (at a specific layer), which stores key-value pairs with keys being cached activations and values learned hidden state vectors that modify the model's behavior. If a hidden state $\mathbf{h}^{\ell-1}$ falls into the ball of radius ε_i centered on a key k_i in a set of stored keys $\mathbb{K}$, then the corresponding value $v_i \in \mathbb{V}$, which is learned through backpropagation, will replace it (where $d(\cdot)$ is some distance function):

$$\mathbf{h}^\ell = \begin{cases} v_i & \text{if } \exists (k_i, v_i) \in \mathbb{K} \times \mathbb{V} \\ & \text{s.t. } d(\mathbf{h}^{\ell-1}, k_i)) < \varepsilon_i \\ f^\ell(\mathbf{h}^{\ell-1}) & \text{otherwise} \end{cases} \tag{7}$$

As new edits come in, the codebook is updated mostly by shrinking existing radii so that the edits do not interfere. However, as we show in 5 GRACE's efficacy in cross-lingual KE highly depends on the sensitive choice of the initial cluster radius $\varepsilon_{\text{init}}$. The gating function should activate only on the edited prompt and its semantic equivalents across languages and not semantically related entities within a language: this, however, is difficult to achieve due to the limited semantic alignment of LLM hidden representations across languages (Kargaran et al., 2025).

A.3 Language selection

We report the languages included in our benchmark in Table 6.

A.4 Relationship selection

For constructing BABELEDITS, we sample the 200 most frequent relationships that we could extract from Wikipedia and BabelNet such as to perform a manual selection of the 100 most adequate of them. We removed relationships that had the following issues:

- Relationships that can have many different answers, like SEMANTICALLY_RELATED (Alma mater, SEMANTICALLY_RELATED, Mean Girls 2) or INSTANCE_OF (1672, INSTANCE_OF, Calendar year)
- Relationships which do not make sense when edited. for examples, if the subject and the object are similar like GIVEN_NAME (Miklós Horthy, GIVEN_NAME, Miklós)
- Relationships that are very specific to a given field, like PARENT_TAXON (Coronaviridae, PARENT_TAXON, Nidovirales)
- Relationships that reflects the structure of Wikipedia or BabelNet rather than the actual world (9/11, WIKIMEDIA_OUTLINE, Outline of the September 11 attacks)

B Additional Results

B.1 Translation quality assessment

We manually compared the quality of our entities translated with the EasyProject method (Chen et al., 2023) with entities directly translated with Google Translate. Since four of the authors speak a different language, we randomly sample up to 100 translations from the test set in each language: German, Italian, French, and Croatian.

For each annotator, the 100 pairs of translations were randomly inverted to make it impossible to guess which one is the raw translation and which one is the result of applying EasyProject.

Language	ISO 639-1 code	# Wikipedia articles (in millions)	Class in Joshi et al. (2020)	Script	Language family
Afrikaans	af	0.09	3	Latin	IE: Germanic
Arabic	ar	1.02	5	Arabic	Afro-Asiatic
Azerbaijani	az	0.18	1	Latin	Turkic
Belarussian	be	0,43	3	Cyrillic	IE: Slavic
Bulgarian	bg	0.26	3	Cyrillic	IE: Slavic
Bengali	bn	0.08	3	Brahmic	IE: Indo-Aryan
Catalan	ca	1.70	4	Latin	IE: Romance
Czech	cs	1.57	4	Latin	IE: Slavic
Danish	da	0.79	3	Latin	IE: Germanic
German	de	2.37	5	Latin	IE: Germanic
Greek	el	0.17	3	Greek	IE: Greek
English	en	5.98	5	Latin	IE: Germanic
Spanish	es	1.56	5	Latin	IE: Romance
Estonian	et	0.2	3	Latin	Uralic
Basque	eu	0.34	4	Latin	Basque
Persian	fa	0.7	4	Perso-Arabic	IE: Iranian
Finnish	fi	0.47	4	Latin	Uralic
French	fr	2.16	5	Latin	IE: Romance
Gujarati	gu	0.03	1	Brahmic	IE: Indo-Aryan
Hebrew	he	0.25	3	Jewish	Afro-Asiatic
Hindi	hi	0.13	4	Devanagari	IE: Indo-Aryan
Croatian	hr	0.54	4	Latin	Slavic
Haitian Creole	ht	0.06	2	Latin	Creole
Hungarian	hu	0.46	4	Latin	Uralic
Armenian	hy	0.89	1	Armenian alphabet	IE: Armenian
Indonesian	id	0.51	3	Latin	Austronesian
Italian	it	1.57	4	Latin	IE: Romance
Japanese	ja	1.18	5	Ideograms	Japonic
Javanese	jv	0.06	1	Brahmic	Austronesian
Georgian	ka	0.13	3	Georgian	Kartvelian
Kazakh	kk	0.23	3	Arabic	Turkic
Korean	ko	0.47	4	Hangul	Koreanic
Lithuanian	lt	0.2	3	Latin	IE: Baltic
Malayalam	ml	0.07	1	Brahmic	Dravidian
Marathi	mr	0.06	2	Devanagari	IE: Indo-Aryan
Malay	ms	0.33	3	Latin	Austronesian
Burmese	my	0.05	1	Brahmic	Sino-Tibetan
Dutch	nl	1.99	4	Latin	IE: Germanic
Norwegian	no	1.53	1	Latin	IE: Germanic
Punjabi	pa	0.04	2	Brahmic	IE: Indo-Aryan
Polish	pl	1.44	4	Latin	IE: Slavic
Portuguese	pt	1.02	4	Latin	IE: Romance
Cusco Quechua	qu	0.02	1	Latin	Quechuan
Romanian	ro	0.42	3	Latin	IE: Romance
Russian	ru	1.58	4	Cyrillic	IE: Slavic
Slovak	sk	0.57	3	Latin	IE: Slavic
Swedish	sv	6.21	4	Latin	IE: Germanic
Serbian	sr	3.73	4	Serbian Cyrillic	IE: Slavic
Swahili	sw	0.05	2	Latin	Niger-Congo
Tamil	ta	0.12	3	Brahmic	Dravidian
Telugu	te	0.07	1	Brahmic	Dravidian
Thai	th	0.13	3	Brahmic	Kra-Dai
Tagalog	tl	0.08	3	Brahmic	Austronesian
Turkish	tr	0.34	4	Latin	Turkic
Ukrainian	uk	1.06	3	Cyrillic	IE: Slavic
Urdu	ur	0.15	3	Perso-Arabic	IE: Indo-Aryan4
Uzbek	uz	0.52	3	Latin	Turkic
Vietnamese	vi	1.24	4	Latin	Austro-Asiatic
Yoruba	yo	0.03	2	Arabic	Niger-Congo
Mandarin	zh	1.09	5	Chinese ideograms	Sino-Tibetan

Table 6: Languages composing the BABELEDITS dataset. Languages in bold are the ones used for evaluation.

We report the results of the translation quality assessment in Table 7 and the annotator instructions in Table 8.

B.2 Additional results

The following additional results can be found at the end of the Appendix:

- Extended results with all evaluation aspects and number of sequential edits in English of Llama 3.1 and Gemma 2 in Table 14.
- Detail of evaluation metrics on each target language after sequential editing in English on the full BABELEDITS dataset with Llama 3.1 in Table 15
- Detail of evaluation metrics on each target language after sequential editing in English on the full BABELEDITS dataset with Gemma 2 in Table 16
- Evaluation of sequential editing in English of Llama and Gemma 2 on the MzsRE dataset (Zhang et al., 2025) on the languages that intersect with our evaluation set (DE, EN, FR, ZH) in Table 17.
- Results for single editing on LLaMa 3.1 in Table 18.
- Results for single editing on Gemma 2 in Table 19.

C Additional Experimental Details

C.1 Computing resources

We perform all of our editing experiments using the `EasyEdit` library (Wang et al., 2024b) on a single NVIDIA A6000/A100/A40 GPU (40 or 48 GB) using `bfloat16` precision. Each run takes between 5 to 20 hours: we estimate our editing experiments to have required circa 2,250 GPU hours.

C.2 Hyperparameter Selection

The Hyperparameter selection was done for each method using 100 randomly sampled parameter sets. The editing is always done in English and the validation criterion is the average reliability across languages.

To pick the hyperparameters we perform a grid search for each method/model/{single, sequential}-editing combination using a random sample of 100 edits from the validation split of BABELEDITS. We perform the editing in English and use average reliability across languages as a validation criterion. We search over the following grids:

- FT-L: Layers: all, Learning Rate: $\{1e\text{-}4, 5e\text{-}4\}$, Norm Constraint: $\{2e\text{-}3, 1e\text{-}4, 2e\text{-}5\}$
- FT-M: Layers: all, Learning Rate: $\{1e\text{-}4, 5e\text{-}4\}$
- r-ROME: Layers: all, KL Factor: $\{0.0625, 0.9, 1\}$,
- GRACE: Layers: all, Learning Rate: $\{0.1, 1.0\}$, Replacement: $\{\text{last}, \text{all}\}$, $\varepsilon_{\text{init}}$: $\{0.1, 1.0, 100\}$
- BABELREFT: Layers: all, Learning Rate: $\{1e\text{-}4, 1e\text{-}3, 2e\text{-}3\}$, Low-rank dimensionality: $\{4, 16, 64\}$

For FT-L, Norm Constraint indicates the L_∞ norm constraint. For r-ROME, KL factor indicates the weight of the KL term in the v optimization term. For GRACE, replacement indicates whether the replaced hidden states are all or just the one at the last token position. The best-found hyperparameters are in Table 9.

C.3 Models Used

We report in Table 10 the models used in our study together with their Huggingface Hub links for download.

C.4 Prompt for GPT-4-based template prompt creation

To verbalize these relations into usable prompts, we provide GPT-4o with the relation r and an example of subject s and o from the previously extracted synsets and ask it to provide a template prompt $\pi(\langle s \rangle, r)$ to be later filled with the appropriate subject. For example, for the relation $r = \text{LOCATEDIN}$, then the GPT-4o output was $\pi(\langle s \rangle, r) =$ *Where is* $\langle s \rangle$ *located in?*. We additionally ask GPT-4o to generate a rephrased version of $\pi(\langle s \rangle, r)$ to create the generality set of BABELEDITS.

We report in Table 11 the full prompt used to ask GPT-4o to create template prompts and rephrase template prompts for BABELEDITS. In Table 12 we present the prompt used to have GPT-4o generate the multi-hop portability prompts.

Language	Preference Ratio (%)	Annotation Size	Different Prompts
Italian	89.0%	100	512
French	90.0%	100	429
German	81.9%	83	193
Croatian	59.2%	71	133

Table 7: Results of the translation quality assessment for 4 languages. Different prompts indicates the number of prompts in the test set for which the extended prompts $\pi(s, r, o')$ obtained with MT applied separately to subject, object and prompt and our markered translations obtained with EasyProject differ.

```
You will be presented with a prompt for a knowledge editing task in the English
Language.  Together with that, you will be provided with two translations
under the column labelled 'A' and 'B'.
Your task is to express a preference for one of the two translations. Compare
the English prompt with the one in your mother tongue and choose the one
between the options 'A' and 'B' which sounds more correct to you both in terms
of how grammatical it is and how well the subject and object are translated.
You must always express a preference. If you are unsure about the nature of
the subject and object of the prompt, you can find Babelnet links to both
in the two columns titled 'BabelNet Subject URL' and 'BabelNet Object URL'.
Simply write A or B in capital letters in the column titled 'Preference'.
```

Table 8: Task descriptions for the annotators which were asked to select between one of the two possible translations of the English prompt.

C.5 Prompts used for downstream evaluation

We evaluate downstream performance using the `lm-eval` library (Biderman et al., 2024), in a zero-shot fashion on the intersection of our 10 languages and the languages, in the Belebele benchmark (all but QU) and XQuAD dataset (AR, DE, EN, ZH).

The prompts used for downstream evaluation for the two downstream tasks (Belebele and XQuAD) are reported in Table 13.

D Scientific artefacts

D.1 BabelNet License

BABELEDITS is a KE benchmark made from BabelNet v5.3 downloaded from `https://babelnet.org`, made available with the BabelNet NonCommercial License (see `https://babelnet.org/full-license`).

D.2 Software Used

This project utilized the following key software libraries:

- The BabelNet Python API (version 1.2.0) was used to access and query BabelNet (Navigli and Ponzetto, 2010). It has a license that allows free usage for research purpose[7]
- Weights & Biases (`wandb`) version 0.18.7 was employed for experiment tracking and visualization citewandb and hyperparameter optimization. The Python SDK has MIT License.
- `hydra` (Yadan, 2019) was used for configuration management (version 1.3.2, MIT License).
- `EasyEdit` (Zhang et al., 2024b) was used for computing performing knowledge editing with the reported baselines (no version naming, MIT License).
- The Google Cloud Translate API (Python SDK version 3.18.0)

The main Python dependencies were the following, and all were used within the boundaries of their license:

- PyReFT (0.0.8, Apache License 2.0)
- Pyvene (0.1.6, Apache License 2.0)
- The HuggingFace datasets library (version 3.1.0, Apache License 2.0)

[7] `https://babelnet.org/license`

Model	Setting	FT-L			FT-M		r-ROME		GRACE				BABELREFT		
		Layer	Learning Rate	Norm Constr.	Layer	Learning Rate	Layer	KL factor	Layer	Learning Rate	Replacement	ε_{init}	Layer	Learning Rate	Low Rank
Llama 3.1	Single	21	5e-4	0.002	15	5e-4	15	0.0625	19	0.1	last	100	12	2e-3	64
	Sequential	19	1e-4	0.002	21	5e-4	17	1.0	21	0.1	all	100	12	2e-3	64
Gemma 2	Single	23	1e-4	0.002	27	5e-4	25	0.9	29	0.1	all	100	22	2e-3	64
	Sequential	31	1e-4	0.002	31	1e-4	5	0.9	31	0.1	all	100	18	2e-3	64

Table 9: Knowledge Editing Methods best-found hyperparameters.

Model	URL
Llama 3.1 8B Instruct	https://huggingface.co/meta-llama/Meta-Llama-3.1-8B-Instruct
Gemma 2 9B Instruct	https://huggingface.co/google/gemma-2-9b-it
NLLB 200 600M	https://huggingface.co/google/nllb-200-distilled-600M

Table 10: URLs of the models used in our study for the KE task (Llama 3.1, Gemma 2) or creating the subject aliasing prompts (NLLB).

```
You are a helpful assistant that is able to leverage its world knowledge to
convert relations extracted from a knowledge graph (for example, WordNet or
Babelnet) into natural language questions. Given the relations provided in
the user input, create a question for each relation.
In the case of the relation PLAYS_FOR, the question could be 'Which team
does <subject> play for?'.
Moreover, create an additional version of the question by rephrasing.
The input is a markdown table with 4 columns: relation_name, count, subject,
object.
When creating the question, ALWAYS keep the <subject> placeholder. The
examples provided as subject and object are there just to help you understand
the relation; do NOT include them in the question, which means that you should
NOT replace the <subject> placeholder with the examples.
You simply need to output the result in tsv format with 6 columns:
relation_name, count, subject, object, question, and rephrase.
For all the columns except question and rephrase, simply copy the values
from the input tsv. Reply directly with the tsv file, without ANY additional
text.
```

Table 11: Prompt used to ask GPT-4o to create template prompts and rephrased template prompts.

- The HuggingFace tokenizers library (version 0.20.4, Apache License 2.0)
- The HuggingFace transformers library (version 4.45.1, Apache License 2.0)
- Eleuther AI LM evaluation Harness (version 0.4.7, MIT License)
- pyahocorasick (2.1.0, BSD-3 Clause License)

D.3 Datasets used

- XQuAD (Artetxe et al., 2020) (License: CC BY-SA 4.0)
- Belebele (Bandarkar et al., 2024) (License: CC BY-SA 4.0)
- MzsRE (Wang et al., 2024c) (No license found)
- BabelNet (Navigli and Ponzetto, 2010), see Section D.1
- Wikipedia (License: CC BY-SA 4.0)

E Usage of AI assistants

We use ChatGPT and Claude 3.5 Sonnet to write parts of this paper, including text or creating/refactoring tables. Throughout development, we used GitHub Copilot as our coding assistant.

You are a helpful assistant that is able to leverage its world knowledge to convert relations extracted from a knowledge graph (for example, WordNet or Babelnet) into natural language questions. In this case we are dealing with joined triples of the form (subject, relation, object, relation_2, object_2). You need to formulate a natural language question which should be answered with object 2. Consider the case of (Messi, PLAYS_FOR, Barcelona, LOCATED_IN, Spain). The question could be 'In which country is the team that Messi plays for located?'. In the generated question, NEVER mention the object (in this case, Barcelona). Let me repeat: Do NOT INCLUDE the object in the question. The input will be a markdown table, with five columns: subject, relation, object, relation_2, object_2. Please reply directly without any additional text, one question per line, no special characters at the beginning of each line and separate each line with a SINGLE newline character and not two. Just a reminder: only one question per line, only one newline character at the end of each line.

Table 12: Prompt used to ask GPT-4o to create the prompts for multi-hop portability.

Task (Language)	Prompt Template
Belebele (all)	P: {{flores_passage}}\nQ: {{question.strip()}}\nA: {{mc_answer1}}\nB: {{mc_answer2}}\nC: {{mc_answer3}}\nD: {{mc_answer4}}\nAnswer:
XQuAD (AR)	سيا: {{context}}\n\nسؤال: {{question}}\n\nإجابة:
XQuAD (DE)	Kontext: {{context}}\n\nFrage: {{question}}\n\nAntwort:
XQuAD (EN)	Context: {{context}}\n\nQuestion: {{question}}\n\nAnswer:
XQuAD (ZH)	语境: {{context}}\n\n问题: {{question}}\n\n回答:

Table 13: Prompts used to evaluate models on different multilingual benchmarks.

	Llama 3.1					Gemma 2				
Edits	**FT-L**	**FT-M**	**r-ROME**	**GRACE**	**BABELREFT**	**FT-L**	**FT-M**	**r-ROME**	**GRACE**	**BABELREFT**
Cross-lingual Knowledge Editing Performance										
↑ *Reliability: edit success*										
100	1.59	21.94	-0.01	9.33	**37.12**	3.16	15.58	0.12	15.51	**42.30**
250	1.46	23.48	-0.02	9.31	**35.89**	2.81	12.74	0.01	16.36	**41.48**
500	1.59	19.98	-0.02	9.27	**37.48**	2.12	10.26	0.01	16.54	**40.90**
Full	1.52	19.51	-0.04	9.26	**36.51**	3.02	9.79	0.01	17.10	**40.72**
↑ *Generality: edit success over paraphrases*										
100	1.29	19.73	-0.02	0.72	**34.40**	2.35	10.28	0.12	8.30	**39.52**
250	1.55	21.88	-0.02	0.54	**33.74**	2.09	7.81	0.01	8.76	**38.46**
500	1.48	17.87	-0.02	0.58	**35.15**	1.41	5.99	0.01	8.86	**38.48**
Full	1.71	17.69	-0.04	0.47	**34.25**	2.37	5.76	0.01	9.36	**38.18**
↑ *Subject-Alias: edit success over prompts with a subject alias*										
100	1.49	17.23	-0.01	2.45	**23.08**	1.72	10.30	0.01	9.86	**26.93**
250	0.52	15.28	-0.01	1.91	**25.65**	1.63	6.11	0.00	10.29	**28.87**
500	0.83	11.05	-0.01	1.47	**28.33**	0.92	5.07	0.00	7.96	**27.95**
Full	0.87	11.93	-0.01	1.32	**27.85**	1.36	4.66	0.00	9.18	**28.81**
↓ *Locality: preservation of unrelated knowledge*										
100	7.35	4.97	21.13	**0.03**	4.41	11.00	6.28	2.64	**0.06**	5.90
250	7.49	5.66	14.63	**0.03**	4.12	9.80	6.07	0.71	**0.10**	5.91
500	8.10	5.97	15.25	**0.02**	4.03	8.23	6.65	0.49	**0.05**	6.15
Full	7.34	5.97	13.13	**0.03**	4.20	11.23	6.75	0.45	**0.09**	6.37
↑ *Multi-Hop: edit success over multi-hop queries*										
100	-0.16	0.25	-0.17	0.00	**1.00**	0.01	0.42	0.07	0.00	**1.83**
250	-0.69	-0.62	-0.70	0.00	**0.84**	0.01	0.20	0.01	0.01	**1.32**
500	-0.53	-0.32	-0.55	0.00	**0.55**	0.03	0.26	0.01	0.00	**0.95**
Full	-0.37	0.12	-0.50	0.00	**1.27**	0.05	0.26	0.01	0.04	**1.64**
Downstream Performance										
↑ *Belebele (accuracy)*										
Original	73.59	73.59	73.59	73.59	73.59	84.68	84.68	84.68	84.68	84.68
100	73.42	73.99	34.89	73.59	73.50	84.48	84.46	24.14	**84.71**	84.59
250	73.73	72.02	22.51	73.50	73.47	84.49	84.56	22.92	**84.71**	84.60
500	73.79	68.06	28.64	73.56	73.50	84.48	84.38	26.07	**84.71**	84.70
Full	72.92	60.26	22.58	**73.50**	73.39	84.64	84.40	28.62	**84.71**	84.70
↑ *XQuAD (EM)*										
Original	29.60	29.60	29.60	29.60	29.60	31.79	31.79	31.79	31.79	31.79
100	18.07	20.00	0.00	**29.60**	29.60	29.64	29.98	0.13	**31.76**	31.76
250	27.21	12.29	0.00	**29.71**	29.68	30.04	29.81	0.21	**31.76**	31.72
500	19.37	1.58	0.00	**29.71**	29.45	29.03	28.78	2.77	**31.76**	31.64
Full	19.35	0.36	0.00	**29.71**	29.18	26.37	29.81	4.33	**31.76**	31.70

Table 14: Comprehensive comparison of cross-lingual knowledge editing and downstream task performance for Llama 3.1 8B and Gemma 2 9B models with sequential editing done in English with an increasing number of sequential edits. Editing metrics are averaged over all target languages and multiplied by 100 for readability. Bold numbers indicate the best performance for each metric and model combination. Downstream performance is averaged over the target languages: Original indicates the model performance before editing.

Method	avg	ar	de	en	fr	hr	it	ja	ka	my	qu	zh
Cross-lingual Knowledge Editing Performance												
↑ *Reliability*												
FT-L	1.52	0.48	1.83	6.56	2.20	0.91	2.47	0.38	0.47	0.26	0.43	0.78
FT-M	19.51	7.82	30.52	63.77	25.87	19.86	33.78	8.42	4.83	0.33	7.96	11.42
r-ROME	-0.04	-0.04	-0.03	-0.00	-0.01	-0.02	-0.20	-0.03	-0.10	-0.04	-0.00	-0.00
GRACE	9.26	-0.00	1.13	99.08	0.38	0.47	0.75	0.00	0.00	0.00	-0.00	0.00
BABELREFT	36.51	20.90	63.04	98.49	49.86	40.92	64.38	19.44	18.38	2.17	10.81	13.19
↑ *Generality*												
FT-L	1.71	0.38	1.83	8.02	2.13	1.14	2.92	0.30	0.50	0.25	0.56	0.83
FT-M	17.69	6.74	27.56	55.06	24.51	17.90	32.28	7.55	4.39	0.21	7.79	10.58
r-ROME	-0.04	-0.02	-0.03	-0.00	-0.01	-0.03	-0.13	-0.06	-0.14	-0.05	-0.00	-0.00
GRACE	0.47	-0.00	0.76	2.67	0.57	0.47	0.47	0.10	0.09	-0.00	0.00	0.00
BABELREFT	34.25	20.90	59.17	93.69	47.64	35.17	61.30	18.66	13.30	2.25	11.61	13.00
↓ *Locality*												
FT-L	7.34	8.88	8.97	9.13	8.50	7.35	8.92	7.74	8.21	1.28	3.70	8.05
FT-M	5.97	7.57	5.96	6.00	5.72	5.10	5.70	6.26	6.38	6.51	3.72	6.79
r-ROME	13.13	9.70	11.70	12.66	12.10	12.45	12.21	12.17	21.16	15.34	14.65	10.25
GRACE	0.03	0.00	0.00	0.30	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00
BABELREFT	4.20	2.50	6.64	9.60	6.18	4.31	6.95	2.56	1.30	0.49	2.16	3.45
↑ *Multi-hop portability*												
FT-L	-0.37	-1.54	-1.26	-0.16	-0.19	-0.01	-0.27	-0.25	-0.38	0.03	0.01	-0.02
FT-M	0.12	-1.54	-0.34	1.65	0.72	0.70	0.79	-0.25	-0.34	-0.04	-0.01	-0.01
r-ROME	-0.50	-1.54	-1.51	-0.60	-0.37	-0.18	-0.52	-0.26	-0.38	-0.05	-0.03	-0.02
GRACE	-0.00	-0.02	0.01	0.00	0.00	0.00	-0.00	0.00	-0.01	-0.00	-0.00	0.00
BABELREFT	1.27	0.53	2.55	2.94	2.29	1.68	1.86	0.65	0.63	-0.00	0.57	0.21
↑ *Subject-alias portability*												
FT-L	0.87	0.25	0.71	4.19	0.25	1.20	2.21	0.16	0.01	0.31	0.00	0.27
FT-M	11.93	2.90	28.66	29.77	12.05	21.53	26.15	2.83	2.39	2.28	0.30	2.40
r-ROME	-0.01	-0.02	-0.00	-0.00	-0.00	-0.00	-0.01	-0.01	-0.05	-0.00	-0.00	-0.00
GRACE	1.32	0.00	0.00	14.57	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00
BABELREFT	27.85	7.65	53.93	87.29	39.05	33.14	55.55	3.53	7.84	4.02	12.19	2.11
Downstream Performance												
↑ *Belebele*												
Original	73.59	73.22	77.11	88.67	82.78	74.00	81.0	77.67	52.33	43.78	-	85.33
FT-L	72.92	74.33	77.22	86.89	82.56	73.00	80.44	76.33	52.22	42.00	-	84.22
FT-M	60.26	57.33	64.11	70.78	69.56	56.78	66.89	60.67	45.11	35.00	-	76.33
r-ROME	22.58	25.11	19.33	20.78	24.00	23.89	23.00	22.33	21.22	24.89	-	21.22
GRACE	73.50	73.22	77.11	88.67	83.00	73.33	80.67	77.67	52.44	43.78	-	85.11
BABELREFT	73.39	73.67	76.78	88.56	82.89	73.22	80.11	77.44	52.22	43.78	-	85.22
↑ *XQuAD*												
Original	29.60	28.91	34.71	32.44	-	-	-	-	-	-	-	22.35
FT-L	19.35	11.01	18.49	28.49	-	-	-	-	-	-	-	19.41
FT-M	0.36	0.00	0.42	0.17	-	-	-	-	-	-	-	0.84
r-ROME	0.00	0.00	0.00	0.00	-	-	-	-	-	-	-	0.00
GRACE	29.71	28.99	34.54	33.03	-	-	-	-	-	-	-	22.27
BABELREFT	29.18	28.74	34.12	31.85	-	-	-	-	-	-	-	22.02

Table 15: Results detailed by language for the sequential edits performed in English on the full BABELEDITS dataset with LLaMA 3.1 7B.

Method	avg	ar	de	en	fr	hr	it	ja	ka	my	qu	zh
Cross-lingual Knowledge Editing Performance												
↑ *Reliability*												
FT-L	3.02	0.76	3.24	18.73	3.06	0.72	5.64	0.29	0.11	-0.02	0.17	0.55
FT-M	9.79	1.53	13.59	62.49	8.82	4.04	13.57	1.23	0.78	-0.02	0.06	1.55
r-ROME	0.01	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.09	0.00	0.00
GRACE	17.10	4.80	23.50	98.55	16.77	10.85	22.88	3.98	2.47	0.00	0.09	4.25
BABELREFT	36.51	20.90	63.04	98.49	49.86	40.92	64.38	19.44	18.38	2.17	10.81	13.19
↑ *Generality*												
FT-L	2.37	0.69	2.58	13.83	2.67	0.79	4.23	0.28	0.14	-0.03	0.23	0.61
FT-M	5.76	1.00	10.08	29.53	6.57	3.66	8.91	1.07	0.45	-0.02	0.04	2.10
r-ROME	0.01	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.14	0.00	0.00
GRACE	9.36	3.56	18.58	30.52	14.00	9.07	17.18	3.35	2.23	0.00	0.00	4.47
BABELREFT	34.25	20.90	59.17	93.69	47.64	35.17	61.30	18.66	13.30	2.25	11.61	13.00
↓ *Locality*												
FT-L	11.23	12.29	13.27	15.65	13.86	13.65	14.33	5.52	14.13	1.41	11.84	7.61
FT-M	6.75	8.90	7.31	8.69	7.64	7.76	8.53	4.88	6.61	0.94	6.94	6.07
r-ROME	0.45	0.27	0.22	0.07	0.14	0.65	0.18	0.69	1.08	0.86	0.35	0.49
GRACE	0.09	0.05	0.22	0.34	0.08	0.04	0.18	0.01	0.00	0.00	0.00	0.04
BABELREFT	4.20	2.50	6.64	9.60	6.18	4.31	6.95	2.56	1.30	0.49	2.16	3.45
↑ *Multi-hop portability*												
FT-L	0.05	0.05	0.06	0.22	0.11	0.01	0.08	0.07	0.01	-0.04	0.01	0.01
FT-M	0.26	0.04	0.38	1.29	0.24	0.11	0.16	0.29	0.01	-0.04	0.16	0.17
r-ROME	0.01	0.00	0.00	0.00	-0.00	0.00	-0.00	-0.01	-0.00	0.13	0.00	0.01
GRACE	0.04	0.00	0.16	0.16	0.00	0.00	0.16	-0.00	-0.00	0.00	0.00	0.00
BABELREFT	1.27	0.53	2.55	2.94	2.29	1.68	1.86	0.65	0.63	-0.00	0.57	0.21
↑ *Subject-alias portability*												
FT-L	1.36	0.24	1.52	8.33	0.49	0.18	4.08	0.07	0.01	0.01	0.00	0.05
FT-M	4.66	0.47	5.53	28.44	5.20	2.93	6.87	0.36	0.36	0.01	0.76	0.37
r-ROME	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	-0.00	0.00	0.00
GRACE	9.18	1.54	10.28	34.75	15.51	14.00	18.89	1.23	1.02	1.02	2.29	0.44
BABELREFT	27.85	7.65	53.93	87.29	39.05	33.14	55.55	3.53	7.84	4.02	12.19	2.11
Downstream Performance												
↑ *Belebele*												
Original	84.68	85.78	88.00	93.22	90.67	86.67	89.67	85.11	74.89	64.00	-	88.78
FT-L	84.64	85.89	88.11	93.78	90.67	86.67	89.67	84.89	75.33	63.11	-	88.33
FT-M	84.40	86.00	87.56	93.89	90.67	86.11	89.56	85.33	75.11	61.33	-	88.44
r-ROME	28.62	24.33	25.22	52.56	30.22	22.78	28.89	23.56	22.67	22.89	-	33.11
GRACE	84.71	86.00	88.11	93.22	90.78	86.78	89.78	85.11	75.11	63.44	-	88.78
BABELREFT	84.70	86.00	88.22	93.33	90.67	86.78	89.78	85.11	75.11	63.22	-	88.78
↑ *XQuAD*												
Original	31.79	24.71	27.31	46.89	-	-	-	-	-	-	-	28.24
FT-L	26.37	21.26	22.44	43.19	-	-	-	-	-	-	-	18.57
FT-M	29.81	20.76	26.47	45.80	-	-	-	-	-	-	-	26.22
r-ROME	4.33	0.25	3.53	12.35	-	-	-	-	-	-	-	1.18
GRACE	31.76	24.62	27.48	47.23	-	-	-	-	-	-	-	27.73
BABELREFT	31.70	24.62	27.23	47.31	-	-	-	-	-	-	-	27.65

Table 16: Results detailed by language for the sequential edits performed in English on the full BABELEDITS dataset with Gemma 2 9B.

Methods	Llama 3.1					Gemma 2				
	avg	de	en	fr	zh	**avg**	de	en	fr	zh
Cross-lingual Knowledge Editing Performance										
↑ *Reliability*										
FT-L	1.50	0.94	3.90	0.78	0.39	2.54	1.23	7.25	1.34	0.34
FT-M	27.53	23.09	64.76	17.52	4.75	25.28	17.63	68.44	12.86	2.20
r-ROME	-0.13	-0.20	-0.24	-0.05	-0.02	0.00	0.00	0.00	0.00	-0.00
GRACE	25.17	0.78	99.51	0.40	0.00	30.13	13.10	98.95	7.11	1.34
BABELREFT	45.24	44.31	97.23	34.25	5.17	47.74	48.89	97.95	37.45	6.68
↑ *Generality*										
FT-L	1.13	0.75	2.65	0.72	0.40	1.89	1.20	5.02	1.07	0.28
FT-M	23.23	20.66	51.87	16.01	4.39	17.31	15.23	41.99	10.17	1.84
r-ROME	-0.11	-0.24	-0.14	-0.04	-0.02	0.00	0.00	0.00	-0.00	-0.00
GRACE	2.57	0.53	9.50	0.26	0.00	12.33	9.09	35.21	4.53	0.51
BABELREFT	42.13	42.57	89.87	31.09	4.99	44.00	46.55	88.80	34.15	6.48
Downstream Performance										
↑ *Belebele*										
Original	83.47	77.11	88.67	82.78	85.33	90.17	88.00	93.22	90.67	88.78
FT-L	83.22	77.00	88.44	82.56	84.89	90.06	88.11	93.44	90.56	88.11
FT-M	71.53	63.33	80.78	68.33	73.67	90.03	87.78	93.33	90.67	88.33
r-ROME	22.89	22.89	22.89	22.89	22.89	25.83	24.56	27.11	26.22	25.44
GRACE	83.36	76.78	88.67	82.78	85.22	90.22	88.11	93.22	90.78	88.78
BABELREFT	83.31	77.00	88.56	82.44	85.22	90.03	87.78	93.11	90.56	88.67
↑ *XQuAD*										
Original	29.83	34.71	32.44	-	22.35	34.15	27.31	46.89	-	28.24
FT-L	4.90	1.26	9.08	-	4.37	33.81	28.07	44.87	-	28.49
FT-M	5.97	4.29	7.56	-	6.05	34.71	31.43	47.06	-	25.63
r-ROME	0.00	0.00	0.00	-	0.00	3.31	0.67	8.99	-	0.25
GRACE	29.94	34.54	33.03	-	22.27	34.15	27.48	47.23	-	27.73
BABELREFT	29.75	34.45	32.52	-	22.27	34.12	27.31	46.81	-	28.24

Table 17: Comparison of knowledge editing methods across Llama 3.1 and Gemma 2 models for sequential editing done in English on the entire MzsRE dataset (742 edits), showing both editing performance and downstream task evaluation.

Methods	avg	ar	de	en	fr	hr	it	ja	ka	my	qu	zh
Cross-lingual Knowledge Editing Performance												
↑ *Reliability: edit success*												
FT-L	4.26	1.84	6.71	7.70	5.82	5.95	7.78	1.93	0.85	0.31	5.43	2.48
FT-M	28.87	25.46	39.29	37.10	35.98	34.98	39.74	**25.61**	20.33	8.57	24.22	26.32
r-ROME	16.24	10.78	24.00	29.76	22.26	17.41	24.29	11.10	2.64	5.31	14.96	16.13
GRACE	**32.58**	19.55	**45.33**	**46.19**	**40.65**	**41.69**	**46.73**	19.19	18.66	**14.57**	**41.37**	24.42
BABELREFT	30.96	**27.02**	43.65	37.48	39.02	39.18	41.54	24.46	**22.10**	9.41	29.01	**27.64**
↑ *Generality: edit success over paraphrases*												
FT-L	4.38	1.93	7.05	8.35	5.96	5.91	8.06	1.99	0.72	0.34	5.36	2.47
FT-M	27.81	**24.61**	38.07	35.42	34.53	33.48	**38.61**	**24.94**	**19.23**	8.24	23.38	**25.36**
r-ROME	15.67	10.48	23.50	28.57	21.69	16.04	24.70	10.40	2.68	5.07	14.14	15.10
GRACE	**32.20**	19.16	**44.92**	**45.63**	**40.21**	**41.37**	**46.27**	19.06	18.25	14.49	**40.97**	23.93
BABELREFT	28.45	22.55	41.59	35.04	37.30	35.89	40.18	22.55	17.64	8.46	26.50	25.26
↓ *Locality: preservation of unrelated knowledge*												
FT-L	2.80	2.69	**3.53**	2.94	**3.22**	3.47	3.68	3.01	2.30	0.77	**2.37**	**2.78**
FT-M	**2.79**	3.04	3.56	**2.86**	3.39	3.19	**3.58**	3.02	1.97	**0.69**	2.48	2.88
r-ROME	2.91	**2.04**	3.61	4.16	3.95	**2.32**	4.10	**2.49**	**1.26**	1.90	2.68	3.49
GRACE	6.62	6.72	6.56	3.01	5.56	7.40	6.43	7.77	5.91	8.45	9.85	5.21
BABELREFT	3.70	2.53	5.00	4.20	4.77	4.38	4.87	2.69	1.96	1.57	5.04	3.68
↑ *Subject-Alias: edit success over prompts with a subject alias*												
FT-L	3.91	2.34	6.16	7.48	5.85	5.25	7.32	1.78	0.69	0.21	3.60	2.29
FT-M	24.24	**21.19**	34.76	34.16	33.04	30.96	36.47	**20.03**	16.74	0.52	19.00	19.80
r-ROME	10.30	8.00	15.97	20.53	15.54	11.39	18.03	6.20	1.54	2.07	8.21	5.79
GRACE	**31.96**	17.11	**45.57**	**48.40**	**43.28**	**42.87**	**47.74**	17.08	**17.12**	**9.48**	**42.01**	**20.87**
BABELREFT	19.16	12.56	32.25	27.42	29.25	25.50	30.15	11.52	10.60	0.55	19.81	11.15
↑ *Multi-hop: edit success over multi-hop queries*												
FT-L	-0.14	-0.30	-0.00	0.09	-0.06	0.03	-0.01	-0.26	-0.39	-0.21	-0.07	-0.35
FT-M	0.67	0.61	0.91	0.85	0.72	0.54	0.81	0.70	0.43	0.81	0.44	0.50
r-ROME	0.30	0.10	0.34	0.51	0.52	0.50	0.51	0.21	-0.14	0.19	0.47	0.08
GRACE	0.37	-0.12	0.66	0.65	0.50	0.43	0.58	0.06	-0.02	0.83	**0.55**	-0.08
BABELREFT	**1.16**	**0.98**	**1.31**	**1.25**	**1.46**	**1.04**	**1.42**	**1.02**	**1.11**	**1.72**	**0.55**	**0.88**
Downstream Performance												
↓ *Delta PPL*												
FT-L	59.25	15.80	2.11	0.89	1.44	6.57	2.02	1.82	2.68	2.13	6.12e2	3.48
FT-M	79.52	84.01	40.24	3.37	16.05	26.93	36.05	15.15	11.95	58.20	5.77e2	5.42
r-ROME	4.02e2	26.00	10.89	6.00	10.17	34.31	15.50	18.19	1.29e3	1.93e2	2.80e3	13.30
GRACE	5.46e4	3.11e4	5.09e4	8.65e4	3.61e4	5.29e4	5.12e4	1.91e4	2.75e4	1.33e4	1.29e5	1.02e5
BABELREFT	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.00**

Table 18: Comprehensive comparison of cross-lingual knowledge editing in the single edit setup and perplexity variation for Llama 3.1 8B. Each column (except 'avg') corresponds to an editing language, and the results are averaged across all the target languages. Column 'avg' averages those results. Values are percentages except for perplexity where they are absolute values, and bold numbers indicate best performance for each metric and language.

Methods	avg	ar	de	en	fr	hr	it	ja	ka	my	qu	zh
Cross-lingual Knowledge Editing Performance												
↑ *Reliability: edit success*												
FT-L	4.19	2.07	8.75	9.13	7.38	4.28	6.42	1.70	0.29	1.21	2.75	2.09
FT-M	24.37	17.79	33.76	36.12	31.67	27.30	33.47	17.55	15.64	8.79	24.61	21.42
r-ROME	6.80	3.99	10.85	13.90	9.50	8.93	11.34	2.86	0.50	0.96	7.82	4.18
GRACE	24.88	17.97	33.89	37.86	31.54	28.33	33.48	18.13	16.50	9.10	24.73	**22.14**
BABELREFT	**30.45**	**23.88**	**43.11**	**42.10**	**39.34**	**37.74**	**40.84**	**20.52**	**21.84**	**9.21**	**34.82**	21.49
↑ *Generality: edit success over paraphrases*												
FFT-L	3.99	1.92	8.08	9.05	6.75	3.88	5.93	1.75	0.21	1.59	2.96	1.83
FT-M	22.36	15.75	31.30	32.85	28.97	24.92	30.85	16.29	13.61	8.53	23.65	19.22
r-ROME	6.49	3.63	10.29	13.08	9.07	8.57	10.94	2.79	0.45	0.94	7.71	3.88
GRACE	23.08	16.21	31.64	35.18	29.06	25.90	30.99	17.11	14.73	**8.97**	24.02	**20.12**
BABELREFT	**27.68**	**19.64**	**40.48**	**39.24**	**36.81**	**34.30**	**38.89**	**18.26**	**17.43**	8.17	**32.62**	18.68
↓ *Locality: preservation of unrelated knowledge*												
FT-L	**2.10**	**2.24**	**2.33**	**2.13**	**2.39**	**2.29**	**2.29**	**2.46**	**2.51**	**0.39**	**1.86**	**2.23**
FT-M	3.30	2.71	3.29	3.56	3.87	3.09	3.50	4.06	2.91	0.82	4.38	4.08
r-ROME	3.28	2.49	3.81	4.27	3.56	3.58	3.26	3.43	2.81	1.43	3.37	4.10
GRACE	3.07	2.59	3.09	3.51	3.64	2.34	3.38	3.41	3.37	0.40	3.95	4.12
BABELREFT	5.04	2.79	6.24	7.05	6.13	5.62	5.96	3.43	4.46	1.01	8.67	4.05
↑ *Subject-Alias: edit success over prompts with a subject alias*												
FT-L	2.66	1.36	5.50	6.84	4.96	2.98	4.01	1.06	0.16	0.02	1.36	0.97
FT-M	17.92	**12.81**	25.08	28.33	26.30	20.32	26.56	12.02	12.01	0.53	20.10	13.00
r-ROME	6.02	3.28	10.47	13.06	7.05	7.79	10.34	3.05	0.80	**0.62**	5.97	3.75
GRACE	18.01	12.69	24.65	27.09	25.32	21.12	24.94	**13.28**	**12.65**	0.39	21.96	**14.06**
BABELREFT	**18.34**	10.96	**28.34**	**30.14**	**28.30**	**24.28**	**29.21**	8.79	9.62	0.26	**25.02**	6.83
↑ *Multi-hop: edit success over multi-hop queries*												
FT-L	0.44	0.18	0.80	0.58	0.67	0.57	0.51	0.19	0.12	0.82	0.22	0.18
FT-M	0.86	0.57	0.98	**1.15**	0.96	0.73	1.00	0.68	0.45	1.50	0.85	**0.57**
r-ROME	0.25	0.13	0.37	0.38	0.32	0.23	0.36	0.15	0.13	0.19	0.31	0.14
GRACE	0.64	0.43	0.75	0.90	0.80	0.66	0.80	0.61	0.38	0.62	0.59	0.44
BABELREFT	**1.07**	**0.66**	**1.22**	1.12	**1.27**	**1.01**	**1.03**	**0.90**	**0.63**	**2.47**	**0.88**	0.55
Downstream Performance												
↓ *Delta PPL*												
FT-L	12.69	-6.37	-19.02	-14.01	-17.13	-16.17	-18.60	4.66	23.14	30.77	171.04	1.34
FT-M	1.38e2	48.18	52.00	34.94	29.17	69.28	61.19	9.10	31.36	65.98	1.09e3	19.72
r-ROME	1.65e8	4.96e6	1.94e7	8.57e6	1.85e7	1.72e8	1.03e7	1.37e7	4.37e5	6.36e5	1.57e9	2.82e6
GRACE	8.20e2	9.22e2	4.66e2	6.80e2	5.31e2	6.39e2	5.12e2	4.93e2	5.94e2	4.92e2	2.80e3	8.83e2
BABELREFT	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**

Table 19: Comprehensive comparison of cross-lingual knowledge editing and perplexity variation for Gemma 2 9B. Each column (except 'avg') corresponds to an editing language, and the results are averaged across all the target languages. Column 'avg' averages those results. Values are percentages except for perplexity where they are absolute values, and bold numbers indicate best performance for each metric and language.